# KỶ YẾU
# HỘI NGHỊ KHOA HỌC TRẺ
# LẦN THỨ 5 - NĂM 2023

## YSC 2023

## TRI THỨC TRẺ TRONG KỶ NGUYÊN SỐ

## LĨNH VỰC CÔNG NGHỆ THÔNG TIN

NHÀ XUẤT BẢN ĐẠI HỌC CÔNG NGHIỆP
THÀNH PHỐ HỒ CHÍ MINH

**BỘ CÔNG THƯƠNG**
**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH**

# KỶ YẾU HỘI NGHỊ KHOA HỌC TRẺ LẦN 5 NĂM 2023 – (YSC2023) TRI THỨC TRẺ TRONG KỶ NGUYÊN SỐ LĨNH VỰC CÔNG NGHỆ THÔNG TIN

**NHÀ XUẤT BẢN ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH**

**HỘI NGHỊ KHOA HỌC TRẺ LẦN 5 NĂM 2023**
**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH**

**----------**

## BAN TỔ CHỨC

| | | | |
|---|---|---|---|
| - | PGS.TS. Huỳnh Trung Hiếu | Phó Hiệu trưởng | Trưởng ban |
| - | TS. Nguyễn Xuân Hồng | Phó Hiệu trưởng | Phó Trưởng ban |
| - | PGS.TS. Đàm Sao Mai | Phó Hiệu trưởng | Phó Trưởng ban |
| - | PGS.TS. Trịnh Ngọc Nam | Trưởng phòng QLKH & HTQT | Phó Trưởng ban |
| - | KS. Huỳnh Phú Vinh | Bí thư Đoàn Thanh niên Trường | Thành viên |
| - | PGS.TS. Đoàn Văn Đạt | Trưởng tiểu ban Hóa – Sinh – Thực phẩm – Môi trường | Thành viên |
| - | TS. Lê Nhật Duy | Trưởng tiểu ban Điện – Điện tử – Công nghệ Thông tin | Thành viên |
| - | TS. Nguyễn Quốc Cường | Trưởng tiểu ban Kinh tế | Thành viên |
| - | PGS.TS. Nguyễn Minh Phú | Trưởng tiểu ban Cơ khí – Xây dựng – Động lực – Nhiệt lạnh | Thành viên |
| - | TS. Phan Thị Tuyết Nga | Trưởng tiểu ban Khoa học Xã hội và Nhân văn | Thành viên |
| - | ThS. Bùi Đình Tiền | Phó Trưởng phòng QLKH & HTQT | Thành viên |
| - | TS. Phạm Trần Bích Thuận | Phó Trưởng phòng QLKH & HTQT | Thành viên |
| - | ThS. Phạm Trung Kiên | Trưởng phòng Tổ chức – Hành chính | Thành viên |
| - | ThS. Phạm Thị Quế Minh | Trưởng phòng Tài chính Kế toán | Thành viên |
| - | TS. Lê Ngọc Sơn | Viện trưởng Viện Đào tạo Quốc tế và Sau Đại học | Thành viên |
| - | ThS. Nguyễn Thị Thương | Giám đốc Trung tâm Thông tin – Truyền thông | Thành viên |
| - | ThS. Nguyễn Phúc Hưng | Giám đốc Trung tâm Quản trị Hệ thống | Thành viên |
| - | ThS. Nguyễn Quý Tuấn | Trưởng phòng Quản trị | Thành viên |
| - | ThS. Dương Tiến Đoàn | Trưởng phòng Dịch vụ | Thành viên |
| - | ThS. Trần Thu Thảo | Trưởng ban Học tập – Nghiên cứu Khoa học Đoàn Trường | Thành viên |
| - | ThS. Nguyễn Minh Tú Anh | Chuyên viên Phòng QLKH & HTQT | Thành viên |
| - | ThS. Nguyễn Phúc Thùy Dương | Chuyên viên Phòng QLKH & HTQT | Thành viên |

## BAN CHUYÊN MÔN THẨM DUYỆT BÀI BÁO

**Tiểu ban Điện – Điện tử – Công nghệ Thông tin**

| | | | |
|---|---|---|---|
| - | TS. Lê Nhật Duy | Khoa Công nghệ Thông tin | Trưởng Tiểu ban |
| - | PGS.TS. Nguyễn Ngọc Sơn | Khoa Công nghệ Điện tử | Thành viên |
| - | TS. Đặng Thị Phúc | Khoa Công nghệ Thông tin | Thành viên |
| - | TS. Nguyễn Hoài Thương | Khoa Công nghệ Điện | Thư ký |

# DANH MỤC BÀI BÁO

**YSC5.F001** ..... **5**

**ChainForHelp: HIGHLY TRANSPARENT PEER-TO-PEER CHARITY FUNDING SYSTEM BASED ON BLOCKCHAIN TECHNOLOGY**

ĐẶNG NGỌC PHONG[1*], NGUYỄN ĐẶNG HOÀNG THI[1], PHAN HẢI TRIỀU[1], GIÁP CHÍ CƯỜNG[1], HUỲNH NAM[1]

**YSC5.002** ..... **15**

**QUY HOẠCH TUYẾN TÍNH NGUYÊN VÀ ỨNG DỤNG TRONG THỰC TẾ**

TRỊNH NGUYỄN HOÀNG VŨ[1*], PHẠM VIẾT DŨNG MINH[1], TRẦN THÁI HÀ[1], TRỊNH DƯƠNG HOAN[1], TRẦN BÌNH MINH[1], LÊ PHÚC LỮ[1], NGUYỄN HỮU TÌNH[1]

**YSC5.F003** ..... **28**

**PHÂN TÍCH DỮ LIỆU BẢO TOÀN TÍNH RIÊNG TƯ TRÊN CÂY MERKLE CỦA BLOCKCHAIN**

LÂM NGỌC LONG[1*], ĐỖ QUỐC TUẤN[1], TRẦN ANH TUẤN[1], NGUYỄN MINH LUẬN[1]
TRƯƠNG MẠNH QUỲNH[1], NGUYỄN THÀNH TÍN[1] , VÕ QUỐC THỊNH[1], ĐÀO XUÂN HOÀNG TUẤN[1], NGUYỄN HỮU TÌNH[1], LÊ PHÚC LỮ[1]

**YSC5.F004** ..... **40**

**CẢI TIẾN THUẬT TOÁN TỐI ƯU HÀM LỒI BẰNG TÌM KIẾM TAM PHÂN VÀ QUY HOẠCH ĐỘNG**

TRẦN LỘC[1*], LÂM QUANG PHÚ[1], TRƯƠNG CÔNG ĐẠT[1], NGUYỄN GIA LÂM[1]
TRẦN TRỌNG TRÍ[1], MAI THỊ DIỆU MY[1], PHAN TẤN TÀI[1], LÊ PHÚC LỮ[1], NGUYỄN HỮU TÌNH[1]

**YSC5.F005** ..... **53**

**CÁC MÔ HÌNH CHIA SẺ DỰA TRÊN MÃ HÓA NGƯỠNG ỨNG DỤNG VÀO BÀI TOÁN CHỮ KÝ SỐ**

TRẦN MINH LONG[1*], PHAN TÔN LỘC NGUYÊN[1], NGUYỄN MINH PHÚC[1], TRẦN NGỌC HUYỀN[1], ĐÀO THỊ MỸ LINH[1], NGUYỄN THỊ TUYẾT LAN[1], NGUYỄN QUANG HUY[1], DƯƠNG HOÀNG HUY[1], NGUYỄN HUY HOÀNG[1], LÊ PHÚC LỮ[1], NGUYỄN HỮU TÌNH[1]

**YSC5.F006** ..... **65**

**ỨNG DỤNG KỸ THUẬT XỬ LÝ HÌNH ẢNH KẾT HỢP VỚI MẠNG HỌC SÂU PHÁT HIỆN UNG THƯ VÚ**

HÀ BẢO ANH[1*], LÊ THÀNH NGHĨA[1], NGUYỄN VĂN NAM[1], LÂM QUANG PHÚ[1]
ĐÀO XUÂN HOÀNG TUẤN[1], HUỲNH TRUNG HIẾU[1]

**YSC5.F007** ..... **99**

**IUH ONE: CÔNG CỤ RÚT GỌN LIÊN KẾT, THEO DÕI LƯỢT TRUY CẬP VÀ ĐÁNH GIÁ TỈ LỆ TƯƠNG TÁC CỦA NGƯỜI DÙNG CHO CÁC WEBSITE**

TON LONG PHUOC[1*], TOAN DUC VO[1], DINH NHAT PHAN[1]

**YSC.F008** ..... **110**

**XÂY DỰNG MỘT ỨNG DỤNG TRỰC TUYẾN VỚI MICROSERVICES VÀ CLUSTERING**

TẠ DUY CÔNG CHIẾN[12], NGUYỄN VĂN TOÀN[3]

**YSC5.F009** ..... **119**

**LINK: ỨNG DỤNG CHAT NỘI BỘ VÀ ĐỒNG BỘ HÓA TIẾN ĐỘ CÔNG VIỆC TRÊN ANDROID**

TRẦN THỊ ANH THI[1], HOÀNG HUY VŨ[1], TRẦN THÀNH NAM[1]

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F010** ..... **126**
**PHÁT TRIỂN GAME "KHU VƯỜN MINI" TRÊN NỀN TẢNG ANDROID KẾT HỢP CÔNG NGHỆ UNITY-2D**
TRẦN THỊ ANH THI[1], PHAN HỒNG HÀ[1]

**YSC5.F011** ..... **133**
**XÂY DỰNG ỨNG DỤNG QUẢN LÝ ĐẶT MÓN ATFOOD - TẠI HỆ THỐNG NHÀ HÀNG MỘC QUÁN TRÊN THIẾT BỊ ĐI ĐỘNG**
TRẦN THỊ ANH THI[1], NGUYỄN THÁI AN[1], ĐINH THỊ PHƯƠNG THẢO[1]

**YSC5.F012** ..... **141**
**FARMING SIMULATOR 3D TT: PHÁT TRIỂN TRÒ CHƠI LÀM VƯỜN OFFLINE TRÊN THIẾT BỊ DI ĐỘNG SỬ DỤNG CÔNG NGHỆ UNITY**
TRẦN THỊ ANH THI[1], LÊ VÕ MINH THUẬN[1], VÕ TRUNG TÍN[1]

**YSC5.F013** ..... **149**
**DỰ ĐOÁN GIÁ CỔ PHIẾU SỬ DỤNG THUẬT TOÁN DI TRUYỀN**
NGUYỄN NĂNG ANH[1*], NGUYỄN TẤN ĐẮT[1], NGUYỄN HUỲNH ĐAN KHÁNH[1]
ĐỖ THÀNH MINH PHÚ[1], CAO HUỲNH VÕ THANH[1], LÊ PHÚC LỮ[1], NGUYỄN HỮU TÌNH[1]

**YSC5.F014** ..... **163**
**T&T HEALTHCARE: CÔNG CỤ HỖ TRỢ SỨC KHỎE THÔNG MINH CHO BỆNH NHÂN ĐƯỜNG HUYẾT DỰA VÀO TRÍ TUỆ NHÂN TẠO**
TÔN LONG PHƯỚC[1*], LÊ TUẤN[1], TĂNG BẢO TRẤN[1]

**YSC5.F015** ..... **176**
**AVTEDU - CÔNG CỤ HỖ TRỢ VÀ QUẢN LÝ HỌC VỤ TẠI TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH**
TÔN LONG PHƯỚC[1*], NGUYỄN VIỆT ANH[1], PHẠM NGUYỄN VĂN TRƯỜNG[1]

**YSC5.F016** ..... **187**
**AMANDA- CÔNG CỤ HỖ TRỢ MUA SẮM TRỰC TUYẾN TRONG LĨNH VỰC THƯƠNG MẠI ĐIỆN TỬ DỰA TRÊN CÔNG NGHỆ BLOCKCHAIN**
TÔN LONG PHƯỚC[1*], HOÀNG VĂN CHINH[1], NGUYỄN THỊ HẢO[1]

**YSC5.F017** ..... **196**
**MEDIHEARTH: GIẢI PHÁP CHĂM SÓC SỨC KHỎE THÔNG MINH CHO BỆNH NHÂN TIM MẠCH DỰA TRÊN CÔNG NGHỆ INTERNET VẠN VẬT**
TÔN LONG PHƯỚC[1*], ĐINH QUANG HUY[1], NGUYỄN THÀNH LUÂN[1]

**YSC5.F018** ..... **209**
**CADIOCARE: XÂY DỰNG HỆ THỐNG HỖ TRỢ CHĂM SÓC SỨC KHỎE THÔNG MINH CHO BỆNH NHÂN TIM MẠCH VỚI SỰ HỖ TRỢ ChatGPT**
TÔN LONG PHƯỚC[1*], VÕ THÀNH NHỚ[1], BÙI SỸ SƠN[1]

**YSC5.F019** ..... **220**
**BUGHOUSE - ỨNG DỤNG QUẢN LÝ HỢP ĐỒNG THÔNG MINH CHO CÁC DỊCH VỤ CHO THUÊ PHÒNG DỰA TRÊN KỸ THUẬT BLOCKCHAIN**
TÔN LONG PHƯỚC[1*], VÕ THỊ TRÀ GIANG[1], ĐOÀN NGỌC QUỐC BẢO[1]

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F020** ..... **232**
**HEALTH MONITORING - HỆ THỐNG CẢNH BÁO CHĂM SÓC SỨC KHỎE DỰA TRÊN NỀN TẢNG IoT**
TÔN LONG PHƯỚC[1*], MAI VĂN TRƯỜNG[1], NGUYỄN HOÀNG DUY[1]

**YSC5.F021** ..... **245**
**HỆ THỐNG HỖ TRỢ HỌC VỤ CHO SINH VIÊN TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH**
TÔN LONG PHƯỚC[1], LÊ VĂN TÙNG[1], PHAN TẤN TÀI[1]

**YSC5.F022** ..... **257**
**TIẾP CẬN HỖ TRỢ CHẨN ĐOÁN UNG THƯ TỪ NHŨ ẢNH SỬ DỤNG PHƯƠNG PHÁP HỌC SÂU**
LƯU LƯƠNG VI NHÂN[1*], QUÁCH XUÂN NAM[1]

**YSC5.F023** ..... **269**
**XÂY DỰNG HỆ THỐNG QUẢN LÝ KHO TIỆN LỢI**
NGUYỄN THỊ HẠNH[1], ĐỖ THỊ TƯỜNG VI[1], HUỲNH TUẤN ANH[1]

**YSC5.F024** ..... **278**
**XÂY DỰNG ỨNG DỤNG HỖ TRỢ TÌM KIẾM THỢ SỬA KHÓA**
NGUYỄN VĂN THẮNG[1*], NGUYỄN MINH VƯƠNG[2], NGUYỄN VĂN TRINH[1]

**YSC5.F025** ..... **286**
**XÂY DỰNG HỆ THỐNG QUẢN LÝ KHO THUỐC THÔNG MINH**
NGUYỄN THỊ HẠNH[1], NGUYỄN HOÀNG SANG[1*], VĂN ANH KHOA[1]

**YSC5.F026** ..... **296**
**TÌM LẠI NỤ CƯỜI XƯA**
HỒ VÕ HOÀNG DUY[1], VÕ QUỐC HUY[1], NGUYỄN XUÂN GIANG[1], NGUYỄN QUANG BẢO[1], DƯƠNG QUANG HUY[1], ĐỖ THU ĐÔNG[1], BÙI THANH HÙNG[1]

**YSC5.F027** ..... **305**
**PTASK: XÂY DỰNG WEBSITE QUẢN LÝ DỰ ÁN TÙY CHỈNH CHO DOANH NGHIỆP**
VÕ MINH PHƯƠNG[1], NGUYỄN ĐỨC HUY[1], NGUYỄN THỊ HẠNH[1*]

**YSC5.F028** ..... **315**
**XÂY DỰNG HỆ THỐNG QUẢN LÝ HỌC PHẦN KHÓA LUẬN TỐT NGHIỆP ÁP DỤNG TẠI TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH**
NGUYỄN THỊ HẠNH[1], NGUYỄN THỊ MINH CHÂU[1*], NGUYỄN THANH SƠN[1], HOÀNG MINH TRÍ[1], NGUYỄN CHI QUỐC ĐANG[1]

**YSC5.F029** ..... **328**
**XÂY DỰNG ỨNG DỤNG VẬN CHUYỂN HÀNG BẰNG XE TẢI**
NGUYỄN THỊ HẠNH, VƯƠNG ÁNH LÂM, NGUYỄN TRUNG QUỐC

**YSC5.F030** ..... **339**
**PHÂN LOẠI BI-RADS SỬ DỤNG PHƯƠNG PHÁP HỌC SÂU**
TRỊNH THỊ BẢO BẢO

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F031** ..... **354**

**GIẢI Ô SỐ SUDOKU KÍCH THƯỚC LỚN VỚI GIẢI THUẬT DI TRUYỀN ĐA DẠNG HÓA QUẦN THỂ KẾT HỢP BACKTRACKING**

TRẦN MẠNH THÌN [1], TRẦN THANH LƯỢNG [1], ĐỖ HÀ PHƯƠNG[1*]

**YSC5.F032** ..... **369**

**ỨNG DỤNG XỬ LÝ SONG SONG NHÂN MA TRẬN LỚN ĐỂ MÃ HÓA CÔNG KHAI TRUYỀN THÔNG ĐIỆP VỚI HỆ MÃ HÓA MCELIECE**

HUỲNH DUY QUANG [1], TRẦN XUÂN THẮNG [1], ĐỖ HÀ PHƯƠNG[1*]

**YSC5.F033** ..... **380**

**IMPROVING THE PROPOSAL AND APPROVAL SYSTEM FOR EMULATION AND REWARD CONTESTS AT IUH**

QUANG-MANH PHAM, TAN-LOI PHAM, PHUOC-LOC NGUYEN [1], HUU-QUANG NGUYEN [2*]

**YSC5.F034** ..... **394**

**ĐÁNH GIÁ KHẢ NĂNG HỎI - ĐÁP CỦA CÁC MÔ HÌNH NGÔN NGỮ CỠ NHỎ TRÊN MIỀN TRI THỨC KHOA HỌC DỮ LIỆU**

NGUYỄN VĂN ANH TUẤN (*), NGUYỄN ĐÌNH THANH, CAO NGUYỄN GIA HƯNG, NGUYỄN VĂN HẬU, DƯƠNG TRẦN KIM NGÂN, TRẦN KỲ, NGUYỄN VĂN HOÀI NAM, TRẦN VĂN QUYẾN, NGUYỄN HỮU TÌNH

**YSC5.F035** ..... **409**

**HỆ THỐNG IOT HỖ TRỢ QUẢN LÝ, CHĂM SÓC VÀ THEO DÕI SỨC KHỎE THÚ CƯNG TỰ ĐỘNG**

TRẦN THỊ MINH KHOA[1,*], MÃ TIỂU LINH[1], TRẦN ANH VŨ[1], ĐẶNG THỊ PHÚC[1]

**YSC5.F036** ..... **425**

**XÂY DỰNG ỨNG DỤNG TRÍCH XUẤT THÔNG TIN TỪ DANH THIẾP SỬ DỤNG KỸ THUẬT HỌC SÂU**

DANG THI PHUC [1](*), LE DINH HIEU[1], LE MINH PHU[1], TRAN THI MINH KHOA[1], LE THI ANH TUYET[1]

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F001**

# ChainForHelp: HIGHLY TRANSPARENT PEER-TO-PEER CHARITY FUNDING SYSTEM BASED ON BLOCKCHAIN TECHNOLOGY

ĐẶNG NGỌC PHONG[1*], NGUYỄN ĐẶNG HOÀNG THI[1], PHAN HẢI TRIỀU[1], GIÁP CHÍ CƯỜNG[1], HUỲNH NAM[1]

*[1]Faculty of Information Technology, Industrial University of Ho Chi Minh City*

*[1*]phongdang.contact@gmail.com, thinguyen280601@gmail.com, trieuphan3701@gmail.com, giapchicuong@gmail.com,huynhnam@iuh.edu.vn;*

**Abstract.** Every year, many individuals and organizations have built and organized charity funds, community contribution activities, and scholarship programs that take place at the Industrial University of Ho Chi Minh City. However, the university currently does not have a platform to post and aggregate information about these charity calls. Therefore, it is essential to build a platform to help call for and manage charity information. ChainForHelp is an application that we built based on blockchain technology to help individuals and organizations make donations, create, and post public information about charitable activities with high transparency and integrity. ChainForHelp consists of two parts: a website application that displays the user interface and a smart contract application that handles requests from users as well as processes data about generated charity activities and stores them on the blockchain network. ChainForHelp is built on technology platforms such as JavaScript, Solidity, ReactJS, and Thirdweb. The application will be deployed to the student community and related organizations at the university to verify its relevance and effectiveness. In the future, we will expand the application to many universities and upgrade more functions to help users use the application more effectively.

**Keywords.** Charity, Transparency, Smart contract, Blockchain, Peer-to-peer network.

## ChainForHelp: HỆ THỐNG GÂY QUỸ TỪ THIỆN NGANG HÀNG VỚI TÍNH MINH BẠCH CAO DỰA TRÊN CÔNG NGHỆ BLOCKCHAIN

**Tóm tắt.** Hằng năm, có rất nhiều cá nhân, tổ chức đã xây dựng và tổ chức các quỹ từ thiện, các hoạt động đóng góp cộng đồng, các chương trình kêu gọi học bổng diễn ra tại Trường Đại học Công nghiệp Thành phố Hồ Chí Minh. Tuy nhiên, trường hiện vẫn chưa có nền tảng nào giúp đăng tải và tổng hợp thông tin về các cuộc kêu gọi từ thiện này. Do đó, việc xây dựng một nền tảng giúp kêu gọi và quản lý thông tin từ thiện là điều cần thiết. ChainForHelp là một ứng dụng mà chúng tôi xây dựng dựa trên công nghệ blockchain nhằm giúp các cá nhân, tổ chức có thể thực hiện việc quyên góp, tạo và đăng tải thông tin về các hoạt động từ thiện với tính minh bạch và toàn vẹn cao. ChainForHelp gồm hai phần: ứng dụng website hiển thị giao diện phía người dùng và ứng dụng hợp đồng thông minh nhằm xử lý các yêu cầu từ người dùng cũng như xử lý dữ liệu của các hoạt động từ thiện được tạo và lưu trữ chúng trên mạng lưới blockchain. ChainForHelp được xây dựng dựa trên các nền tảng công nghệ như: JavaScript, Solidity, ReactJS và Thirdweb. Ứng dụng sẽ được triển khai cho cộng đồng sinh viên và các tổ chức liên quan trong trường nhằm kiểm chứng tính phù hợp và mức độ hiệu quả. Trong tương lai, chúng tôi sẽ mở rộng ứng dụng sang nhiều trường đại học và nâng cấp thêm nhiều chức năng hỗ trợ người dùng sử dụng ứng dụng hiệu quả hơn.

**Từ khóa.** Từ thiện, Minh bạch, Hợp đồng thông minh, Blockchain, Mạng ngang hàng.

## 1 INTRODUCTION

Every year, hundreds of charity campaigns related to finance take place at the Industrial University of Ho Chi Minh City. With the development of technology, online payment methods are increasingly being

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

used[1]. However, there are still many limitations and difficulties in carrying out online charity activities, mainly in transparency and financial management. With the goal of making online charity activities more transparent and effective, and applying Blockchain technology - a technology widely used in currency-related fields[2] - we decided to research and study the platform for charity calls using Blockchain technology at the Industrial University of Ho Chi Minh City. The aim of the project is to build a platform for charity calls using Blockchain technology, making fundraising and sponsorship distribution more transparent, reliable, and effective. The platform is the interaction between smart contracts, the features of the website, and the ETH blockchain to make fundraising and sponsorship distribution easier, convenient, and transparent.

To achieve the research goal, the project will be conducted using the product development research method, combined with experimentation and analysis. Specifically, the project will include the following stages:

Stage 1. Researching Blockchain technology and its applications in the field of charity

Stage 2. Research and learn how to use related technologies to develop web3 applications

Stage 3. Design and build the application

Stage 4. Evaluate the results and provide future development directions

## 2 THEORETICAL BASIS

### 2.1 Blockchain Technology and Its Application in the Field of Charity

Blockchain technology is a decentralized database that stores information in blocks of information that are linked and expand over time. This technology can enhance transparency, security, and efficiency in transactions and data management[4].

Blockchain technology has many applications in different fields, including the charity sector. Some of the potential uses of blockchain technology in the charity sector are:

Create a transparent and trusted fundraising platform that allows donors to track the destination of their donations and the results of their charitable activities.

Create an efficient and fair donation distribution system, helping recipients receive quick and complete support, avoiding corruption and waste.

Creating a connected and collaborative philanthropic community, enabling charities to share information, experiences, and resources with each other, strengthens the power of philanthropy[3].

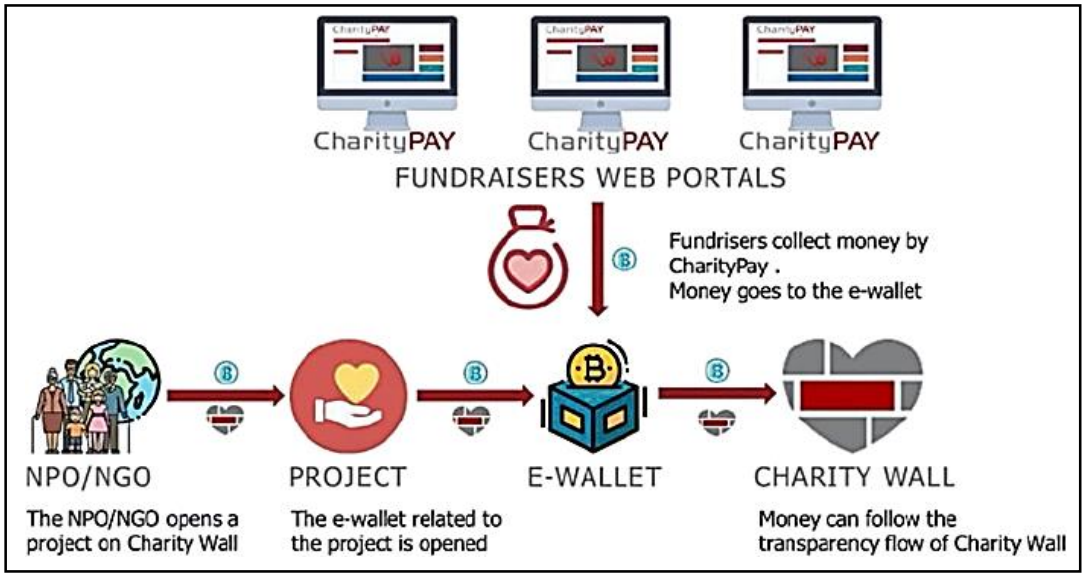



Figure 1: Managing Charity 4.0 with Blockchain[3]

### 2.2 Smart Contract

A smart contract is an application or program that runs on the blockchain, a decentralized database that stores information in linked blocks of information. Smart contracts can automatically execute terms or agreements between parties (computer systems) thanks to blockchain technology. These permissions are

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

predefined by computer code, and all nodes in the network must copy and enforce those rules.
Smart contracts have many advantages, such as[5][6]:

- Automation: Everything is automated, so there are no delays or mistakes.
- High transparency: The regulations are clearly written and unedited, so interference behind the scenes is impossible.
- Easy to control: Just open it up to see it again. Documents will be encrypted on a shared ledger, so no loss will occur.
- Security Guarantee: Blockchain always ensures the safety of the documents in the contract.
- Accuracy: Common errors on paper will not appear on smart contracts.




Figure 2: The Awesomeness of Smart Contracts

## 2.3 Related Technologies

### 2.3.1 ReactJS

ReactJS[7] is a JavaScript library used to build user interfaces for single-page web applications. ReactJS allows for creating reusable and dynamically updating interface components according to data. ReactJS also supports features like event handling, state management and combines with other technologies like HTML, CSS, GraphQL, Redux, etc.
To define a React component class, you need to extend React.Component and override the render() method. The render() method will return a JSX snippet that describes the component's interface. You can also use other lifecycle methods to run the code at specific times in the process. Each component also has props and states to manage the input data and the internal state of the component.
ReactJS is one of the most popular JavaScript libraries available today. It is used by many big companies like Facebook, Instagram, Netflix, Airbnb, etc.

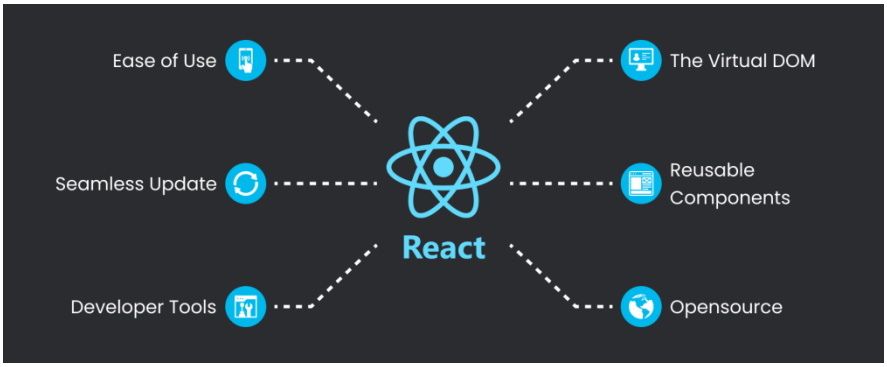



Figure 3: ReactJS Advantages

### 2.3.2 Solidity

Solidity[9] is a programming language that is being widely used in the development of blockchain applications. With properties such as simplicity, ease of use, and safety and security features. Solidity becomes one of the top choices for developing blockchain applications.

The properties of Solidity include high interpretability, staticness of data types, function support, inheritance, polymorphism, and abstraction. Solidity also supports features such as account and transaction management, access decentralization, and smart contracts.

Solidity's current applications include the development of blockchain applications such as Ethereum, those based on smart contracts, and financial applications. With these applications, Solidity makes it possible for programmers to create smart contracts and blockchain-based applications quickly and easily[8].

Technologies related to Solidity include blockchain platforms such as Ethereum, EOS, Tezos and technologies related to blockchain application development such as Web3.js, Truffle, and Ganache. These technologies provide users with the necessary tools to develop and deploy blockchain applications using Solidity.

Going forward, Solidity is expected to continue to grow and improve to meet the growing needs of the blockchain market. Many developers are using Solidity to create new applications and the motivation to improve Solidity is still huge. With the rise of blockchain applications and the development of related technologies, Solidity could become one of the most important programming languages for the development of blockchain applications in the future[10].




Figure 4: The Characteristic of Solidity Programming

### 2.3.3 Thirdweb

The ThirdWeb framework is an open-source framework developed on the Ethereum platform, intended to make it easy for developers to build decentralized applications (dApps). ThirdWeb was developed by a team of developers at HashEx, a company specializing in the development of Blockchain applications.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

ThirdWeb Framework has the special property that it supports popular programming languages such as Solidity, Vyper, Javascript, Python, etc. This gives developers flexibility in choosing programming languages and increases the applicability of ThirdWeb.
The ThirdWeb Framework has many practical applications today, such as building decentralized finance applications, intellectual property management systems, applications in the healthcare industry, etc. ThirdWeb is also compatible with various protocols on the Ethereum platform, such as ERC-20, ERC-721, ERC-1155, etc.
Technologies related to the Thirdweb framework include blockchain-related technologies and Ethereum-related protocols. ThirdWeb also uses many new technologies such as IPFS, Web3.js, Ethers.js, etc., to enhance the stability and speed of the application.
The future of Thirdweb framework is very promising, because decentralized applications are becoming a trend that many people are interested in. Thirdweb is increasingly being chosen by developers and businesses to build their decentralized applications. The features and stability of the Thirdweb framework are continuously improved, promising to bring better solutions to developers and users in the future.

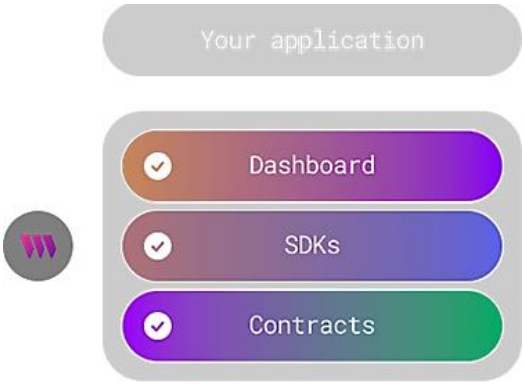



Figure 5: Thirdweb Functions

### 2.3.4 Metamask

Metamask is a cryptocurrency wallet software and web browser based on the Ethereum blockchain. Launched in 2016, Metamask allows users to store, manage, and trade tokens and cryptocurrencies based on the Ethereum blockchain on their web browser.
The essence of Metamask is its ability to integrate with other services on the web, such as Ethereum-based dApps (decentralized applications), allowing users to access these services conveniently and securely. Metamask also allows for the creation and management of diverse wallet accounts, allowing users to store various cryptocurrencies at once.
Metamask's applications are diverse, from conducting cryptocurrency and token transactions, to accessing dApp services such as NFT marketplaces, games, and complex financial applications such as lending and borrowing. In addition, Metamask offers security features such as two-factor authentication and limiting gas fees in transactions to reduce costs for users.
Metamask is built on blockchain and open-source technologies, allowing users to easily learn and change its functions. Metamask also integrates the latest technologies, such as EIP-1559, and is continuing to develop new features to improve the user experience and enhance security.
As for the future of Metamask, it seems that it will continue to evolve and enhance its features to meet the growing needs of its users. In addition, with the rapid development of DeFi and the growth of blockchain applications, Metamask will also play an important role in bringing blockchain technology closer to end users[11].
Here's how Metamask's passwordless cryptographically secure login flow works:
**Step 1:** Modify the user model (Back-end)
The user model needs two new required fields: publicAddress and nonce. With publicAddress being unique. You can keep the usual username, email, and password fields—especially if you want to implement your MetaMask login alongside your email/password login—but they are optional.

**Step 2:** Generate Nonces (Back-end)
For each user in the database, generate a random string for the nonce field.
**Step 3:** User Fetches Their Nonce (Front-end)
The user enters the publicAddress in the login form, when they click the login button, the application will call the api to retrieve the nonce associated with their publicAddress. If the previous request did not return any results, it means that the current public address is not registered.
**Step 4:** User Signs the Nonce (Front-end)
After the UI receives the nonce in the response of the previous API call. MetaMask shows a confirmation popup to sign the message. The nonce will be displayed in this pop-up to let the user know that they are not signing some malicious data. When the user accepts it, the callback function will be called with the signed message (called the signature) as an argument. The front end then makes another API call to POST/api/authentication, passing the body with both the signature and the public address.
**Step 5:** Signature Verification (Back-end)
When the back-end receives a POST/api/authentication request, the back-end first fetches the user in the database corresponding to the public address provided in the request body. Specifically, it fetches the associated nonce.
Having the nonce, publicAddress and signature, the back end can then cryptographically verify that the nonce was signed correctly by the user. If this is the case, then the user has proven ownership of the public address and we can consider the person authenticated. Then the JWT or session identifier can be returned to the UI.
**Step 6:** Change the Nonce (Back-end)
To prevent users from logging in again with the same signature (in case the signature is compromised), Metamask ensures that the next time the same user wants to log in, they need to sign a new nonce. This is achieved by generating another random nonce for this user and saving it in the database.

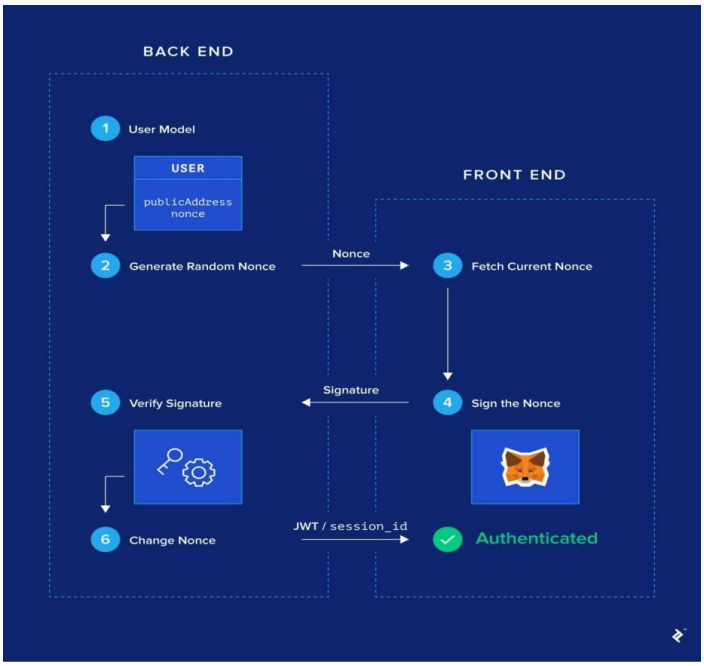



Figure 6: Overview of The Six Steps of the MetaMask Login Flow[17]

### 2.3.5 Firebase Storage

Firebase Storage[12] is a service built for the purpose of storing and managing user-generated content such as images, videos, or file data.
Some strong points such as:

- Robust: Firebase Storage performs uploads and downloads regardless of network connection quality and can restart when paused, saving time and bandwidth.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

- Secure: Built-in Firebase Authentication for security, so it's easy to manage access to files.
- Scalable: Firebase Storage is built on top of Google Cloud Platform, so the scalability can be up to Petabytes of data.

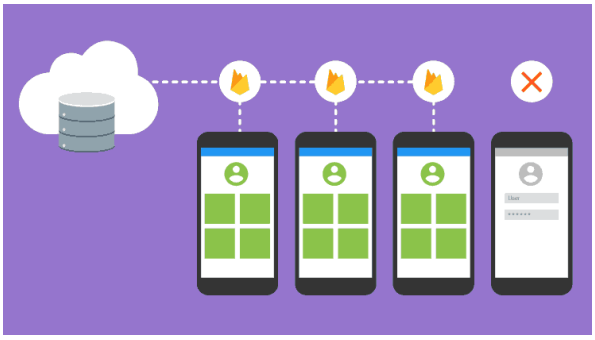


Figure 7: Store and serve content with ease on firebase storage[12]

# 3 DESIGN AND BUILD CHAINFORHELP APPLICATION

## 3.1 Software Architecture

This section will describe in detail the interaction between the front-end and the smart contract(back-end) of the system using Metamask wallet[12][13].

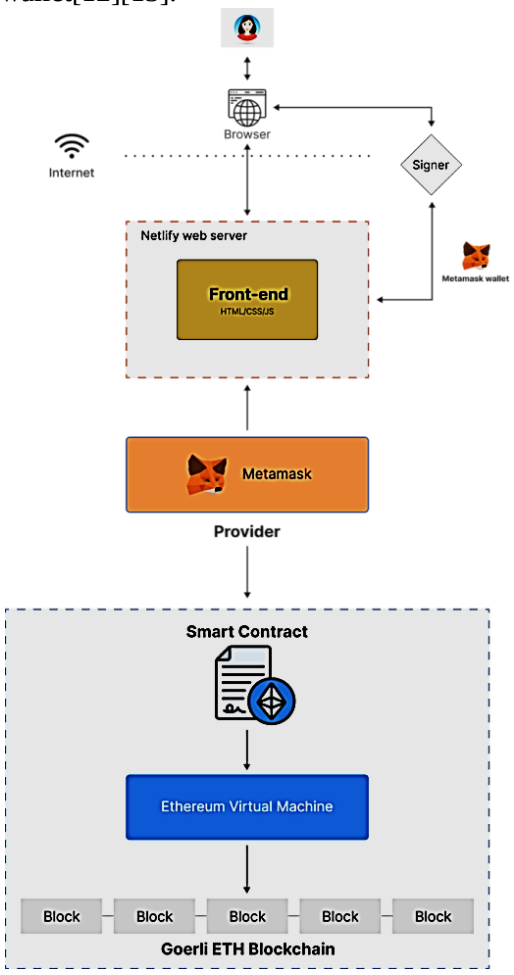



Figure 8: ChainForHelp Architecture

Front-end: It is a web application deployed on Netlify server. Provide a user interface that allows users to interact with the application easily. The interface is intuitively built with search and filter features to increase the comfort of the user experience. Going into the client section, in order to fully interact with the ChainForHelp application, users first need to have a Metamask wallet account and have a balance inside it. The condition of having a balance in the account will not be required when you just want to "roam around" in this application. However, when you want to create a project to raise capital or support other projects here, you are required to have a balance in your account.

❖ Smart Contract:

- The Smart Contract is used as a back-end server to process user requests and transfer the generated campaign data to the Ethereum Virtual Machine (EVM). Then, the EVM will perform the processing to store the data on the Goerli ETH blockchain (a test network blockchain of ETH). Some data is stored as:
  - Owner: The wallet address of the user who creates the charity project.
  - Title: The title of the charity project.
  - Description: A detailed description of the project to convince potential supporters to contribute.
  - Target: The donation goal of the project.
  - Deadline: The deadline for the charity campaign.
  - Amount collected: The total amount of donates collected during the campaign.
  - Images: Images attached to the project for display purposes.
  - Donators: A list of wallet addresses of individuals who have contributed to the project.
  - Donations: The amount contributed by each individual.
  - Category: Type of campaign
- The functions of the smart contract include:
  - createCampaign: Creates a new charity campaign.
  - donateToCampaign: Enables users to contribute to an existing charity campaign.
  - getDonators: Returns the list of wallet addresses of individuals who have contributed to a specific campaign.
  - getCampaigns: Returns a list of all existing charity campaigns.
  - filterCampaign: Returns an alphabetical list of campaigns.

After writing the smart contract, ThirdWeb, a third-party platform, is used to deploy the contract to a decentralized network. Once the deployment process is complete, all activities after the client-side are processed entirely on the blockchain network, and ThirdWeb is no longer involved.

## 3.2 Technologies used for ChainForHelp

In the ChainForHelp application[14], ReactJS are used to design the user interface along with technologies such as TailwindCSS and ThirdWeb to create a user-friendly interface.

ChainForHelp's Smart Contract is written by Solidity programming language. Thirdweb is powered by a Smart Contract written by Solidity that can be run on the internet easily with the command npm run deploy. In addition, it also provides packages to easily interact with ReactJS technology related interface.

Next, in order for users to be able to donate to charity, they will need to use Metamask wallet, when users start trading smart contracts on ChainForHelp, Metamask will automatically prompt them confirm transaction details, including gas fee and amount of cryptocurrency sent. After the user confirms the transaction, Metamask digitally signs it and broadcasts it to the Ethereum network. Additionally, Metamask also gives users a high degree of control over their transactions, allowing them to regulate gas prices. and limits to prioritize transaction speed or reduce fees, and choose from a variety of Ethereum networks, including mainnet, test network, and custom network. Finally, Google Cloud Storage is used to store images of campaigns created on the ChainForHelp application.

## 3.3 Results

In this study, we completed the ChainForHelp application with the following functionalities:

- ❖ Login: by connecting their Metamask wallet account to the application.
- ❖ Create a fundraising project: the application will provide a form and require users to fill in the required information. As this is a Blockchain and SmartContract-based application, there will be a

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

"barrier" of trust from others towards your fundraising project, so it is recommended to write your story as honestly as possible to achieve a successful fundraising goal.

- ❖ Support other fundraising projects: users will use their wallet balance to support other ongoing fundraising projects, provided that the deadline for the project has not passed.
- ❖ Use user support tools: there are two user support tools in this application: the Search bar and the Filter tool. The Search bar will help users search for the names of existing projects on the application. The Filter tool, this feature will help users narrow their search range through various filtering conditions. These two tools will increase search speed and user-friendliness.

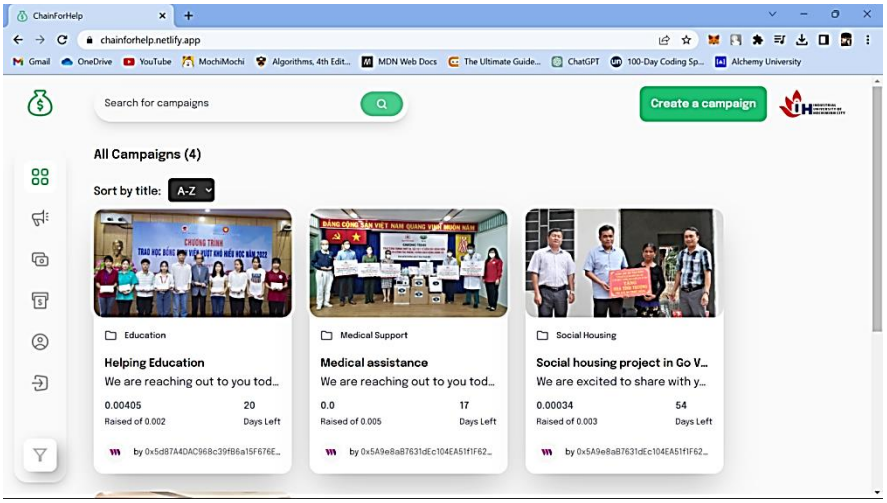



Figure 9: ChainForHelp Homepage

## 4 CONCLUSION AND OUR VISION

After analyzing and assessing the societal situation, specifically at the Industrial University of Ho Chi Minh City, we have developed a solution to address the aforementioned challenges through the ChainForHelp platform. This article presents the two main components of the ChainForHelp system.

Firstly, we have built a user-friendly website application that facilitates seamless and transparent transactions between benefactors and charity organizations.

Secondly, we utilized the Solidity programming language and the Thirdweb Framework to construct smart contracts. These contracts are responsible for processing the campaign data generated on the website and storing it securely on the Ethereum blockchain network.

Currently, the application fulfills our desired features, but due to budget constraints, the smart contract has only been deployed on the ETH test network. It serves to verify the application's features among a group of users. In the upcoming version, we plan to deploy it to the ETH main network to gather further feedback and evaluate the application's overall performance.

Today, digital currency is still quite new to many people, and to be able to conduct cryptocurrency transactions, they need to install a lot of things that make users afraid when they intend to use the application. Our vision is to extend the application to other organizations and universities, assisting them in achieving transparency in charitable activities. Additionally, it is also possible to disseminate more widely to Vietnamese people about blockchain technology - a technology that will be the foundation of many applications in the near future.

## REFERENCES

[1] J. Wu, K. Lin, D. Lin, Z. Zheng, H. Huang, and Z. Zheng, "Financial Crimes in Web3-Empowered Metaverse: Taxonomy, Countermeasures, and Opportunities," IEEE Open Journal of the Computer Society, vol. 4, 2023.

[2] Chen, C., Zhang, L., Li, Y., Liao, T., Zhao, S., Zheng, Z., Huang, H., & Wu, J. (2022). When Digital Economy Meets Web3.0: Applications and Challenges. IEEE Open Journal of the Computer Society, 3, 225-235.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

[3] A. Rangone and L. Busolli, "Managing charity 4.0 with Blockchain: a case study at the time of Covid-19," International Review on Public and Nonprofit Marketing, vol. 18, pp. 491-521, 2021.

[4] "Công nghệ Blockchain và ứng dụng trong thị trường chuỗi cung ứng". [Online]. Available: https://vjcc.org.vn/tri-thuc/cong-nghe-blockchain-va-ung-dung-trong-thi-truong-chuoi-cung-ung.html. [Accessed: 02/03/2023].

[5] "What Are Smart Contracts? - Binance Academy". [Online]. Available: https://academy.binance.com/en/articles/what-are-smart-contracts. [Accessed: 03/03/2023].

[6] "Hợp đồng thông minh là gì? - Đánh giá - Tin tức FPT Shop". [Online]. Available: https://fptshop.com.vn/tin-tuc/danh-gia/hop-dong-thong-minh-la-gi-138549. [Accessed: 03/03/2023].

[7] "React – A JavaScript library for building user interfaces". [Online]. Available: https://reactjs.org/. [Accessed: 05/03/2023].

[8] "Solidity Tutorial". [Online]. Available: https://101blockchains.com/solidity-tutorial/. [Accessed: 08/03/2023].

[9] "What is Solidity Programming: Data Types, Smart Contracts, and EVM?". [Online]. Available: https://www.simplilearn.com/tutorials/blockchain-tutorial/what-is-solidity-programming. [Accessed: 08/03/2023].

[10] "A Beginners Guide to Web3". [Online]. Available: https://surajondev.com/2022/05/17/a-beginners-guide-to-web3/. [Accessed: 08/03/2023].

[11] "Metamask - What is it, and How Does it Work?". [Online]. Available: https://originstamp.com/blog/metamask-what-is-it-and-how-does-it-work/. [Accessed: 09/03/2023].

[12] "Firebase Storage - Cloud Storage for Firebase". [Online]. Available: https://firebase.google.com/products/storage. [Accessed: 01/04/2023].

[13] R. Qin, W. Ding, J. Li, S. Guan, G. Wang, Y. Ren and Z. Qu, "Web3-Based Decentralized Autonomous Organizations and Operations: Architectures, Models, and Mechanisms," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 53, no. 4, pp. 1-14, 2023.

[14] "Build and Deploy a Web3 Crowdfunding Platform (Kickstarter) As Your First Blockchain Application". [Online]. Available: https://github.com/adrianhajdin/project_crowdfunding. [Accessed: 20/01/2023].

[15] "The Architecture of a Web 3.0 application". [Online]. Available: https://www.preethikasireddy.com/post/the-architecture-of-a-web-3-0-application. [Accessed: 15/03/2023].

[16] "Top Web3 Architecture Layers Explained: Frontend, Backend, and Data". [Online]. Available: https://itnext.io/top-3-web-3-0-architecture-layers-explained-frontend-backend-and-data-e10200f7fc76. [Accessed: 15/03/2023].

[17] "One-click Login With Blockchain: A MetaMask Tutorial". [Online]. Available: https://www.toptal.com/ethereum/one-click-login-flows-a-metamask-tutorial/. [Accessed: 09/03/2023].

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.002**

# QUY HOẠCH TUYẾN TÍNH NGUYÊN VÀ ỨNG DỤNG TRONG THỰC TẾ

TRỊNH NGUYỄN HOÀNG VŨ[1*], PHẠM VIẾT DŨNG MINH[1], TRẦN THÁI HÀ[1], TRỊNH DƯƠNG HOAN[1], TRẦN BÌNH MINH[1], LÊ PHÚC LỮ[1], NGUYỄN HỮU TÌNH[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
**trinhnguyenhoangvu306@gmail.com*

**Tóm tắt.** Quy hoạch tuyến tính (QHTT) thuộc lĩnh vực của bài toán tối ưu, chuyên khảo sát các vấn đề có ràng buộc và hàm mục tiêu ở dạng tuyến tính. QHTT có thể được sử dụng để mô hình hóa và giải quyết nhiều vấn đề tối ưu trong các lĩnh vực, chẳng hạn như tài chính, sản xuất, vận tải, lý thuyết trò chơi. Trên thực tế, nhiều bài toán có dữ liệu đầu vào, kết quả đầu ra đều ở dạng số nguyên và đòi hỏi phải xử lý khó khăn hơn so với số thực, gọi là Quy hoạch nguyên. Ở chủ đề nghiên cứu này, chúng tôi sẽ tìm hiểu về phương pháp nhánh & cận và thuật toán lát cắt Gomory trong Quy hoạch nguyên, ứng dụng vào bài toán xếp lịch, bài toán pha cắt vật tư, bài toán người giao hàng. Ngoài ra, nhóm cũng xem xét thêm về bài toán Quy hoạch tuyến tính đa mục tiêu.

**Từ khóa.** Bài toán tối ưu, QHTT nguyên, Kỹ thuật nhánh & cận, Lát cắt Gomory, Bài toán xếp lịch, TSP.

## INTEGER LINEAR PROGRAMMING AND ITS REAL APPLICATIONS

**Abstract.** Linear programming (LP) belongs to the domain of optimization problems, specializing in investigating problems with constraints and objective functions in linear form. LP can be used to model and solve many optimization problems in fields such as finance, manufacturing, transportation, game theory. In reality, many problems have inputs, outputs in the form of integers and require more difficult processing than real numbers, which is called Integer Programming (LIP). In this research, we will learn about the branch & bound method and the Gomory's cut algorithm in LIP, and then apply to the scheduling problem, the material cutting problem, and the traveling salesman problem TSP. In addition, we also considers more about the Multi-objective Linear Programming problems.

**Keywords.** Optimization problem, Integer linear programming, Branch & bound method, Gomory's cut algorithm, Scheduling problem, TSP.

## 1 ĐẶT VẤN ĐỀ

Tối ưu hóa là một vấn đề quan trọng trong cuộc sống. Chúng ta thường tìm cách tối ưu hóa thời gian, tiền bạc và năng lượng để đạt được mục tiêu của mình một cách hiệu quả nhất. Trong Toán học và Khoa học máy tính, Quy hoạch tuyến tính là một phương pháp tối ưu được sử dụng rộng rãi để giải quyết các bài toán liên quan đến việc phân bổ tài nguyên, người giao hàng,... một cách hiệu quả nhất. Quy hoạch tuyến tính có thể giúp chúng ta tìm ra giải pháp tối ưu cho các vấn đề trong cuộc sống thực tế.

Ví dụ, một công ty có thể có nguồn lực hạn chế như nguyên liệu, lao động và thời gian. Quy hoạch tuyến tính có thể được sử dụng để xác định kế hoạch sản xuất tối ưu nhằm tối đa hóa lợi nhuận trong khi vẫn nằm trong giới hạn của các nguồn lực sẵn có. Tương tự, một dịch vụ giao hàng có thể sử dụng Quy hoạch tuyến tính để xác định tuyến đường ngắn nhất nhằm tiết kiệm thời gian và nhiên liệu. Các tổ chức tài chính sử dụng Quy hoạch tuyến tính để xác định danh mục các sản phẩm tài chính có thể cung cấp cho khách hàng. Quy hoạch tuyến tính cũng có thể áp dụng vào cuộc sống cá nhân. Ví dụ, một cá nhân có thể có thời gian và tiền bạc hạn chế cho các hoạt động giải trí. Quy hoạch tuyến tính có thể được sử dụng để phân bổ thời gian và tiền bạc tối ưu nhằm tối đa hóa lợi nhuận trong giới hạn ngân sách. Tuy nhiên, vấn đề được phát

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

sinh trong các vấn đề tối ưu là về tính nguyên của các biến. Ví dụ, nếu bạn giải quyết một bài toán liên quan đến việc sản xuất sản phẩm, số lượng sản phẩm sản xuất phải là một số nguyên.
Quy hoạch tuyến tính nguyên là một dạng đặc biệt của Quy hoạch tuyến tính, đòi hỏi xử lý khó khăn hơn do việc giới hạn các giải pháp chỉ ở các số nguyên có thể khiến cho không gian tìm kiếm trở nên rất lớn và phức tạp. Chính vì vấn đề này, nhóm của chúng tôi đã nghiên cứu về các giải pháp để giải quyết vấn đề trên bằng phương pháp nhánh & cận, cắt Gomory, đơn hình. Vận dụng giải quyết các bài toán xếp lịch, bài toán pha cắt vật tư, bài toán người giao hàng và các bài tập phổ biến khác.

## 2 QUY HOẠCH TUYẾN TÍNH NGUYÊN

### 2.1 Thế nào là Quy hoạch tuyến tính?

Quy hoạch tuyến tính (Linear Programming - LP) là một thuật toán nhằm tìm ra một kế hoạch hay phương án tốt nhất từ vô số những lựa chọn khác nhau. Đây là bài toán tối ưu, trong đó hàm mục tiêu (objective function) và các điều kiện ràng buộc đều là tuyến tính. Phương án tối ưu là phương án thỏa mãn được các mục tiêu đề ra, phụ thuộc vào các hạn chế và ràng buộc. Quy hoạch tuyến tính giúp giải quyết nhiều vấn đề rất thực tiễn trong cuộc sống như: lập kế hoạch, xếp lịch,... Để giải quyết bài toán, ta sẽ mô tả các ràng buộc dưới dạng các biến số thực không âm (đôi khi là số nguyên) rồi sử dụng các thuật toán trong LP để tìm phương pháp tốt nhất. Trong phần tiếp theo, ta sẽ xem xét một số bài toán tối ưu trong thực tế và cách tối ưu của chúng bằng việc lập mô hình quy hoạch tuyến tính.
Với các biến số thực $x_1,\ x_2, \dots ,\ x_n$ trong bài toán, ta lập ra hàm nhiều biến $f(x_1, x_2, \dots, x_n)$ có dạng $c_1x_1 + c_2x_2 + \cdots + c_nx_n$ trả về kết quả là số thực (để đơn giản, ta ký hiệu là có dạng $f$). Ta cần tìm giá trị lớn nhất hoặc nhỏ nhất của hàm này, tương ứng là $f \to max$ hoặc $f \to min$.

### 2.2 Giới thiệu các bài toán phổ biến trong thực tế

**Dạng bài phân bổ tài nguyên**: Để đảm bảo được dinh dưỡng, một người trung bình cần bổ sung mỗi ngày:

- Không ít hơn 200 gram chất bột đường (carbohydrate).
- Không ít hơn 20 gram chất đạm (protein).

Hàm lượng dinh dưỡng các chất này trong 4 loại lương thực hiện có trong một cửa hàng được liệt kê ở bảng:

| Loại thức ăn | Bột đường (gram) | Đạm (gram) | Giá (nghìn đồng) |
|---|---|---|---|
| **Cà rốt** | 300 | 30 | 50 |
| **Gạo** | 400 | 10 | 20 |
| **Khoai tây** | 400 | 20 | 10 |
| **Bột mì** | 500 | 50 | 30 |

Như vậy người ấy cần tối thiểu là bao nhiêu tiền để đảm bảo được dinh dưỡng cho bữa ăn?
*Phân tích.* Bài toán dinh dưỡng được mô hình dưới dạng bài toán quy hoạch tuyến tính như sau:
Gọi các biến thực $x_1,\ x_2,\ x_3,\ x_4$ là số gram cà rốt, gạo, khoai tây và bột mì cần mua. Ta có hàm mục tiêu là tổng số tiền:

$$f = 50x_1 + 20x_2 + 10x_3 + 30x_4$$

Các ràng buộc bao gồm:

- Đảm bảo về hàm lượng bột đường: $300x_1 + 400x_2 + 400x_3 + 500x_4 \geq 200$
- Đảm bảo về hàm lượng đạm: $30x_1 + 10x_2 + 20x_3 + 50x_4 \geq 20$
- Ngoài ra còn có ràng buộc không âm cho các biến: $x_1, x_2, x_3, x_4 \geq 0$

Như vậy, bài toán quy hoạch tuyến tính cho dinh dưỡng sẽ là:

$$f = 50x_1 + 20x_2 + 10x_3 + 30x_4 \to min$$
$$300x_1 + 400x_2 + 400x_3 + 500x_4 \geq 200$$
$$300x_1 + 400x_2 + 400x_3 + 500x_4 \geq 200$$
$$x_1, x_2, x_3, x_4 \geq 0, x_1, x_2, x_3, x_4 \in \boldsymbol{R}$$

**Dạng bài sản xuất:** Một nhà máy thực phẩm sản xuất ra 4 loại sản phẩm với chi phí sản xuất là:

- Đũa tre: 5 triệu/tấn.
- Tăm tre: 6 triệu/tấn.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

- Chân hương: 4.5 triệu/tấn.
- Xiên thịt: 6 triệu/tấn

Biết rằng để sản xuất được 1 tấn của lần lượt các loại sản phẩm trên thì nhà máy cần tiêu tốn nguyên liệu và nhân công như sau:

| | **Đũa tre** | **Tăm tre** | **Chân hương** | **Xiên thịt** |
|---|---|---|---|---|
| **Tre (tấn)** | 1.1 | 1.5 | 1.4 | 1.2 |
| **Hương liệu (tấn)** | 0.03 | 0.06 | 0.04 | 0.04 |
| **Than (tấn)** | 1 | 2 | 2.2 | 2.1 |
| **Số giờ làm việc** | 200 | 300 | 220 | 240 |
| **Giá bán ra (triệu/tấn)** | 23 | 36 | 26.5 | 30 |

Kế hoạch trong tháng tới nhà máy có 1000 giờ làm việc và tối đa 5 tấn tre, 0.2 tấn hương liệu, 10 tấn than, nhà máy có vốn là 25 triệu đồng. Hỏi nhà máy cần lập kế hoạch sản xuất như thế nào để tối đa lợi nhuận?

*Phân tích.* Giả định sản lượng sản xuất của mỗi loại lần lượt là các biến thực $x_1, x_2, x_3, x_4$. Hàm mục tiêu:

$f = (23-5)x_1 + (36-6)x_2 + (26.5-4.5)x_3 + (30-6)x_4 = 18x_1 + 30x_2 + 22x_3 + 24x_4$

- Ràng buộc về nguyên liệu:
  - Đối với Tăm tre: $1.1x_1 + 1.5x_2 + 1.4x_3 + 1.2x_4 \leq 5$
  - Đối với Hương liệu: $0.03x_1 + 0.06x_2 + 0.04x_3 + 0.04x_4 \leq 0.2$
  - Đối với Than: $x_1 + 2x_2 + 2.2x_3 + 2.1x_4 \leq 10$
- Ràng buộc về số giờ làm việc: $200x_1 + 300x_2 + 220x_3 + 240x_4 \leq 1000$
- Ràng buộc về chi phí sản xuất: $5x_1 + 6x_2 + 4.5x_3 + 6x_4 \leq 25$
- Ràng buộc về số lượng: $x_1, x_2, x_3, x_4 \geq 0$

Như vậy, ta có bài toán:

$$f = 18x_1 + 30x_2 + 22x_3 + 24x_4 \rightarrow Min$$
$$1.1x_1 + 1.5x_2 + 1.4x_3 + 1.2x_4 \leq 5$$
$$0.03x_1 + 0.06x_2 + 0.04x_3 + 0.04x_4 \leq 0.2$$
$$x_1 + 2x_2 + 2.2x_3 + 2.1x_4 \leq 10$$
$$200x_1 + 300x_2 + 220x_3 + 240x_4 \leq 1000$$
$$5x_1 + 6x_2 + 4.5x_3 + 6x_4 \leq 25$$
$$x_1, x_2, x_3, x_4 \geq 0,\ x_1, x_2, x_3, x_4 \in \boldsymbol{R}$$

## 2.3 Thế nào là quy hoạch tuyến tính nguyên?

Quy hoạch tuyến tính nguyên (Integer Linear Programming - ILP) là một phương pháp trong Quy hoạch tuyến tính, trong đó các biến quyết định được giới hạn trong tập hợp các số nguyên thay vì các số thực. Với ILP, chúng ta cần tìm giá trị của các biến nguyên sao cho hàm mục tiêu tuyến tính đạt giá trị lớn nhất hoặc nhỏ nhất và các ràng buộc tuyến tính được thỏa mãn. Các vấn đề quy hoạch tuyến tính nguyên thường được sử dụng để giải quyết các bài toán tối ưu về lựa chọn và phân bổ tài nguyên có số lượng hữu hạn. Ví dụ, trong một bài toán vận tải, số lượng hàng hóa cần vận chuyển từ một kho đến một điểm đến có thể được biểu diễn bằng các số nguyên. Khi giải quyết bài toán này bằng phương pháp quy hoạch tuyến tính thông thường, các biến quyết định có thể có giá trị là số thực, nhưng khi sử dụng phương pháp ILP thì các biến quyết định sẽ được giới hạn phải là các số nguyên.

ILP được sử dụng rộng rãi trong nhiều lĩnh vực, bao gồm kinh tế học, sản xuất, quản lý chuỗi cung ứng, và trong lập kế hoạch sản xuất và phân bổ tài nguyên. Phương pháp này cho phép giải quyết những bài toán tối ưu hóa phức tạp, đòi hỏi phải xác định các ràng buộc nguyên và các giới hạn về các biến quyết định. Việc sử dụng phương pháp ILP giúp tối ưu hóa kết quả và tăng tính hiệu quả trong việc quản lý và điều phối các tài nguyên. Các dạng Quy hoạch nguyên phổ biến:

- Quy hoạch nhị phân: các biến nhận giá trị 0, 1 (liên quan đến việc chọn hay không chọn).
- Quy hoạch nguyên không âm: các biến liên quan đến bài toán thực tế thường phải không âm.
- Quy hoạch hỗn hợp: trong bài toán có thể có biến nhận giá trị nguyên, có biến là số thực.

## 2.4 Các ví dụ cho bài toán quy hoạch nguyên

**Bài toán phân công nhân sự:** Trong thực tế, ở một doanh nghiệp sẽ có nhiều nhiệm vụ khác nhau theo từng dự án: trong một dự án xây dựng ta cần có kiến trúc sư, công nhân xây dựng, nhà thầu, quản lý dự án,...

Giả sử doanh nghiệp xây dựng P cần xây dựng một trung tâm mua sắm, doanh nghiệp hiện có N nhân viên và hiện đang có M nhiệm vụ thiếu nhân viên, trong đó năng suất của mỗi nhân viên $i$ trong từng nhiệm vụ là khác nhau, mỗi nhiệm vụ cần có số nhân viên tối thiểu, mỗi nhân viên không đảm nhiệm quá một số lượng nhiệm vụ. Giao nhiệm vụ cho N nhân viên sao cho dự án đạt năng suất tối đa.

*Phân tích.* Các cách phân công này được biểu diễn thành ma trận A = $[a_{ij}]$ gồm N hàng và M cột, trong đó: N là số lượng nhân viên; M là số nhiệm vụ đang thiếu nhân lực hiện tại; $a_{ij}$ là năng suất làm việc của nhân viên thứ i ở nhiệm vụ j.

| Nhân viên | **Nhiệm vụ** | | | | **Số nhiệm vụ tối đa** |
|---|---|---|---|---|---|
| | **Nhiệm vụ 1** | **Nhiệm vụ 2** | | **Nhiệm vụ M** | |
| Nhân viên 1 | $a_{11}$ | $a_{12}$ | … | $a_{1M}$ | $c_1$ |
| Nhân viên 2 | $a_{21}$ | $a_{22}$ | … | $a_{2M}$ | $c_2$ |
| … | … | … | … | … | … |
| Nhân viên N | $a_{N1}$ | $a_{N2}$ | … | $a_{NM}$ | $c_N$ |
| Số nhân viên cần thiết | $b_1$ | $b_2$ | … | $b_M$ | |

Đặt $x_{ij}$ là biến nhị phân, $x_{ij} = 1$ nếu công việc $j$ được giao cho nhân viên i và $x_{ij} = 0$ nếu ngược lại (i = 1,2,..,N) (j = 1,2,…,M). Ta có:

- Năng suất tối đa có thể đạt được là:

$$f = \sum_{i=1,j=1}^{N,M} a_{ij}\, x_{ij} \rightarrow max$$

Các ràng buộc:

- Về số lượng nhân viên cần thiết: $\sum_{i=1}^{N} x_{ij} \leq b_j \ (j = 1, 2, \dots, M)$
- Về số nhiệm vụ tối đa của từng nhân viên: $\sum_{j=1}^{M} x_{ij} \leq c_i \ (i = 1, 2, \dots, N)$

Như vậy, vấn đề đã được đưa về thành một bài toán Quy hoạch tuyến tính nguyên.

**Bài toán người giao hàng TSP:** Một nhân viên giao hàng được yêu cầu thăm mỗi thành phố trong n thành phố, được đánh chỉ dấu từ 1 đến n. Anh ấy bắt đầu từ một "thành phố gốc" được đánh dấu là 0, đi qua mỗi thành phố khác trong số n thành phố chính xác một lần, và trở về thành phố 0. Trong quá trình đi lại, anh ấy phải quay trở lại 0 chính xác t lần, bao gồm cả lần quay trở về cuối cùng (ở đây t có thể thay đổi), và anh ấy không được đi quá p thành phố trong một chuyến đi: một chuyến đi, được hiểu là một chuỗi các lần thăm các thành phố mà không dừng lại tại thành phố 0. Yêu cầu tìm một lịch trình như vậy để giảm thiểu tổng khoảng cách mà nhân viên bán hàng đi qua.

*Phân tích.* Khởi tạo giá trị biến:

- $x_{ij}$ là biến nhị phân, người giao hàng đi từ thành phố i đến thành phố j khi và chỉ khi $x_{ij} = 1$, ngược lại.
- $d_{ij}$ là khoảng cách đi từ thành phố i đến thành phố j
- $u_i$ là thành phố i là thành phố thứ $u_i$ trong một chuyến đi bao gồm thành phố i. Như vậy với $x_{ij} = 1$, ta có $u_j = u_i + 1$.

Vì giảm thiểu tổng khoảng cách mà nhân viên bán hàng đi nên ta có hàm mục tiêu tuyến tính như sau:

$$f = \sum_{0 \leq i \neq j \leq n} \sum d_{ij} x_{ij} \rightarrow min$$

Các ràng buộc:

- Về số lần thăm của 1 thành phố: $\sum_{i=0}^{n} x_{ij} = 1, (i \neq j, j = 1, 2, \dots, n)$ và $\sum_{j=0}^{n} x_{ij} = 1, (j \neq i, i = 1, 2, \dots, n)$
- Về số lần quay lại thành phố gốc: $\sum_{i=1}^{n} x_{i0} = t$
- Về số lần thăm trong 1 chuyến đi: $u_i \leq p, (i = 1, 2, \dots, n)$

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

- Nếu t cố định, để bài toán tồn tại giải pháp phải có ràng buộc $tp \geq n$, với $t = 1$ thì $p \geq n$.
Như vậy, vấn đề đã được đưa về thành bài toán Quy hoạch tuyến tính Nguyên.

### 2.5 Các khó khăn và kết quả mong muốn

Các thuật toán cơ bản của Quy hoạch tuyến tính đều đưa ra được lời giải tối ưu cho bài toán quy hoạch tuyến tính. Tuy nhiên, kết quả của bài toán không là số nguyên. Điều này có thể dẫn đến sự thiếu chính xác trong kết quả, ảnh hưởng nhiều đến tính ứng dụng của nó.
Giả sử nếu ta đang tối ưu hóa việc đặt hàng trong một nhà kho với các mặt hàng khác nhau. Ta muốn tối ưu hóa sao cho chi phí tổng cộng là thấp nhất. Ta xác định rằng số lượng đặt hàng của mỗi mặt hàng phải là một số nguyên không âm. Nếu kết quả tối ưu cho số lượng đặt hàng của một hoặc nhiều mặt hàng không phải là số nguyên, ta sẽ phải đưa ra quyết định để làm tròn lên hoặc xuống số lượng đó để đảm bảo rằng nó là một số nguyên. Nhưng việc làm tròn số này có thể dẫn đến thay đổi đáng kể trong chi phí tổng cộng và không đảm bảo rằng thỏa mãn tất cả các điều kiện ràng buộc của bài toán. Do đó, việc giải quyết bài toán Quy hoạch tuyến tính nguyên sẽ trở nên khó khăn và đòi hỏi các phương pháp khác nhau để giải quyết định bài toán nguyên.

## 3 CÁC CÁCH GIẢI CHÍNH CHO QUY HOẠCH TUYẾN TÍNH NGUYÊN

Trước hết, ta giới thiệu qua về hai phương pháp quan trọng để giải bài toán Quy hoạch tuyến tính là Đơn hình và Hình học, sau đó tiếp đến các phương pháp cho bài toán Quy hoạch nguyên: phương pháp cắt Gomory, và Kỹ thuật nhánh & cận.

### 3.1 Phương pháp đơn hình

Năm 1947, nhà toán học George Bernard Danzig đưa ra phương pháp đơn hình: ý tưởng cơ bản là bắt đầu xét từ một phương án cực biên (phương án chấp nhận được), ta xem nó có là phương án tốt nhất hay chưa, nếu chưa là phương án tốt nhất ta lần lượt xét đến các phương án cực biên liền kề sao cho làm tăng giá trị hàm mục tiêu. Quá trình tiến hành đến lúc thu được phương án tối ưu hoặc giá trị hàm mục tiêu không hữu hạn.

| $B$ | $\boldsymbol{c}^B$ | $\boldsymbol{b}^B$ | $\boldsymbol{A}^B_{k_1}$ | ... | $\boldsymbol{A}^B_{k_m}$ | $\boldsymbol{A}^B_{k_{m+1}}$ | ... | $\boldsymbol{A}^B_{k_v}$ | ... | $\boldsymbol{A}^B_{k_n}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | $c_{k_1}$ | ... | $c_{k_m}$ | $c_{k_{m+1}}$ | ... | $c_{k_v}$ | ... | $c_{k_n}$ |
| $\boldsymbol{A}_{k_1}$ | $c_{k_1}$ | $b_1$ | 1 | ... | 0 | $a_{1k_{m+1}}$ | ... | $a_{1k_v}$ | ... | $a_{1k_n}$ |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| $\boldsymbol{A}_{k_r}$ | $c_{k_r}$ | $b_r$ | 0 | ... | 0 | $a_{rk_{m+1}}$ | ... | $a_{rk_v}$ | ... | $a_{rk_n}$ |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| $\boldsymbol{A}_{k_m}$ | $c_{k_m}$ | $b_m$ | 0 | ... | 0 | $a_{mk_{m+1}}$ | ... | $a_{mk_v}$ | ... | $a_{mk_n}$ |
| $Z$ | min/max | | $\Delta_{k_1}$ | ... | $\Delta_{k_m}$ | $\Delta_{k_{m+1}}$ | ... | $\Delta_{k_v}$ | ... | $\Delta_{k_n}$ |

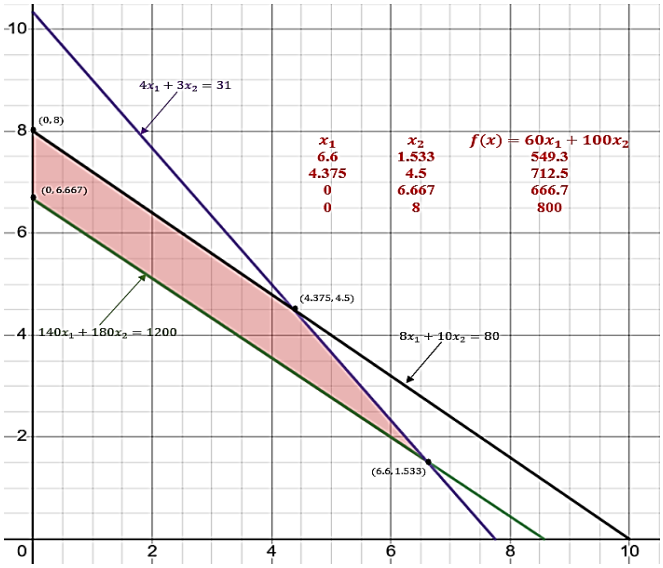
Hình 1: Ảnh mô tả bảng đơn hình

**Ý tưởng của thuật toán**
**Bước 1:** Khởi tạo: thành lập phương án cực biên.
**Bước 2:** Kiểm tra lời giải: xét xem phương án cực biên hiện hành đã là phương án tối ưu hay chưa bằng dấu hiệu tối ưu. Nếu phương án cực biên này là phương án tối ưu thì kết thúc. Ngược lại, nếu bài toán có phương án mới tốt hơn thì sang **bước 3.**
**Bước 3:** Xây dựng một phương án cực biên mới sao cho giá trị hàm mục tiêu lớn hơn hoặc bằng giá trị hàm mục tiêu của phương án cực biên trước đó.
**Bước 4:** Quay lại bước 2: Tiếp tục quá trình kiểm tra và tìm kiếm lời giải tối ưu cho đến khi đạt được lời giải tối ưu.

### 3.2 Phương pháp hình học

Phương pháp hình học là một kỹ thuật trong lĩnh vực tối ưu hóa sử dụng các khái niệm hình học như đường thẳng, mặt phẳng, để thực hiện các phép cách, phép vạch trên không gian nghiệm và kỹ thuật hình học để giải quyết các bài toán quy hoạch tuyến tính. Phương pháp này kết hợp giữa các yếu tố hình học và các

ràng buộc tuyến tính trong bài toán để tìm ra giải pháp tối ưu. Đây là một cách tiếp cận trực quan và hiệu quả để giải quyết bài toán quy hoạch tuyến tính. Phương pháp này sử dụng cho bài toán có hai biến.

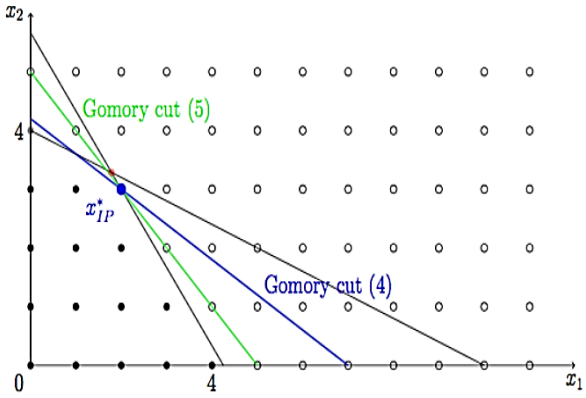



Hình 2: Ảnh minh họa phương pháp hình học

Dưới đây là thuật toán cơ bản của phương pháp hình học:
**Bước 1:** Chuẩn bị dữ liệu: Xác định các hệ số và ràng buộc của bài toán quy hoạch tuyến tính.
**Bước 2:** Vẽ đồ thị cho bài toán bằng cách sử dụng các biến quyết định.
**Bước 3:** Tìm tất cả các điểm cực biên, là các giao điểm giữa hai đường thẳng.
**Bước 4:** Thay giá trị của tọa độ các điểm cực biên tìm được vào hàm mục tiêu và so sánh.

### 3.3 Phương pháp cắt Gomory

Phương pháp cắt Gomory là một phương pháp phát triển các ràng buộc mới để giải quyết các bài toán quy hoạch tuyến tính nguyên. Phương pháp này dựa trên việc tạo ra các ràng buộc mới từ các ràng buộc hiện có và tìm kiếm giải pháp tối ưu cho bài toán mới.




Hình 3: Ảnh mô tả phương pháp cắt Gomory

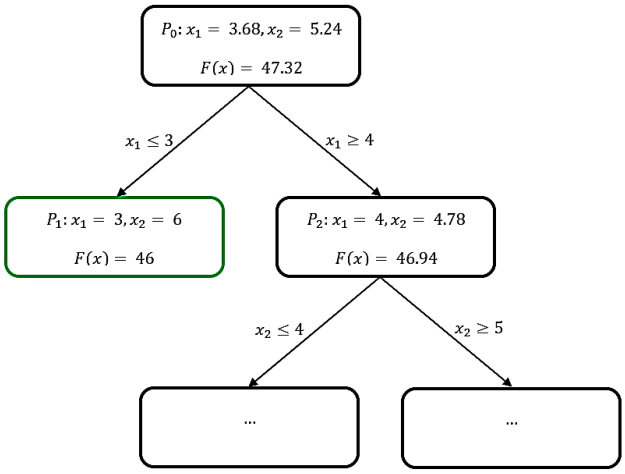
Dưới đây là các bước cụ thể để thực hiện thuật toán cắt Gomory:
**Bước 1:** Khởi tạo:
Giải bài toán quy hoạch tuyến tính ban đầu và tìm một lời giải nguyên xuất phát (nếu có).
Lưu trữ giá trị nguyên tối ưu là giá trị tốt nhất hiện tại.
**Bước 2:** Kiểm tra nguyên:

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Kiểm tra xem lời giải hiện tại có phải là lời giải nguyên tối ưu không.
Kiểm tra từng biến trong lời giải để xác định xem có biến không nguyên nào không.
Nếu không có biến không nguyên, lời giải hiện tại là lời giải nguyên tối ưu và kết thúc.
Nếu có biến không nguyên, chọn một biến không nguyên để tiếp tục bước tiếp theo.
**Bước 3:** Xác định ràng buộc cắt nguyên:
Chọn một biến không nguyên trong lời giải hiện tại.
Xây dựng một ràng buộc cắt nguyên dựa trên giá trị của biến chọn và các hệ số của nó trong các ràng buộc ban đầu.
Ràng buộc cắt nguyên có dạng: $a_1x_1 + a_2x_2 + \cdots + a_nx_n \geq b$, trong đó $a_i$ là các hệ số và b là giá trị không nguyên của biến.
**Bước 4:** Thêm ràng buộc cắt nguyên:
Thêm ràng buộc cắt nguyên vào bài toán Quy hoạch tuyến tính ban đầu.
Giải lại bài toán để tìm lời giải mới với các ràng buộc cắt nguyên đã thêm.
**Bước 5:** Cập nhật:
Nếu lời giải mới tốt hơn lời giải tốt nhất hiện tại, cập nhật lời giải tốt nhất.
Quay lại bước 2 để kiểm tra lời giải mới.

## 3.4 Phương pháp nhánh & cận

Phương pháp nhánh & cận: là một thuật toán được sử dụng để giải các bài toán tối ưu tổ hợp, bao gồm cả bài toán quy hoạch tuyến tính nguyên (ILP - Integer Linear Programming). Phương pháp này tìm kiếm không gian tìm nghiệm của bài toán một cách hệ thống bằng cách chia nhỏ không gian tìm kiếm ban đầu thành các phần nhỏ hơn và loại bỏ những phần không cần thiết để cải thiện hiệu suất giải quyết bài toán.




Hình 4: Ảnh minh họa phương pháp nhánh & cận

**Các bước thực hiện thuật toán**
Dưới đây là thuật toán cơ bản của phương pháp nhánh & cận:
**Bước 1:** Khởi tạo một bộ lời giải tạm thời, thường là một lời giải không tối ưu.
**Bước 2:** Tính toán giá trị của hàm mục tiêu tại lời giải tạm thời.
**Bước 3:** Kiểm tra xem lời giải tạm thời có phải là lời giải tối ưu hay không. Nếu đúng, kết thúc thuật toán. Nếu không, tiếp tục bước 4.
**Bước 4:** Chọn một biến nguyên trong lời giải tạm thời mà không phải là số nguyên, phân chia không gian tìm kiếm thành hai phần con, một phần con chứa các giá trị lớn hơn hoặc bằng giá trị nguyên của biến được chọn, và một phần con chứa các giá trị nhỏ hơn.
**Bước 5:** Áp dụng thuật toán tìm kiếm tối ưu trên từng phần con được phân chia (các vấn đề con).
**Bước 6:** Lặp lại bước 2 đến bước 5 cho đến khi lời giải tối ưu được tìm thấy.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

### 3.5 Về bài toán Quy hoạch tuyến tính đa mục tiêu

Quy hoạch tuyến tính đa mục tiêu là bài toán có hai hoặc nhiều hàm mục tiêu, và ta cần tối ưu tất cả chúng đồng thời. Nó có ứng dụng trong các lĩnh vực đời sống một cách thiết thực, vì rõ ràng nhiều lúc ta cần phải tìm min, max cùng lúc nhiều giá trị liên quan đến các đại lượng cần khảo sát. Trong khoảng 50 năm, đã có hơn 70 thuật toán khác nhau cho bài toán QHTT hai hoặc nhiều hàm mục tiêu. So với QHTT thông thường, nếu xét quy hoạch nguyên thì dạng tối ưu đa biến này khó tiếp cận hơn. Các phương pháp giải:

- *Two pharses*: giải trước bài toán ứng với một hàm mục tiêu, sau đó thêm kết quả tìm được vào các ràng buộc và giải tiếp cho hàm mục tiêu còn lại.
- *Combining*: kết hợp hai hàm mục tiêu lại bởi một trọng số alpha thích hợp. Sau đó, giải cho từng giá trị alpha thay đổi một khoảng nhỏ; trên cơ sở đó chọn ra alpha tốt nhất. Ta cũng có thể giải riêng bài toán ứng với từng hàm $z_1, z_2, ..., z_k$, được các giá trị $d_1, d_2, ..., d_k$, sau đó ta sử dụng ý tưởng chung là gộp lại thành một hàm mục tiêu duy nhất và giải tiếp. Có nhiều cách để gộp lại:
  + Chandra Sen: max z = tổng $z_i/|d_i|$ - tổng $z_j/|d_j|$ với max được lấy dấu + còn min được lấy dấu trừ.
  + Trung bình điều hòa: đặt SL, SS bằng tổng các hàm mục tiêu lấy max và lấy min, đặt $Hm_1$, $Hm_2$ = trung bình điều hòa của các $d_i$ (lấy max) và $d_i$ (lấy min). Khi đó ta có:

$$\max z = SL/Hm_1 - SS/Hm_2$$

  + Trung bình bình phương: tương tự ta lấy căn bậc hai của tổng bình phương $Qm_1$, $Qm_2$.

## 4 THỰC NGHIỆM VỚI CÁC BÀI TOÁN CỤ THỂ

Trong phần này, nhóm đã xây dựng thuật toán dựa trên phương pháp nhánh cận kết hợp với thư viện PuLP để giải một số bài toán điển hình trên nền tảng Python. Các mã nguồn nhóm đã thực hiện có thể xem tại link bên dưới: https://github.com/VuTrinhNguyenHoang/Integer-Linear-Programming.

### 4.1 Dạng bài toán vận tải trong du lịch

Trong sa mạc, có những con lạc đà, loại có một bướu và có hai bướu, giúp vận chuyển hàng hóa giữa hai thành phố với các thông tin như sau:

| Loại lạc đà | Khả năng chở | Nhu cầu cỏ khô | Nhu cầu nước | Chi phí |
|---|---|---|---|---|
| Một bướu | 140 kg | 4 bó | 8 lít | 60$ |
| Hai bướu | 180 kg | 3 bó | 10 lít | 100$ |

Có một đoàn du khách cần vận chuyển ít nhất 1200kg và trong kho có lượng cỏ khô dự trữ là 31 bó, lượng nước dự trữ là 80 lít. Tìm phương án thuê lạc đà để tổng chi phí là ít nhất, vẫn đảm bảo các ràng buộc nêu trên.

Phân tích. Gọi các biến nguyên $x_1, x_2$ là số lượng lạc đà một bướu và lạc đà hai bướu.

Hàm mục tiêu: $f = 60x_1 + 100x_2$

- Du khách cần vận chuyển ít nhất 1200kg: $140x_1 + 180x_2 \geq 1200$
- Trong kho hiện có lượng cỏ khô dự trữ là 31 bó: $4x_1 + 3x_2 \leq 31$
- Lượng nước dự trữ là 80 lít: $8x_1 + 10x_2 \leq 80$
- Ngoài ra còn có các ràng buộc không âm: $x_1, x_2 \geq 0$

Như vậy, bài toán sẽ có dạng sau: $x_1, x_2 \geq 0, x_1, x_2 \in \boldsymbol{Z}$

$$f = 60x_1 + 100x_2 \rightarrow min$$
$$140x_1 + 180x_2 \geq 1200$$
$$4x_1 + 3x_2 \leq 31$$
$$8x_1 + 10x_2 \leq 80$$

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Thực thi chương trình:

```
max/min?: min
Number of variables = 2
Number of binding functions = 3
Objective function: 60 100
Binding function 1: 140 180 >= 1200
Binding function 2: 4 3 <= 31
Binding function 3: 8 10 <= 80
Binding condition of variable 1: 0 none
Binding condition of variable 2: 0 none
```

Kết quả:

```
The result for the objective function is
 560.0
The value of the hidden x: [6. 2.]
Time: 0.006988048553466797 (s)
```

### 4.2 Bài toán pha cắt vật tư

Có một số thanh cốt thép dài 11.7m. Để thi công lắp đặt cốt thép đầm, cột cho một tầng của một tòa nhà bê tông cốt thép thì cần phải có 210 đoạn dài 2.1m; 161 đoạn dài 2.9m; 176 đoạn dài 3.2m; 48 đoạn dài 4.2m. Vậy nên cắt cốt thép như thế nào để tốn ít thanh nguyên nhất?

*Phân tích.* Đầu tiên, với thanh 11.7m thì ta sẽ có nhưng phương án cắt sau đây:

| **Các phương án** | **Số lượng các đoạn** | | | | **Mẫu thừa** |
|---|---|---|---|---|---|
| | 2.1m | 2.9m | 3.2m | 4.2m | |
| $x_1$ | 0 | 0 | 1 | 2 | 0.1 |
| $x_2$ | 0 | 1 | 0 | 2 | 0.4 |
| $x_3$ | 1 | 0 | 0 | 2 | 1.2 |
| $x_4$ | 0 | 0 | 2 | 1 | 1.1 |
| $x_5$ | 0 | 1 | 1 | 1 | 1.4 |
| $x_6$ | 0 | 2 | 0 | 1 | 1.7 |
| $x_7$ | 2 | 1 | 0 | 1 | 0.4 |
| $x_8$ | 3 | 0 | 0 | 1 | 1.2 |
| $x_9$ | 1 | 0 | 3 | 0 | 0 |
| $x_{10}$ | 1 | 1 | 2 | 0 | 0.3 |
| $x_{11}$ | 2 | 0 | 2 | 0 | 1.1 |
| $x_{12}$ | 1 | 2 | 1 | 0 | 0.6 |
| $x_{13}$ | 2 | 1 | 1 | 0 | 1.4 |
| $x_{14}$ | 4 | 0 | 1 | 0 | 0.1 |
| $x_{15}$ | 0 | 4 | 0 | 0 | 0.1 |
| $x_{16}$ | 1 | 3 | 0 | 0 | 0.9 |
| $x_{17}$ | 2 | 2 | 0 | 0 | 1.7 |
| $x_{18}$ | 4 | 1 | 0 | 0 | 0.4 |
| $x_{19}$ | 5 | 0 | 0 | 0 | 1.2 |

Gọi $x_i$ là số thanh nguyên pha cắt theo phương án i, $x_i \in Z$. Ta có hàm mục tiêu: $f = \sum_{i=1\to 19} x_i \to min$

Các ràng buộc:

- Có 210 đoạn dài 2.1m: $0x_1 + 0x_2 + x_3 + 0x_4 + 0x_5 + 0x_6 + 2x_7 + 3x_8 + x_9 + x_{10} + 2x_{11} + x_{12} + 2x_{13} + 4x_{14} + 0x_{15} + x_{16} + 2x_{17} + 4x_{18} + 5x_{19} \geq 210$
- Có 161 đoạn dài 2.9m: $0x_1 + x_2 + 0x_3 + 0x_4 + x_5 + 2x_6 + x_7 + 0x_8 + 0x_9 + x_{10} + 0x_{11} + 2x_{12} + x_{13} + 0x_{14} + 4x_{15} + 3x_{16} + 2x_{17} + 1x_{18} + 0x_{19} \geq 161$
- Có 176 đoạn dài 3.2m: $x_1 + 0x_2 + 0x_3 + 2x_4 + x_5 + 0x_6 + 0x_7 + 0x_8 + 3x_9 + 2x_{10} + 2x_{11} + x_{12} + x_{13} + x_{14} + 0x_{15} + 0x_{16} + 0x_{17} + 0x_{18} + 0x_{19} \geq 176$
- Có 48 đoạn dài 4.2m: $2x_1 + 2x_2 + 2x_3 + x_4 + x_5 + x_6 + x_7 + x_8 + 0x_9 + 0x_{10} + 0x_{11} + 0x_{12} + 0x_{13} + 0x_{14} + 0x_{15} + 0x_{16} + 0x_{17} + 0x_{18} + 0x_{19} \geq 210$

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Thực thi chương trình:

```
max/min?: min
Number of variables = 19
Number of binding functions = 4
Objective function: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Binding function 1: 0 0 1 0 0 0 2 3 1 1 2 1 2 4 0 1 2 4 5 >= 210
Binding function 2: 0 1 0 0 1 2 1 0 0 1 0 2 1 0 4 3 2 1 0 >= 161
Binding function 3: 1 0 0 2 1 0 0 0 3 2 2 1 1 1 0 0 0 0 0 >= 176
Binding function 4: 2 2 2 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 >= 48
Binding condition of variable 1: 0 none
Binding condition of variable 2: 0 none
Binding condition of variable 3: 0 none
Binding condition of variable 4: 0 none
Binding condition of variable 5: 0 none
Binding condition of variable 6: 0 none
Binding condition of variable 7: 0 none
Binding condition of variable 8: 0 none
Binding condition of variable 9: 0 none
Binding condition of variable 10: 0 none
Binding condition of variable 11: 0 none
Binding condition of variable 12: 0 none
Binding condition of variable 13: 0 none
Binding condition of variable 14: 0 none
Binding condition of variable 15: 0 none
Binding condition of variable 16: 0 none
Binding condition of variable 17: 0 none
Binding condition of variable 18: 0 none
Binding condition of variable 19: 0 none
```

Kết quả

```
The result for the objective function is 144.0
The value of the variable x: [19.  5.  0.  0.  0.  0.  0.  0. 38.  0.  0.  0.  0. 43. 39.  0.  0.  0
 ,
  0.]
Time: 8.832241773605347 (s)
```

Như vậy lời giải của bài toán sẽ là:

| Các phương án | Số lượng cắt | Số lượng các đoạn | | | |
|---|---|---|---|---|---|
| | | 2.1m | 2.9m | 3.2m | 4.2m |
| $x_1$ | 19 | 0 | 0 | 19 | 38 |
| $x_2$ | 5 | 0 | 5 | 0 | 10 |
| $x_9$ | 38 | 38 | 0 | 114 | 0 |
| $x_{14}$ | 43 | 172 | 0 | 43 | 0 |
| $x_{15}$ | 39 | 0 | 156 | 0 | 0 |
| **Tổng** | **144** | **210** | **161** | **176** | **48** |

Tuy nhiên, có một đáp án khác cho bài toán này, đó là:

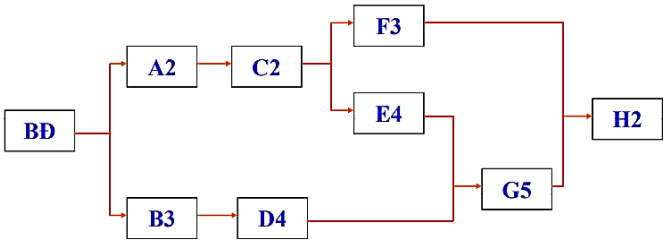
| Các phương án | Số lượng cắt | Số lượng các đoạn | | | |
|---|---|---|---|---|---|
| | | 2.1m | 2.9m | 3.2m | 4.2m |
| $x_1$ | 23 | 0 | 0 | 23 | 46 |
| $x_3$ | 1 | 1 | 0 | 0 | 2 |
| $x_9$ | 37 | 37 | 0 | 111 | 0 |
| $x_{14}$ | 42 | 168 | 0 | 42 | 0 |
| $x_{15}$ | 40 | 0 | 160 | 0 | 0 |
| $x_{18}$ | 1 | 4 | 1 | 0 | 0 |
| **Tổng** | **144** | **210** | **161** | **176** | **48** |

Hai lời giải trên tuy cho chúng ta cùng một đáp số là 144 nhưng có sự khác biệt về số lượng phương án lựa chọn, chứng tỏ rằng không nhất thiết phải có một phương án tối ưu duy nhất.

### 4.3 Bài toán rút ngắn dự án đầu tư

Rút ngắn thời gian của dự án xây dựng nhà công nghiệp của công ty ABC với những điều kiện sau:

- Thời gian thực thi dự án là 12 tuần
- Chi phí rút ngắn là tối thiểu




Bảng mô tả công việc


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| Công việc | Thời gian (tuần) | | Chi phí (ngàn đồng) | | Thời gian rút ngắn được | Chi phí rút ngắn đơn vị |
|---|---|---|---|---|---|---|
| | Bình thường | Rút ngắn | Bình thường | Rút ngắn | | |
| A | 2 | 1 | 22.000 | 23.000 | 1 | 1.000 |
| B | 3 | 1 | 30.000 | 34.000 | 2 | 2.000 |
| C | 2 | 1 | 26.000 | 27.000 | 1 | 1.000 |
| D | 4 | 3 | 48.000 | 49.000 | 1 | 1.000 |
| E | 4 | 2 | 56.000 | 58.000 | 2 | 1.000 |
| F | 3 | 2 | 30.000 | 30.500 | 1 | 500 |
| G | 5 | 2 | 80.000 | 86.000 | 3 | 2.000 |
| H | 2 | 1 | 16.000 | 19.000 | 1 | 3.000 |

*Phân tích.* Gọi $X$ là thời điểm kết thúc sớm (EF) của một công việc.
- $X_A$ là thời điểm kết thúc sớm của công việc A.
...
- $X_H$ là thời điểm kết thúc sớm của công việc H.
Gọi Y là thời gian rút ngắn (tuần) của từng công việc.
- $Y_A$ là thời gian rút ngắn của công việc A.
...
- $Y_H$ là thời gian rút ngắn của công việc H.
Vì mục tiêu của bài là chi phí rút ngắn tối thiểu nên ta có hàm mục tiêu:

$$f = 1000Y_A + 2000Y_B + 1000Y_C + 1000Y_D + 1000Y_E + 500Y_F + 2000Y_G + 3000Y_H \rightarrow min$$

Các ràng buộc:
- Ràng buộc về quan hệ trước sau: EF của công việc sau ≥ EF của công việc trước + $(t - Y)$

| Công việc | Ràng buộc về quan hệ trước sau | Công việc | Ràng buộc về quan hệ trước sau |
|---|---|---|---|
| BĐ | $X_{BĐ} = 0$ | F | $X_F - X_C + Y_F \geq 3$ |
| A | $X_A - X_{BĐ} + Y_A \geq 2$ | G | $X_G - X_D + Y_G \geq 5$ |
| B | $X_B - X_{BĐ} + Y_B \geq 3$ | | $X_G - X_E + Y_G \geq 5$ |
| C | $X_C - X_A + Y_C \geq 2$ | H | $X_H - X_F + Y_H \geq 2$ |
| D | $X_D - X_B + Y_D \geq 4$ | | $X_H - X_G + Y_H \geq 2$ |
| E | $X_E - X_C + Y_E \geq 4$ | | |

- Ràng buộc về thời gian rút ngắn: $Y_A \leq 1, Y_B \leq 2, Y_C \leq 1, Y_D \leq 1, Y_E \leq 2, Y_F \leq 1, Y_G \leq 3, Y_H \leq 1$

Sau khi thiết lập bài toán xong, thực thi chương trình:

```
max/min?: min
Number of variables = 16
Number of binding functions = 10
Objective function: 1000 2000 1000 1000 1000 500 2000 3000 0 0 0 0 0 0 0 0
Binding function 1: 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 >= 2
Binding function 2: 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 >= 3
Binding function 3: 0 0 1 0 0 0 0 0 -1 0 1 0 0 0 0 0 >= 2
Binding function 4: 0 0 0 1 0 0 0 0 0 -1 0 1 0 0 0 0 >= 4
Binding function 5: 0 0 0 0 1 0 0 0 0 0 -1 0 1 0 0 0 >= 4
Binding function 6: 0 0 0 0 0 1 0 0 0 0 -1 0 0 1 0 0 >= 3
Binding function 7: 0 0 0 0 0 0 1 0 0 0 0 -1 0 0 1 0 >= 5
Binding function 8: 0 0 0 0 0 0 0 1 0 0 0 0 -1 0 1 0 >= 5
Binding function 9: 0 0 0 0 0 0 0 0 1 0 0 0 0 0 -1 0 1 >= 2
Binding function 10: 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 -1 1 >= 2
Binding condition of variable 1: 0 1
Binding condition of variable 2: 0 2
Binding condition of variable 3: 0 1
Binding condition of variable 4: 0 1
Binding condition of variable 5: 0 2
Binding condition of variable 6: 0 1
Binding condition of variable 7: 0 3
Binding condition of variable 8: 0 1
Binding condition of variable 9: 0 none
Binding condition of variable 10: 0 none
Binding condition of variable 11: 0 none
Binding condition of variable 12: 0 none
Binding condition of variable 13: 0 none
Binding condition of variable 14: 0 none
Binding condition of variable 15: 0 none
Binding condition of variable 16: 0 12
```

```
The result for the objective function is 5000.0
The value of the variable x: [ 0.  0.  0.  1.  2.  0.  1.  0.  2.  3.  4.  6.  6. 10. 10. 12.]
Time: 0.007541894912719726 (s)
```

Như vậy, ta có kết quả cho bài toán này như sau:

| **Thời điểm kết thúc công việc** | | **Thời gian rút ngắn (tuần)** | | **Chi phí rút ngắn** |
|---|---|---|---|---|
| $X_A$ | 2 | $Y_A$ | 0 | 0 |
| $X_B$ | 3 | $Y_B$ | 0 | 0 |
| $X_C$ | 4 | $Y_C$ | 0 | 0 |
| $X_D$ | 6 | $Y_D$ | 1 | 1.000 |
| $X_E$ | 6 | $Y_E$ | 2 | 2.000 |
| $X_F$ | 10 | $Y_F$ | 0 | 0 |
| $X_G$ | 10 | $Y_G$ | 1 | 2.000 |
| $X_H$ | 12 | $Y_H$ | 0 | 0 |
| **Tổng chi phí rút ngắn (tối thiểu)** | | | | **5.000** |

Tương tự bài toán trên, ta cũng có các lời khác cho việc rút ngắn công việc với những phương án rút gọn rất khác nhau. Việc chọn giải pháp nào cho phù hợp nhất thì còn phụ thuộc vào điều kiện của người sử dụng.

# 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 5.1 Kết luận

Với bài báo này, nhóm đã giới thiệu tổng quan về Quy hoạch tuyến tính nói chung và Quy hoạch nguyên nói riêng, các bài toán điển hình cũng như cách tiếp cận. Thông qua các bài toán ứng dụng ở mục trước, nhóm đã thấy được một số ưu điểm của phương pháp nhánh cận:

- Hiệu quả: cho kết quả tốt trong thời gian tối thiểu, phụ thuộc vào số lượng biến, số lượng hàm ràng buộc, không gian tìm kiếm và kết quả ban đầu mà chương trình cho kết quả nhanh hay chậm. Nhưng nhìn chung, thời gian thực thi cho ra kết quả của phương pháp nhánh cận có phần tốt hơn.
- Khả năng tìm kiếm toàn diện: phương pháp nhánh cận có thể tìm kiếm toàn bộ giá trị nguyên có thể của các biến quyết định, do đó có nhiều kết quả tối ưu với cùng một vấn đề.
- Tính linh hoạt: khi áp dụng phương pháp nhánh cận, ta có thể sử dụng những chiến lược nhánh cận khác nhau để giải quyết từng nhánh con, cho phép tùy chỉnh với từng bài toán cụ thể.

## 5.2 Hướng phát triển

Nhóm mong muốn tự cài đặt được cả thuật toán đơn hình và hình học trong thư viện PuLP. Để làm được điều này, cần phải tìm hiểu rõ hơn cơ chế, các bước xử lý, tuy nhiên điều này hoàn toàn khả thi.

# TÀI LIỆU THAM KHẢO

[1] Nguyễn Đức Phương, *Bài giảng Quy hoạch tuyến tính*, Đại học Công nghiệp Thành phố Hồ Chí Minh, 2017.

[2] Robert J. Vanderbei, Linear Programming - Foundations and Extensions, Fifth Edition. Springer Publication, 2020.

[3] George B. Dantzig, Mukund N. Thapa, *Linear Programming, Introduction*. Springer Publication, 1997.

[4] Đỗ Thị Xuân Lan, *Quy hoạch tuyến tính số nguyên*, Tin học trong quản lý, Đại học Bách khoa Thành phố Hồ Chí Minh, 2010.

[5] Thu Huyền, Quy hoạch tuyến tính và những ứng dụng, 2021. [Online] Available at: https://odt.vn/tin-tuc/quy-hoach-tuyen-tinh.html.

[6] Tucker, A. W On directed graphs and integer programs, Princeton–IBM Math. Re-search Project Techn. Report, 1960.

[7] Nguyễn Đình Định, *Ứng dụng mô hình Quy hoạch nguyên tuyến tính trong thiết kế phần mềm cắt thép thanh*, Tạp chí khoa học, Đại học Hồng Đức – số 29.2016.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

[8] Gomory, R. E. *An algorithm for integer solutions to linear programs*, Re-search Project Techn. Report No. 1, 1958.

[9] Luenberger, David G and Yinyu Ye, *Linear and Nonlinear Programming*. Springer, 2008.

[10] Dantzig, Fulkerson and Johnson On a linear-programming, combinatorial approach to the traveling salesman problem. JORSA, 7 (1959).

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F003**

# PHÂN TÍCH DỮ LIỆU BẢO TOÀN TÍNH RIÊNG TƯ TRÊN CÂY MERKLE CỦA BLOCKCHAIN

LÂM NGỌC LONG[1*], ĐỖ QUỐC TUẤN[1], TRẦN ANH TUẤN[1], NGUYỄN MINH LUẬN[1], TRƯƠNG MẠNH QUỲNH[1], NGUYỄN THÀNH TÍN[1] , VÕ QUỐC THỊNH[1], ĐÀO XUÂN HOÀNG TUẤN[1], NGUYỄN HỮU TÌNH[1], LÊ PHÚC LỮ[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*[*]lamngoclong1004@gmail.com*

**Tóm tắt.** Blockchain là một công nghệ đang phát triển rất nhanh và được sử dụng trong nhiều lĩnh vực như tiền ảo, chuỗi cung ứng, giáo dục, thương mại, y tế... Bên cạnh đó, việc truy xuất dữ liệu bảo toàn tính riêng tư (Private Information Retrieval - PIR) vẫn luôn là một thách thức lớn. Như thế, việc áp dụng PIR cho cây Merkle của Blockchain là một ý tưởng tự nhiên và quan trọng, được nhóm chúng tôi đưa ra khai thác và giải quyết. Việc này cho phép các thao tác trên cây Merkle được thực hiện mà không cần tiết lộ thông tin riêng tư của người dùng. Bằng cách nghiên cứu cách thức truy xuất Merkle proof của một giao dịch, nhóm đưa ra cách áp dụng PIR cho từng node trong cây Merkle, chính là các hash và cũng hướng đến cải tiến hiệu suất bằng việc phân hoạch tập hợp đỉnh trong cây một cách đồng bộ (Balanced Unrelated Partitioning - BUP).

**Từ khóa.** PIR, Cây Merkle, Hash, Blockchain, BUP.

## PRIVACY-PRESERVING DATA ANALYSIS ON BLOCKCHAIN'S MERKLE TREE

**Abstract.** Blockchain is a technology which is developing very fast and can be applied in many fields such as crypto currency, logistic, education, commerce, healthcare,... Besides, private Information Retrieval (PIR) is still a big challenge. So the application of PIR to the Merkle tree of Blockchain is really a natural and important idea, which is exploited and solved by this research. This allows operations on the Merkle tree to be performed without revealing the user's private information. By studying how the Merkle proof of a transaction is retrieved, our team shows how to apply PIR to each node in the Merkle tree, which are hashes and aim to improve performance by using balanced unrelated partition (BUP).

**Keywords.** PIR, Merkle tree, Hash, Blockchain, BUP.

## 1 GIỚI THIỆU

Theo [1], quyền riêng tư và bảo vệ thông tin cá nhân là một trong những vấn đề quan trọng nhất trong thời đại số hóa hiện nay. Với sự phát triển nhanh chóng của công nghệ và mạng lưới kết nối, khối lượng dữ liệu cá nhân được thu thập và chia sẻ đã tăng vọt, từ các hoạt động trực tuyến hàng ngày cho đến các dự án lớn về phân tích dữ liệu. Tuy nhiên, việc thu thập và sử dụng dữ liệu cá nhân cũng đặt ra nhiều nguy cơ và mối đe dọa đến quyền riêng tư và an ninh thông tin cá nhân. Trong ngữ cảnh này, phân tích dữ liệu đã trở thành một công cụ mạnh mẽ để rút ra thông tin quan trọng và đưa ra quyết định hợp lý. Từ việc nghiên cứu xu hướng tiêu dùng đến phân tích tình hình kinh doanh, phân tích dữ liệu đã đóng vai trò ngày càng quan trọng trong việc tạo ra giá trị và nâng cao hiệu suất hoạt động của nhiều lĩnh vực như tài chính, y tế và quản lý chuỗi cung ứng.

Hiện nay, có những phương pháp truy xuất dữ liệu đã và đang được phát triển hiện nay như: giả mờ dữ liệu [3] là một phương pháp để xử lý dữ liệu không chắc chắn hoặc không rõ ràng. Với những ưu điểm như linh hoạt và xử lý đa dạng dữ liệu nhưng cũng có những khuyết điểm như đòi hỏi tính toán phức tạp, độ chính xác và khả dụng không cao trong các tập dữ liệu lớn và phức tạp. Ta cũng có phương pháp Anonim hóa dữ

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

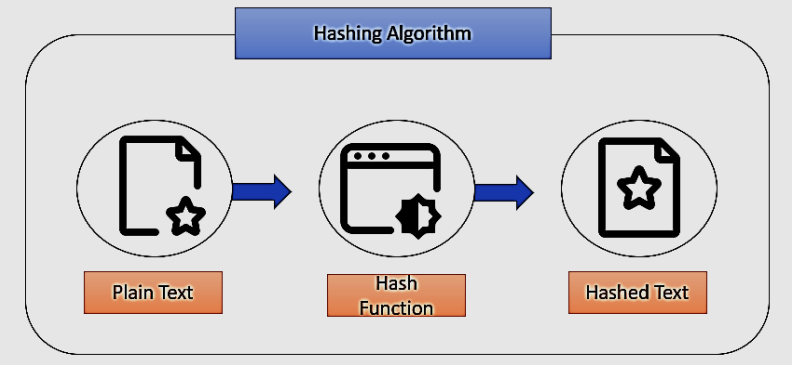
liệu [4] là quá trình xử lý dữ liệu để loại bỏ hoặc ẩn danh thông tin cá nhân, nhằm bảo vệ quyền riêng tư và tránh tiềm ẩn rủi ro.

Trong lĩnh vực công nghệ, công nghệ Blockchain (còn gọi là Chuỗi khối) đã thu hút sự chú ý đặc biệt. Blockchain, với tính chất phi tập trung và tính bảo mật cao, đã mang lại nhiều tiềm năng ứng dụng trong việc bảo vệ thông tin cá nhân và đảm bảo tính toàn vẹn của dữ liệu. Tuy nhiên, việc bảo vệ tính riêng tư của người dùng trên Blockchain vẫn là một thách thức lớn. Đó chính là chủ đề quan trọng mà chúng tôi sẽ phân tích và đưa ra giải pháp giải quyết trong nghiên cứu này.

# 2 CƠ SỞ LÝ THUYẾT VÀ CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 Hashing

Hashing (hàm băm) là một kỹ thuật rất quan trọng và có tính ứng dụng cực kỳ cao, đặc biệt trong lĩnh vực bảo mật. Hashing là quá trình biến đầu vào là một nội dung có kích thước, độ dài bất kỳ rồi sử dụng những thuật toán, công thức toán học để biến thành đầu ra tiêu chuẩn có độ dài nhất định. Nó là một kiểu mã hóa một chiều và có thể mô tả như ảnh bên dưới:




Hình 1: Mô tả cách hoạt động của mã hóa.

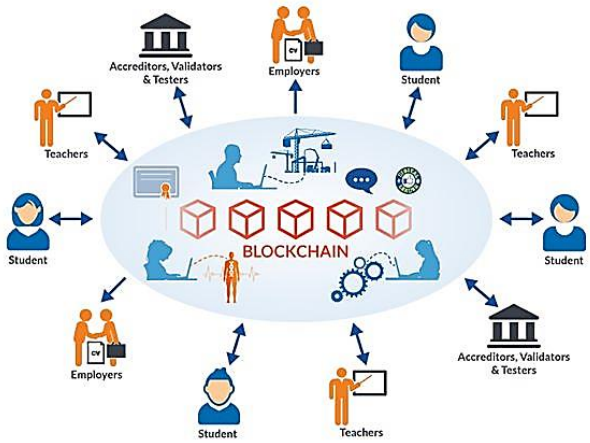
Các hàm băm phổ biến: MD5, SHA1, SHA2. Ngoài ra, hàm băm có thể dùng để:

- *Kiểm tra sự toàn vẹn của tệp tin*: Như đặc điểm của hàm băm, cùng một giá trị sẽ cho ra cùng một giá trị băm. Vậy nên ta có thể đối chiếu tệp tin ta tải trên mạng về với bản gốc bằng cách so sánh giá trị băm của chúng với nhau. Nếu chúng có chung giá trị băm tức là tệp tin trùng với bản gốc, nếu không tệp tin của bạn đã bị sửa đổi hoặc bị hỏng. Một số trường hợp, tệp tin của bạn tải về bị can thiệp bởi bên thứ ba trước khi đến thiết bị của bạn và chúng có thể cài mã độc vào tệp tin đó. Việc kiểm tra giá trị băm giúp đảm bảo tệp tin an toàn.
- *Xác minh mật khẩu*: trong những thiết kế cơ sở dữ liệu hiện đại, thứ lưu trong đó không phải mật khẩu dưới dạng văn bản đơn thuần mà là giá trị hash của chúng. Khi nhập, mật khẩu sẽ được chạy qua hàm hash, sau đó sẽ được so sánh với giá trị băm trong cơ sở dữ liệu để quyết định user có được chứng thực để sử dụng dịch vụ không. Điều này làm giảm đáng kể thiệt hại khi cơ sở dữ liệu bị tấn công, khi những gì bị lộ ra ngoài là những giá trị băm chứ không phải mật khẩu gốc. Để an toàn hơn, hệ thống còn thêm giá trị salt vào mật khẩu gốc, rồi cho chạy qua hàm băm, sau đó mới lưu vào cơ sở dữ liệu. Vậy nên kể cả khi giá trị băm của mật khẩu bị lộ và bị giải mã, kẻ tấn công vẫn chưa thể có được mật khẩu thực sự ban đầu.

## 2.2 Công nghệ Blockchain

### 2.2.1 Giới thiệu

Theo [8], Blockchain bắt nguồn từ cuối những năm 1970, khi một nhà khoa học máy tính tên là Ralph Merkle được cấp bằng sáng chế cho cây Băm hay cây Merkle. Những cây này là một cấu trúc khoa học máy tính để lưu trữ dữ liệu bằng cách liên kết các khối có sử dụng mật mã. Vào cuối những năm 1990, Stuart Haber và W. Scott Stornetta đã sử dụng cây Merkle để triển khai một hệ thống trong đó dấu thời gian của tài liệu không thể bị làm giả. Đây là trường hợp chuỗi khối xuất hiện đầu tiên trong lịch sử.




Hình 2: Mô phỏng các ứng dụng của BlockChain

### 2.2.2 Các ứng dụng

Công nghệ blockchain được sử dụng cho nhiều mục đích khác nhau từ cung cấp dịch vụ tài chính đến quản trị hệ thống bỏ phiếu. Dưới đây là các ứng dụng phổ biến nhất của blockchain:

- *Tiền điện tử*: Việc sử dụng blockchain phổ biến nhất hiện nay là tiền điện tử như Bitcoin hoặc Ethereum. Khi mọi người mua, trao đổi hoặc chi tiêu tiền điện tử, các giao dịch được ghi lại trên một blockchain.
- *Ngân hàng*: Ngoài tiền điện tử, blockchain đang được sử dụng để xử lý các giao dịch bằng tiền tệ fiat như USD và EUR. Công nghệ này giúp việc gửi tiền qua ngân hàng nhanh hơn và các giao dịch được xác minh nhanh hơn ngoài giờ làm việc bình thường.
- *Chuyển giao tài sản*: Blockchain cũng có thể được sử dụng để ghi lại và chuyển quyền sở hữu các tài sản khác nhau. Công nghệ này hiện đang rất phổ biến với các tài sản kỹ thuật số như NFT - một đại diện cho quyền sở hữu nghệ thuật kỹ thuật số và video.
- *Smart contracts*: Một ứng dụng khác của blockchain là các hợp đồng tự thực hiện thường được gọi là "hợp đồng thông minh". Các hợp đồng kỹ thuật số này được ban hành tự động sau khi các điều kiện được đáp ứng.

### 2.2.3 Các ưu nhược điểm và cách thức hoạt động

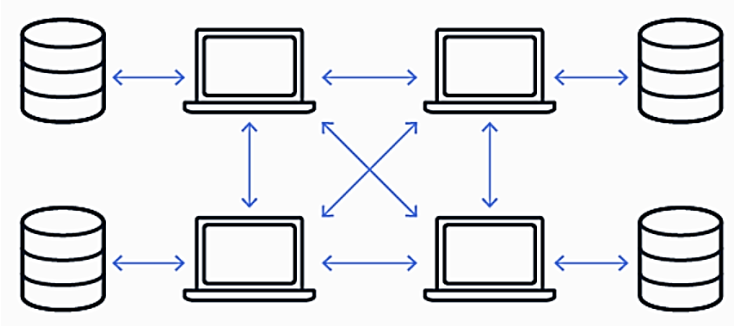
Ưu điểm của blockchain có thể kể đến như sau: độ chính xác cao hơn của các giao dịch, không cần trung gian, chuyển tiền hiệu quả hơn, phi tập trung, bất biến và đồng thuận. Bên cạnh đó, nhược điểm của blockchain đó là tốc độ xử lý chậm như quá trình xác nhận các giao dịch trong blockchain yêu cầu sự đồng thuận của nhiều tham gia, điều này khiến quá trình xử lý giao dịch diễn ra chậm hơn so với các hệ thống truyền thống. Chi phí cao do các giao dịch trong blockchain đòi hỏi một số nguồn tài nguyên lớn như điện năng và sức mạnh tính toán, dẫn đến việc tăng chi phí giao dịch, đặc biệt là trên các mạng công cộng quy mô lớn như Bitcoin và Ethereum.

Các thành phần chính của Blockchain bao gồm: sổ cái phân tán, smart contracts và mật mã hóa công khai. Các bước mà một Blockchain hoạt động:

- Bước 1 – Ghi lại giao dịch
- Bước 2 – Đạt được sự đồng thuận
- Bước 3 – Liên kết các khối
- Bước 4 – Chia sẻ sổ cái

Các loại Blockchain: mạng lưới chuỗi khối công khai, chuỗi khối công khai, chuỗi khối hỗn hợp và các mạng lưới chuỗi khối liên hợp.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh



Hình 3: Mô tả hệ thống phân phối bản sao mới nhất của sổ cái trung tâm cho toàn bộ người tham gia.

## 2.3 Tổng quan về cây Merkle

### 2.3.1 Giới thiệu về cây Merkle

Cây Merkle được đặt theo tên nhà khoa học cùng tên đã phát minh vào năm 1987 trong bài báo "A Digital Signature Based on a Conventional Encryption Function" (Chữ kí số dựa trên chức năng mã hóa thông thường). Đây là một cấu trúc dữ liệu cây nhị phân, trong đó mỗi đỉnh được biểu diễn bằng mã hash của nội dung trong các đỉnh con của nó.

Cây Merkle cho phép biểu diễn một lượng lớn dữ liệu và đảm bảo tính toàn vẹn dữ liệu. Mọi thay đổi trong các đỉnh con sẽ dẫn đến thay đổi của thông tin tại đỉnh gốc của cây. Điều này giúp kiểm tra tính toàn vẹn của dữ liệu một cách hiệu quả với độ phức tạp O(log(n)) với n là số lượng dữ liệu lưu trữ tại các lá của cây. Cây Merkle rất hữu ích vì nó cho phép người dùng xác minh một giao dịch cụ thể mà không cần tải xuống toàn bộ chuỗi khối.

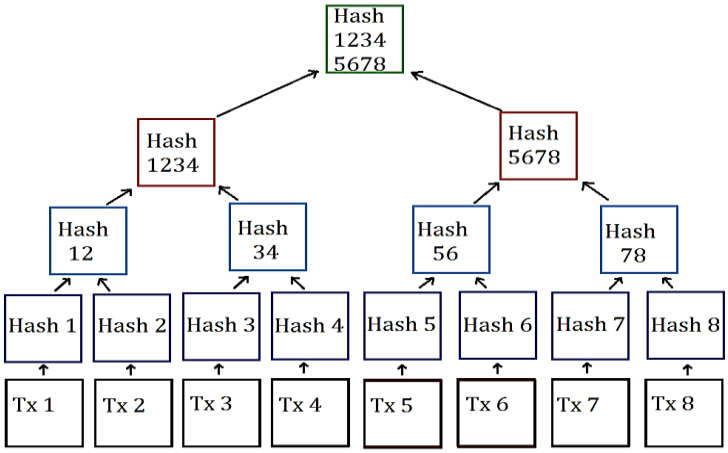
Cây Merkle có nhiều ứng dụng quan trọng trong thực tế. Ví dụ, nó được sử dụng để đồng bộ dữ liệu trong Amazon DynamoDB, lưu trữ ghi log của Google's Certificate Transparency, và cả lưu trữ các giao dịch trong các Blockchain.

### 2.3.2 Cách thực hoạt động

Theo [11], để dễ hiểu ta hình dung cấu trúc này giống như một cái cây. Trong sơ đồ bên dưới, "T" là một giao dịch, "H" là một hàm băm. Các hàm băm ở hàng dưới cùng được gọi là các "lá", các giá trị băm ở giữa là "các nhánh" và hàm băm ở trên cùng là "gốc". Gốc Merkle của một khối nhất định được lưu trữ trong tiêu đề. Ví dụ, gốc Merkle có thể là:

*e045b18e7a3d708d686717b4f44db2099aabcad9bebf968de5f7271b458f71c8.*

Gốc Merkle được kết hợp với các thông tin khác (phiên bản phần mềm, hàm băm của khối trước, nhãn thời gian (timestamp), mục tiêu độ khó và số được sử dụng một lần (nonce)) và sau đó chạy qua hàm băm để tạo ra hàm băm duy nhất của khối.




Hình 4: Mô phỏng cây Merkle trong BlockChain.

Ví dụ: giả sử muốn xác minh rằng giao dịch T có trong khối ở sơ đồ trên. Nếu có hàm băm gốc ($H_{12345678}$), ta sẽ tìm nó như sau: truy vấn $H_4$ trong mạng lưới và nó trả về các giá trị $H_3$, $H_{12}$ và $H_{5678}$. Cây Merkle cho phép xác minh rằng mọi thứ được tính bằng ba giá trị băm đã biết là $H_{12}$, $H_3$, $H_{5678}$ và gốc $H_{12345678}$, $H_4$ (hàm băm duy nhất bị thiếu) phải có trong hệ thống.

## 2.4 Federated Learning

### 2.4.1 Giới thiệu về Federated Learning

Trong phần này, ta giới thiệu thêm về một vấn đề bảo toàn tính riêng tư liên quan đến lĩnh vực Machine learning. Hiện nay, việc huấn luyện các mô hình trí tuệ nhân tạo với phương pháp Machine learning (học máy) cổ điển đang phải đối mặt với hai thử thách chính. Một là trong hầu hết các ngành công nghiệp dữ liệu thường bị cô lập, phân tán nhiều nơi. Hai là ngày càng nhiều các chính sách bảo mật dữ liệu. Để giải quyết những khó khăn này, một phương pháp học máy là Federated Learning hay còn được gọi là học cộng tác là một kỹ thuật học máy cho phép huấn luyện mô hình trí tuệ nhân tạo với dữ liệu trên các thiết bị, máy chủ, trung tâm dữ liệu phân tán mà không cần phải tập trung dữ liệu lại một nơi như cách chúng ta thường làm với phương pháp Học máy cổ điển nhưng vẫn có hiệu suất không quá khác biệt so với phương pháp học máy cổ điển [7].

### 2.4.2 Các đặc điểm chính của Federated Learning

Để đảm bảo kết quả cuối cùng của mô hình toàn cục có kết quả tốt, Federated Learning học tập dựa vào quá trình lặp được chia nhỏ thành một tập hợp các tương tác giữa Node và Server gọi là vòng lặp Federated Learning. Mỗi quy trình vòng lặp bao gồm việc gửi mô hình toàn cục cho các Node tham gia huấn luyện với dữ liệu cục bộ để tạo ra mô hình cục bộ và tổng hợp các mô hình cục bộ này để tạo phiên bản mô hình toàn cục mới [7].

Quá trình lặp học liên kết gồm các quy trình chính như sau:

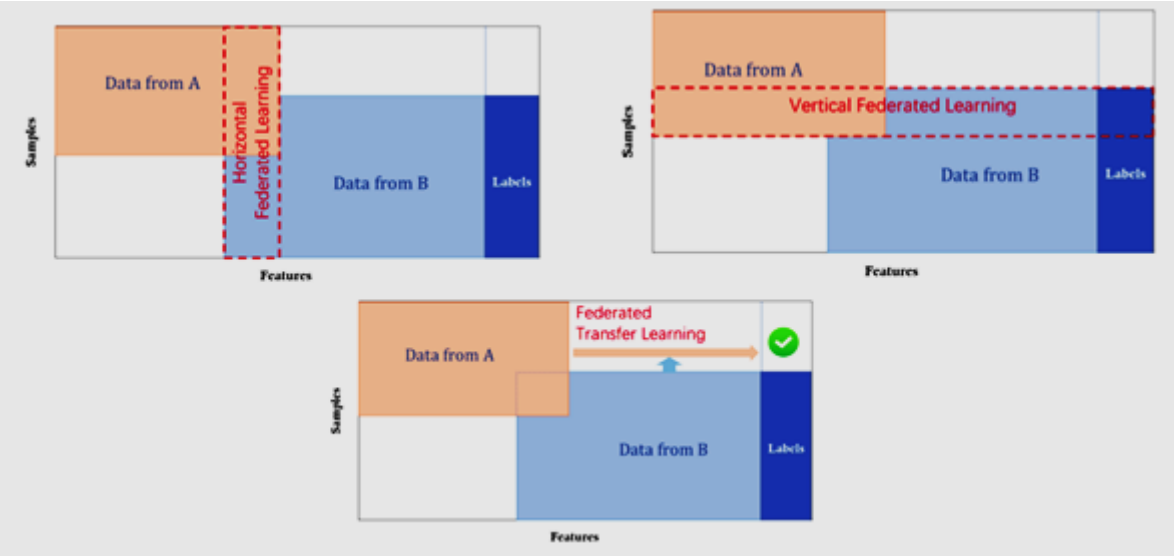
1. Khởi tạo: Dựa trên tập nhỏ dữ liệu ở Server, một mô hình học máy được chọn và huấn luyện.
2. Chọn các Node học: Một tỉ lệ các Node (tập con của toàn bộ Node đang tham gia học) sẽ được chọn để tham gia huấn luyện. Các Node không được chọn khác thì đợi các vòng lặp sau.
3. Cấu hình: Server sẽ gửi các siêu tham số cho các Node để huấn luyện với dữ liệu cục bộ (mini_batch, local_iteration,..)
4. Phản hồi: các Node sau khi học sẽ gửi mô hình cục bộ trở lại Server để tiến hành tổng hợp. Server sẽ tổng hợp các kết quả này lại tạo phiên bản mô hình toàn cục mới. Đối với các Node được chọn nhưng bị lỗi (mất kết nối,..) không phản hồi được ở vòng này thì sẽ được yêu cầu gửi phản hồi ở các vòng lặp sau.
5. Kết thúc: Một khi đạt được kết quả mong đợi (số vòng lặp toàn cục hoặc là một ngưỡng hiệu suất cụ thể) thì Server sẽ hoàn thiện mô hình toàn cục và kết thúc quá trình lặp.

### 2.4.3 Các loại Federated Learning

Trong Federated Learning dữ liệu thường có phân phối và đặc tính rất đa dạng, để phù hợp với các bài toán hơn thì chúng ta có các hình thức Federated Learning dựa trên phân vùng và đặc tính của dữ liệu.

- Horizontal Federated Learning: trong trường hợp này, feature space sẽ như nhau và hai bên học sẽ tiến hành đóng góp thêm mẫu để làm giàu dữ liệu.
- Vertical Federated Learning: còn được gọi là học liên kết dựa trên feature áp dụng cho các trường hợp hai bộ dữ liệu có chung (ID space) nhưng khác nhau về (feature space).
- Federated Transfer Learning: hai tập dữ liệu khác nhau sẽ không có bất cứ điểm chung nào trong feature space và sample space.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 5: Mô phỏng các loại Federated Learning.

## 3 BÀI TOÁN TRUY XUẤT DỮ LIỆU BẢO TOÀN TÍNH RIÊNG TƯ (PRIVATE INFORMATION RETRIEVAL - PIR)

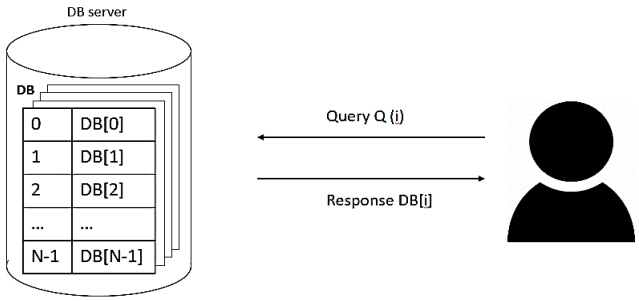
### 3.1 Private information retrieval và ứng dụng

Bài toán PIR − private information retrieval, đặt ra khi một người dùng muốn tải một nội dung từ cơ sở dữ liệu công khai sao cho không để lộ thông tin họ đã tải những gì.

PIR cung cấp một cơ chế bảo mật quyền riêng tư cho việc truy cập thông tin được lưu trữ trên máy chủ từ xa. Giao thức PIR đảm bảo rằng máy chủ không biết gì về truy vấn của người dùng, trong khi người dùng có thể nhận được thông tin mong muốn mà không tiết lộ sự quan tâm của mình cho máy chủ.

Bài toán được giới thiệu lần đầu vào năm 1995 bởi Chor và cộng sự; phát triển thêm một nhánh vào năm 1997 bởi Kushilevitz. Từ đó đến nay, đã có nhiều mô hình đã được khai thác, gần nhất các bài báo trong chuỗi hội nghị khoa học ISIT 2020-2021.

- Server: có chứa n bit/item dữ liệu ($x_1$, $x_2$, ..., $x_n$) với n rất lớn.
- User: muốn truy xuất một tập con k bits xi với i thuộc {1,2,..., n} sao cho giữ bí mật được thông tin của K (có thể xem K là các thông tin có liên quan với nhau).




Hình 6: Mô phỏng của Private information retrieval

Trong trường hợp toàn bộ dữ liệu được lưu trữ trên một server, giải pháp gần như duy nhất để đảm bảo được tính riêng tư là tải hết toàn bộ dữ liệu đó về (căn cứ trên các lý thuyết thông tin và các sức mạnh tính toán của máy tính lượng tử). Vì thế, để bài toán có ý nghĩa, xét hai khía cạnh kinh điển của PIR:

- Hạn chế về khả năng tính toán của server để có thể triển khai các mô hình mã hóa.
- Phân tán dữ liệu ra nhiều server độc lập, không giao tiếp để có thể triển khai các mô hình truy xuất thích hợp bằng cách gây nhiễu.

Các hệ thống truy xuất thông tin truyền thống yêu cầu người dùng tiết lộ truy vấn cho máy chủ, điều này gây ra mối quan ngại về quyền riêng tư, đặc biệt là khi xử lý dữ liệu nhạy cảm hoặc cá nhân. PIR giải quyết vấn đề này bằng cách cho phép người dùng truy xuất thông tin một cách riêng tư trong khi giữ truy vấn của mình không bị máy chủ biết. Giao thức PIR đóng vai trò quan trọng trong các ứng dụng yêu cầu quyền riêng tư và bảo mật, chẳng hạn như cơ sở dữ liệu y tế, hồ sơ tài chính và hệ thống nhắn tin riêng tư. Có các phương pháp khác nhau để đạt được truy xuất thông tin riêng tư, bao gồm:

- Truy xuất thông tin riêng tư tính toán (cPIR): Các giao thức cPIR dựa trên các kỹ thuật mật mã để che giấu truy vấn của người dùng. Thông thường, chúng sử dụng các kỹ thuật như mã hóa, truyền thông không phụ thuộc và tính toán đa bên an toàn để đảm bảo quyền riêng tư.
- Truy xuất thông tin riêng tư đối xứng: Những giao thức này sử dụng các hệ mã hóa đối xứng, trong đó người dùng mã hóa truy vấn của mình và gửi đến máy chủ. Máy chủ, không biết nội dung truy vấn, truy xuất dữ liệu được yêu cầu và gửi lại cho người dùng, người có thể giải mã để nhận thông tin mong muốn.
- Truy xuất thông tin riêng tư công khai: sử dụng mật mã học khóa công khai, trong đó người dùng mã hóa truy vấn của mình bằng khóa công khai của máy chủ. Máy chủ, sở hữu khóa bí mật tương ứng, có thể giải mã truy vấn, truy xuất dữ liệu được yêu cầu và gửi cho người dùng.

## 3.2 Private information retrieval trong merkle proofs

Khi cần kiểm tra xem một giao dịch có thuộc cây Merkle không, người dùng có thể sử dụng Merkle proof. Merkle proof cho phép người dùng kiểm tra tính toàn vẹn của một giao dịch bằng cách kiểm tra các giá trị hash trung gian trong cây Merkle. Bằng cách cung cấp Merkle proof, người dùng chỉ cần kiểm tra một số lượng nhỏ các giá trị hash để xác minh tính hợp lệ của một giao dịch mà không cần phải kiểm tra toàn bộ cây. Nhờ tính chất toàn vẹn và hiệu quả của nó, Merkle proof đóng vai trò quan trọng trong việc đảm bảo tính toàn vẹn dữ liệu và kiểm tra item trong các ứng dụng thực tế.

Trong lĩnh vực này [2], C. Y. Merkle đã đề xuất một phương pháp quan trọng để đảm bảo tính riêng tư trên Blockchain. Tuy nhiên, dù đã có nhiều nỗ lực để bảo vệ tính riêng tư trên Blockchain, vẫn còn nhiều thách thức và vấn đề cần được giải quyết. Việc triển khai và sử dụng cây Merkle trên Blockchain không đảm bảo hoàn toàn tính riêng tư, đặc biệt là trong trường hợp các giao dịch được kết hợp và phân tích. Một người dùng có thể bị nhận diện thông qua các thông tin được liên kết với địa chỉ công khai của mình trong cây Merkle. Vấn đề này gây ra lo ngại lớn về quyền riêng tư và an ninh thông tin trong việc sử dụng Blockchain. Với mục tiêu giải quyết những khuyết điểm của các phương pháp truy xuất dữ liệu hiện nay và tăng tính bảo mật trong việc sử dụng cây Merkel trên Blockchain, ta đưa ra giải pháp "*Privacy-preserving data analysis on Blockchain merkle tree*". Để đạt được mục tiêu này, ta tiến hành các nhiệm vụ sau:

- Đảm bảo quyền riêng tư cho thông tin cá nhân: Chúng tôi sẽ tìm hiểu và áp dụng các phương pháp và kỹ thuật để bảo vệ thông tin cá nhân trong quá trình phân tích dữ liệu trên Merkle Tree.
- Xây dựng hệ thống bảo vệ quyền riêng tư sử dụng Merkle Tree trên Blockchain: thiết kế và triển khai một hệ thống đáng tin cậy và an toàn, sử dụng Merkle Tree trên nền tảng Blockchain, nhằm đảm bảo tính toàn vẹn và bảo mật dữ liệu cá nhân. Tuân thủ các quy định chính sách và quyền riêng tư hiện hành.
- Đánh giá hiệu suất và tính tin cậy: tiến hành các thử nghiệm và đánh giá hiệu suất, tính tin cậy của phương pháp bảo vệ quyền riêng tư đề xuất, so sánh với các phương pháp hiện có.
- Qua đây, chúng tôi hy vọng có thể giới thiệu bài báo nghiên cứu và đề xuất một phương pháp cân bằng giữa quyền riêng tư của người truy cập thông tin và khả năng sử dụng hiệu quả trên nền tảng Blockchain, từ nâng cao sự tin cậy và chấp nhận của công nghệ Blockchain trong nhiều lĩnh vực và ứng dụng trong đời sống. Góp phần vào sự phát triển và ứng dụng tiếp thị của Blockchain trong tương lai.

## 3.3 Giải pháp thực hiện PIR trong Merkle Proofs

Ta đặt ra câu hỏi: "Nếu máy chủ có cây Merkle và người dùng có một giao dịch T cùng với chỉ mục của nó, liệu người dùng có thể truy xuất chứng minh Merkle của T mà không cần tiết lộ T?". Trong quá trình truy xuất riêng tư của chứng minh Merkle, người dùng muốn lấy được chứng minh Merkle của giao dịch T, trong khi vẫn giữ được tính riêng tư của T. Để làm điều này, người dùng có thể sử dụng các kỹ thuật mã hóa và các chức năng hash không thể đảo ngược.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**Giải pháp #1:** người dùng có thể tải xuống toàn bộ cây Merkle từ máy chủ. Nhìn chung, giải pháp này không yêu cầu chi phí tính toán và đảm bảo được tính toàn vẹn và riêng tư của giao dịch T. Nhưng nhược điểm lớn nhất chính là việc tiêu tốn tài nguyên khi user phải lưu trữ toàn bộ cây Merkle trên thiết bị của mình và điều này cũng yêu cầu chi phí lớn để thực hiện.
**Giải pháp #2:** Áp dụng kỹ thuật PIR để tải xuống từng giá trị hash một cách riêng tư từ cây Merkle, giải pháp này được nghiên cứu rộng rãi từ những năm 1977. Trong giải pháp này, thay vì tải xuống toàn bộ cây Merkle, người dùng yêu cầu máy chủ chỉ cung cấp chứng minh Merkle cho giao dịch T cụ thể. Chứng minh này sẽ chứa các giá trị hash trung gian và các phép tính hash được thực hiện trên cây Merkle, mà không tiết lộ thông tin về giao dịch T. Người dùng sau đó sử dụng chứng minh Merkle để xác minh tính hợp lệ của giao dịch T mà không cần phải truy xuất toàn bộ cây Merkle. Bằng cách chỉ yêu cầu máy chủ cung cấp các giá trị hash trung gian cần thiết và thực hiện các phép tính hash tương ứng, người dùng có thể kiểm tra tính toàn vẹn của giao dịch T một cách hiệu quả.
**Giải pháp #3:** (parallel) layer-based – giải pháp này tận dụng cấu trúc lớp của cây Merkle và xem mỗi lớp như một cơ sở dữ liệu độc lập. Cụ thể:

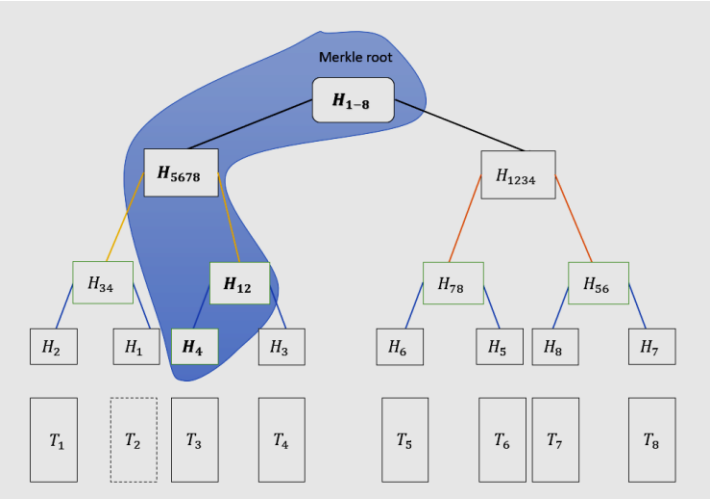
- Bước 1: Xử lý từng lớp như một cơ sở dữ liệu riêng biệt. Mỗi lớp chứa một số lượng hash tương ứng với số lượng đỉnh trên lớp đó.
- Bước 2: Áp dụng PIR (Private Information Retrieval) để tải xuống một giá trị hash từ mỗi lớp. PIR cho phép người dùng truy xuất thông tin từ một cơ sở dữ liệu mà không tiết lộ thông tin truy vấn của mình.
- Bước 3: Quá trình truy xuất được thực hiện song song, ví dụ như sử dụng nhiều lõi xử lý (multi-core) để cải thiện hiệu suất.

Nhìn chung, giải pháp dựa trên lớp giúp giảm kích thước cơ sở dữ liệu và cải thiện hiệu suất thông qua việc áp dụng PIR trên từng lớp của cây Merkle. Tuy nhiên, người dùng phải chờ đợi cho đến khi quá trình truy xuất đến lớp cuối cùng, do quá trình truy xuất bắt đầu từ lớp trên cùng và tiếp tục xuống lớp cuối cùng, người dùng phải chờ đợi cho đến khi truy xuất tới lớp dưới cùng (có kích thước $n = 2^h$), nên có thể gây mất điểm về thời gian đáp ứng trong một số trường hợp.
**Giải pháp #4:** (parallel) partition-based - giải pháp này nhằm giải quyết vấn đề truy xuất riêng tư của chứng minh Merkle mà không tiết lộ thông tin giao dịch T. Cụ thể:
Bước 1: Phân chia các giá trị hash thành $h$ tập hợp disjoin cân bằng. Mục đích là chia cây Merkle thành các phân đoạn có số lượng hash gần như nhau.
Bước 2: Áp dụng PIR (Private Information Retrieval) trên mỗi tập hợp. PIR cho phép người dùng truy xuất thông tin từ một cơ sở dữ liệu mà không tiết lộ thông tin truy vấn của mình.
Tổng quan về giải pháp (parallel) partition-based giúp cải thiện hiệu suất so với giải pháp trước đó (giải pháp #3) thông qua việc áp dụng PIR trên từng tập hợp của các giá trị hash. Tuy nhiên, việc phân vùng các giá trị hash thành các tập hợp cân bằng có thể là một thách thức đối với việc triển khai.

## 3.4 Swapped Merkle Tree

Swapped Merkle Tree [5] - một biến thể của cây Merkle trong đó các nút lá được hoán đổi vị trí với nhau. Swapped Merkle Tree đảm bảo tính toàn vẹn và cho phép truy xuất các Merkle proof theo root-to-leaf path. Nguyên tắc hoạt động cụ thể như sau:
*Hoán đổi các nút lá*: Trong Swapped Merkle Tree, các nút lá được hoán đổi vị trí với nhau. Điều này có nghĩa là khối dữ liệu tại mỗi lá đã được đảo ngược vị trí so với cây Merkle thông thường. Cây hoán đổi (Swapped tree): Kết quả của việc hoán đổi các nút lá là một cây Merkle mới, được gọi là Swapped Merkle Tree. Trong Swapped tree, các Merkle proof được xây dựng bằng cách sử dụng đường đi từ gốc đến lá (root-to-leaf path) trên cây hoán đổi. Mỗi Merkle proof vẫn giữ được tính chất xác thực tính toàn vẹn của dữ liệu mà không cần tiết lộ thông tin về cây hoán đổi cho bên thứ ba. Câu hỏi đặt ra là "Vấn đề đặt ra là làm thế nào để truy xuất một đường đi từ gốc đến lá trên Swapped Merkle Tree một cách riêng tư và bảo mật, mà không tiết lộ thông tin về cây hoán đổi cho bên thứ ba".

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 7: Minh họa Swapped Merkle Tree

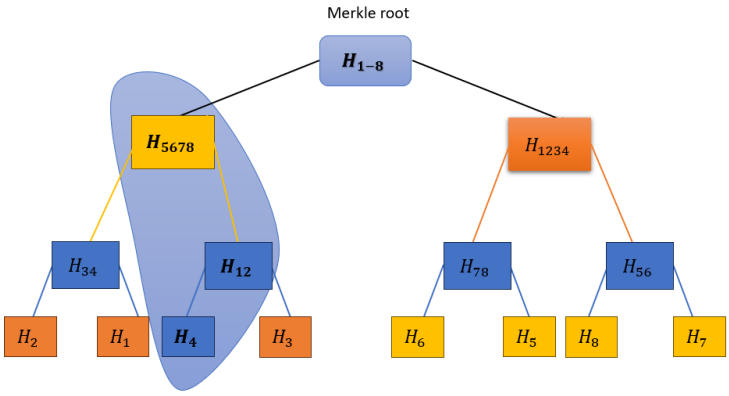
*Giải pháp*: Một giải pháp để truy xuất một đường đi từ gốc đến lá trên Swapped Merkle Tree một cách riêng tư là sử dụng giải pháp dựa trên phân vùng song song.
*Phân vùng các giá trị băm*: Đầu tiên, chúng ta chia các giá trị băm trên cây thành các tập hợp con không giao nhau và cân bằng nhau. Mỗi tập hợp con này sẽ chứa một phần của cây hoán đổi. Áp dụng PIR cho mỗi tập hợp con: Tiếp theo, chúng ta áp dụng Private Information Retrieval (PIR) cho mỗi tập hợp con. PIR đảm bảo rằng chúng ta có thể truy xuất thông tin từ mỗi tập hợp con một cách riêng tư và bảo mật mà không cần tiết lộ thông tin về cây hoán đổi.

## 3.5 Cài đặt mô hình song song trên cây Merkle dùng phân hoạch BUP

Giả sử người dùng có một giao dịch cụ thể và biết vị trí (index) của giao dịch đó trong cây Merkle. Người dùng muốn tải về một cách bảo mật các chứng minh Merkle của giao dịch đó (vì khi biết vị trí, người dùng biết các giá trị hash cần tải về). Một giải pháp trực tiếp là tải về toàn bộ cây Merkle, tuy nhiên, điều này đòi hỏi nhiều tài nguyên và có phần lãng phí. Để giải quyết vấn đề này, chúng ta có thể thực hiện việc hoán đổi vị trí của tất cả các đỉnh từ trái sang phải ở mỗi tầng, từ đó tạo ra một mô hình mới trong đó tất cả các chứng minh Merkle đều nằm trên cùng một đường đi từ gốc đến lá.
Cách cài đặt như sau: Với mỗi tầng trong cây Merkle, chúng ta biết rằng sẽ có một đỉnh cần tải về. Chúng ta sử dụng công cụ SealPIR [6] để thực hiện PIR (Private Information Retrieval) cho từng tầng. Công cụ này có khả năng hỗ trợ thực hiện song song cho các tầng. Tuy nhiên, điều này không cải thiện tốc độ lắm, vì mỗi tầng có số lượng đỉnh tăng gấp đôi và thời gian thực hiện PIR cho tầng cuối cùng sẽ lâu nhất (trong khi tầng đầu tiên chỉ có 2 đỉnh).
Vì vậy, để tăng tốc độ, chúng ta cần phân chia đồng đều các đỉnh trong cây Merkle và sử dụng BUP (Balanced Unrelated Partitioning) để thực hiện PIR. Điều này sẽ cải thiện tốc độ từ $2^h$ xuống khoảng $\frac{2^{h+1}}{h}$, tăng tốc độ lên khoảng $\frac{h}{2}$ lần. Việc phân chia đồng đều các đỉnh là cần thiết vì:

- Nó giúp tiết kiệm thời gian.
- Nó đảm bảo tính riêng tư bằng cách không tiết lộ vị trí (index) của giao dịch đang được quan tâm. Nếu không, nếu có hai đỉnh trên cùng một đường đi từ gốc đến lá, điều này sẽ cho phép máy chủ đoán biết vị trí (index) của giao dịch mà người dùng muốn kiểm tra.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 8: Minh họa cho một cách phân hoạch vi phạm tính không là tổ tiên

# 4 KẾT QUẢ VÀ THẢO LUẬN

## 4.1 So sánh cây Merkle có PIRvà cây Merkle bình thường

### 4.1.1 Về mục đích sử dụng

- Cây Merkle trong Blockchain: Mục đích chính của nó là xác minh tính toàn vẹn của dữ liệu trong Blockchain và cung cấp tính năng như Proof of Existence. Nó giúp đảm bảo rằng dữ liệu không bị thay đổi và các giao dịch trong Blockchain không bị gian lận.
- Cây Merkle có PIR: Mục đích chính là bảo vệ quyền riêng tư của người dùng khi truy vấn dữ liệu từ một nguồn dữ liệu công khai. Nó cho phép người dùng truy vấn dữ liệu mà không tiết lộ thông tin về truy vấn của họ cho người quản lý nguồn dữ liệu.

### 4.1.2 Xây dựng cây

- Cây Merkle trong Blockchain: Cây Merkle trong Blockchain được xây dựng bằng cách tìm giá trị hash của mọi mục dữ liệu và nhóm các mục đó thành từng cặp từ trái sang phải đây sẽ là các nút lá, tính toán hash của từng cặp áp dụng hàm băm và tạo ra nút cha mới, tiếp tục nhóm các nút cha này lại và tiếp tục quá trình này cho đến khi chỉ còn lại nút duy nhất, được gọi là root hash.
- Cây Merkle có PIR: Cây Merkle có PIR cũng được xây dựng tương tự như cây Merkle trong Blockchain. Tuy nhiên, cây Merkle có PIR được sử dụng để bảo vệ quyền riêng tư trong quá trình truy vấn dữ liệu. Dữ liệu được phân chia thành các phần tử, mỗi phần tử được hash và ghép thành các cặp, và quá trình này được lặp lại cho đến khi chỉ còn lại một giá trị root hash.

### 4.1.3 Quyền riêng tư và truy cập dữ liệu

- Cây Merkle trong Blockchain: Cây Merkle trong Blockchain không cung cấp bảo vệ quyền riêng tư trong việc truy cập dữ liệu. Mục đích chính của nó là đảm bảo tính toàn vẹn của dữ liệu và xác minh giao dịch.
- Cây Merkle có PIR: Cây Merkle có PIR được thiết kế để bảo vệ quyền riêng tư của người dùng khi truy vấn dữ liệu từ một nguồn dữ liệu công khai. Nó cho phép người dùng truy vấn dữ liệu mà không tiết lộ thông tin về truy vấn của họ cho người quản lý nguồn dữ liệu.

## 4.2 Kết quả nghiên cứu

Trong bài báo nghiên cứu này, chúng tôi đã khám phá ứng dụng của công nghệ Blockchain trong nhiều lĩnh vực đa dạng như tiền ảo, chuỗi cung ứng, giáo dục, thương mại, y tế, và các lĩnh vực khác. Với sự phát triển nhanh chóng của Blockchain, việc bảo toàn tính riêng tư của dữ liệu trở thành một thách thức lớn. Để giải quyết vấn đề này, chúng tôi đã đề xuất và nghiên cứu việc áp dụng PIR (Private Information Retrieval) cho cây Merkle của Blockchain, một ý tưởng tự nhiên và quan trọng.

Ứng dụng PIR cho cây Merkle cho phép các thao tác trên cây được thực hiện mà không cần tiết lộ thông tin riêng tư của người dùng. Bằng cách nghiên cứu cách thức truy xuất Merkle proof của một giao dịch, chúng tôi đã đề xuất cách áp dụng PIR cho từng node trong cây Merkle, đó là các giá trị hash. Điều này giúp người dùng kiểm tra tính toàn vẹn của giao dịch một cách riêng tư và an toàn.

Chúng tôi cũng hướng đến cải tiến hiệu suất của phương pháp này bằng việc áp dụng phân hoạch tập hợp đỉnh trong cây một cách đồng bộ (Balanced Unrelated Partitioning - BUP). Điều này giúp tối ưu hóa quá trình truy xuất thông tin và đáp ứng nhanh chóng yêu cầu của người dùng, đồng thời giảm thiểu tải lưu trữ và tăng cường hiệu suất trên hệ thống.

Kết quả nghiên cứu này đóng góp một góc nhìn mới về việc bảo toàn tính riêng tư trong việc truy xuất dữ liệu trên Blockchain thông qua ứng dụng PIR cho cây Merkle. Hi vọng rằng những phát hiện và kiến thức từ bài nghiên cứu này sẽ góp phần thúc đẩy sự phát triển và ứng dụng tiếp theo của công nghệ Blockchain trong các lĩnh vực quan trọng của đời sống và kinh tế. Qua bài nghiên cứu, nhóm chúng tôi đã phân tích và đã đưa ra các đề xuất phù hợp nhằm bảo toàn dữ liệu trên cây Merkle và hướng tới việc sử dụng phân hoạch tập hợp đỉnh trong cây một cách đồng bộ (Balanced Unrelated Partitioning - BUP) nhằm cải thiện hiệu suất.

Dưới đây là bảng so sánh giữa các giải pháp mà chúng tôi đã đề xuất ở trước đó:

| Giải pháp | Ưu điểm | Nhược điểm |
|---|---|---|
| Giải pháp #1 | - Đảm bảo tính toàn vẹn và riêng tư của giao dịch | - Tiêu tốn tài nguyên lớn khi phải lưu trữ toàn bộ cây Merkle trên thiết bị người dùng. |
| Giải pháp #2 | - Sử dụng kỹ thuật PIR giúp người dùng truy xuất thông tin một cách riêng tư từ cây Merkle. | - Có thể yêu cầu thời gian đáng kể để tải xuống từng giá trị hash một cách riêng lẻ nếu cây Merkle lớn. |
| Giải pháp #3 | - Tận dụng cấu trúc lớp của cây Merkle và cải thiện hiệu suất thông qua việc áp dụng PIR trên từng lớp của cây. | - Yêu cầu chờ đợi cho đến khi quá trình truy xuất đến lớp cuối cùng, có thể gây mất điểm về thời gian đáp ứng. |
| Giải pháp #4 | - Cải thiện hiệu suất so với giải pháp #3 thông qua việc áp dụng PIR trên các tập hợp cân bằng của các giá trị hash. | - Thách thức trong việc phân vùng các giá trị hash thành các tập hợp cân bằng có thể gây khó khăn cho việc triển khai giải pháp này. |

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này, chúng tôi đưa ra giải pháp để giải quyết vấn đề bảo toàn tính riêng tư trên cây Merkle của Blockchain nhằm bảo mật thông tin cho người dùng một cách hiệu quả. Bên cạnh đó các giải pháp còn giúp tăng hiệu suất, cải thiện tiến trình cho hệ thống, cụ thể là việc phân hoạch tập hợp đỉnh trong cây một cách đồng bộ (BUP). Hướng nghiên cứu trong tương lai là tìm hiểu, kết hợp thêm nhiều công nghệ thực tiễn để có thể tiến hành triển khai thực nghiệm với một ứng dụng cụ thể trên Web3, sử dụng ý tưởng bảo mật này.

## TÀI LIỆU THAM KHẢO

[1] Birnhack, Michael D, *A quest for a theory of privacy: context and contro*l, *Jurimetrics*, vol. 51, no. 4, 2011, pp. 447–79. JSTOR.

[2] Merkle, R, *A Certified Digital Signature*, p.218-238.

[Online] Available at: https://doi.org/10.1007/0-387-34805-0_21, 1989.

[3] Atanassov, K.T, *My Personal View on Intuitionistic Fuzzy Sets Theory. In: Bustince, H., Herrera, F., Montero, J. (eds) Fuzzy Sets and Their Extensions: Representation, Aggregation and Models. Studies in Fuzziness and Soft Computing*, vol 220. Springer, Berlin, Heidelberg, 2008.

[4] Altman, N. S, *Data Anonymization Techniques: A Comprehensive Survey, ACM Computing Surveys*, 46(4), 1-33, 2013.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

[5] Son Hoang Dau, *On a Tree Coloring Problem and Its Application in Private Retrieval of Merkle Proofs*, VMMACS 2021.

[6] Hao Chen, Kim Laine, and Srinath Setty, *PIR with compressed queries and amortized query processing*, Sebastian Angel, IEEE Symposium on Security and Privacy (SP), 2018.

[7] Dinh Tuan Anh, *Ap dung hoc lien ket vao danh gia diem tin dung*, Graduated thesis from Industrial University of Ho Chi Minh City, 2023.

[8] *Công nghệ Blockchain là gì?* [Online] Available at: https://aws.amazon.com/vi/what-is/blockchain

[9] *Hash là gì và dùng hash để làm gì?* [Online] Available at: https://codelearn.io/sharing/hash-la-gi-va-hash-dung-de-lam-gi

[10] Jasmine, *Blockchain là gì và nó được ứng dụng thế nào trong cuộc sống?*, 2021.
[Online] Available at: https://fptshop.com.vn/tin-tuc/danh-gia/blockchain-la-gi-va-no-duoc-ung-dung-the-nao-trong-cuoc-song-138522

[11] Tường Vy, Cây Merkle (Merkle Tree) trong công nghệ chuỗi khối là gì? Nội dung về cây Merke, 2020. [Online] Available at: https://vietnambiz.vn/cay-merkle-merkle-tree-trong-cong-nghe-chuoi-khoi-la-gi-20200120232652793.htm

**YSC5.F004**

# CẢI TIẾN THUẬT TOÁN TỐI ƯU HÀM LỒI BẰNG TÌM KIẾM TAM PHÂN VÀ QUY HOẠCH ĐỘNG

TRẦN LỘC[1*], LÂM QUANG PHÚ[1], TRƯƠNG CÔNG ĐẠT[1], NGUYỄN GIA LÂM[1],
TRẦN TRỌNG TRÍ[1], MAI THỊ DIỆU MY[1], PHAN TẤN TÀI[1], LÊ PHÚC LỮ[1], NGUYỄN HỮU TÌNH[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
**tranloc120603@gmail.com*

**Tóm tắt.** Trong lĩnh vực tối ưu, các bài toán liên quan đến hàm lồi cũng như độ hiệu quả về thời gian tính toán của chúng chiếm một vị trí quan trọng. Tính chất của hàm lồi là đạt cực đại hoặc cực tiểu tại một vị trí duy nhất trong miền xác định, mà ở đây ta quan tâm miền của bài toán là tập hợp rời rạc. Khi đó, có hai cách tiếp cận nổi bật là thuật toán chia để trị và quy hoạch động, đều có đặc điểm chung là phân tích vấn đề lớn thành các vấn đề con đồng dạng nhưng có kích thước nhỏ hơn để giải quyết độc lập. Trong bài báo này, nhóm sẽ tìm hiểu tập trung về một dạng chia để trị độc đáo là tìm kiếm tam phân cũng như đề xuất các cách cải tiến quy hoạch động dùng bao lồi và Alien trick. Từ đó, nhóm sẽ ứng dụng các kỹ thuật này vào giải quyết một số bài toán tối ưu nổi bật trong khoa học máy tính.

**Từ khóa.** Hàm lồi, Chia để trị, Tìm kiếm tam phân, Quy hoạch động bao lồi, Alien trick.

## IMPROVEMENT OF CONVEX OPTIMIZATION USING TERNARY SEARCH AND DYNAMIC PROGRAMMING

**Abstract.** In optimization scope, the class of problems involving convex functions as well as their computational time efficiency play an important role. The property of a convex function is to reach a maximum or a minimum at a single position in a given domain, where we are interested in a discrete set. There are two prominent approaches, divide and conquer algorithms and dynamic programming, both of which have the common feature of analyzing a large problem into smaller subproblems to be solved independently. In this article, the group will focus on a unique divide-and-conquer form of ternary search and propose the ways to improve dynamic programming using convex hulls and the Alien trick. From there, we will apply these techniques to solve some outstanding optimization problems in computer science.

**Keywords.** Convex function, Divide-and-Conquer, Ternary search, Convex DP, Alien trick.

## 1 GIỚI THIỆU

Trong lĩnh vực tối ưu, các bài toán liên quan đến hàm lồi xuất hiện khá phổ biến và có vai trò qua trọng. Ta xem xét giải quyết chúng dựa trên tính chính xác của thuật toán cũng như độ hiệu quả về thời gian tính toán. Trong nghiên cứu này, chúng tôi sẽ đưa ra các nội dung:

1. *Tổng quan lý thuyết, các tính chất về hàm lồi*: Trình bày tổng quan về các lý thuyết, tính chất. Từ các tính chất đó đề xuất các cách tối ưu Hàm lồi để giải quyết bài toán chính.
2. *Giới thiệu phương pháp tìm kiếm tam phân*: Phân tích, trình bày việc tối ưu tìm kiếm cực trị so với cách làm thông thường. Ưu điểm của tìm kiếm tam phân so với tìm kiếm nhị phân, tính toán độ phức tạp tìm kiếm tam phân để qua đó thấy sự hiệu quả trong việc tìm cực trị của Hàm lồi. Tính ứng dụng của tìm kiếm tam phân thông qua Ternary Search Tree.
3. *Quy hoạch động:* Trình bày các lý thuyết, bài toán, vấn đề mô tả cho phương pháp giải bằng quy hoạch động, quy hoạch động bao lồi. Qua tính chất của Hàm lồi, đề xuất cách tối ưu quy hoạch động bằng các thủ thuật như "Aliens Trick", từ đó giảm thiểu thời gian tính toán, cho ra kết quả nhanh chóng và một số bài toán mang tính ứng dụng thực tế qua thủ thuật "Aliens Trick".

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

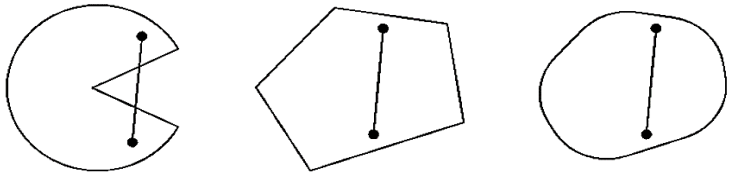
Chúng tôi sẽ cải tiến thuật toán tối ưu hàm lồi bằng cách kết hợp tìm kiếm tam phân và quy hoạch động. Phương pháp này đã được chứng minh là hiệu quả trong việc tối ưu hóa các bài toán, thường được đặt trong khoa học máy tính và các cuộc thi giải thuật. Từ đó, góp phần đề xuất các cách tối ưu các vấn đề thực tế.

# 2 CƠ SỞ LÝ THUYẾT

## 2.1 Hàm lồi và các tính chất

Một tập hợp $X$ trong không gian vectơ là tập lồi nếu với bất kì $a, b \in X$, đoạn thẳng nối $a$ và $b$ cũng thuộc $X$. Điều này có nghĩa là với mọi $\lambda \in [0, 1]$**,** ta có: [1]

$$\lambda.a + (1 - \lambda).b \in X, \forall\ a, b \in X \quad (1)$$

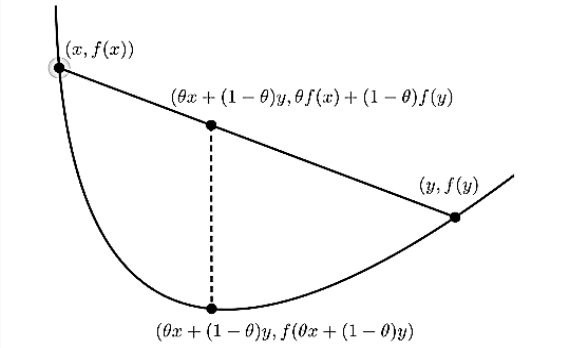


Hình 1: Ví dụ về tập lồi

Tập hợp đầu tiên (bên trái) không phải là tập lồi do tồn tại các đoạn thẳng không nằm trong tập hợp. Hai tập còn lại là tập lồi vì mọi đường thẳng nối hai điểm bất kỳ trong tập đều nằm trong tập hợp đó. Trong toán học, một hàm có giá trị thực định nghĩa một khoảng cách chiều n được gọi là lồi nếu đoạn thẳng ở giữa, nối bất kỳ hai điểm nào của đồ thị của hàm số nằm phía trên đồ thị giữa hai điểm.
Tương tự, một hàm là Hàm lồi nếu tập hợp các điểm ở trên hoặc phía trên đồ thị hàm số (epigraph) là một tập lồi. Nó được định nghĩa trên một tập hợp các điểm trong không gian Euclid bằng cách yêu cầu rằng mỗi đoạn thẳng giữa hai điểm bất kỳ trên tập hợp đó phải nằm hoàn toàn trên hoặc bên trên tập hợp đó.
Hàm đạo hàm cấp hai của một biến đơn là hàm lồi nếu và chỉ nếu đạo hàm cấp hai của nó là không âm trên toàn bộ miền giá trị của nó. Một hàm lồi có thể được biểu diễn như sau:

$$f(\lambda.x + (1-\lambda).y) \leq \lambda.f(x) + (1-\lambda).f(y), \forall\ \lambda \in [0,1] \quad (2)$$

Hình biểu diễn trong hệ trục $(x, f(x))$. Ta thấy hàm $f$ là hàm lồi thì chắc chắn rằng mọi giá trị của $f$ trong đoạn $[x, y]$ luôn nhỏ hơn (nằm dưới) đoạn thẳng nối giữa 2 điểm $(x, f(x))$ và $(y, f(y))$.




Hình 2: Ví dụ về Hàm lồi

Hàm $f$ được gọi lồi chặt nếu bất đẳng thức (2) là dấu $< \lambda \in (0,1)$. Hàm lõm (Concave) được định nghĩa ngược lại với hàm lồi. Một hàm số $f$ là lõm nếu như $-f$ là một hàm lồi. Đường tiếp tuyến: đường tiếp tuyến của một hàm lồi trên một điểm bất kỳ trên đồ thị của nó luôn nằm phía dưới đồ thị của hàm số đó. Đạo hàm: đạo hàm của một hàm lồi là một hàm luôn không giảm trên miền xác định của hàm số ban đầu.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

## 2.2 Hàm lồi trên tập rời rạc

Xét hàm số $f(x)$ trên tập hợp số nguyên dương $\{0,1,\dots,n\}$. Khi đó, để kiểm tra xem hàm này có đơn điệu không, có lồi/lõm không, ta không thể dùng phép tính đạo hàm được, ta sẽ kiểm tra trực tiếp như sau:

- Hàm đơn điệu biến: $\delta(f) = f(n+1) - f(n)$, nếu luôn có $\delta(f) < 0$ thì hàm nghịch biến, còn nếu luôn có $\delta(f) > 0$ thì có hàm đồng biến. Đại lượng này tương tự đạo hàm cấp 1 của hàm $f$.
- Hàm lồi: $\delta\big(\delta(f)\big) = f(n+1) - f(n) - \big(f(n) - f(n-1)\big) = f(n+1) + f(n-1) - 2f(n)$ và nếu $\delta\big(\delta(f)\big) > 0$ thì ta có hàm lồi, còn nếu luôn âm thì ta có hàm lõm. Đại lượng này tương tự đạo hàm cấp 2 của hàm $f$ và thường được dùng trong các bài toán thực tế để kiểm tra tính chất của hàm trên tập rời rạc (do việc tìm cực trị cũng thường xét đối với các hàm này).

Điểm uốn: nếu một hàm số có đạo hàm liên tục trên một miền xác định, thì điểm uốn của hàm số đó phải nằm giữa hai điểm cực trị của hàm số đó. Kết hợp hàm lồi: một tổng có trọng số của các hàm lồi cũng là một hàm lồi. Tích của hàm lồi và một số dương là một hàm lồi [3].

## 2.3 Tìm kiếm tam phân

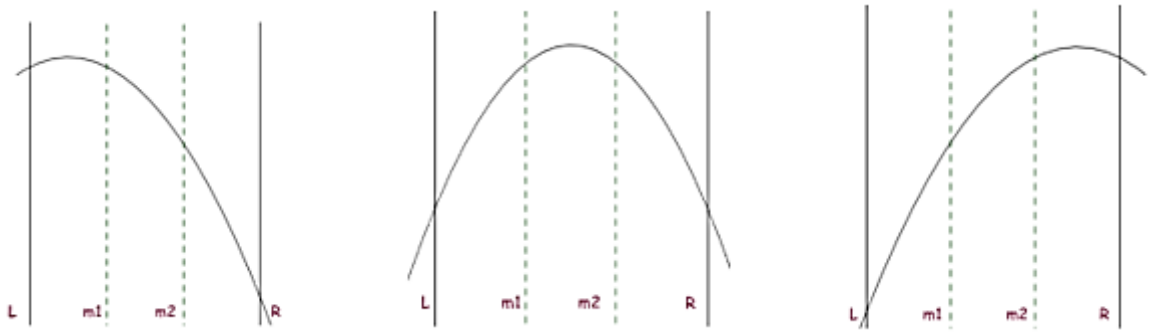
### 2.3.1 Định nghĩa

Tìm kiếm tam phân (Ternary search) là một thuật toán chia để trị (Divide and conquer) được sử dụng để giải quyết bài toán tối ưu hàm đơn điệu (Unimodal function). Một hàm đơn điệu là một hàm mà chỉ có một điểm cực trị duy nhất và giá trị của hàm tăng hoặc giảm đơn điệu từ hai phía của điểm cực trị đó.

Thuật toán tìm kiếm tam phân hoạt động bằng cách chọn hai điểm chốt trong khoảng tìm kiếm ban đầu và so sánh giá trị của hàm tại hai điểm chốt này để loại bỏ một phần của khoảng tìm kiếm không chứa điểm cực trị. Quá trình này được lặp lại cho đến khi khoảng tìm kiếm thu hẹp xuống dưới một ngưỡng cho trước. Điều này được thực hiện bằng cách chia khoảng tìm kiếm thành ba phần bằng cách sử dụng tỷ lệ cắt và còn lại, và xác định khoảng chứa điểm cực trị dựa trên so sánh giá trị của hàm tại các điểm chốt [4].

Cho một hàm $F(x)$ trong đoạn $[l,r]$ thỏa mãn: $F$ tăng chặt tới một cực đại (điểm $H$) rồi giảm chặt. Yêu cầu tìm điểm đạt giá trị lớn nhất (điểm $H$).

Xét hai vị trí $m_1$, $m_2$ trong $[l,r]$ sao cho $l < m_1 < m_2 < r$. Rõ ràng cực trị có thể nằm 1 trong 3 phần:




Hình 3: Các hình ảnh mô tả cho ba trường hợp của thuật toán

- $[l, m_1]$. Khi ta chắc chắn sẽ có $F(m_1) > F(m_2)$
- $[m_1, m_2]$. Ta chưa thể rút ra kết luận gì về $F(m_1)$ và $F(m_2)$
- $[m_2, r]$. Tương tự trường hợp đầu, chắc chắn $F(m_1) < F(m_2)$

Ngược lại, bằng việc so sánh $F(m_1)$ và $F(m_2)$, ta có thể rút ra kết luận như sau:

- Nếu $F(m_1) < F(m_2)$: Ta biết chắc chắn $H$ nằm trong $[m_1, r]$.
- $F(m_1) > F(m_2)$: Ta biết chắc chắn $H$ nằm trong $[l, m_2]$.
- $F(m_1) = F(m_2)$: $H$ nằm trong $[m_1, m_2]$. (Lưu ý: trường hợp này khi xét 2 số thực, thường bỏ qua trường hợp này, để tránh sai số, và do trên thực tế 2 số thực hầu như không bao giờ bằng nhau).

Do đó, dựa vào việc so sánh $F$ ở hai điểm $m_1$, $m_2$ ta có thể thay đổi và giảm không gian tìm kiếm $[l,r]$ xuống một khoảng không gian nhỏ hơn $[l', r']$. Nếu ta chọn:

- $m_1 = l + \frac{(r-l)}{3}$

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

- $m_2 = r - \frac{(r-l)}{3}$

thì sau mỗi lần, độ lớn của đoạn $[l, r]$ giảm xuống còn $\frac{2}{3}$ lần.

Nếu ta lặp đi lặp lại $K$ lần, thì độ lớn của $[l, r]$ sẽ chỉ còn là $(\frac{2}{3})^K$. Ví dụ với $l = -10^9$, $r = 10^9$, ta lặp lại $K = 100$ lần, thì đoạn $[l, r]$ thu về chỉ còn độ dài là

$$(\frac{2}{3})^{100} * (2.10^9) < 5.10^{-9},$$

đủ chính xác với hầu hết các bài. Độ phức tạp là $O(logT)$ với $T$ là độ chính xác mà ta cần thực hiện.

### 2.3.2 So sánh tìm kiếm tam phân và tìm kiếm nhị phân

Giống: Cả hai thuật toán tìm kiếm nhị phân (Binary search) và tìm kiếm tam phân (Ternary search) đều là các thuật toán tìm kiếm trên đoạn số (Range query). Tìm kiếm tam phân cũng tương tự như tìm kiếm nhị phân nhưng tìm kiếm tam phân là cải tiến của tìm kiếm nhị phân.

| | **Ternary Search** | **Binary Search** |
|---|---|---|
| **Phương pháp** | Chia không gian tìm kiếm thành 3 phần bằng cách sử dụng hai chỉ số, tức là $l$ và $r$. Nó so sánh phần tử mục tiêu với hai phần tử ở giữa (một ở gần $l$ và một ở gần $r$) và dựa trên kết quả so sánh đó để xác định phần nào của không gian tìm kiếm chứa phần tử mục tiêu. | Chia không gian tìm kiếm thành hai phần và so sánh phần tử mục tiêu với phần tử ở giữa của mảng con hiện tại. |
| **Độ phức tạp** | $O(log_{3/2}n)$ | $O(log\ n)$ |
| **Hiệu suất** | Tìm kiếm tam phân hoạt động trên các mảng đã được sắp xếp. Nó hiệu quả hơn binary search trong một số trường hợp, nhưng không phải lúc nào cũng như vậy. Tìm kiếm tam phân tạo ra nhiều phân vùng hơn, do đó có thể tốn nhiều thời gian hơn để thực hiện so sánh. Tìm kiếm tam phân có thể tìm kiếm cực trị trên hàm lồi. | Tìm kiếm nhị phân hoạt động tốt trên các mảng đã được sắp xếp. Nó là một thuật toán đơn giản và hiệu quả, nhưng yêu cầu mảng đầu vào đã được sắp xếp trước.<br>Tìm kiếm nhị phân không tìm được cực trị trên hàm Lồi. |

## 2.4 Quy hoạch động

Quy hoạch động là một phương pháp tính toán được sử dụng để giải quyết các vấn đề tối ưu hóa trong lĩnh vực khoa học máy tính và toán học. Phương pháp này tập trung vào việc phân tách các vấn đề lớn thành các vấn đề con nhỏ hơn, từ đó tìm ra giải pháp tối ưu cho từng vấn đề con, sau đó kết hợp chúng lại để tạo ra một giải pháp tối ưu cho toàn bộ vấn đề lớn hơn [5]. Dưới đây là một bài toán minh họa:

*Ví dụ:* Cho một bảng gồm $N$ hàng và $M$ cột. Các hàng và cột được đánh số từ trên xuống dưới và từ trái qua phải. Ô hàng thứ $i$ và cột thứ $j$ được đánh dấu là ô $(i, j)$. Có $K$ vật cản nằm trên một số ô nhất định. Ta đang đứng ở ô $(1, 1)$. Hãy tính toán xem có bao nhiêu cách di chuyển đến ô $(N, M)$ biết chỉ được phép di chuyển sang phải hoặc xuống dưới và không được di chuyển đến ô có vật cản.

Gọi $F[x, y]$ là số cách để đi đến được ô $(x, y)$. Để di chuyển đến ô $(x, y)$ nào đó, ta có thể di chuyển từ ô $(x-1, y)$ xuống dưới hoặc ô $(x, y-1)$ sang phải. Vậy nếu ta biết được số cách di chuyển đến ô $(x-1, y)$ và ô $(x, y-1)$ thì ta sẽ có số cách di chuyển đến ô $(x, y)$ hay

$$F[x, y] = F[x-1, y] + F[x, y-1].$$

Nếu ô $(x, y)$ có vật cản thì $F[x, y] = 0$.

Chẳng hạn ta có bảng 4 hàng 6 cột và hai ô có vật cản là $(3, 3)$, $(2, 5)$ được mô tả dưới ảnh sau:

| Cột \ Hàng | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| 1 | Start | | | | | |
| 2 | | | | | X | |
| 3 | | | X | | | |
| 4 | | | | | | End |

Hình 4: Bảng di chuyển 4 hàng 6 cột trong bài toán trên

Dưới đây là đoạn mã C++ để giải bài trên:

```
//Đánh dấu các ô hàng 1 chỉ có một cách di chuyển
    for (int i = 1; i <= m; i++) {
        if (check(1, i)) F[1, i] = 1;
    }
//Đánh dấu các ô cột 1 chỉ có một cách di chuyển
    for (int i = 1; i <= n; i++) {
        if (check(i, 1)) F[i, 1] = 1;
    }
//Tính số cách di chuyển cho các ô còn lại
    for (int i = 2; i <= n; i++) {
        for (int j = 2; j <= m; j++) {
            if (check(i, j)) F[i, j] += F[i-1, j] + F[i, j-1];
        }
    }
```

Độ phức tạp thuật toán: $O(N * M)$.

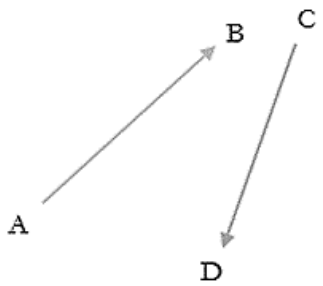
Với check$(i, j)$ sẽ trả về True nếu không có vật cản ở ô $(i, j)$ và False nếu có vật cản. Bài toán trên đã được giải bằng phương pháp quy hoạch động dựa trên phân tích vấn đề lớn bằng vấn đề nhỏ hơn. Cụ thể là tính được số cách di chuyển cho hai ô $(x-1, y)$ và ô $(x, y-1)$ thì ta sẽ có được số cách di chuyển đến ô $(x, y)$. Đồng thời kết quả $F[x, y]$ này cũng có thể được dùng lại như là một vấn đề nhỏ hơn để tính cho các ô khác và xây dựng chung cho kết quả bài toán cuối cùng. Vậy kết quả bài toán trên sẽ là $F[N, M]$.

## 3 ỨNG DỤNG VÀO BÀI TOÁN TỐI ƯU HÀM LỒI

### 3.1 Ứng dụng tìm kiếm tam phân vào tối ưu hàm lồi

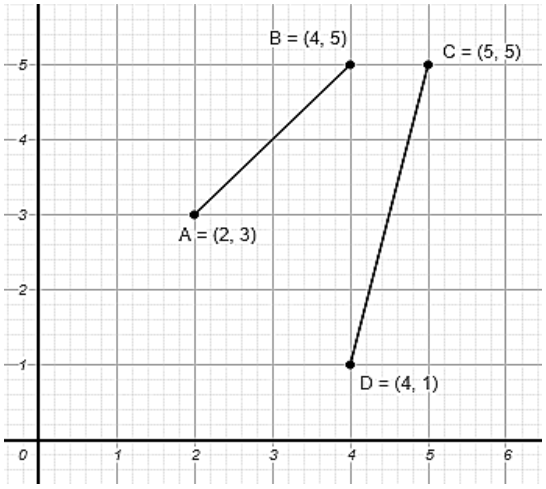
Ta có thể thiết lập hàm mục tiêu cho bài toán tối ưu, sau đó kiểm tra xem nó có lồi/lõm không. Một khi kiểm tra được, ta sử dụng tìm kiếm tam phân như đã mô tả ở trên để có thể xác định được cực trị cần tìm (chú ý rằng nó có thể là số nguyên hoặc số thực).

Ví dụ: Hai người cùng di chuyển cùng nhau, người thứ nhất di chuyển từ $A$ tới $B$ và người còn lại di chuyển từ $C$ tới $D$. Ban đầu, người thứ nhất xuất phát ở vị trí $A$, người thứ hai xuất phát ở vị trí $C$. Họ duy trì tốc độ di chuyển của mình sao cho khi người thứ nhất tới điểm $B$ thì người thứ hai cũng tới điểm $D$. Các điểm $A, B, C, D$ được cho trong không gian hai chiều. Bạn hãy tìm khoảng cách Euclidean nhỏ nhất trong quá trình di chuyển của hai người đó. Bạn sẽ được cho tọa độ $(x, y)$ của các điểm đó và hãy trả lời câu hỏi của bài toán[6].




Ví dụ bài toán có các điểm có tọa độ: $A(2, 3)$, $B(4, 5)$, $C(5, 5)$, $D(4, 1)$.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 5: Ví dụ bài toán ứng dụng tìm kiếm tam phân trong hàm lồi

Giả sử thời gian di chuyển từ lúc bắt đầu đến lúc kết thúc là 1s. Phương trình khoảng cách Euclid giữa hai người trong quá trình di chuyển theo đơn vị thời gian $t$ là:

$$f(x) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} = \sqrt{45t^2 - 42t + 13} \; với \; t \in [0,1] \quad (3)$$

Ta thấy đạo hàm cấp hai $f''(x)$ luôn $> 0$ với $t \in [0,1]$ nên hàm khoảng cách giữa hai người di chuyển trên là một Hàm lồi. Ta áp dụng tìm kiếm tam phân để tìm $min(f(x))$ trên:

```
def calcDistance(m):
    return math.sqrt(45 * m * m -42 * m + 13)
lo, hi, ans, time = 0, 1, 1e9, 1
while (lo < hi):
    m1 = lo + (hi - lo)/3
    m2 = hi - (hi - lo)/3
    if calcDistance(m1) < calcDistance(m2):
        hi = m2
        cur = calcDistance(m1)
    else:
        lo = m1
        cur = calcDistance(m2)
    if abs(cur - ans) < 1e-6:
        break
    ans = cur
    print(str(time) + ' ' + str(lo) + ' ' + str(hi) + ' ' + str(ans))
    time += 1
```

| **lo:hi** | **Lần 1** | **Lần 2** | **Lần 3** | **Lần 4** | **…** | **Lần 15** | **Lần 16** |
|---|---|---|---|---|---|---|---|
| **calcDistance(m1):calcDistance(m2)** | 0:1 | 0:0.666667 | 0.222222:0.666667 | 0.37037:0.666667 | … | 0.466028:0.468311 | 0.466028:0.46755 |
| **ans** | 2 | 1.1795055 | 1.822358 | 1.788931 | … | 1.788858 | 1.788855 |

Với bài toán trên, ta tìm kiếm kết quả bằng tìm kiếm tam phân 16 lần để có được khoảng cách chính xác 6 chữ số thập phân. Đồng thời, trong mỗi lần tính toán, phạm vi tìm kiếm cũng được rút ngắn lại $\frac{1}{3}$ so với phạm vi trước đó nên ta có thể nhanh chóng tìm được cực tiểu của hàm lồi.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

| | Lần 1 | Lần 2 | Lần 3 | Lần 4 | … | Lần 15 | Lần 16 |
|---|---|---|---|---|---|---|---|
| **lo:hi** | 0:1 | 0:0.666667 | 0.222222:0.666667 | 0.37037:0.666667 | … | 0.466028:0.468311 | 0.466028:0.46755 |
| **Phạm vi [lo:hi] so với khoảng cách tìm kiếm ban đầu** | 100% | 66.67% | 44.45% | 29.63% | … | 0.34% | 0.23% |

## 3.2 Ternary Search Tree

Trong khoa học máy tính, cây tìm kiếm tam phân (TST) là một loại cây Trie (đôi khi được gọi là cây tiền tố) trong đó các nút được sắp xếp theo một cách tương tự như cây tìm kiếm nhị phân (Binary Search Tree), nhưng có tối đa ba nút con thay vì hai như giới hạn của cây nhị phân. Giống như các cây tiền tố khác, TST có thể được sử dụng như một cấu trúc ánh xạ kết hợp với khả năng tìm kiếm chuỗi tăng dần. Tuy nhiên, TST hiệu quả về không gian hơn so với cây tiền tố tiêu chuẩn, nhưng tốc độ tìm kiếm chậm hơn. Các ứng dụng phổ biến của cây tìm kiếm tam phân bao gồm kiểm tra chính tả và tự động hoàn chỉnh [7].

TST được xây dựng dựa trên các ký tự của từ hoặc chuỗi cần lưu trữ. Mỗi nút trong cây đại diện cho một ký tự và có ba con trỏ: con trỏ sang cây con có các ký tự nhỏ hơn, con trỏ sang cây có con có các ký tự lớn hơn và con trỏ sang cây con có cùng ký tự.

Quá trình xây dựng cây tìm kiếm tam phân bắt đầu từ gốc. Với mỗi từ hoặc chuỗi được chèn vào cây, chúng ta so sánh ký tự hiện tại với ký tự của nút hiện tại. Nếu ký tự nhỏ hơn, chúng ta di chuyển đến cây con có các ký tự nhỏ hơn; nếu ký tự lớn hơn, chúng ta di chuyển đến cây con có các ký tự lớn hơn; nếu ký tự bằng, chúng ta di chuyển đến cây con có cùng ký tự. Nếu cây con không tồn tại, chúng ta tạo ra một cây con mới cho ký tự đó. Quá trình này được lặp lại cho tất cả các ký tự trong từ hoặc chuỗi.

Hình dưới đây cho thấy một cây tìm kiếm tam phân với các chuỗi
"cute", "cup", "at", "as", "he", "us" và "i".




Hình 6: Ví dụ cây Ternary Search

Với cây tìm kiếm tam phân, việc tìm kiếm một từ hoặc chuỗi cũng tương tự như quá trình xây dựng cây. Chúng ta so sánh ký tự hiện tại với ký tự của nút hiện tại và di chuyển tương ứng. Nếu ký tự không tồn tại trong cây hoặc ta đã đi qua tất cả các ký tự mà không tìm thấy từ hoặc chuỗi cần tìm, thì từ hoặc chuỗi đó không tồn tại trong cây.

Cây tìm kiếm tam phân cung cấp hiệu suất tìm kiếm tốt hơn so với cây Trie vì nó không cần lưu trữ các con trỏ rỗng cho các ký tự không tồn tại trong từ điển. Đồng thời, nó giữ được sự tổ chức phân cấp của các từ hoặc chuỗi. Tuy nhiên, cây tìm kiếm tam phân có thể sử dụng nhiều không gian hơn so với cây Trie vì nó cần lưu trữ các con trỏ cho các cây con [8].

## 3.3 Quy hoạch động bao lồi

Quy hoạch động bao lồi (Kỹ thuật bao lồi) là một lớp của thuật toán quy hoạch động. Vấn đề bao gồm duy trì, tức là theo dõi bao lồi đối với dữ liệu đầu vào thay đổi động, tức là khi các yếu tố dữ liệu đầu vào có thể được chèn, xóa hoặc sửa đổi. Thuật toán này chỉ có thể áp dụng khi các điều kiện nhất định được đáp ứng. Kỹ thuật bao lồi là kĩ thuật (hoặc là cấu trúc dữ liệu) dùng để xác định hiệu quả, có tiền xử lý, cực trị của một tập các hàm tuyến tính tại một giá trị của biến độc lập. Mặc dù tên gọi giống nhưng kỹ thuật này lại khá khác biệt so với thuật toán bao lồi [9] của hình học tính toán [10].

Đây là một ví dụ bài toán Quy hoạch động bao lồi: The Fair Nut and Rectangles [11].

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Tóm tắt đề: Cho $n$ mảnh đất hình chữ nhật, tọa độ 4 đỉnh của mảnh đất hình chữ nhật thứ $i$ là $(0,0)$, $(x_i,0)$, $(x_i,y_i)$, $(0,y_i)$. Mỗi hình chữ nhật có kèm theo một số nguyên dương $a_i$, chi phí mua khi dùng mảnh đất này. Không có bất kỳ hình chữ nhật nào nằm hoàn toàn bên trong một hình chữ nhật khác. Bạn hãy chọn ra một tập hợp mảnh đất trong $n$ mảnh đất đó sao cho phần diện tích mà tập hình chữ nhật đó đè lên trừ đi tổng giá trị chi phí $a_i$ trong tập bạn chọn là lớn nhất. Trong đó $1 <= n <= 10^6$. Mỗi hình chữ nhật được cung cấp 3 số nguyên $x_i$, $y_i$, $a_i$ $(1 <= x_i, y_i <= 10^9, a_i <= x_i * y_i)$. Hãy tìm ra kết quả lớn nhất đó.
Giả sử ta có thông tin các mảnh đất được cho như sau:

4 4 8, 1 5 0, 5 2 10.

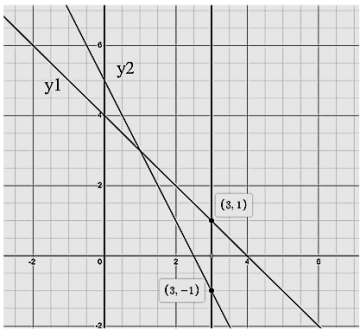
Đáp số bài toán sẽ là 9 (Chọn hình chữ nhật 1 và 2, diện tích tập đó - tổng $a_i$ của hình 1 và 2: 17 - 8 = 9).
*Phân tích.* Sắp xếp $n$ hình chữ nhật theo trình tự tăng dần của $x_i$. Do không có hình chữ nhật nào lồng nhau nên giá trị của $y_i$ sẽ giảm dần theo trình tự trên. Gọi $dp[i]$ là giá trị lớn nhất đạt được từ việc chọn một tập hợp hình chữ nhật trong $i$ hình chữ nhật đầu tiên và tập hợp này có hình chữ nhật thứ $i$.
Ta có thể tính được:

$$dp[i] = x_i * y_i - a_i + max(-x * y_i + dp[j]), (j < i) \quad (4)$$

Trong đó, $x_i * y_i - a_i$ là giá trị bắt buộc lấy hình chữ nhật thứ $i$ để tính cho $dp[i]$, $dp[j]$ là giá trị của một tập hợp hình chữ nhật trước đó đã tính và trừ $x_i * y_i$ là phần chung của hình chữ nhật hiện tại với tập hợp hình chữ nhật trong $dp[j]$ đó.
Với công thức này, ta có thể giải quyết nhanh với đoạn mã sau với độ phức tạp là $O(n^2)$

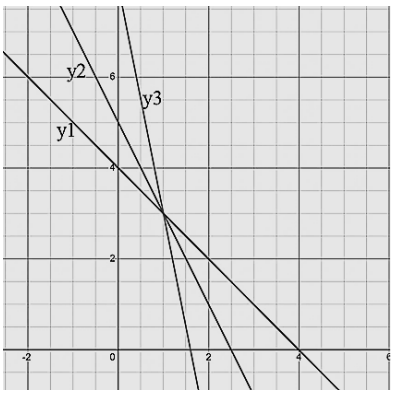
```
for (int i = 1; i <= n; i++) {
        for (int j = 0; j < i; j++) {
            dp[i] = max(dp[i], a[i].x * a[i].y - a[i].a + (- a[j].x * a[i].y + dp[j]));
        }
}
```

Kết quả sẽ là giá trị lớn nhất của mảng dp.
Nhưng nếu ta áp dụng Quy hoạch động bao lồi cho bài này, độ phức tạp sẽ giảm xuống còn O(n). Vì để tính được giá trị lớn nhất cho dp[i] thì ngoài $a[i].x * a[i].y - a[i].a$ cố định, phần $-a[j].x * a[i].y + dp[j]$ là lớn nhất. Giá trị này có dạng đường thẳng $y = m.x + c$. Ta sẽ xây dựng một tập hợp các đường thẳng này bằng Queue (thứ tự theo trình tự push vào, do đó hệ số gốc sẽ giảm dần) để tính giá trị dp[i] tiếp theo chỉ dựa vào $a[i].y$ để tìm giá trị lớn nhất. Đồng thời trong quá trình tính toán, ta phải duy trì được tập hợp này hiệu quả. Nó có nghĩa là khi tính toán cho giá trị dp[i] nào đó hay thêm mới một đường thẳng, ta phải loại bỏ những đường thẳng $y = m.x + c$ không còn ý nghĩa. Xét dữ liệu sau:
1 6 2, 2 4 3, 5 3 8, 6 2 4
Ta đang tính cho dp[3], lúc này tập đường thẳng ta có $y_1 = -x + 4$, $y_2 = -2x + 5$. Kiểm tra các đường dư thừa trước khi tính toán: Vị trí cắt nhau của hai đường đầu trong Queue có tọa độ x trong đồ thị nhỏ hơn giá trị $a[3].y$ nên không loại bỏ đường thẳng nào. Nếu có loại bỏ đường đầu tiên của Queue đến khi thỏa mãn. Với giá trị $a[3].y = 3$ ta có giá trị lớn nhất tại y đầu tiên trong Queue, $y_1 = 1$ suy ra $dp[3] = 8$.




Hình 7: Đường thẳng $y_1$ và $y_2$

Sau đó ta thêm vào Queue đường thẳng $y_3 = -5x + dp[3] = -5x + 8$ và kiểm tra loại bỏ các đường thẳng dư thừa trong tập hợp.




Hình 8: Đường thẳng $y_1$, $y_2$, $y_3$

Ta thấy, giá trị lớn nhất khi thay x thuộc $[-\infty, 1]$ là $y_3$, x thuộc $[1, +\infty]$ là $y_1$. Cho nên $y_2$ là dư thừa và ta loại nó trong Queue. Vì mỗi kết quả dp[i] bài này đều được tính toán dựa vào đường thẳng đầu tiên của Queue. Các đường thẳng thêm và bị xóa không quá 1 lần cho nên độ phức tạp cho bài trên là $O(n)$.

### 3.4 Aliens Trick

Trong phần này, chúng tôi sẽ giới thiệu một kỹ thuật tối ưu hóa quy hoạch động rất hữu ích thường được bắt gặp trong các bài toán quy hoạch động khó với độ phức tạp thời gian cao. Kỹ thuật này được sử dụng để giảm kích thước không gian tìm kiếm trong các cấu hình quy hoạch động, bằng cách khai thác tính chất của hàm lồi và một số hàm chi phí. Phương pháp này còn được giới thiệu thông qua bài toán "Người ngoài hành tinh" IOI 2016 (Olympic Tin học quốc tế) dưới cái tên "Aliens Trick".

Aliens Trick là một kỹ thuật tối ưu quy hoạch động được ứng dụng rộng rãi ở Trung Quốc, với tên gọi là WQS-Binary-Search, WQS xuất phát từ Qingshi Wang. Tuy nhiên, kỹ thuật này chính thức được cộng đồng quốc tế gọi tên là "Parameter search". Trong lập trình thi đấu, kỹ thuật này được biết đến nhiều hơn với cái tên Aliens Trick, do được sử dụng để giải quyết vấn đề Người ngoài hành tinh IOI năm 2016. Tuy nhiên, nó không chỉ là tối ưu hóa DP, nó rất hữu ích để tối ưu hóa nhiều thứ khác [12].

Để hiểu rõ hơn về Aliens Trick, chúng ta có thể bắt đầu với một bài toán quy hoạch động đơn giản. Ban đầu, độ phức tạp của bài toán là $O(N^4)$, tuy nhiên, khi áp dụng kỹ thuật này, độ phức tạp giảm xuống chỉ còn $O(N * logVal)$, giúp tối ưu hóa thời gian xử lý và hiệu quả giải quyết bài toán.

#### 3.4.1 Mô tả bài toán

Vấn đề sau được tham khảo từ bài toán "Người ngoài hành tinh" IOI 2016 [14] đề bài được mô tả như sau: Bạn được cho một mảng v chứa các số nguyên (có thể âm) có độ dài N ($N \leq 10^5$) và một số K ($K \leq N$). Hãy chọn tối đa K phân đoạn không chồng lấn của dãy số ban đầu sao cho tổng các phần tử trong các phân đoạn đó là lớn nhất có thể. Từ bài toán trên ta có thể dễ dàng suy ra được công thức quy hoạch động đơn giản như sau:

$$dp[n][k] = \max \begin{Bmatrix} n-1 \\ \mathrm{dp[n-1][k]}, \ max_{i=k} \{dp[i-1][k-1] + \sum_{k=i}^{n} v[k]\} \end{Bmatrix} \qquad (5)$$

Trong đó dp[n][k] ($n \leq N, k \leq K$) đại diện cho tổng tốt nhất khi chỉ xem xét n phần tử đầu tiên và sử dụng đúng k phân đoạn. Việc triển khai trực tiếp công thức trên một cách mù quáng có thể dẫn đến độ phức tạp thời gian là $O(N^4)$ nếu như K có kích thước tương đương với N. Tuy nhiên ta vẫn có hai cách để tối ưu khi thực hiện công thức trên, đầu tiên ta có thể sử dụng tổng tiền tố để tiền xử lý $\sum_{k=1}^{n} v[k]$ như vậy mỗi lần cần lấy tổng v[k] ta chỉ mất $O(1)$ và giảm thời gian khi thực hiện công thức trên xuống $O(N^3)$ sau đó ta lại kết hợp với kỹ thuật bao lồi như đã trình bày ở mục 3.3 bằng các đảo dấu công thức trên và sử dụng min để tìm kết quả tối ưu qua đó công thức trên có thể được thực hiện trong $O(N^2)$.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Tuy nhiên, kết quả này vẫn chưa đạt đến tốc độ đủ nhanh, chúng ta cần phải tối ưu với độ phức tạp là $O(N\log^2 N)$ hoặc $O(N\log N)$, nếu muốn vượt qua độ phức tạp $O(N^2)$ như đã đề cập ở trên. Vì vậy đây chính là lúc ta cần phải áp dụng Aliens Trick.

Bí quyết đằng sau của "Aliens Trick" là chúng ta có thể thêm một chi phí phạt được ký hiệu là λ cho mỗi phân đoạn được chọn. Nếu $\lambda = 0$ giải pháp tối ưu sẽ là chọn một phân đoạn sao cho mỗi phần tử đều dương, nhưng bằng cách tăng giá trị của chi phí phạt λ giải pháp tối ưu sẽ liên quan đến việc chọn ít phân đoạn hơn. Bây giờ, chúng ta chỉ cần tìm một giá trị λ mà nó cho phép chúng ta chọn nhiều phân đoạn nhất có thể, nhưng vẫn đảm bảo ít hơn K.

Ta có thể định nghĩa các ký hiệu như sau, λ là chi phí mà chúng ta sẽ bỏ ra khi thêm một phân đoạn mới và việc tăng λ sẽ làm giảm số phân đoạn trong một giải pháp tối ưu hoặc giữ nguyên, nhưng không bao giờ làm tăng nó. Điều này cho thấy ta có thể sử dụng tìm kiếm nhị phân giá trị nhỏ nhất của λ mà tạo ra một giải pháp có ít hơn K phân đoạn. Từ đó ta có được công thức quy hoạch động mới như sau:

$$dp_\lambda[n] = max\left\{dp_\lambda[n-1],\ \max_{i=1}^{n-1}\{(\sum_{k=i}^{n} v_k) + dp_\lambda[i-1] - \lambda\}\right\} \quad (6)$$

Trong đó $dp_\lambda[n]$ là kết quả tối ưu cho tiền tố có độ dài n của mảng v ban đầu của chúng ta, trong đó việc thêm một mảng con sẽ gây ra một chi phí λ . Bên việc sử dụng mảng quy hoạch động (dp) ta sẽ lưu trữ thêm một mảng phụ trợ khác là $cnt_\lambda[n]$ để lưu trữ số lượng mảng con đã sử dụng cho kết quả tối ưu $dp_\lambda[n]$. Nếu chúng ta đang sử dụng quá nhiều mảng con ($> K$), chúng ta nên tăng λ để không khuyến khích việc mở thêm các mảng con mới và nếu chúng ta đang sử dụng quá ít ($< K$), chúng ta nên giảm λ sau cùng ta thực hiện tìm kiếm nhị phân giá trị λ qua đoạn mã như sau:

```
minbound = -1e18, maxbound = 1e18
while maxbound - minbound > 1e-6:
    λ = (maxbound - minbound)/2
    #tính toán dp và  aux cho λ
    if cnt[n] <= k:
        minbound = λ
    else:
        maxbound = λ
#tính toán dp và cnt cho giá trị λ cuối cùng
return dp[n] + cnt[n] * λ # lưu ý rằng nếu có ít hơn k giá trị dương thì
cnt[n] < k
```

### 3.4.2 Cơ sở toán học của bài toán

Đối với vấn đề ban đầu của chúng ta, việc tăng λ không bao giờ làm tăng số lượng mảng con được sử dụng có lẽ là một sự thật rất dễ hiểu, nhưng chúng ta muốn tìm một chứng minh cụ thể cho điều này và đặt ra một tiêu chí tổng quát để sử dụng phương pháp tối ưu hóa đỉnh trong việc giảm số chiều quy hoạch động. Tiêu chí này có tính chất lồi (hoặc lõm) theo một cách nào đó. Hãy ký hiệu ans[k] là kết quả cho vấn đề, sử dụng chính xác k mảng con. Quan sát quan trọng để chứng minh tính đúng đắn của phương pháp giải của chúng ta là dãy ans[k] có tính chất lõm, nghĩa là ans[k] - ans[k-1] < ans[k-1] - ans[k-2]. Một cách hiểu tự nhiên hơn và cách mà hầu hết mọi người cảm nhận tính lõm/lồi là nếu ta có k mảng con và thêm một mảng con nữa, nó sẽ giúp ích hơn so với khi tôi có k+1 mảng con và thêm một mảng con nữa, tức là: càng có nhiều mảng con trong giải pháp, thì giá trị được cải thiện càng ít đi. Bây giờ, chúng ta hãy xem làm thế nào tính chất lõm này giúp chúng ta chứng minh tính chính xác của giải pháp của chúng ta.

Giả sử λ = 0. Giải pháp của chúng ta chỉ đơn giản là tìm giá trị lớn nhất toàn cục của dãy lõm, được ký hiệu là ans[p]'. Lưu ý rằng bất kể giá trị của λ là gì, tính lõm của dãy sẽ không thay đổi. Hãy tạm dừng chú ý về dãy lõm và tập trung vào hàm lõm. Ví dụ $f(x) = \lambda x - x^2$ là một ví dụ tốt ($f''(x) > 0$). Bằng cách thay đổi λ, chúng ta có thể di chuyển đỉnh của hàm đến bất kỳ tọa độ x nào và hàm vẫn duy trì tính lõm. Bây giờ, hãy quay trở lại với dãy "rời rạc" hơn của chúng ta. Chúng ta đã có một thuật toán để tìm p và ans[p] sao cho ans[p] là giá trị lớn nhất trong dãy, nhưng chúng ta không quan tâm đến giá trị lớn nhất của dãy, chúng ta quan tâm đến ans[k] với một giá trị k cụ thể. Vì vậy, chúng ta có thể buộc k trở thành giá trị lớn nhất của dãy bằng cách thêm một hàm tuyến tính vào dãy của chúng ta (ans[k] $->$ ans[k] + λk), tương tự như việc thay đổi đỉnh của hàm liên tục, đó chính là cách ta đã thực hiện trong giải pháp của chúng ta.

Thuật toán sẽ trả về giá trị lớn nhất của dãy tại k với giá trị ans[k] + λk và chúng ta chỉ cần trừ đi λk để nhận được giá trị mong muốn: ans[k].
Về tiêu chí tổng quát, bạn có thể đã đoán được: nếu (ans[k]) 1 <= k <= n là dãy các giá trị đáp án cho các giá trị k đã cho, thì dãy đó phải có tính lõm hoặc lồi, tức là:

$$\forall i \in (1 \ldots n), ans[i] - ans[i-1] \leq ans[i+1] - ans[i] \quad (7)$$

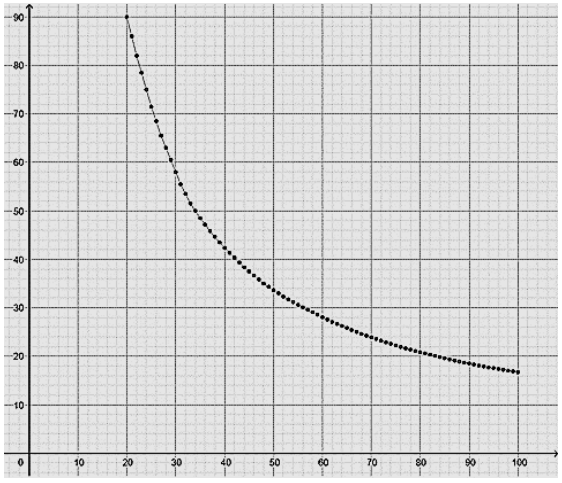
hoặc

$$\forall i \in (1 \ldots n), ans[i] - ans[i-1] \geq ans[i+1] - ans[i] \quad (8)$$

### 3.4.3 Aliens Trick với bài toán thực tế

Aliens Trick là một kỹ thuật dùng để tối ưu, giảm độ phức tạp của bài toán xuống bằng cách khai thác các tính chất của Hàm lồi và các hàm chi phí. Qua đó nhanh chóng tìm được kết quả của bài toán, vấn đề một cách hiệu quả hơn.
Không chỉ trong lĩnh vực khoa học máy tính, nghiên cứu, giải thuật mà bên ngoài thực tế cũng có rất nhiều vấn đề tìm lời giải đáp. Nhóm sẽ đưa ra một vấn đề có thể gặp rất nhiều trong thực tế dưới đây để áp dụng Aliens Trick tìm ra lời giải đáp.
Đặt vấn đề: Trong sản xuất dây chuyền, việc tạo một luồng công việc tối ưu và hiệu quả là điều mà các nhà làm sản xuất mong muốn. Do một số sản phẩm có tính chất đặc thù nên các sản phẩm ấy phải được sản xuất theo thứ tự của các giai đoạn nhà máy, phải sản xuất xong giai đoạn trước mới đến được giai đoạn tiếp theo. Thời gian xử lý trung bình cho 1 sản phẩm mỗi giai đoạn là khác nhau, nên lượng nhân công phân bổ vào từng giai đoạn cũng khác nhau. Mỗi giai đoạn có tính chất công việc cao nên cần nhân công có kinh nghiệm riêng về giai đoạn đó, do đó người làm sản xuất phải tuyển số lượng người có kinh nghiệm theo từng giai đoạn. Giả sử người làm sản xuất đó chưa có kinh nghiệm vận hành chu trình sản xuất, nên không biết tính toán số lượng người hợp lý cho từng giai đoạn, sao cho tổng quá trình sản xuất hiệu quả, và thời gian sản xuất sản phẩm nhanh nhất có thể với số lượng nhân công giới hạn.
Cụ thể: Một công ty X sản xuất một sản phẩm P. Công ty này muốn tạo ra các khối hàng có 1000 sản phẩm để đóng gói chuyển đi phân phối. Quá trình sản xuất có N giai đoạn, với 1000 sản phẩm này thì thời gian trung bình sản xuất của N giai đoạn sẽ là, A[1], A[2],...A[N] giờ cho 1000 sản phẩm. Công ty này có số lượng nhân công là K người. Hãy tìm thời gian ngắn nhất công ty này sản xuất lô hàng 1000 sản phẩm theo thứ tự các giai đoạn, số lượng nhân công hợp lý cho từng giai đoạn sao cho hiệu quả.
Cách giải tham lam: Phân bổ ít nhất mỗi giai đoạn 1 nhân công, vậy còn dư K - N nhân công. Ta tính toán để phân bổ K - N nhân công này vào N giai đoạn cho rút ngắn thời gian sản xuất. Gọi $C[i]$ là số lượng nhân công phân bổ cho giai đoạn i, thời gian làm xong giai đoạn i đó là $\frac{A[i]}{C[i]}$ giờ. Nếu ta tăng số lượng nhân công tại một vị trí i nào đó lên 1. Thì thời gian hoàn thành giai đoạn i sẽ giảm xuống: $\frac{A[i]}{C[i]} - \frac{A[i]}{C[i]+1} = \frac{A[i]}{C[i](C[i]+1)}$ giờ. Vậy đối với K - N nhân công này, ta sẽ chọn ra một giai đoạn nào j nào đó $(1 \leq j \leq N)$ sao cho $\frac{A[j]}{C[j](C[j]+1)}$ là lớn nhất để tăng nhân công lên 1. Qua đó rút ngắn thời gian giai đoạn j cũng như cho toàn bộ quá trình sản xuất. Nhưng vì K - N nhân công khá lớn nên việc thêm từng người nhân công cho từng giai đoạn để xem thấy sự hiệu quả là khá lâu. Vì vậy ta cần phải tối ưu cách làm tham lam này để tìm kết quả nhanh hơn.
Một số tính chất bài toán: Gọi $f(i)$ $(i \in N)$ là thời gian ngắn nhất để hoàn thành toàn bộ quá trình sản xuất 1000 sản phẩm có i nhân công. Ta có $m_i$ là hệ số góc tại vị trí $x = i$, $m_i = f(i-1) - f(i)$ (hệ số góc $m_i$ được tính ngược lại, nó sẽ là $f(i) - f(i-1)$). Hay nói cách khác $m_i$ sẽ là lượng thời gian giảm khi tăng i người thành $i + 1$ người trong việc tính toán thời gian toàn bộ quá trình. Rõ ràng ta thấy, f là hàm có giá trị giảm dần, $f(i) > f(i+1)$. Hay $f(i) - f(i+1) \leq f(i-1) - f(i)$
=> $m_{i+1} \leq m_i$ (từ bên trái sang phải theo x thì độ dốc ít hơn phía trước) hay nói cách khác, f là hàm lồi.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh



Hình 9: Mô tả hàm f

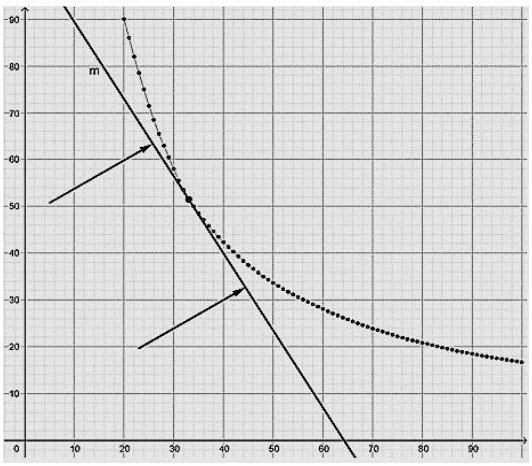
Nếu công ty đó có tuyển rất nhiều người thì việc hoàn thành các giai đoạn rất dễ dàng. Nhưng nếu ta phải trả một chi phí nào đó cho mỗi người nhân công thì công ty sẽ không sử dụng hết những người đó. Gọi $\mathrm{G(m)}$ là thời gian hoàn thành toàn bộ quá trình nhanh nhất khi mỗi nhân công nghỉ ngơi m giờ trong toàn bộ quá trình làm và $\mathrm{G_c(m)}$ là số nhân công tối đa để hoàn thành công việc để đạt được trong thời gian tối ưu $\mathrm{G(m)}$. Vậy việc thêm một nhân công cho giai đoạn thứ i nào đó được thực hiện khi và chỉ khi

$$\frac{A[i]}{C[i](C[i]+1)} \geq m < C^2 + C - \frac{A[i]}{m} \leq 0, \text{ vậy } C[i] = \left[\frac{\sqrt{1+\frac{4*A[i]}{m}}-1}{2}\right] + 1.$$

Với một hệ số góc m (như trên đã đề cập, đây là hệ số góc được tính ngược lại) và hai hàm G, $\mathrm{G_c}$ trên đã cho thì đường thẳng hệ số góc m sẽ chạm bên dưới Hàm lồi f tại một điểm và điểm đó có tọa độ là $(\mathrm{G_c(m)}, \mathrm{f(G_c(m))})$.

Với hàm G không tăng, ta sẽ đảm bảo được khi giảm giá trị m đến một giá trị nào đó ta sẽ đạt được $\mathrm{G_c(m)} = \mathrm{K}$. Vậy ta thực hiện tìm kiếm nhị nhân để tìm ra được giá trị m sao cho đạt được $\mathrm{G_c(m)} = \mathrm{K}$. Ta có, $\mathrm{f(G_c(m))} = \mathrm{G(m)} - \mathrm{m} * \mathrm{G_c(m)}$ với $\mathrm{G_c(m)} = \mathrm{K}$ thì ta có kết quả thời gian ngắn nhất cho toàn bộ quá trình sản xuất là $\mathrm{f(K)} = \mathrm{G(m)} - \mathrm{m} * \mathrm{K}$.

Từ $m$, số người nhân công xấp xỉ cho lần i là $C[i] = \left[\frac{\sqrt{1+\frac{4*A[i]}{m}}-1}{2}\right] + 1.$




Hình 10: Đường thẳng m giao điểm với hàm f

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Từ cách giải tham lam có độ phức tạp O(K logN) xuống còn O(logT + N). Với T là độ chính xác cần khi ta tìm kiếm nhị phân ra giá trị m. Vấn đề trên được viết lại để gần hơn với các vấn đề thực tế có thể giải quyết, vấn đề gốc được lấy từ bài Building a Tall Barn USACO 2017 January Contest [15] [16].

# 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 4.1 Kết luận

Tóm lại, trong bài báo này nhóm nghiên cứu đã giới thiệu một phương pháp cải tiến cho thuật toán tối ưu hàm lồi bằng cách sử dụng tìm kiếm tam phân. Thuật toán tìm kiếm tam phân sẽ xác định vị trí tối ưu của hàm lồi trên một khoảng xác định đồng thời giảm số lượng phép tính cần thực hiện. Kết quả của thực nghiệm cho thấy phương pháp cải tiến này mang lại hiệu suất tốt hơn so với các phương pháp truyền thống trước đây cho việc tối ưu hàm lồi. Bên cạnh đó nhóm còn giới thiệu hai kỹ thuật khác để tối ưu hóa các bài toán quy hoạch động điển hình là "Aliens Trick" và "Quy hoạch động bao lồi". Điều này đặc biệt hữu ích trong các lĩnh vực như tối ưu hóa hàm mục tiêu, phân tích dữ liệu và lập kế hoạch quyết định.

## 4.2 Hướng phát triển

Tuy nhiên, trong bài báo không đề cập đến các hạn chế và giới hạn của các phương pháp cải tiến. Việc nghiên cứu và so sánh với các thuật toán khác cũng có thể làm rõ thêm về hiệu suất áp dụng của các thuật toán trên trong các bài toán cụ thể. Vì vậy, cần tiếp tục nghiên cứu để có thể đánh giá sự ổn định và độ chính xác của thuật toán trong các tình huống đa dạng và phức tạp hơn.

# TÀI LIỆU THAM KHẢO

[1] Stephen P. Boyd and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2004, pp. 37-39.

[2] Hazewinkel and Michiez. *Convex function (of a real variable)*, Encyclopedia of Mathematics, 2001.

[3] Stephen P. Boyd and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2004.

[4] Tìm kiếm tam phân - Ternary Search Thư viện VNOI, [Online] https://vnoi.info/wiki/translate/emaxx/Tim-kiem-tam-phan-Ternary-Search.md

[5] Cormen, T. H; Leiserson, C. E.; Rivest, R. L.; Stein, C. (2001), Introduction to Algorithms (2nd ed.), MIT Press & McGraw–Hill pp.334.

[6] Closest Distance [Online], https://lightoj.com/problem/closest-distance

[7] Jon Bentley and Bob Sedgewick, *Ternary Search Trees*, Dr.Dobb's, 1998.

[8] Akshat Anand, Application of Tries and Ternary Search trees 2020. [Online] Available at: https://medium-com.translate.goog/data-science-community-srm/application-of-tries-and-ternary-search-trees-d38663c5a550?_x_tr_sl=en&_x_tr_tl=vi&_x_tr_hl=vi&_x_tr_pto=sc

[9] Bao lồi (Convex Hull), [Online].https://vnoi.info/wiki/translate/wcipeg/Convex-Hull

[10] Kỹ thuật bao lồi (Convex Hull Trick) [Online].
https://vnoi.info/wiki/translate/wcipeg/Convex-Hull-Trick.md

[11] E. The Fair Nut and Rectangles https://codeforces.com/contest/1083/problem/E

[12] J. Bergstra, R. Bardenet, Y. Bengio and B. Kégl , "Algorithms for Hyper-Parameter Optimization" *25th Annual Conference on Neural Information Processing Systems (NIPS 2011)*, Dec 2011,pp. 1–5.

[13 ] Website ioinformatics.org, ioi2016problem6. [Online].ioi2016problem6.pdf (ioinformatics.org)

[14]Website Wikipedia (achive.org), *Proof by intimidation*. [Online]. Proof by intimidation - Wikipedia (archive.org)

[15] USACO 2017 January Contest, Platinum Problem 2. Building a Tall Barn

[16] My Take on Aliens' Trick https://mamnoonsiam.github.io/posts/attack-on-aliens.html

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F005**

# CÁC MÔ HÌNH CHIA SẺ DỰA TRÊN MÃ HÓA NGƯỠNG ỨNG DỤNG VÀO BÀI TOÁN CHỮ KÝ SỐ

TRẦN MINH LONG[1*], PHAN TÔN LỘC NGUYÊN[1], NGUYỄN MINH PHÚC[1], TRẦN NGỌC HUYỀN[1], ĐÀO THỊ MỸ LINH[1], NGUYỄN THỊ TUYẾT LAN[1], NGUYỄN QUANG HUY[1], DƯƠNG HOÀNG HUY[1], NGUYỄN HUY HOÀNG[1], LÊ PHÚC LỮ[1], NGUYỄN HỮU TÌNH[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
**minhlong2002@gmail.com*

**Tóm tắt.** Mã hóa, bảo mật luôn đóng vai trò quan trọng trong nhiều lĩnh vực của đời sống, đặc biệt là khi một dữ liệu nhạy cảm cần chia sẻ trong một nhóm người. Bài toán đặt ra là sau khi mã hóa sử dụng khóa công khai, để có thể giải mã, cần có sự tham gia của một số lượng đủ lớn của những người trong nhóm (mã hóa ngưỡng). Nhóm nghiên cứu sẽ tìm hiểu ý nghĩa của bài toán, cách thức hoạt động của các mô hình có liên quan như: Secret Shamir Sharing, Blakley, Định lý Số dư Trung Hoa (CRT) và nền tảng Toán khác. Nhóm sẽ tiến hành cài đặt và đánh giá, so sánh giữa các schemes nêu trên, đồng thời kết hợp thêm mã hóa bất đối xứng, từ đó tìm cách ứng dụng vào bài toán chữ ký số.

**Từ khóa.** Mã hóa ngưỡng, Secret Shamir Sharing, Blakley, Định lý Số dư Trung Hoa, chữ ký số.

## THRESHOLD CRYPTOGRAPHY SYSTEM APPLIED TO DIGITAL SIGNATURE PROBLEMS

**Abstract.** Encryption and security always play a crucial role in many areas of life, especially when sensitive data needs to be shared among a group of people. The problem is that after encrypting using public-key encryption, to be able to decrypt, the participation of a large enough amount of people in the group is required (threshold encryption). In this research, we will investigate the definition of the problem, the algorithms such as Secret Shamir Sharing, Blakley, Chinese Remainder Theorem (CRT), and other mathematical approaches. We will implement and evaluate, compare between the above schemes, and combine asymmetric encryption and apply them to the digital signature problem.

**Keywords.** Threshold cryptography, Secret Shamir Sharing, Blakley, CRT, Digital Signature.

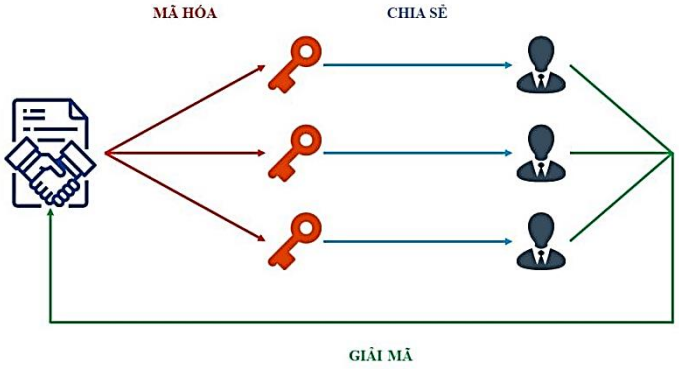
## 1 GIỚI THIỆU

### 1.1 Mô hình mã hóa ngưỡng (threshold encryption model)

#### 1.1.1 Định nghĩa

*Mô hình mã hóa ngưỡng* là một trong những mô hình mật mã học mà trong đó thông tin sẽ được mã hóa tạo thành các khóa khác nhau rồi phân phối cho nhiều bên sở hữu. Để giải mã thông tin đã được mã hóa cần sự tham gia của một số bên giữ khóa nhất định (nhiều hơn hoặc bằng ngưỡng định trước) cùng hợp tác. Một trong những ưu điểm cơ bản của mô hình này là nó có thể ra tạo sự thuận tiện trong quá trình quản lý khóa. Với các hệ thống mật mã truyền thống, nếu người giữ khóa bị thất lạc khóa hoặc bị tấn công, toàn bộ hệ thống có thể cần phải đổi khóa mới để đảm bảo an toàn. Trong mô hình mã hóa ngưỡng, khóa có thể được xây dựng lại bằng cách sử dụng các phần còn lại, cho phép hệ thống hoạt động tiếp, không gián đoạn. *Mô hình mã hóa ngưỡng* có thể được ứng dụng trong nhiều lĩnh vực, bao gồm giao tiếp an toàn, kiểm soát truy cập và chữ ký số. Nó cũng là một thành phần quan trọng của tính toán đa bên, cho phép nhiều bên cùng tính toán một chức năng mà không tiết lộ đầu vào riêng tư của họ.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 1: Ví dụ về mô hình mã hóa ngưỡng (ngưỡng là 3)

### 1.1.2 Tính chất

*Tính bảo mật:* được thiết kế để bảo vệ các thông tin bí mật bằng cách phân chia chúng thành nhiều phần và phân phối chúng đến nhiều người dùng khác nhau. Điều này đảm bảo rằng không có ai có thể khôi phục lại thông tin bí mật nếu không có đủ số lượng người dùng và các phần của thông tin.
*Tính tin cậy:* phân chia thông tin thành nhiều phần khác nhau sẽ đảm bảo rằng thông tin không bị mất hoặc hư hỏng khi một phần bị mất hoặc bị thay đổi. Các phần thông tin được giữ bởi nhiều người dùng khác nhau, nên nguy cơ mất mát hoặc hư hỏng được giảm thiểu.
*Tính linh hoạt:* cho phép sử dụng các thông tin bí mật khác nhau với số lượng người được phân chia và ngưỡng khác nhau tùy theo yêu cầu của người dùng. Điều này làm cho mã hóa ngưỡng trở nên linh hoạt và có thể được áp dụng cho nhiều mục đích khác nhau.
*Tính hiệu suất:* có hiệu suất cao vì nó cho phép nhiều người dùng có thể truy cập vào thông tin bí mật cùng một lúc, giảm thiểu thời gian và tài nguyên cần thiết để quản lý thông tin bí mật.
*Tính an toàn:* có mức độ bảo mật cao hơn so với các hệ thống mật mã đơn người dùng truyền thống. Trong mô hình mã hóa ngưỡng, thông tin được chia thành nhiều khóa khác nhau, mỗi khóa chỉ nắm giữ một phần của thông tin. Điều này làm cho việc tấn công để truy cập đến toàn bộ thông tin trở nên khó khăn.

## 1.2 Chữ ký số (digital signature)

### 1.2.1 Định nghĩa

*Chữ ký số* (Digital signature) là một phần của chữ ký điện tử, đây một dạng dữ liệu văn bản xác thực và thường được đính kèm theo các loại dữ liệu siêu văn bản khác như hình ảnh, âm thanh, video,… Nhằm mục đích xác nhận chủ sở hữu của một vấn đề.
*Chữ ký số* khóa công khai (hay hạ tầng khóa công khai) là một mô hình ứng dụng mã hóa bất đối xứng (Asymmetric encryption model) vào chữ ký số để tạo sự tin cậy trong việc xác thực độ uy tín của chữ ký thông qua cặp khóa công khai – bí mật. Qua đó, người dùng có thể ký gián tiếp các văn bản điện tử hay trao đổi các thông tin một cách bí mật, uy tín và an toàn.

### 1.2.2 Tính chất

*Tính toàn vẹn:* do được áp dụng mô hình mã hóa một chiều (One-way encryption model) nên thừa hưởng tính vẹn toàn từ mô hình này. Nhờ đó, thông tin đảm bảo được toàn vẹn, vì nếu thông tin được sửa chữa trong quá trình gửi đi thì dẫn đến mã băm sẽ thay đổi và không thể xác thực được thông tin.
*Tính một chiều:* do được áp dụng mô hình mã hóa một chiều (One-way encryption model) nên thừa hưởng tính một chiều từ mô hình này. Nhờ đó, nếu quá trình mã hóa và truyền thông tin bị lộ đoạn mã băm thì các tin tặc vẫn không thể giải mã và tìm được về thông tin quan trọng ban đầu.
*Tính xác thực chủ sở hữu, xác định nguồn gốc:* được áp dụng mô hình mã hóa bất đối xứng. Nhờ đó, bên gửi đi có thể dùng khóa riêng tư của mình để mã hóa đoạn mã băm (hay mã gốc) tạo thành chữ ký số và gửi đi; bên nhận nhận được chữ ký, dùng khóa công khai của bên gửi giải mã để thu được mã gốc (quá trình xác thực chủ sở hữu), sau đó so sánh mã gốc nhận được với mã gốc ban đầu (xác định nguồn gốc).
*Tính không thể phủ nhận:* được áp dụng mô hình mã hóa bất đối xứng. Nhờ đó, thông tin được mã hóa và giải mã ngay từ chính cặp khóa công khai – bí mật của người gửi nên người gửi không thể phủ nhận.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

# 2 CÁC LÝ THUYẾT CƠ SỞ

## 2.1 Tổng hợp các lý thuyết

Các mô hình chia sẻ bí mật dựa trên Threshold cryptosystem sử dụng nhiều phép toán rời rạc và số học:

- Phép toán modular (Modulus): Phép toán này được sử dụng để đảm bảo tính bảo mật của thông tin bí mật được chia sẻ. Các phép tính toán trên các phần tử được thực hiện trong không gian modulo để tránh việc thông tin bí mật bị lộ ra bên ngoài.
- Phép toán nghịch đảo modular (Modular Inverse): Đây là một phép toán rất quan trọng trong mô hình chia sẻ bí mật. Phép toán này được sử dụng để tính toán các giá trị khóa riêng tư và khóa công khai cho các thành viên trong tập hợp con.
- Phép toán XOR (Exclusive OR): Đây là một phép toán rời rạc đơn giản được sử dụng để tạo ra các bộ khóa và chia sẻ bí mật. Ví dụ: a XOR b = c, trong đó c được tính bằng cách so sánh từng bit của a và b. Trong C/C++ XOR là một toán tử logic được sử dụng để kiểm tra xem một số điều kiện có đúng hay không.
- Phép toán AND (Logical AND): Đây là một phép toán logic được sử dụng để thực hiện các tính toán trên các bit. Ví dụ: a AND b = c, trong đó c là giá trị được tính bằng cách so sánh từng bit của a và b và trả về giá trị 1 nếu cả hai bit đều là 1.
- Phép toán OR (Logical OR): Đây là một phép toán logic được sử dụng để thực hiện các tính toán trên các bit. Ví dụ: a OR b = c, trong đó c là giá trị được tính bằng cách so sánh từng bit của a và b và trả về giá trị 1 nếu có ít nhất một bit là 1.
- Ma trận: thực hiện các tính toán đại số tuyến tính và kiểm tra tính đúng đắn của quá trình chia sẻ bí mật. Trong Python tính toán trên ma trận thường được thực hiện bằng cách dùng thư viện NumPy.
- Phép toán nhân trên các số nguyên rời rạc: Phép toán này được sử dụng để tạo ra các khóa riêng tư và khóa công khai cho các thành viên trong tập hợp con.
- Phép toán chia trên các số nguyên rời rạc: Phép toán này được sử dụng để tính toán các ngưỡng được thiết lập trước đó, và chỉ khi số thành viên trong tập hợp con đạt đủ ngưỡng thì mới có thể khôi phục lại thông tin bí mật.
- Phép toán cộng và trừ trên các số nguyên rời rạc: Các phép toán này được sử dụng để tính toán các giá trị đại diện cho thông tin bí mật, và đảm bảo tính toàn vẹn của thông tin khi chia sẻ nó đến các thành viên trong tập hợp con.
- Giải hệ phương trình tuyến tính: Xây dựng dựa vào ma trận. Được sử dụng cho mục đích giải mã bí mật Blakley sử dụng phép toán rời rạc được sử dụng để giải hệ phương trình này và tìm ra vector bí mật ban đầu.
- Đa thức Lagrange: Để khôi phục bí mật từ các phần được chia sẻ, các bên sử dụng đa thức Lagrange. Đa thức này được tính bằng cách sử dụng các giá trị rời rạc và các điểm được chọn để chia sẻ bí mật ban đầu.

## 2.2 Số học

### 2.2.1 Đồng dư

Cho $m \in N^*; a, b \in Z$. Nếu a và b khi chia cho m có cùng số dư ta nói: a và b đồng dư theo modulo m.
Kí hiệu: $a \equiv b \pmod m$. Hệ thức: $a \equiv b \pmod m$ gọi là đồng dư thức. Các tính chất

$$a \equiv b \pmod m \Leftrightarrow a \pm c \equiv b \pm c \pmod m.$$
$$a \equiv b \pmod m \Rightarrow f(a) \equiv f(b) \pmod m$$
$$a \equiv b \pmod m;\ b \equiv c \pmod m \Rightarrow a \equiv c \pmod m$$

### 2.2.2 Định lý số dư Trung Hoa (Chinese Remainder Theorem - CRT)

#### 2.2.2.1 Nội dung và ứng dụng

*Định lý số dư Trung Hoa* (Chinese Remainder Theorem - CRT) là một định lý toán học liên quan đến số học modulo và được sử dụng rộng rãi trong lý thuyết số, mã hóa thông tin và các lĩnh vực liên quan đến toán học ứng dụng. Định lý nói về nghiệm của hệ phương trình đồng dư bậc nhất:

$$\begin{cases} x \equiv a_1 \ (mod\ m_1) \\ x \equiv a_2 \ (mod\ m_2) \\ \quad \dots \dots \dots \\ x \equiv a_3 \ (mod\ m_3) \end{cases}$$

Trong đó: m1, m2,…, mk đôi một nguyên tố cùng nhau. Cụ thể là: hệ luôn có nghiệm duy nhất theo modulo m1.m2….mk . Phương pháp giải:
Bước 1: Đặt $m = m_1 m_2 \dots m_k = N_i m_i$ với $i = 1,2,3,\dots,k$ .
Bước 2: Tìm các nghiệm $N_i^{-1}$ của phương trình $N_i x \equiv 1 \pmod m$ (sử dụng thuật toán Euclid mở rộng)
Định nghĩa modulo nghịch đảo: Nếu gcd(a,b) = 1 => a*x (mod b) = 1 thì x là nghịch đảo của a theo mod b. Kí hiệu là: $a^{-1} \pmod b$. Tìm x: a.x + b.y = 1
Bước 3: Tìm được một nghiệm của hệ là: $x_0 = \sum_i N_i N_i^{-1} m_i$.
Bước 4: Kết luận nghiệm: $x \equiv x_0 \pmod m$.
Ví dụ: (bài toán Hàn Tín Điểm binh) Hàn Tín là một danh tướng thời Hán Sở, từng được phong tước vương thời Hán Cao Tổ Lưu Bang, ông có khả năng tính chính xác số binh lính bằng cách: Cho binh lính lần lượt xếp thành hàng 3, hàng 5, hàng 7, rồi lấy số binh lính dư ra ở mỗi cách xếp hàng để tính ra được số lính. Cụ thể: binh lính xếp hàng 3 thì dư 2 người, xếp hàng 5 thì dư 3 người, xếp hàng 7 thì dư 5 người.
Ta được một hệ phương trình đồng dư là: 3, 5, 7 đôi một nguyên tố cùng nhau, có UCLN là 1.

- Bước 1: m = 3.5.7 = 105; $\boldsymbol{m_1}$ = 5.7 = 35; $\boldsymbol{m_2}$ = 3.7=21; $\boldsymbol{m_3}$ = 3.5 = 15
- Bước 2: Tìm được gcd(35,3) = 1. Tìm modulo nghịch đảo của 35 trong mod 3, được kết quả là 2.
  + Tìm gcd(21,5) = 1. Tìm modulo nghịch đảo của 21 trong 5, được 1.
  + Tìm gcd(17,5) = 1. Tìm modulo nghịch đảo của 17 trong 5, được 3.
- Bước 3: $x_0$ = 2.2.35 + 3.1.21 + 5.1.15 = 278.
- Bước 4: $\boldsymbol{x \equiv 278\ (mod\ 105)}$, do 278 % 105 = 68 nên $\boldsymbol{x \equiv 68\ (mod\ 105)}$.

Bên dưới là các ứng dụng:

- Để giải quyết những bài toán về sự tồn tại và đếm các số nguyên thỏa mãn một hệ các điều kiện đồng dư, quan hệ chia hết,…
- Có nhiều ứng dụng trong các bài toán về số nguyên lớn áp dụng vào lí thuyết mật mã, trong đó tiêu biểu là lý thuyết mật mã RSA.
- Sử dụng định lý để chứng minh công thức Euler, hay giải bài toán mở rộng của định lý Wilson và đếm số nghiệm của phương trình đồng dư.
- Tùy vào mục đích sử dụng, còn một số ứng dụng khác,....

Ta cũng có thể áp dụng CRT vào mã hóa:

- CRT được sử dụng để giải mã thông điệp được mã hóa bằng thuật toán RSA, giúp tăng tốc độ xử lý và đảm bảo tính toán chính xác, tối ưu quá trình giải mã và mã hóa.
- CRT giúp tăng tốc độ tính toán trong mã hóa, giảm thiểu áp lực tính toán trên các số lớn và không gian lưu trữ yêu cầu cho khối dữ liệu mã hóa.
- CRT là một công cụ toán học được sử dụng rộng rãi trong các hệ thống mã hóa hiện đại, trong đó quan trọng là mã hóa bất đối xứng, phổ biến nhất là mã hóa RSA.

### 2.2.2.2 Mã nguồn C/C++

Đường dẫn Github: https://github.com/TranNgocHuyen19/YSC-2023.

## 2.3 Mô hình mã hóa bất đối xứng (asymmetric encryption model)

### 2.3.1 Định nghĩa

*Mô hình mã hóa bất đối xứng* (Asymmetric encryption model) là một trong những mô hình mật mã học mà trong đó thông tin sẽ được mã hóa thông qua một cặp khóa gồm *khóa công khai* (Public key) và *khóa bí mật* (Private key). Khóa công khai có thể được chia sẻ rộng rãi và được sử dụng để mã hóa thông tin, trong khi khóa bí mật chỉ có thể được sở hữu bởi chủ sở hữu của nó và được sử dụng để giải mã thông tin.
*Phương pháp mã hóa bất đối xứng* thường được sử dụng để tăng tính bảo mật và độ an toàn cho các thông tin quan trọng trong giao thức truyền tin qua mạng Internet. Với mã hóa bất đối xứng, người gửi thông tin

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

mã hóa dữ liệu của họ bằng khóa công khai của người nhận, và chỉ người nhận có thể giải mã thông tin đó bằng khóa bí mật của mình.

### 2.3.2 Ưu điểm

- *Bảo mật cao:* Mã hóa bất đối xứng sử dụng một cặp khóa gồm khóa công khai và khóa bí mật. Với việc sử dụng khóa công khai để mã hóa và khóa riêng tư để giải mã, thông tin được bảo vệ tốt hơn so với mã hóa đối xứng.
- *Dễ dàng chia sẻ khóa công khai:* Khóa công khai có thể được chia sẻ rộng rãi mà không ảnh hưởng đến tính bảo mật của thông tin được mã hóa.
- *An toàn trong truyền thông:* Mã hóa bất đối xứng được sử dụng rộng rãi trong truyền thông an toàn qua mạng, xác thực người dùng, chữ ký số,…

### 2.3.3 Nhược điểm

- *Tốc độ chậm:* Tốc độ mã hóa bất đối xứng thường chậm hơn so với mã hóa đối xứng.
- *Khóa riêng tư phải được giữ tốt:* Nếu khóa riêng tư bị rò rỉ, thông tin sẽ dễ dàng bị xâm nhập và đánh cắp. Do đó, việc bảo vệ khóa riêng tư là rất quan trọng trong mã hóa bất đối xứng.
- *Khó thực hiện trên các thiết bị có tài nguyên hạn chế:* Việc thực hiện mã hóa bất đối xứng trên các thiết bị có tài nguyên hạn chế như điện thoại thông minh, máy tính nhúng, IoT,... có thể gặp khó khăn do tính toán phức tạp và tốc độ xử lý chậm.

## 2.4 Phương pháp mã hóa RSA

### 2.4.1 Giới thiệu sơ lược

*Phương pháp mã hóa RSA* là một trong những phương pháp mã hóa bất đối xứng phổ biến nhất được sử dụng hiện nay. Mã hóa RSA là một trong những phương pháp mã hóa công khai được phát triển bởi Ron Rivest, Adi Shamir và Leonard Adleman vào năm 1977. RSA là viết tắt các chữ cái đầu tiên trong tên của ba nhà khoa học này.

Mã hóa RSA dựa trên nguyên tắc rằng việc phân tích một số nguyên thành các thừa số nguyên tố là rất khó khăn, đặc biệt là khi số đó rất lớn. Do đó, phương pháp mã hóa RSA khai thác sử dụng tính chất này của các số nguyên tố để tạo ra cặp khóa công khai và khóa bí mật.

### 2.4.2 Giải thuật

*Thuật toán mã hóa RSA* hoạt động dựa trên nguyên tắc về tính khó giải của việc phân tích một số nguyên lớn thành các *thừa số nguyên tố*. Mã hóa RSA bao gồm các quá trình sau:

Quá trình tạo khóa:

Bước 1: Chọn hai số nguyên tố lớn p và q ngẫu nhiên ($p \neq q$).

Bước 2: Tính $n = p \cdot q$.

Bước 3: Tính hàm số mũ số nguyên tố Euler của n: $\varphi(n) = (p-1) \cdot (q-1)$.

Bước 4: Chọn số e sao cho $1 < e < \varphi(n)$ và e là số nguyên tố cùng nhau với $\varphi(n)$.

Bước 5: Tính số nguyên tố cùng nhau với $\varphi(n)$ và thỏa mãn $e \cdot d \equiv 1 \pmod{\varphi(n)}$. Số d là khóa bí mật.

- ✓ *Khóa công khai bao gồm:*
  - *n* – môđun.
  - *e* - số mũ công khai (số mũ mã hóa).
- ✓ *Khóa bí mật bao gồm:*
  - *n* - môđun, xuất hiện cả trong *khóa công khai* và *khóa bí mật*.
  - *d* - số mũ bí mật (số mũ giải mã).
- ✓ *Một dạng khác của khóa bí mật bao gồm:*
  - *p và q* - hai số nguyên tố chọn ban đầu.
  - *d mod (p - 1), d mod (q - 1)* - thường được gọi là *dmp1, dmq1*.
  - *mod p* - thường được gọi là *iqmp*.

Dạng này cho phép thực hiện giải mã và ký nhanh hơn với việc sử dụng định lý số dư Trung Quốc. Ở dạng này, tất cả thành phần của khóa bí mật phải được giữ bí mật. A gửi khóa công khai cho B, và giữ bí mật khóa bí mật của mình. Ở đây, p và q giữ vai trò rất quan trọng. Chúng là các phân tố của n và cho phép tính

d khi biết e. Nếu không sử dụng dạng sau của khóa bí mật (dạng CRT) thì p và q sẽ được xóa ngay sau khi thực hiện xong quá trình tạo khóa.
Quá trình mã hóa: Giả sử A muốn gửi đoạn thông tin M cho B. Đầu tiên A chuyển M thành số m (m < n) theo một phương thức được thỏa thuận trước. Lúc này A có m và biết n, e do B gửi. B sẽ tính c là bản đã mã hóa của m theo công thức:

$$\boldsymbol{c = m^e mod\ n}$$

Hàm trên có thể tính dễ dàng bằng cách sử dụng phương pháp tính hàm mũ (theo môđun) thông qua thuật toán bình phương và nhân. Cuối cùng A gửi c cho B.
Quá trình giải mã: B nhận c từ A và biết khóa bí mật d. B có thể tìm được m từ c theo công thức sau:

$$\boldsymbol{m = c^d mod\ n}$$

Biết m, B tìm lại M theo phương pháp đã thỏa thuận trước đó. Quá trình giải mã hoạt động vì ta có:

$$\boldsymbol{c^d \equiv (m^e)^d \equiv m^{e \cdot d}}$$

Do e * d ≡ 1 (mod p - 1) và e * d ≡ 1 (mod q - 1), (theo Định lý Fermat nhỏ) nên:

$$m^{e*d} = m \bmod p \text{ và } m^{e*d} = m \bmod q$$

Do p và q là hai số nguyên tố cùng nhau, áp dụng định lý số dư Trung Quốc, ta có:

$$m^{e*d} = m \bmod (q * p) \text{ hay } c^d = m \bmod n$$

### 2.4.3 Mã C/C++

Đường dẫn Github: https://github.com/phantonlocnguyen/RSA

## 2.5 Mô hình mã hóa một chiều (One-way encryption model)

### 2.5.1 Giới thiệu

*Mô hình mã hóa một chiều (One-way encryption model)* là một trong những mô hình mật mã học mà trong đó thông tin sẽ được mã hóa thành một mật mã duy nhất và không thể giải mã tìm lại thông tin đầu. Nó có các đặc điểm sau đây:

- *Tính toàn vẹn*: tạo ra một đoạn giá trị mã hóa cố định duy nhất và không trùng lặp cho mỗi thông tin đầu vào khác nhau.
- *Tính một chiều:* thông tin sau khi được mã hóa sẽ không thể giải mã tìm lại thông tin ban đầu.

Về hàm Hash (hàm băm): là hàm thực hiện quá trình biến một dữ liệu đầu vào có độ dài bất kỳ thành một chuỗi đầu ra đặc trưng có độ dài cố định. Các giá trị được trả về bởi hàm băm được gọi là giá trị băm, mã băm, thông điệp băm, hoặc đơn giản là "Hash". Ta mô tả cụ thể như bên dưới:

### 2.5.2 Giải thuật MD5

Dưới đây là các bước thực hiện giải thuật MD5:
Định nghĩa một macro để thực hiện phép xoay trái (rotate left) trên một giá trị 32 - bit.
Định nghĩa hàm 'md5_hash' nhận các tham số đầu vào: một con trỏ trỏ tới một mảng các byte đầu vào ('initial_msg'), một kích thước của đầu vào ('initial_len'), và một con trỏ trỏ tới một mảng các byte để lưu giá trị băm ('digest').

```
void md5_hash(const uint8_t * initital_msg, size_t initial_len, uint8_t *digest)
```

Khởi tạo các giá trị hằng số cho bước khởi tạo của thuật toán MD5.
Tính toán độ dài đầu vào của đơn vị bit.
Chuẩn bị đầu vào cho thuật toán MD5 bằng cách tạo một mảng 'msg' có độ dài phù hợp với quy tắc của thuật toán. Mảng này chứa đầu vào gốc, bit "1" được thêm vào cuối, và các phần tử còn lại được điền bằng 0. Độ dài bit cuối cùng của đầu vào được ghi vào cuối mảng.
Tạo con trỏ 'M' tới mảng 'msg' với với kiểu dữ liệu là uint32_t.
Chia nhỏ phần đầu vào thành các khối 512 - bit (64 - byte). Con trỏ 'X' trỏ tới một khối trong mảng 'M'.
Khởi tạo các biến đệm ban đầu.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Thực hiện một vòng lặp để tính toán các giá trị mới của biến đệm A, B, C, D dựa trên các phép toán logic và phép xoay trái:

- Cập nhật các giá trị của biến đệm với các giá trị mới tính toán.
- Chuyển đổi giá trị của các biến đệm thành dạng byte và lưu vào mảng *'digest'* để đưa ra kết quả cuối: Hàm *'main()'* chứa ví dụ sử dụng hàm MD5. Khởi tạo một chuỗi *'message'* và một mảng *'digest'* để lưu giá trị băm. Sau đó, gọi hàm *'md5_hash()'* với đầu vào là chuỗi và lưu kết quả vào mảng *'digest'*. Cuối cùng, in ra giá trị MD5 dưới dạng mã hexa.

### 2.5.3 Mã C/C++

Đường dẫn Github: https://github.com/phantonlocnguyen/HASH_MD5

## 3 CÁC CÁCH TIẾP CẬN CHO MÃ HÓA NGƯỠNG

### 3.1 Mã hóa Shamir

### 3.1.1 Định nghĩa

Mã hóa Shamir *(Shamir's Secret Sharing Scheme – SSS)* là một thuật toán được đặt theo tên của nhà mật mã học Adi Shamir người Israel, ông là người đã phát minh ra nó. Thuật toán được sử dụng để phân phối một bí mật thành nhiều phần, được gọi là các chia sẻ bí mật, trong đó mỗi chia sẻ bí mật được phân phối cho một nhóm người tham gia. Mục đích của Shamir's Secret Sharing Scheme là bảo vệ tính riêng tư và đảm bảo an toàn trong việc truyền và xác minh thông tin. Các phần của chia sẻ bí mật được tập hợp lại với nhau để xây dựng lại bí mật ban đầu, tổng số các chia sẻ bí mật nhỏ hơn tổng số các chia sẻ bí mật ban đầu được chia sẻ gọi là ngưỡng, là điều bắt buộc để xây dựng lại bí mật. Điều này giúp tránh thất bại trong việc giải mã thông tin được lưu giữ chặt chẽ nếu chỉ một hoặc một vài bên không có mặt. Thuật toán *Shamir's Secret Sharing Scheme – SSS* dựa trên nguyên tắc đa thức và áp dụng nó vào việc phân phối bí mật. Mỗi chia sẻ được tạo ra bằng cách chọn ngẫu nhiên một điểm trên đường cong của một đa thức với hệ số ngẫu nhiên. Để xây dựng lại bí mật, người dùng cần có đủ số lượng chia sẻ và phải biết đường cong và ngưỡng được sử dụng để tạo ra các chia sẻ.

### 3.1.2 Công thức

### 3.1.2.1 Mã hóa Shamir

Công thức Chia sẻ bí mật Shamir gồm các bước sau:

1. Xác định ngưỡng K để tái tạo lại chia sẻ bí mật ban đầu. Sau đó ta lập đa thức độc lập tuyến tính với bậc đa thức là $\boldsymbol{K-1}$.
2. Chọn số S (secret) là bí mật cần chia sẻ, số nguyên $a_1$, $a_2$,... $a_m$ ($1 < a_1, a_2, ...a_m < S$).
3. Tính toán hàm đa thức $\boldsymbol{f(x)}$ mod S bằng công thức:

$$\boldsymbol{f(x) = secret + a_1 x + a_2 x^2 + \ldots + a_m . x^{(K-1)}}$$

Trong đó:

- m là hằng số ngẫu nhiên (là các hệ số tuyến tính $a_1$, $a_2$,... $a_m$).
- secret là 1 hằng số bí mật cần được mã hóa.
- K là ngưỡng chia sẻ bí mật.

4. Chọn N (là số các chia sẻ bí mật được quy định ban đầu) điểm ngẫu nhiên trên đường cong $\boldsymbol{f(x)}$ bằng cách tính toán các giá trị mà tại đó điểm N phải là nghiệm của hàm $\boldsymbol{f(x)}$ (x là giá trị bất kì, $x \rightarrow i$, đây là điều kiện cần) và điểm này phải có ít nhất tối thiểu $N > K$ (là điều kiện đủ) điểm hình thành các chia sẻ bí mật.
5. Các điểm N này sẽ được chia sẻ cho các bên cần truy cập bí mật.
6. Để tái tạo bí mật, tối thiểu K (như đã quy định ban đầu, K là ngưỡng chia sẻ bí mật) phần chia sẻ bí mật được cung cấp cho các bên, sau đó áp dụng công thức nghịch đảo Lagrange để tính toán giá trị của bí mật gốc.

*Ví dụ minh họa:*

1. Ban đầu, để mã hóa mật mã, ta xây dựng đa thức bậc (K - 1).
2. Đặt đa thức là $\boldsymbol{f(x) = a + bx}$. Ở đây, phần hằng số 'a' = secret là mã bí mật của chúng ta.
3. Gọi b là một số bất kỳ, giả sử b = 2.

4. Do đó, đối với đa thức y = 7 + 2x này, ('a' = secret = 7, b = 2) chúng ta tạo ra được N = 4 điểm từ nó.
5. Gọi 4 điểm đó là (1, 9), (2, 11), (3, 13), (4, 15). Rõ ràng, chúng ta có thể tạo đa thức ban đầu từ bất kỳ từ 2 trong số 4 điểm này và trong đa thức kết quả, hằng số 'a' là mã bí mật bắt buộc.

### 3.1.2.2 Giải mã Shamir

Chia sẻ bí mật Shamir gồm các bước sau:

1. Lấy K (là ngưỡng chia sẻ bí mật) phân đoạn đã được chia sẻ, mỗi phân đoạn bao gồm một cặp giá trị (x, y).
2. Để xây dựng lại đa thức đã cho hay tìm lại đa thức ban đầu, chúng ta hình thành các đẳng thức của Lagrange.
   Công thức tổng quát của đa thức Lagrange như sau:

$$\boldsymbol{l_i\,(x) = \frac{x - x_0}{x_i - x_0} \times \ldots \times \frac{x - x_{i-1}}{x_i - x_{i-1}} \times \frac{x - x_{i+1}}{x_i - x_{i+1}} \times \ldots \times \frac{x - x_{k-1}}{x_i - x_{k-1}}}$$

$$\boldsymbol{f(x) = \sum_{i=0}^{n} l_i y_i\,(x)}$$

3. Dựa vào công thức trên, giải các đẳng thức cơ sở của đa thức Lagrange tìm ra phương trình $\boldsymbol{f(x)}$ từ đó tìm ra bí mật ban đầu.

*Ví dụ cụ thể:* Trong 1 trò chơi đi tìm kho báu, để có thể mở được kho báu đó cần nhập đúng mã bí mật là 2023. Ban tổ chức đã mã hóa mã bí mật gốc này thành 4 chia sẻ bí mật khác nhau, để mở được chiếc ổ khóa đó cần có ít nhất 3 mã chia sẻ bí mật trong 4 mã chia sẻ bí mật đó. Nhưng có một điều ban tổ chức không biết cách nào để tạo được các chia sẻ bí mật, bạn hãy giúp ban tổ chức mã hóa bí mật đó và xác định chia sẻ bí mật đó là trùng khớp với mã bí mật ban đầu.

*Phân tích.* Theo đề bài ra, ta có: K = 3, S = 2023, N = 4. Áp dụng công thức tổng quát: $f(x) = secret + a_1x + a_2x^2 + \ldots + a_m.x^{(K-1)}$ để tạo chia sẻ bí mật. Ta chọn đa thức $f(x) = 2023 + 2x + 3x^2$ Là hàm đa thức bậc 2 để mã hóa bí mật.

- Với $\boldsymbol{f(1) = 2023 + 2.1 + 3.1^2 = 2028}$. (1)
- Với $\boldsymbol{f(2) = 2023 + 2.2 + 3.2^2 = 2039}$. (2)
- Với $\boldsymbol{f(3) = 2023 + 2.3 + 3.3^2 = 2056}$. (3)
- Với $\boldsymbol{f(5) = 2D023 + 2.5 + 3.5^2 = 2108}$. (4)

Từ (1), (2), (3), (4) ta được tập nghiệm của phương trình f(x) là S = (x, y) = {(1, 2028); (2, 2039); (3, 2056); (3, 2108)}. Như vậy bạn đã giúp ban tổ chức tạo ra 4 chia sẻ bí mật.

Tiếp đến, chọn 3 điểm trong 4 điểm trên để tìm ra đa thức ban đầu lấy lại mã bí mật gốc mở khóa kho báu trên. Ta có K = 3, gọi lần lượt các điểm (1, 2028); (2, 2039); (3, 2056) là $l_0$ , $l_1$ , $l_2$ .

Áp dụng công thức đa thức Lagrange ta được:

$$\boldsymbol{l_0 = \frac{x - x_1}{x_0 - x_1} \times \frac{x - x_2}{x_0 - x_2} = \frac{x - 2}{1 - 2} \times \frac{x - 3}{1 - 3} = \frac{(x - 2)(x - 3)}{2}}$$

$$\boldsymbol{l_1 = \frac{x - x_0}{x_1 - x_0} \times \frac{x - x_2}{x_1 - x_2} = \frac{x - 1}{2 - 1} \times \frac{x - 3}{2 - 3} = \frac{-(x - 1)(x - 3)}{1}}$$

$$\boldsymbol{l_2 = \frac{x - x_0}{x_2 - x_0} \times \frac{x - x_1}{x_2 - x_1} = \frac{x - 1}{3 - 1} \times \frac{x - 2}{3 - 2} = \frac{(x - 1)(x - 2)}{2}}$$

$$\boldsymbol{f(x) = l_0\,y_0 + l_1\,y_1 + l_2\,y_2}$$

$$\boldsymbol{f(x) = \frac{2028.(x-2)(x-3)}{2} + \frac{-2039.(x-1)(x-3)}{1} + \frac{2056.(x-1)(x-2)}{2}}$$

$$\boldsymbol{= 1014.\left(x^2 - 5x + 6\right) - 2039.\left(x^2 - 4x + 3\right) + 1028.\left(x^2 - 3x + 2\right)}$$

$$\boldsymbol{= 3x^2 - 2x + 2023}$$

Từ kết quả trên ta thấy con số 2023 trong phương trình chính là mã bí mật để mở khóa kho báu.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.1.3 Mã C/C++

*Đường dẫn Github: https://github.com/JosephPhuc/YSC-2023*

## 3.2 Blakley

### 3.2.1 Định nghĩa

Giả sử có 10 nhà khoa học phát minh ra một phát minh vĩ đại nhưng lại mang tiềm năng sức mạnh quân sự khủng khiếp. Bên cạnh đó họ lại là các nhà khoa học từ khắp thế giới, mỗi nhà khoa học lại thuộc một quốc gia khác nhau. Sau khi biểu quyết họ quyết định cất giấu nghiên cứu này. Nhưng có cách nào để tất cả mọi người đều nắm bí mật nhưng lại không sử dụng được nó, chỉ khi đủ số người đồng ý mới sử dụng. Đây là nơi mà mã hóa ngưỡng nói chung tỏa sáng và thực hiện được mục đích của mình.
Mã hóa ngưỡng là mô hình mã hóa mà ở đó: (với n không bé hơn t)

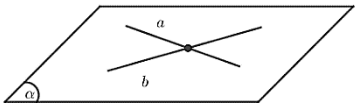
- Bí mật được chia làm n phần,
- Chỉ khi có t phần hoặc hơn trong n phần đó thì mới có thể xác định chính xác bí mật đó. Khi này số "t" được gọi là ngưỡng.

Quay lại bài toán của các nhà khoa học, nếu sử dụng mã hóa ngưỡng, bí mật ở đây sẽ được chia thành 10 phần và ngưỡng có thể là 6,7,...,10 và vấn đề đã được giải quyết. Ai trong số 10 nhà khoa học cũng nắm giữ 1 phần thông tin về bí mật đó cũng như nếu không có t người họp lại và chia sẻ bí mật của họ thì bí mật sẽ không thể được khôi phục và sử dụng cho mục đích xấu. Tất nhiên đó là nếu họ không muốn.
Mô hình chia sẻ bí mật Blakley (hay mô hình mã hóa ngưỡng tuyến tính Blakley) là một mô hình chia sẻ bí mật giữa bên không có sự tin cậy và cần chia sẻ bí mật cũng như số lượng thông tin về bí mật đều nhau. Mô hình mã hóa được đặt tên theo chính tác giả của nó George Robert Blakley (1). Điểm khác biệt của mô hình Blakley so với mô hình Shamir đó chính là ở việc mô hình này sử dụng giao điểm của các siêu phẳng thay vì sử dụng tập hợp các điểm như mô hình Shamir. Khác với mô hình Shamir, mô hình Blakley sử dụng các đa thức tuyến tính để tính toán ra điểm bí mật. Điểm bí mật của Blakley là giao điểm của các siêu phẳng trong một không gian t chiều. Chính vì cách tiếp cận này mà mô hình Blakley không được phổ biến bằng mô hình Shamir do độ phức tạp của hướng tiếp cận toán học.

### 3.2.2 Toán học trong mô hình Blakley

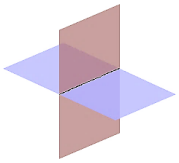
Thay vì sử dụng tập hợp các điểm tạo thành 1 đường thẳng hay đa thức như mô hình Shamir, trong mô hình Blakley ta lại sử dụng nhiều siêu phẳng để tìm ra một giao điểm của các siêu phẳng đó. Để hiểu rõ hơn về vấn đề này, ta sẽ đi qua những không gian 2 và 3 chiều:
Trong không gian 2 chiều:

- Một đường thẳng chính là một siêu phẳng
- Ta có thể thấy được dễ dàng rằng ta sẽ có 1 giao điểm cho 2 đường thẳng.

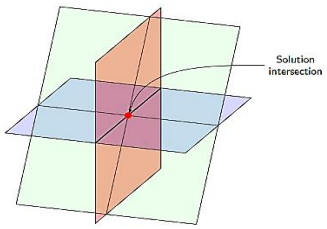



Trong không gian 3 chiều:

- Phần giao nhau của 2 siêu phẳng là một đường thẳng.
- Sẽ luôn vô số siêu phẳng khác khi giao nhau với 2 siêu phẳng mà phần giao nhau của chúng là một điểm:



Minh họa cho phần giao của hai siêu phẳng




Minh họa cho giao điểm của 3 siêu phẳng

Như vậy, ta có thể thấy được rằng, trong không gian t chiều thì chắc chắn sẽ có t siêu phẳng giao nhau tại 1 điểm. Tiếp theo, ta sẽ nói về mã hóa Blakley:

Mã hóa Blakley đưa chúng ta về bài toán tạo ra n siêu phẳng từ điểm bí mật ban đầu. Hay có thể hiểu cách khác rằng tạo ra ma trận C sao cho ma trận C thỏa điều kiện sau:

Cx mod P = y, với

- C: ma trận chứa hệ số của các siêu phẳng khuyết hệ số tự do,
- y: vectơ chứa hệ số tự do tương ứng với các siêu phẳng đó,
- P: số nguyên tố bất kì mà ta chọn.

Để mã hóa Blakley như đã nói ta sẽ chia bí mật thành các siêu phẳng. Các siêu phẳng này sau đó sẽ được chia sẻ cho những cổ phần hay những người nắm giữ bí mật. Giả sử ta có bí mật ban đầu là $x_0$, ngưỡng t và cần chia bí mật ra thành n phần:

- Chọn một số nguyên tố P ($P \geq \boldsymbol{x_0}$)
- Trong trường số P, chọn giá trị ngẫu nhiên hoặc tùy ý cho các giá trị còn lại của điểm bí mật . Từ đây ta sẽ có điểm bí mật trong không gian t chiều.
- Trong trường số P, ta sẽ chọn giá trị cho các hệ số của siêu phẳng sao cho thỏa hệ đồng dư:
- $F(x) \equiv n \pmod P$ với

  F(x): phương trình siêu phẳng khuyết hệ số tự do

  n: hệ số tự do của siêu phẳng đó

Tương tự như vậy ta sẽ tạo ra n siêu phẳng cho n cổ phần hay người giữ bí mật.
Sau khi mã hóa ta sẽ có được n siêu phẳng trong không gian t chiều tương ứng cho mỗi người giữ bí mật hay cổ phần. Chỉ khi có t hoặc nhiều hơn người giữ bí mật đồng ý chia sẻ bí mật với nhau thì bí mật ban đầu mới có thể khôi phục lại. Ta sẽ đi vào một ví dụ cụ thể để dễ dàng hình dung hơn về cách mã hóa của mô hình Blakley. Giả sử ta có:

- Bí mật ban đầu $\boldsymbol{x_0} = 12$,
- Số nguyên tố P = 23,
- Ngưỡng t = 3,
- Số phần bí mật n = 5.

Đầu tiên, ta sẽ chọn $y_0$ và $z_0$ cho điểm bí mật của ta: $y_0 = 20$ và $z_0 = 15$. Ta sẽ có điểm bí mật ban đầu là A (12, 20, 15). Tiếp theo, ta sẽ bắt đầu sử dụng điểm bí mật để tạo ra các phương trình đồng dư thỏa điểm bí mật ban đầu:

- $x + y + z \equiv 1 \pmod{23}$
- $x + 2y + 3z \equiv 5 \pmod{23}$
- $x + 3y + 6z \equiv 1 \pmod{23}$
- $x + 4y + 10z \equiv 12 \pmod{23}$
- $x + 5y + 15z \equiv 15 \pmod{23}$

Ta cũng có thể hiểu thuật toán Blakley bằng theo ma trận:

$$(\mathbf{1\ 1\ 1\ \ 1\ 2\ 3\ \ 1\ 1\ 1\ \ 3\ 4\ 5\ \ 6\ 10\ 15}) \times (\boldsymbol{x_0\ y_0\ \ z_0}) \equiv (\mathbf{1\ 5\ 1\ 12\ 15}) \pmod{23}$$

Và ta đã có hoàn thành xong việc mã hóa theo thuật của Blakley. Giờ việc ta cần làm là giao những phần bí mật cho những người giữ bí mật.
Lưu ý. Trong khi tạo ra các siêu phẳng, ta có thể chọn những hệ số ngẫu nhiên trong trường P. Nhưng để dễ dàng trong việc nghiên cứu cũng như việc thời gian trong lập trình minh họa, chúng ta sẽ sử dụng ma trận Pascal cho thuận tiện.

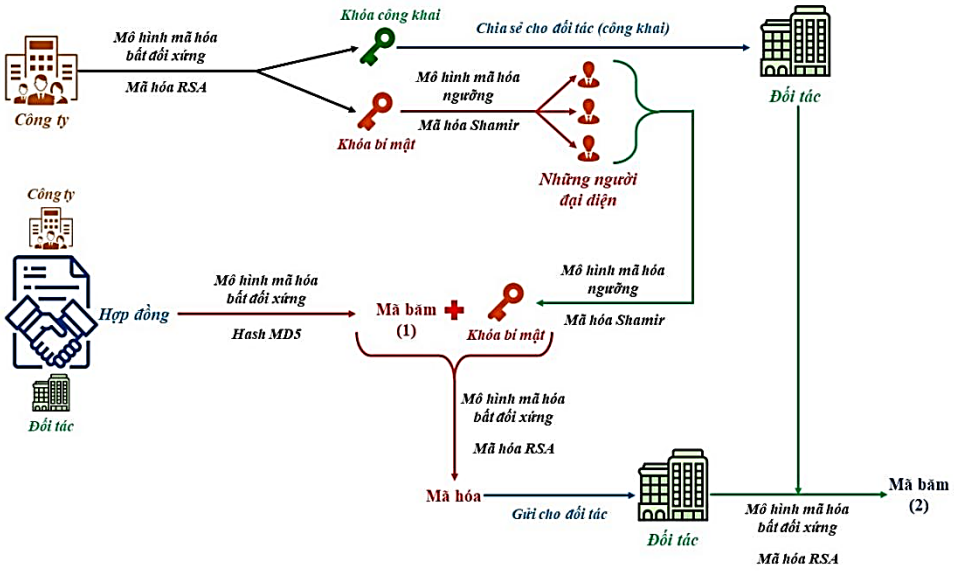
### 3.2.3 Mã python

Đường dẫn Github: https://github.com/HH214129/BlakleyYSC2023

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 3.3 Ứng dụng vào mô hình chữ ký số

### 3.3.1 Mô hình và tính chất riêng




Hình 2: Sơ đồ mô hình chữ ký số có áp dụng mã hóa ngưỡng trong thực hiện hợp đồng

Việc áp dụng mô hình mã hóa ngưỡng (Threshold encryption model) vào chữ ký số sẽ gia tăng thêm các tính chất bảo mật cho chữ ký số (thừa hưởng các tính chất từ mô hình này). Từ đó, tạo nên tính chất đồng thuận tập trung cho chữ ký số. Chỉ khi được đồng thuận bởi một ngưỡng người đại diện nhất định thì mới tìm được khóa bí mật để thực hiện ký số lên đoạn mã băm.

### 3.3.2 Mã C/C++

Đường dẫn Github: https://github.com/phantonlocnguyen/CKS

### 3.3.3 Ứng dụng

Việc mã hóa thông tin theo dạng "đầu – cuối" của mô hình trên có thể đảm bảo được an toàn thông tin trong việc gửi – nhận dữ liệu trên các giao thức truyền tin trực tuyến.

Ngày nay, có nhiều dịch vụ trực tuyến được cung cấp thông qua Internet, chẳng hạn như dịch vụ tài chính - ngân hàng điện tử, dịch vụ ký kết hợp đồng thông minh, dịch vụ thương mại điện tử… Đặc biệt, dịch vụ mạng xã hội là một dịch vụ ngày càng được phát triển và phổ biến trong đời sống, tuy nhiên, chúng có nhược điểm là thường chứa một lượng lớn các thông tin quan trọng của người dùng và những thông tin này truyền qua lại trên các giao thức mạng Internet giữa máy khách và máy chủ (Client – Server).

Do đó, áp dụng mô hình trên vào các hoạt động truyền gửi dữ liệu có thể đảm bảo tính an toàn của dữ liệu: thông tin được vẹn toàn, không bị đánh cắp hay sửa đổi trong quá trình chuyển đi, xác thực lại các thông tin sau khi nhận được, thông tin được mã hóa phân chia ra nhiều máy chủ khác nhau giúp việc tấn công đánh cắp thông tin từ một máy chủ thì không thể giải mã,…

Trong ký kết hợp đồng thông minh bằng chữ ký số, nhờ có tính chất đồng thuận tập trung mà từ đó việc ký kết một hợp đồng quan trọng phải cần có sự thống nhất của toàn bộ người trong hội đồng cấp cao của công ty, tránh một bên cá nhân tự ý quyết định. Việc tấn công lấy thông tin chữ ký số cũng trở nên phức tạp hơn do cần phải tấn công nhiều bên liên quan có nắm giữ chìa khóa.

Trong lĩnh vực sản xuất để bảo mật mã sản phẩm từ quá trình sản xuất cho đến khi nó đến tay người dùng ngăn ngừa được hàng giả, hàng kém chất lượng. Ngoài ra, còn có thể được ứng dụng trong việc bảo mật thư từ, văn bản pháp lý, xác thực thông tin người dùng,…

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

## 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này, nhóm tác giả đã tìm hiểu định nghĩa và các tính chất của các mô hình chia sẻ dựa trên mã hóa ngưỡng và tìm hiểu ứng dụng của chúng vào bài toán chữ ký số. Bài báo đã tập trung vào việc nghiên cứu một số mô hình chia sẻ phổ biến như mô hình chia sẻ Shamir, mô hình chia sẻ Blakley và Định lý số dư Trung Hoa.

Mô hình chia sẻ Shamir là một phương pháp mã hóa ngưỡng được đề xuất bởi Adi Shamir vào năm 1979. Mô hình này cho phép chia sẻ một bí mật thành nhiều phần và chỉ cần một số lượng nhất định (gọi là ngưỡng) của các phần tử để tái tạo lại bí mật ban đầu. Điều này giúp bảo mật thông tin, vì người dùng cần sự hợp tác của nhiều bên để khôi phục bí mật.

Mô hình chia sẻ Blakley, được đề xuất bởi Robert C. Blakley vào năm 1979, là một phương pháp chia sẻ bí mật dựa trên nguyên tắc của đại số tuyến tính. Thay vì sử dụng đa thức như trong mô hình chia sẻ Shamir, mô hình chia sẻ Blakeley sử dụng ma trận và các phép tính trên ma trận để thực hiện việc chia sẻ và tái tạo bí mật.

Định lý số dư Trung Hoa là một định lý toán học quan trọng trong lĩnh vực mã hóa và hệ mật. Định lý này cho phép xác định một số lượng nhất định các phần tử trong một tập hợp con của các phần tử trong tập hợp lớn hơn, dựa trên các mệnh đề về phép chia và số dư. Định lý số dư Trung Hoa đã được áp dụng trong các mô hình chia sẻ mã hoá ngưỡng để đảm bảo tính toàn vẹn và bảo mật cho chữ ký số.

Bên cạnh việc nghiên cứu và phân tích các tính chất của các mô hình chia sẻ dựa trên mã hóa ngưỡng, nhóm tác giả cũng đã tiến hành mô phỏng lại các giải thuật được sử dụng trong các mô hình này bằng ngôn ngữ C/C++ và Python. Họ đã sử dụng các thư viện và chức năng phù hợp để hiện thực hóa các phép toán số học trên ma trận như phép cộng, phép nhân, và phép chia. Việc áp dụng mô hình mã hóa ngưỡng vào bài toán chữ ký số cung cấp thêm lợi ích bảo mật. Thay vì chỉ có một người duy nhất tạo ra chữ ký số, mô hình chia sẻ dựa trên mã hóa ngưỡng cho phép nhiều bên tham gia vào quá trình tạo ra và xác nhận chữ ký. Điều này tăng cường tính phân tán và khó khăn trong việc tấn công và giả mạo chữ ký số.

## TÀI LIỆU THAM KHẢO

[1] A. Alpher, Frobnication, *Journal of Foo*, vol. 12, no. 1, pp. 234-778, 2002.

[2] A. Alpher and J. P. N. Fotheringham-Smythe, Frobnication revisited, *Journal of Foo*, vol. 13, pp. 234-778, 2003.

[3] A. Alpher, J. P. N. Fotheringham-Smythe, and G. Gamow, Can a machine frobnicate?, *Journal of Foo*, vol. 14, no. 1, pp. 234-778, 2004.

[4] V. Arnold, K. Vogtmann, and A. Weinstein, *Mathematical Methods of Classical Mechanics*, ser. Graduate Texts in Mathematics. Springer, 1989.

[5] *FLEXChip Signal Processor (MC68175/D)*, Motorola, 1996.

[6] M.-T. Pham, O. J. Woodford, F. Perbet, A. Maki, and B. Stenger. (2012) Toshiba CAD model point clouds dataset. [Online].

Available: www.toshiba.eu/eu/Cambridge-Research-Laboratory/Computer-Vision-Group/Stereo-Points/

[7] M.-T. Pham, O. J. Woodford, F. Perbet, A. Maki, B. Stenger, and R. Cipolla, A new distance for scale-invariant 3D shape recognition and registration, in *Proc. Int. Conf. on Computer Vision*, 2011, pp. 145-152.

[8] M.-T. Pham, O. J. Woodford, F. Perbet, A. Maki, B. Stenger, and R. Cipolla, An image processing method and system, US Patent 20130016913 (A1), 2013.

[9] L. A. Santalo, Integral geometry and geometric probability, in *Encyclopedia of Mathematics and its Applications*, G. C. Rota, Ed. Addison-Wesley, 1976, vol. 1.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F006**

# ỨNG DỤNG KỸ THUẬT XỬ LÝ HÌNH ẢNH KẾT HỢP VỚI MẠNG HỌC SÂU PHÁT HIỆN UNG THƯ VÚ

HÀ BẢO ANH[1*], LÊ THÀNH NGHĨA[1], NGUYỄN VĂN NAM[1], LÂM QUANG PHÚ[1]
ĐÀO XUÂN HOÀNG TUẤN[1], HUỲNH TRUNG HIẾU[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
**baoanhcr7123456@gmail.com, lethanhnghia147@gmail.com, nguyenvannam14056969@gmail.com, lamquanphu2176@gmail.com, hoangtuan.salmon@gmail.com, hthieu@ieee.org*

**Tóm tắt.** Ung thư vú không những là một bệnh ung thư hay gặp nhất ở phụ nữ mà còn là một trong những nguyên nhân gây tử vong đối với phụ nữ tại nhiều nước. Hiện nay, chụp X-quang vú (mammography) là một kỹ thuật được dùng để sàng lọc và phát hiện sớm các bệnh lý về tuyến vú và ung thư vú, bệnh được phát hiện càng sớm thì khả năng điều trị khỏi bệnh sẽ càng cao, bên cạnh những loại lợi ích mang lại thì chụp X-quang vú cũng có những hạn chế nhất định bao gồm không xác định được trạng thái lành hay ác tính của một tổn thương ở vú, việc đọc kết quả nhũ ảnh đòi hỏi nhân viên y tế phải có kinh nghiệm và trình độ chuyên môn cao, thời gian xử lý, chi phí. Với sự phát triển của trí tuệ nhân tạo hiện nay, việc ứng dụng các mô hình máy học kết hợp với các kỹ thuật xử lý ảnh trong phân tích ảnh đã mang lại những kết quả vượt trội trong nhiều ứng dụng khác nhau. Do đó bài toán phát hiện ung thư vú đã được thực hiện theo cách tiếp cận mới này. Nghiên cứu này có thể giúp giai đoạn sàng lọc và phát hiện ung thư vú có thể thực hiện một cách nhanh chóng, độ chính xác cao, chi phí thấp, hỗ trợ các y bác sĩ trong việc có thể đưa ra các chẩn đoán bệnh kịp thời. Trong quá trình xử lý, nhóm sẽ thực hiện trên tập dữ liệu gồm những ảnh X-quang nhũ ảnh. Bước đầu tiên các nhũ ảnh bằng các phương pháp xử lý ảnh, kế tiếp các hình ảnh đó sau khi được tách ra sẽ được dùng làm đầu vào cho các mô hình phân loại. Từ đó ta có thể đánh giá độ chính xác cũng như độ tối ưu của từng mô hình khác nhau và lựa chọn ra mô hình tốt nhất.

**Từ khóa**. Ung thư vú, Ảnh chụp quang tuyến vú, Mô hình học sâu, Công nghệ hỗ trợ sức khỏe.

# APPLICATION OF IMAGE PROCESSING TECHNIQUES COMBINED WITH DEEP LEARNING NETWORKS FOR BREAST CANCER DETECTION

**Abstract.** Breast cancer is the most common cancer in women and one of the leading causes of death for women in many countries. Currently, mammography is used for screening and early detection of mammary gland diseases and breast cancer. Besides the benefits, mammography also has certain limitations, including not determining the benign or malignant state of a breast lesion; the reading of mammograms requires human medical staff must have experience and high professional qualifications, processing time, and cost. With the current development of artificial intelligence, applying machine learning models combined with image processing techniques in image analysis has brought outstanding results in many different applications. Therefore, the problem of detecting breast cancer has been carried out according to this new approach. This research can help the screening and detection of breast cancer quickly, with high accuracy and low cost, helping doctors make a diagnosis. During the processing, the team will work on a data set of X-ray mammograms. The first step is mammograms by image processing methods, then the images, after being extracted, will be used as input for classification models. From there, we can evaluate the accuracy and optimization of each different model and choose the best model.

**Keywords.** Breast cancer, Mammography, Deep learning, Healthcare technology.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

## 1 GIỚI THIỆU

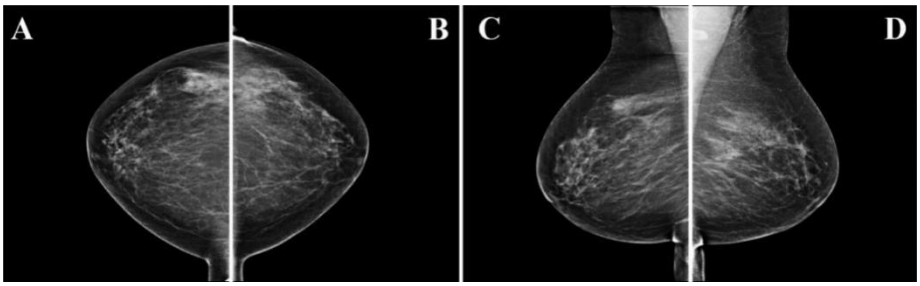
Ung thư vú là một trong những loại ung thư phổ biến nhất hiện nay thường xuất hiện ở phụ nữ. Theo Tổ chức Y tế thế giới (WHO) vào năm 2020 có khoảng 2,3 triệu phụ nữ trên thế giới được chẩn đoán mắc bệnh ung thư vú, trong đó có 685.000 ca tử vong trên toàn cầu. Nguyên nhân của nó vẫn chưa được hiểu rõ, mặc dù gen và hormone dường như đóng vai trò chính. Ung thư vú phát triển ban đầu như một biến đổi di truyền dẫn đến một loạt thay đổi phân tử trong các tế bào biểu mô lót các ống dẫn hoặc tiểu thùy của vú. Theo WHO ung thư vú phát sinh trong các tế bào biểu mô của ống dẫn 85% hoặc tiểu thùy 15% trong mô tuyến của vú. Ban đầu, khối ung thư phát triển giới hạn trong ống hoặc tiểu thùy không gây ra triệu chứng và có khả năng lây lan tối thiểu, theo thời gian, những khối ung thư này có thể tiến triển và xâm lấn mô vú xung quanh sau đó lan đến các hạch bạch huyết gần đó hoặc đến các cơ quan khác trong cơ thể khiến tình trạng sức khỏe của bệnh nhân tiến triển xấu. Để tránh những hậu quả nghiêm trọng của giai đoạn tiếp theo, việc phát hiện và ngăn chặn sự phát triển của tế bào ung thư này càng sớm càng có lợi không những giúp tăng cơ hội chữa khỏi mà còn cải thiện chất lượng cuộc sống của bệnh nhân. Hiện nay, các kỹ thuật hình ảnh như chụp cộng hưởng từ (MRI), chụp cắt lớp điện toán phát xạ đơn photon (SPECT), chụp cắt lớp vi tính (CT) trong đó chụp X-quang nhũ ảnh (mammography) là một kỹ thuật được dùng để sàng lọc và phát hiện sớm các bệnh lý về tuyến vú và ung thư vú, bệnh được phát hiện càng sớm thì khả năng điều trị khỏi bệnh sẽ càng cao. Bên cạnh những lợi ích mang lại thì chụp X-quang nhũ ảnh cũng có những hạn chế nhất định bao gồm không xác định được trạng thái lành hay ác tính của một tổn thương ở vú, việc đọc kết quả nhũ ảnh đòi hỏi nhân viên y tế phải có kinh nghiệm và trình độ chuyên môn cao, thời gian xử lý. Hơn nữa, quy trình đọc ảnh thủ công có thể mang lại kết quả chủ quan, dễ sai sót và gây quá tải cho bệnh viện. Trong số các kỹ thuật hình ảnh y học, chụp X-quang nhũ ảnh là phương pháp tiêu chuẩn vàng để chẩn đoán ung thư vú ở giai đoạn đầu khi chưa có dấu hiệu bệnh. Độ chính xác của kỹ thuật hình ảnh phụ thuộc vào nhiều yếu tố. Ví dụ, nó có độ nhạy thấp đối với bộ ngực có mật độ cao của phụ nữ trẻ. Chụp X-quang tuyến vú cũng có độ đặc hiệu thấp. Nhờ quá trình chụp ảnh nhanh chóng, chụp quang tuyến vú là hình ảnh phổ biến nhất nhưng chúng chỉ có thể được sử dụng ở giai đoạn chẩn đoán ban đầu. Bên cạnh những lợi ích mang lại thì chụp X-quang nhũ ảnh cũng có những hạn chế nhất định bao gồm không xác định được trạng thái lành hay ác tính của một tổn thương ở vú. Để đưa ra chẩn đoán ung thư vú, có một số công việc cần thực hiện bằng cách sử dụng hình ảnh để nghiên cứu bệnh một cách toàn diện. Việc phân tích kỹ lưỡng không chỉ giúp xác định sự tồn tại của ung thư mà còn có thể dự đoán tỷ lệ sống sót, khả năng tái phát, tiến triển và hiệu quả của quá trình điều trị. Như đã đề cập ở phần trước, có một số nhiệm vụ khác nhau như phân loại mật độ vú, phát hiện và phân loại vôi hóa, phát hiện và phân loại bất đối xứng vú, cũng như phát hiện và phân loại khối u. Nhìn chung, các nhiệm vụ này đòi hỏi nhiều thời gian và công sức. Các bác sĩ chuyên khoa phải dành rất nhiều thời gian để đọc và đánh giá các hình ảnh của mỗi trường hợp. Khi đối mặt với hàng nghìn ca bệnh, điều này có thể gây áp lực và quá tải cho bệnh viện. Hơn nữa, việc phân tích hình ảnh trở nên lặp đi lặp lại và đơn điệu sau một thời gian dài. Để tăng hiệu quả trong lĩnh vực này, cộng đồng nghiên cứu đã tập trung vào các giải pháp hỗ trợ máy tính, đặc biệt là sử dụng các cải tiến trong Deep Learning. Những tiến bộ này đã giúp cải thiện khả năng phân tích hình ảnh và tạo ra hiệu quả cao hơn trong quá trình chẩn đoán ung thư vú.

## 2 CÁC NGHIÊN CỨU VÀ CÔNG NGHỆ LIÊN QUAN

Trong nghiên cứu [1] nhóm tác giả đã sử dụng các phương pháp như Mạng thần kinh nhân tạo (ANN), Cây quyết định và Máy vectơ hỗ trợ (SVM) để trích xuất, phân loại và tóm tắt các mẫu ẩn trong hình ảnh từ các tập dữ liệu ung thư vú khác nhau. Kết quả đạt được cho thấy độ chính xác khá cao và nghiên cứu này đề xuất tiếp tục cải tiến bằng cách sử dụng các phương pháp học sâu. Trước đó năm 2018 trong bài nghiên cứu [2], Mohammed A. Al-masni và các đồng nghiệp đã đề xuất một phiên bản YOLO được sửa đổi để phát hiện và phân loại các khối u trong ảnh X-quang vú. Mô hình này có khả năng đồng thời phát hiện và phân loại ung thư vú, và kết quả thực nghiệm cho thấy độ đặc hiệu (Specificity) lên đến 94% đối với hai lớp Benign và Malignant. Trong bài nghiên cứu [3], nhóm tác giả đã sử dụng mạng GAN[4] và đề xuất phương pháp Autoencoder-GAN (AGAN) để tạo ra dữ liệu giả tạo đa dạng cho mạng CNN. Kết quả đánh giá cho thấy mô hình đạt độ chính xác 89,71% trong quá trình đánh giá. Trong bài nghiên cứu [5], nhóm tác giả nghiên cứu và ứng dụng hai mô hình CNN nổi tiếng đó là VGG-16 [6] và ResNet50 [7] để phân loại chụp quang tuyến vú trong bộ dữ liệu IRMA. Hai mô hình phân loại hình ảnh là lành tính hoặc ác tính. Cả hai đều chụp ảnh với độ phân giải 224x224. So sánh với nhau, VGG-16 có độ chính xác cao hơn 94% so

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

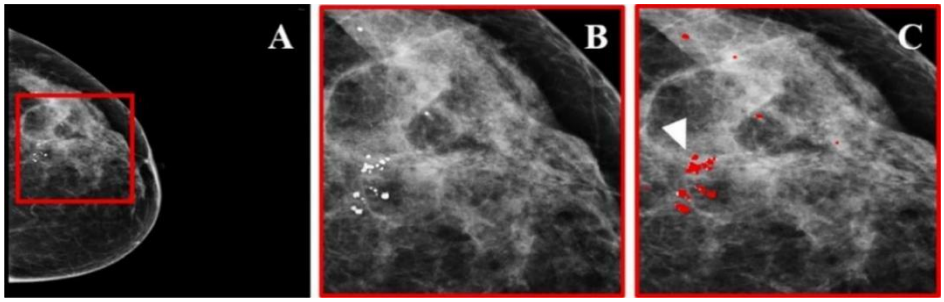
với ResNet50 91,7%. Điều này có thể được giải thích là do kiến trúc VGG-16 phức tạp hơn nhiều so với ResNet50. Qua các nghiên cứu trên đã cho ta cái nhìn tổng quan về các phương pháp thực hiện đã và đang từng bước ứng dụng học sâu vào bài toán phân loại tổn thương trên ảnh X-quang nhũ ảnh. Nghiên cứu này tập trung ứng dụng kết hợp các kỹ thuật xử lý ảnh với các mô hình học sâu để giải quyết vấn đề trên. Tiến hành phân tích đánh giá các kết quả thực nghiệm, kiểm thử và chọn ra mô hình học sâu tối ưu với độ chính xác và thời gian thực.




Hình 1: Ví dụ về các chế độ xem có sẵn từ phiên chụp nhũ ảnh: (A) Chế độ xem CC bên phải. (B) Chế độ xem CC bên trái. (C) Chế độ xem MLO bên phải. (D) Chế độ xem MLO bên trái [2]

Với sự phát triển của kỹ thuật học máy, các ứng dụng phân tích dữ liệu y sinh phát triển rất mạnh mẽ và mang lại các kết quả đáng khích lệ. Đặc biệt, các phương pháp học sâu đang được áp dụng trong các ứng dụng y tế khác nhau như phát hiện và xác định vị trí bất thường trong X-quang ngực. Chụp quang tuyến vú được chụp bằng tia X năng lượng thấp, nhạy cảm với vôi hóa. Loại hình ảnh này đã được xác nhận là có kết quả tốt hơn nhiều trong việc phát hiện các vôi hóa vi mô (MCs) và cụm vôi hóa (MCC) [8]. Hai hình ảnh của mỗi vú được chụp, tạo thành hai hình chiếu khác nhau: Cranio-Caudal (CC) và Medio-Lateral Oblique (MLO). Trong chụp quang tuyến vú CC, hình ảnh được chụp từ phía trên, trong khi ở MLO, hình chiếu từ một bên theo một góc sao cho có thể nhìn thấy cơ ngực được thể hiện qua Hình 1.

Các bác sĩ sẽ xem xét các hình ảnh X-quang tuyến vú thu được để xem xét có bất kì dấu hiệu bất thường nào hay không, sau đó đưa ra các cách điều trị thích hợp trong trường hợp phát hiện ung thư. Các vôi hóa vi mô (MC) là những chất lắng đọng canxi cực nhỏ thường xuất hiện ở vú. Chúng xuất hiện dưới dạng những điểm sáng trên phim chụp quang tuyến vú do hệ số suy giảm tia X của canxi cao hơn so với mô bình thường [9]. Hầu hết MC đều lành tính và không cần bất kỳ sự can thiệp nào (Hình 2A). Tuy nhiên, Cụm vôi hóa vi mô (MCC) được coi là tiền thân của bệnh ung thư (Hình 2). MC lành tính thường to hơn, tròn hơn, số lượng ít hơn và có kích thước, hình dạng đồng nhất. MC, nghi ngờ ung thư sớm, mọc thành cụm, nhỏ, có hình dạng và kích thước không đều và phân nhánh theo hướng. Một phát hiện đáng lo ngại khác trong chụp quang tuyến vú là một khối. Một khối ở vú có liên quan đến sưng cục bộ, lồi hoặc khối u bên trong vú. Các khối thường xuất hiện trong chụp quang tuyến vú dưới dạng các vùng tương đối dày đặc.



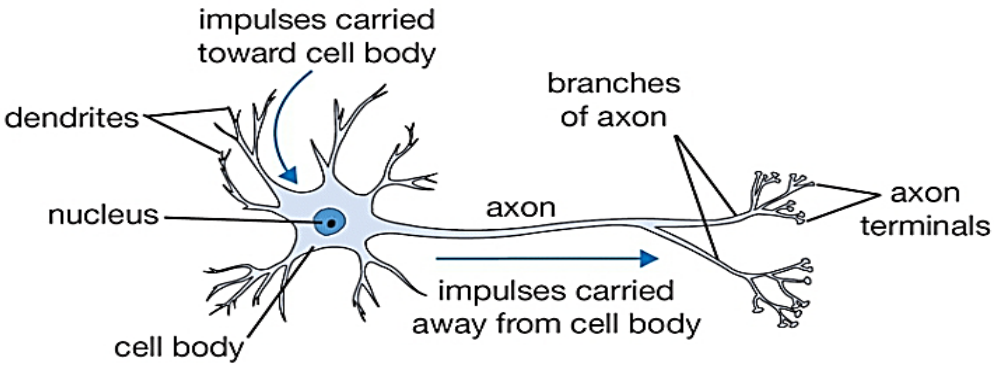

Hình 2. Chụp quang tuyến vú của một phụ nữ có Vôi hóa (MC) [2]

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Để giải quyết những thách thức của việc đánh giá ảnh X-quang tuyến vú, các hệ thống Máy tính hỗ trợ chẩn đoán (CAD) đang được phát triển với các công cụ tự động hoặc bán tự động để hỗ trợ các bác sĩ phát hiện và phân loại các tổn thương vú [10]. Các phương pháp tiếp cận như sử dụng kỹ thuật so khớp mẫu (Template Matching) và phân đoạn thường không hiệu quả, đặc biệt trong trường hợp các khu vực đáng ngờ bị che khuất bởi mô dày đặc hoặc da dày hơn bình thường [11]. Việc phân loại các bất thường ở nhũ ảnh là lành tính hay ác tính cũng rất khó khăn. Nó thường dẫn đến một số trường hợp Kết quả dương tính giả (FP) và hạn chế khả năng ứng dụng lâm sàng của các hệ thống CAD [12]. Liên quan đến một loại ung thư vú đặc biệt gọi là "Ung thư biểu mô ống dẫn" (DCIS), chụp nhũ ảnh cho thấy đây là một phương pháp chẩn đoán lý tưởng. Trong một hồ sơ thống kê, Chụp quang tuyến vú và Chụp cộng hưởng từ MRI có thể xác nhận DCIS tương ứng là 78,9% và 68,4% của một nhóm bệnh nhân được lấy mẫu [13]. Hiện nay, các kỹ thuật chụp nhũ ảnh cổ điển đã được cải tiến để nâng cao kỹ thuật chẩn đoán. Hai cải tiến nổi bật là quét 3D vú trong "Chụp nhũ ảnh tổng hợp kỹ thuật số" và "Chụp nhũ ảnh kỹ thuật số tăng cường độ tương phản" (CEDM), CEDM là truyền tĩnh mạch chất cản quang i-ốt cùng với kiểm tra chụp nhũ ảnh [14]. Kỹ thuật siêu âm thường không được sử dụng như một xét nghiệm tầm soát ung thư vú thông thường. Nhưng nó có thể hữu ích khi xem xét một số thay đổi ở vú, chẳng hạn như khối u đặc biệt là những khối u có thể sờ thấy nhưng không nhìn thấy trên phim chụp quang tuyến vú. Siêu âm có thể đặc biệt hữu ích ở những phụ nữ có mô vú dày đặc, điều này có thể khiến bạn khó nhìn thấy những vùng bất thường trên phim chụp quang tuyến vú. Nó cũng có thể được sử dụng để nhìn rõ hơn khu vực đáng ngờ đã được nhìn thấy trên hình chụp quang tuyến vú. Không giống như các kỹ thuật chụp quang tuyến vú, siêu âm hình ảnh MRI được đề xuất cho các bước chẩn đoán sau này, như là một phương pháp bổ sung, khi các tổn thương và khối u cần được nghiên cứu thêm. MRI có thể chỉ ra các chi tiết về các đối tượng nhắm mục tiêu bao gồm kích thước, hình dạng và vị trí nhờ các lần quét đa mặt cắt 3D [15]. Việc chụp ảnh MRI có thể tạo ra hình ảnh chất lượng và có độ đặc hiệu cao là rất tốn kém. Mặc dù MRI có thể tìm thấy một số bệnh ung thư không thấy trên chụp quang tuyến vú, nhưng nó cũng có nhiều khả năng tìm thấy những thứ hóa ra không phải là ung thư được gọi là dương tính giả. Điều này có thể dẫn đến việc một số phụ nữ phải làm các xét nghiệm và/hoặc sinh thiết mà cuối cùng lại không cần thiết. Đây là lý do tại sao MRI không được khuyến nghị làm xét nghiệm sàng lọc cho phụ nữ có nguy cơ mắc ung thư vú trung bình.

## 3 PHƯƠNG PHÁP TIẾP CẬN

### 3.1 Perceptrons



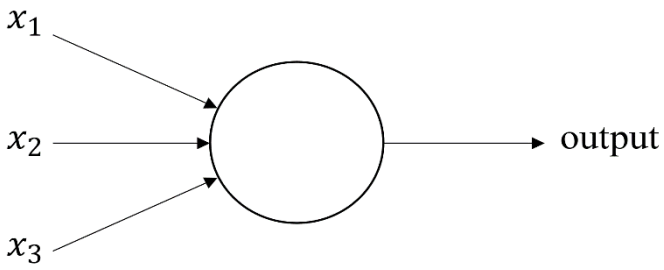

Hình 3: Nơ ron sinh học [21]

Một mạng nơ ron được cấu thành bởi các nơ ron đơn lẻ được gọi là các perceptrons. Trước tiên, ta tìm hiểu xem perceptron là gì sau đó sẽ tiến tới mô hình của mạng nơ ron. Nơ ron nhân tạo được lấy cảm hứng từ nơ ron sinh học như Hình 3.

Ta có thể thấy một nơ ron có thể nhận nhiều đầu vào (từ các dendrites) và cho ra một kết quả duy nhất (đi theo axon). Mô hình của perceptron cũng tương tự như vậy, được mô tả trong Hình 4.

Một perceptron sẽ nhận một hoặc nhiều đầu vào $(x_1, x_2, x_3, \dots)$ dạng nhị phân và cho ra một kết quả ouput dạng nhị phân duy nhất. Các đầu vào được điều phối tầm ảnh hưởng tới đầu ra thông qua tham số trọng lượng tương ứng $(w_1, w_2, w_3, \dots)$ của nó, còn kết quả đầu ra được quyết định dựa vào một ngưỡng quyết định nào đó.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 4: Nơ ron trong mạng ANN

$$output = \begin{cases} 0 \ \ if \ \ \sum_i w_i x_i \ \leq threshold \\ 1 \ \ if \ \ \sum_i w_i x_i \ > threshold \end{cases} \quad (1)$$

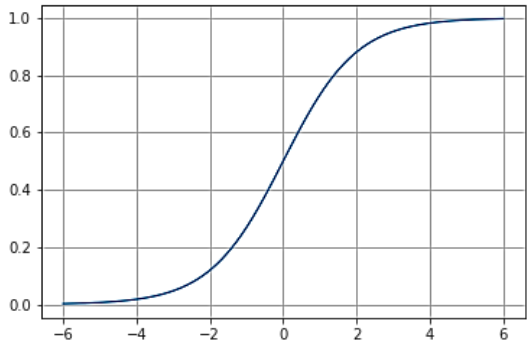
Đặt $b = -threshold$ , ta có thể viết lại thành:

$$output = \begin{cases} 0 \quad if \ \sum_i w_i x_i + b \leq 0 \\ 1 \quad if \ \sum_i w_i x_i + b \ > 0 \end{cases} \quad (2)$$

### 3.2 Hàm kích hoạt (Activation function)

Với đầu vào và đầu ra dạng nhị phân, ta rất khó có thể điều chỉnh một lượng nhỏ đầu vào để đầu ra thay đổi chút ít, nên để linh động, ta có thể mở rộng chúng ra cả khoảng $[0, 1]$. Lúc này đầu ra được quyết định bởi một hàm Sigmoid $\sigma(z)$ có công thức:

$$\sigma(z) = \frac{1}{1 + e^{-z}} \quad (3)$$




Hình 5: Đồ thị hàm Sigmoid

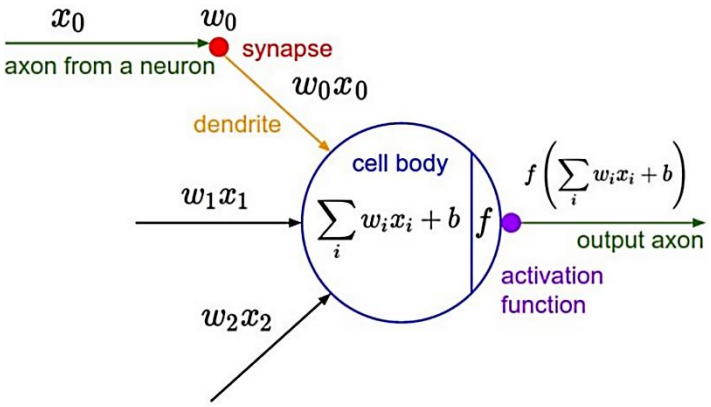
Đặt $z = w^T x$ thì công thức của perceptron lúc này sẽ có dạng:

$$\sigma(z) = \frac{1}{1 + e^{-(w^T x)}} \quad (4)$$

Một cách tổng quát, hàm tính toán đại diện cho perceptron được biểu diễn qua một hàm kích hoạt f(z) như sau:

$$output = f(z) = f(w^T x) \quad (5)$$

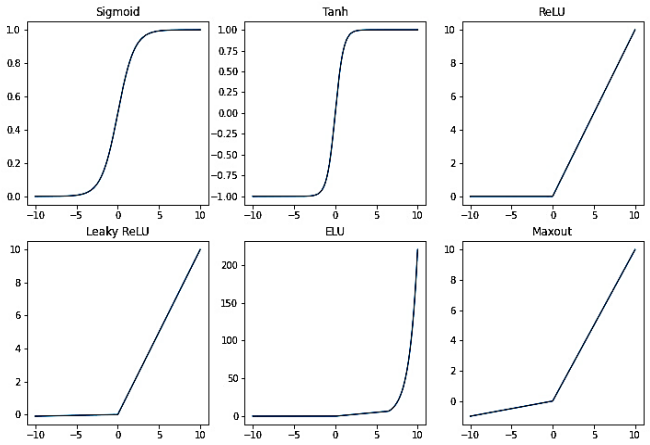
Bằng cách biểu diễn như vậy, ta có thể mô hình hóa nơ ron sinh học như sau:




Hình 6: Mô hình hóa nơ ron [22]

Một điểm cần lưu ý là các hàm kích hoạt buộc phải phi tuyến. Vì nếu nó là tuyến tính thì khi kết hợp với phép toán tuyến tính f(z) thì kết quả thu được cũng sẽ là một thao tác tuyến tính dẫn tới chuyện nó trở nên vô nghĩa. Ngoài hàm Sigmoid ra, ta còn có thể một số hàm như Tanh, ReLU, để thay thế hàm Sigmoid bởi dạng đồ thị của nó cũng tương tự như Sigmoid.

Hàm kích hoạt trong mạng nơ ron là hàm một hàm phi tuyến tính (nonlinearity) nhận đầu vào là các giá trị ở tầng trước, cho ra một tín hiệu (được kích hoạt) tương ứng với giá trị đầu vào. Mỗi hàm thường có ưu nhược điểm riêng, tuy nhiên trong mạng nơ ron, đặc biệt là các mạng học sâu, người ta ưa dùng hàm ReLU hoặc Leaky ReLU do khối lượng tính toán thấp đồng thời là luồng gradient không bị bão hòa ở phía phần dương, cùng với đó nó cho tốc độ hội tụ nhanh hơn so với hàm Sigmoid và hàm Tanh.

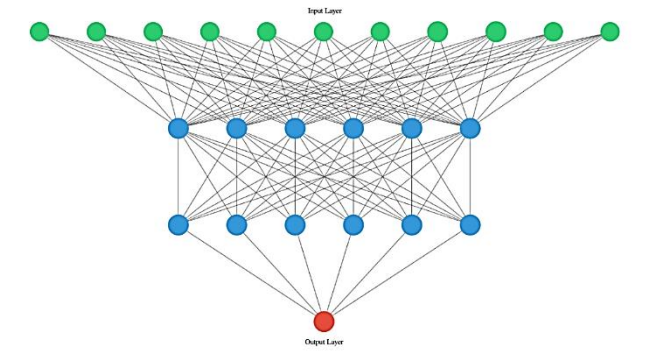



Hình 7: Các hàm kích hoạt

## 3.3 Mạng nơ ron cơ bản – Multilayer perceptrons (MLP)

Mạng nơ ron nhân tạo có tên tiếng Anh là Artificial Neural Networks (ANN) được tổ chức từ nhiều perceptrons (nodes) được chia thành các lớp (layer). Tại mỗi lớp các nodes có liên kết trọng số với các nodes ở lớp trước và sau. ANNs tạo nên một hệ thống xử lý thông tin, mô phỏng hệ thống nơ ron trong bộ não con người. Việc xử lý thông tin tại mỗi nơ ron gồm 2 phần: xử lý tín hiệu vào (tại input layer) và đưa tín hiệu ra (output layer). Hai lớp này tương tác với nhau thông qua một hoặc nhiều lớp ẩn (hidden layers) như trong Hình 2.8.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 8: Mô hình ANN [20]

Các phần tử trong các lớp khác nhau đều được liên kết có trọng số. Giá trị của các trọng số này có thể hiệu chỉnh bởi các quy tắc học (learning rules) thông qua các giá trị đầu vào mà nó sử dụng. Việc một ANN có thể học từ các dữ liệu mẫu cũng tương tự việc một đứa trẻ có thể phân biệt được hoa hồng thông qua các hình ảnh ví dụ về loài hoa hồng vậy. Việc huấn luyện mạng ANNs là việc đi tìm bộ trọng số kết nối thích hợp với dữ liệu đầu vào.

ANNs được huấn luyện theo ba kỹ thuật cơ bản đó là: học có giám sát (Supervised Learning), học không giám sát (Unsupervised Learning) và học tăng cường (Reinforcement Learning).

- **Học có giám sát (Supervised Learning):** Một thành phần quan trọng trong phương pháp này đó là người thầy với kiến thức vô cùng phong phú thể hiện qua tập dữ liệu huấn luyện. ANNs sẽ phải tìm cách thay đổi các tham số (parameter) và các ngưỡng (threshold) để ánh xạ các đầu vào thành các đầu ra mong muốn.
- **Học không giám sát (Unsupervised Learning):** Khác biệt với học có giám sát ở chỗ là đầu ra đúng tương ứng cho mỗi đầu vào là không biết trước. Trong học không có giám sát, một tập dữ liệu đầu vào được thu thập. Nó thường đối xử với các đối tượng đầu vào như là một tập các biến ngẫu nhiên. Sau đó, một mô hình mật độ kết hợp sẽ được xây dựng cho tập dữ liệu đó.
- **Học tăng cường (Reinforcement Learning):** Chiến lược xây dựng của phương pháp này thông qua những quan sát thông qua cơ chế thưởng phạt. Mạng nơ ron của nó sẽ phải quan sát môi trường xung quanh, thu thập thông tin và đưa ra các quyết định. Nếu những quan sát của nó không tốt, mạng nơ ron sẽ phải điều chỉnh các trọng số của nó để đưa ra các quyết định khác trong thời gian tới. Phương pháp học tăng cường thường phổ biến trong các robot.

## 3.4 Hàm mất mát (Loss function)

Khi khởi tạo bộ tham số đầu tiên cho mạng, kết quả tính ra ở output sẽ có sự sai khác so với output mong muốn (expected output), việc đào tạo mô hình cần làm là tối ưu hóa sao cho sự sai khác với expected output là nhỏ nhất. Việc tối ưu này dựa trên hàm mất mát của mạng (loss function). Hàm mất mát thể hiện tiêu chí của mô hình học, xem mô hình cần học những gì, phân tách các lớp ra sao.

Cho tập dữ liệu:

$$\{(x_i, y_i)\}_{i=1}^{N}$$

Trong đó $x_i$ là vector đặc trưng, $y_i$ là nhãn tương ứng (một số nguyên), N là số lượng mẫu dữ liệu trong tập dữ liệu. Hàm mất mát trên tập dữ liệu được định nghĩa là trung bình mất mát trên từng mẫu dữ liệu:

$$L = \frac{1}{N}\sum_{i}^{N} L_i(f(x_i, W), y_i) \quad (6)$$

Bên cạnh đó, để tránh mô hình bị overfitting với dữ liệu huấn luyện, người ta thường dùng những hàm phạt (regularization) để giúp mô hình đơn giản hơn và mang tính tổng quát cao hơn.

$$L = \frac{1}{N}\sum_{i}^{N} L_i(f(x_i, W), y_i) + \lambda R(W) \quad (7)$$

Hàm phạt R(W) có nhiều kiểu như chuẩn L1, L2,... Ở đây λ là một siêu tham số đại diện cho mức độ phạt mô hình để tránh việc các tham số weight (W) ảnh hưởng đa số đến hàm loss. Điều đó khiến các tham số weight đó cần phải điều chỉnh trong quá trình huấn luyện nhiều hơn hay ít hơn.
Trong huấn luyện mạng nơ ron, người ta có thể tự định nghĩa các hàm mất mát, nhưng thông thường, đối với bài toán phân loại người ta thường sử dụng 2 loại hàm mất mát sau:

- **Hàm mất mát Softmax:** Hàm này cực đại hóa xác suất nhãn thật sự của mẫu dữ liệu cho trước các đặc trưng của mẫu dữ liệu đó. Khi đó ta cần tìm cực tiểu của hàm

$$L_i = -\log\left(P(Y = y_i|X = x_i)\right) \tag{8}$$

Một cách khác để biểu diễn hàm này là sử dụng hàm Softmax thay cho công thức xác suất có điều kiện, từ đó công thức của hàm có dạng:

$$L_i = -\log\left(\frac{e^{s_{y_i}}}{\sum_j e^{s_j}}\right) \tag{9}$$

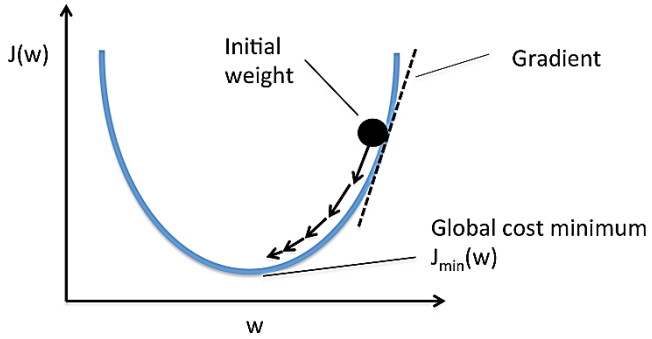
- **Hàm mất mát Cross Entropy**: Hàm này so sánh sự sai khác giữa 2 phân phối xác suất. Đầu tiên, ta chuyển vector score $s = f(x_i, W)$ về một phân phối xác suất bằng cách sử dụng hàm softmax, ta chuyển nhãn lớp của dữ liệu về dạng one-hot encoding (vị trí nhãn lớp có giá trị 1, các vị trí khác có giá trị 0). So sánh 2 phân phối xác suất, khi đó hàm loss được định nghĩa là:

$$L_i = -\sum_{c=1}^{M} y_{o,c}\log(p_{o,c}) \tag{10}$$

Trong đó: M là số lượng nhãn lớp,
y là 1 nếu c là nhãn của mẫu dữ liệu o,
y là 0 nếu ngược lại.
p là xác suất mô hình dự đoán mẫu dữ liệu có nhãn là c.

## 3.5 Gradient descent

Như ta đã biết gradient và sự biến thiên của hàm số thì hàm số sẽ tăng nhanh nhất theo hướng của gradient (gradient ascent) và giảm nhanh nhất theo hướng ngược của gradient (gradient descent - GD). Như vậy, một cách trực quan ta có thể nhận xét rằng nếu ta cứ đi ngược hướng đạo hàm mãi thì ta sẽ tới được điểm cực tiểu của hàm số. Việc này cũng tương tự như đặt một viên bi trên một con dốc nào đó thì nó sẽ lăn xuống dốc theo hướng nghiêng của dốc.



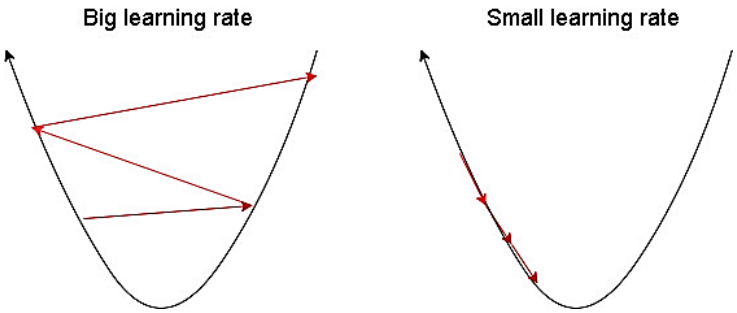

Hình 9: Mô phỏng giải thuật gradient descent [9]

Giả sử ta cần tìm tham số $\theta \in R^n$ để tối thiểu hoá hàm mất mát $J(\theta)$. Đầu tiên ta sẽ đặt θ tại một điểm bất kì nào đó, sau đó giải thuật gradient descent (GD) được thực hiện bằng cách cập nhật dần các tham số θ ngược với hướng của gradient $\nabla_\theta J(\theta)$ tại điểm hiện tại cho tới khi nó hội tụ về điểm nhỏ nhất. Tại mỗi bước cập nhật, ta sẽ dịch tham số bằng một lượng $\eta\nabla_\theta J(\theta)$ với tốc độ học (learning rate) $\eta > 0$ thể hiện cho việc dịch chuyển nhiều tới đâu:

$$\theta^{k+1} = \theta^k - \eta\nabla\theta J(\theta) \tag{11}$$


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

$\theta^k$ ở đây ký hiệu cho tham số bước cập nhật lần k khi thực hiện giải thuật GD.
*Việc chọn η có ý nghĩa rất lớn trong phương pháp này vì nó quyết định tới tính sống còn của giải thuật. Nếu η quá lớn thì mô hình không hội tụ được về đích, nhưng nếu η quá nhỏ thì nó lại mất nhiều thời gian để chạy giải thuật này như minh họa ở hình 9.*



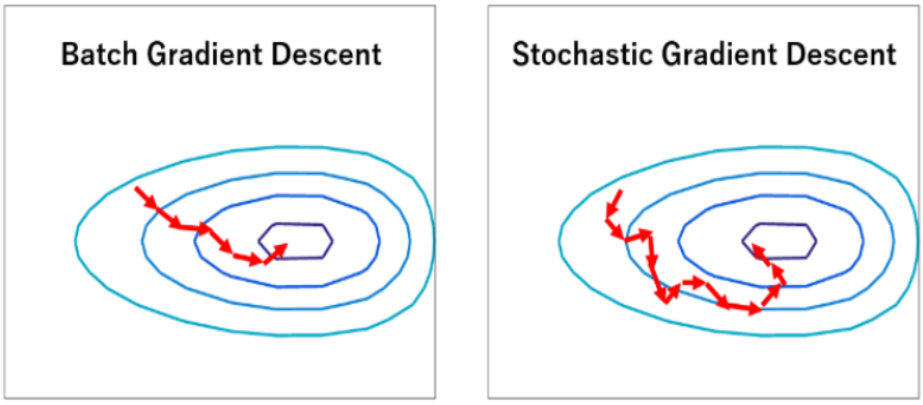

Hình 10: Mô phỏng so sánh giữa việc chọn learning rate

Ngoài ra, nếu để ý thấy thì nếu hàm mất mát J(θ) mà không lồi (non-convex) thì ta rất dễ bị rơi vào điểm tối thiểu cục bộ (local minimum) thay vì tiến tới điểm tối thiểu toàn cục (global minimum). *Việc chọn η lúc này có vai trò rất lớn vì nếu η hợp lý thì ta có thể vượt qua được điểm tối ưu cục bộ để tiến tiếp tới điểm tối ưu toàn cục.*

- *Giải thuật GD có các biến thể là* Stochastic Gradient Descent (SGD), Batch Gradient Descent (BGD) và Mini-Batch Gradient Descent (MGD).
- Stochastic Gradient Descent: Thay vì sử dụng toàn bộ tập dữ liệu để cập nhật tham số thì ta có thể sử dụng từng dữ liệu một để cập nhật. Phương pháp như vậy được gọi là *GD ngẫu nhiên* (Stochastic Gradient Descent). Về cơ bản ở mỗi lần cập nhật tham số, ta duyệt toàn bộ các cặp mẫu $(x^{(i)}, y^{(i)})$ và cập nhật tương tự như Batch Gradient Descent (BGD) như sau [16]:

$$\theta^{k+1} = \theta^k - \eta \nabla \theta J(\theta^k; x^i, y^i) \quad (12)$$

Vì sử dụng từng mẫu đơn một nên tốc độ tính toán đạo hàm sẽ nhanh hơn rất nhiều so với BGD nhưng nó phải trả cái giá là tốc độ hội tụ bị giảm đi. Một lưu ý khi cài đặt giải thuật này là mỗi bước cập nhật ta nên xáo trộn dữ liệu rồi mới lấy ra cập nhật.




Hình 11: So sánh BGD và SGD [9]

Việc này giúp giảm được sự đi lòng vòng về đích của giải thuật vì ta cho phép khả năng cập nhật ngẫu nhiên cho nó tức là sẽ có cơ hội nhảy được 1 bước xa hơn khi tính toán như Hình 10.
Mini-Batch Gradient Descent: Do SGD chạy chậm nên người ta thường sử dụng một phương pháp kết hợp giữa BGD và SGD là sử dụng từng nhóm dữ liệu để cập nhật tham số. Tức là ta sẽ chia dữ liệu ra thành nhiều lô khác nhau và mỗi lần cập nhật dữ liệu, thay vì sử dụng từng mẫu một ta sẽ sử dụng cả lô dữ liệu

một. Phương pháp như vậy được gọi là Mini-batch GD hay viết tắt là MGD. Như vậy ta thấy rằng nếu dữ liệu ta chỉ có 1 lô thì MGD chính là BGD, nếu mỗi lô chỉ có đúng 1 mẫu thì MGD sẽ là SGD.
Giả sử lô thứ i được kí hiệu là $(x^{(i)}, y^{(i)})$ thì công thức cập nhật được viết như sau:

$$\theta^{k+1} = \theta^k - \eta \nabla \theta J(\theta^k; x^i, y^i) \tag{13}$$

Cũng như SGD ta cũng sẽ xáo trộn dữ liệu trước khi phân lô cập nhật tham số.

## 3.6 Lan truyền xuôi (Feed Forward)

Dữ liệu từ tập huấn luyện thông qua lớp đầu vào sẽ được chuyển vào lớp tiếp theo. Tại mỗi nơ ron của mỗi lớp sẽ tiến hành thực hiện việc tính toán thông qua các hàm kết hợp, giá trị này sau khi truyền qua hàm kích hoạt là đầu ra mỗi nơ ron. Việc tính toán sẽ thực hiện trên tất cả các nơ ron của mạng và từ lớp đầu vào cho ra tới giá trị của lớp đầu ra. Sai số được tính bằng cách so sánh giá trị thực xuất ra của mạng với giá trị mong muốn, trong đó sai số của quá trình huấn luyện thường được lấy bằng tổng bình phương tất cả các sai số thành phần. Mỗi nơ ron ở một tầng nào đó sẽ nhận đầu vào là tất cả các đầu ra của nơ ron ở tầng trước đó mà không theo chiều ngược lại. Hay nói cách khác, việc suy luận trong mạng nơ ron là quá trình suy luận tiến hay lan truyền xuôi (feed forward):

$$z_i^{l+1} = \sum_{j=1}^{n^l} w_{ij}^{l+1} a_j^l + b_i^{l+1} \tag{14}$$

$$a_i^{l+1} = f(z_i^{l+1}) \tag{15}$$

Trong đó, $n^l$ số lượng nơ ron ở tầng l tương ứng và $a_j^l$ là nơ ron thứ j của tầng l. Còn $w_{ij}^{l+1}$ là tham số trọng lượng của đầu vào $a_j^l$ đối với nơ ron thứ i của tầng $l+1$ và $b_i^{l+1}$ là độ lệch (bias) của nơ ron thứ i của tầng $l+1$. Đầu ra của nơ ron này được biểu diễn bằng $a_i^{l+1}$ ứng với hàm kích hoạt $f(z_i^{l+1})$ tương ứng.
Riêng với tầng vào, thông thường $a^1$ cũng chính là các đầu vào x tương ứng của mạng. Để tiện tính toán, ta coi $a_0^l$ là một đầu vào và $w_{0j}^{l+1} = b_0^{l+1}$ là tham số trọng lượng của đầu vào này. Lúc đó ta có thể viết lại công thức trên dưới dạng vector:

$$z_i^{l+1} = w_{ij}^{l+1} . a^l \tag{16}$$

$$a_i^{l+1} = f(z_i^{l+1}) \tag{17}$$

Nếu nhóm các tham số của mỗi tầng thành một ma trận có các cột tương ứng với tham số mỗi nút mạng thì ta có thể tính toán cho toàn bộ các nút trong một tầng bằng vector:

$$z^{l+1} = w^{l+1} . a^l \tag{18}$$

$$a^{l+1} = f(z^{l+1}) \tag{19}$$

## 3.7 Lan truyền ngược (Backpropagation)

Vấn đề đặt ra với mạng nơ ron là nó bao gồm rất nhiều tham số, có thể lên đến hàng trăm triệu. Như thế, việc tính toán đạo hàm riêng với từng tham số bằng công thức là điều không khả thi. Giải thuật lan truyền ngược ra đời nhằm khắc phục nhược điểm này, bằng việc áp dụng quy tắc đạo hàm hàm hợp (chain rule) qua một đồ thị tính toán, giải thuật lan truyền ngược có khả năng tính toán đạo hàm riêng của hàm Loss với từng tham số, làm cho quá trình tối ưu của mạng nơ ron được khả thi.
Để tính đạo hàm của hàm lỗi $\nabla J(W)$ trong mạng nơ ron, ta sử dụng một giải thuật đặc biệt là giải thuật lan truyền ngược (backpropagation). Nhờ có giải thuật này mà mạng nơ ron thực thi hiệu quả được và ứng dụng ngày một nhiều cho tới tận ngày này.
Về cơ bản phương pháp này được dựa theo quy tắc đạo hàm hàm hợp và phép tính ngược đạo hàm để thu được đạo hàm theo tất cả các tham số cùng lúc chỉ với 2 lần duyệt mạng.
Giải thuật lan truyền ngược được thực hiện như sau:

1. Lan truyền xuôi:
2. Lần lượt tính các $a^l$ từ $l = 2 \rightarrow L$ theo công thức:

$$z^l = w^l . a^{l-1} \tag{20}$$

$$a^l = f(z^l) \tag{21}$$

Trong đó, tầng vào $a^1$ chính bằng giá trị đầu vào của mạng x.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

3. Tính đạo hàm z ở tầng ra:

$$\frac{\partial J}{\partial z^L} = \frac{\partial J}{\partial a^L}\frac{\partial a^L}{\partial z^L} \quad (22)$$

Với $a^L, z^L$ vừa tính được ở bước 1.

4. Lan truyền ngược:

Dựa trên sai số được tính từ quá trình lan truyền xuôi, mạng sẽ cập nhật lại các trọng số theo nguyên tắc lan truyền ngược sai số. Trong đó, kỹ thuật cơ bản được áp dụng trong quá trình cập nhật trọng số đó là gradient descent. Tính đạo hàm theo $z$ ngược lại từ $l = (L-1) \rightarrow 2$ theo công thức:

$$\frac{\partial J}{\partial z^L} = \frac{\partial J}{\partial z^{L+1}}\frac{\partial z^{L+1}}{\partial a^L}\frac{\partial a^L}{\partial z^L} = (w^{L+1})^T\frac{\partial J}{\partial z^{L+1}}\frac{\partial a^L}{\partial z^L} \quad (23)$$

Với $z^L$ tính được ở bước 1 và $\frac{\partial J}{\partial z^{L+1}}$ tính được ở vòng lặp ngay trước.

5. Tính đạo hàm:

Tính đạo hàm theo tham số $w$ bằng công thức:

$$\frac{\partial J}{\partial w^L} = \frac{\partial J}{\partial z^L}\frac{\partial z^L}{\partial w^L} = \frac{\partial J}{\partial z^L}(a^{L-1})^T \quad (24)$$

Với $a^{L-1}$ tính được ở bước 1 và $\frac{\partial J}{\partial z^{L+1}}$ tính được ở bước 3.

Như vậy, để huấn luyện hay để một mạng học từ dữ liệu thì mạng thực hiện 2 bước lan truyền xuôi và lan truyền ngược sai số. Quá trình này thực hiện cho tới khi sai số đạt được một ngưỡng nào đó hoặc thực hiện qua số bước lặp được người huấn luyện mạng đặt ra.

## 3.8 Convolutional Neural Network (CNN)

### 3.8.1 Khái niệm convolutional Neural Network (CNN)

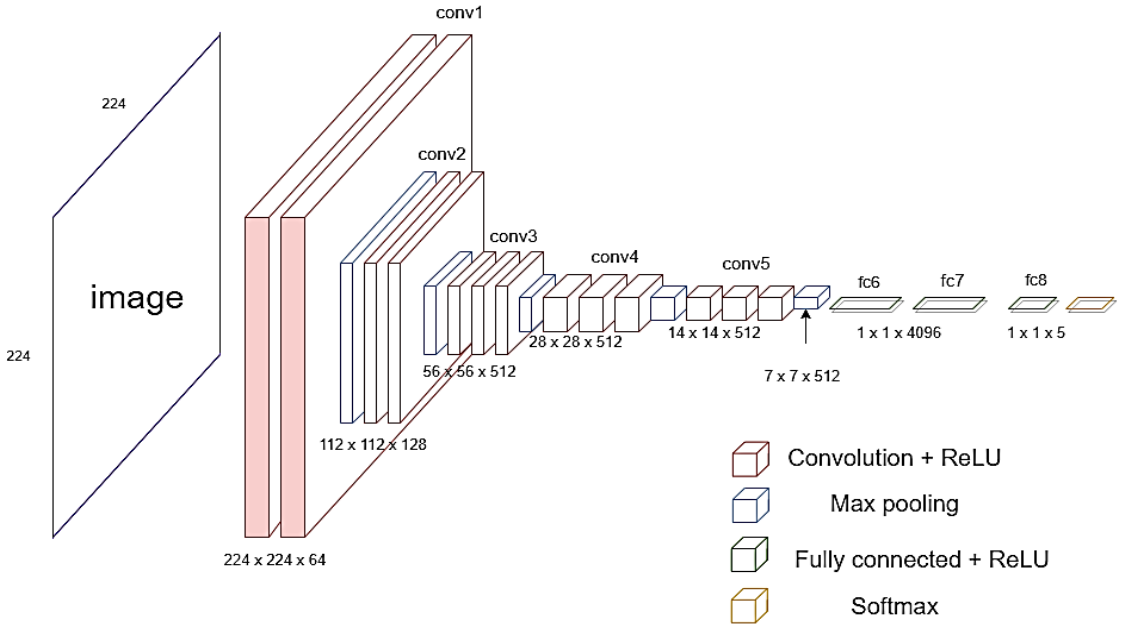
Convolutional Neural Network (CNN) hay còn gọi là mạng nơ ron chập, là một trong những mô hình Deep Learning tiên tiến giúp cho chúng ta xây dựng được những hệ thống thông minh với độ chính xác cao. Cụ thể hơn, mạng chập sử dụng phép chập để trích trọn đặc trưng của một bức ảnh để phù hợp với những yêu cầu cụ thể trong lĩnh vực thị giác máy tính (Computer Vision). Trong những thập kỉ trước, mạng trí tuệ nhân tạo (Artificial Neural Network – ANN) đã được nghiên cứu nhiều và ứng dụng rộng rãi trong các bài toán nhận dạng. Tuy nhiên, ANN không thể hiện tốt lắm đối với các dữ liệu hình ảnh. Chính sự liên kết quá đầy đủ về thông tin ảnh đã tạo nên những hạn chế cho mô hình. Đặc biệt, dữ liệu hình ảnh có kích thước khá lớn. Ví dụ một tấm ảnh xám có kích thước $32 \times 32$ (pixels) sẽ cho ra vector đặc trưng có $32 \times 32 = 1024$ chiều, còn đối với ảnh màu cùng kích thước sẽ là $1024 \times 3 = 3072$ chiều. Điều này cũng có nghĩa là cần tới 3072 trọng số kết nối giữa lớp đầu vào và một nút (node) ở lớp n kế tiếp. Số lượng trọng số sẽ càng nhân rộng hơn nữa nếu số lượng node trong lớp n tăng lên và số lượng các lớp trong mạng tăng lên. Như vậy chỉ với một bức ảnh nhỏ $32 \times 32$ thì cũng cần đến một mô hình khá đồ sộ. Điều này khiến cho việc thao tác với các ảnh có kích thước lớn hơn trở nên khó khăn chẳng hạn như các ảnh HD $(1080 \times 720)$ hoặc Full HD $(1920 \times 1080)$. Dựa trên tư tưởng này, CNN ra đời với một kiến trúc khác so với mạng truyền thẳng. Thay vì toàn bộ ảnh nối với một node thì chỉ có một phần cục bộ trong ảnh nối đến một node trong lớp tiếp theo (local connectivity). Dữ liệu hình ảnh thông qua các lớp của mô hình này sẽ được "học" ra các đặc trưng để tiến hành phân lớp một cách hiệu quả. CNN có kiến trúc đặc biệt hơn so với mạng ANN và không giống một mạng thần kinh truyền thống. Các lớp của một CNN sắp xếp theo 3 chiều: chiều rộng, chiều cao và chiều sâu. Trong mô hình CNN, các lớp liên kết được với nhau thông qua cơ chế Convolution. Lớp tiếp theo là kết quả Convolution từ lớp trước đó, nhờ vậy mà ta có được các kết nối cục bộ. Nghĩa là mỗi nơ ron ở lớp tiếp theo sinh ra từ các bộ lọc (filter) áp đặt lên một vùng ảnh cục bộ của nơ ron lớp trước đó. Mỗi lớp như vậy được áp đặt các bộ lọc khác nhau, thông thường có vài trăm đến vài nghìn bộ lọc như vậy. Một số lớp khác như Pooling/Subsampling dùng để chắt lọc lại các thông tin hữu ích hơn (loại bỏ các thông tin nhiễu). Trong suốt quá trình huấn luyện, CNN sẽ tự động học được các thông số cho các bộ lọc. Với đặc điểm cho phép trích chọn các đặc trưng của ảnh thông qua các lớp Convolution đồng thời kết hợp với đặc điểm của các lớp ẩn trong mạng nơ ron (Neural Network – NN), việc ứng dụng mạng CNN trong

các bài toán phân tích ảnh đã mở ra một hướng đi mới đầy triển vọng để giải quyết các bài toán liên quan đến phân tích ảnh.

### 3.8.2 Cấu trúc mạng CNN

Như đã đề cập ở trên, một CNN là một dãy các lớp và các lớp của một CNN chuyển một khối kích hoạt sang một lớp khác thông qua một hàm khác biệt. Về cơ bản mô hình mạng nơ ron xoắn bao gồm các lớp sau:

- Lớp Convolution
- Lớp Pooling
- Lớp Fully Connected



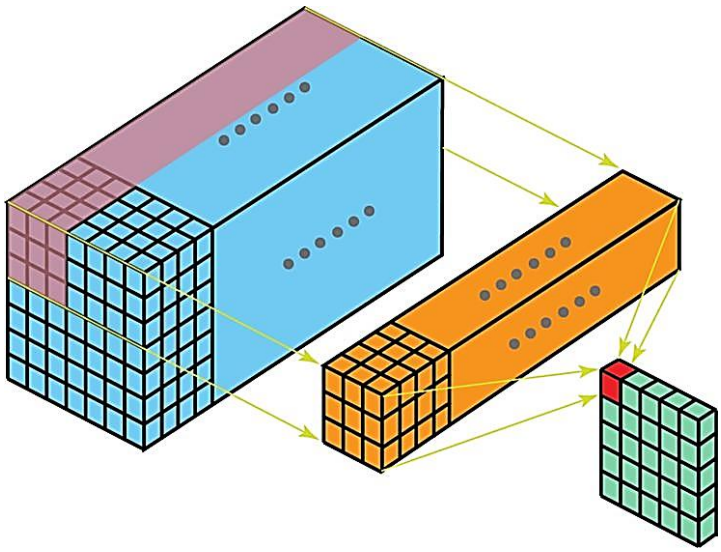

Hình 12: Mạng nơ ron tích chập

Các lớp này được xếp chồng lên nhau để tạo thành kiến trúc CNN đầy đủ. Sự sắp xếp về số lượng và thứ tự giữa các lớp này sẽ tạo ra những mô hình khác nhau phù hợp cho các bài toán khác nhau.
Về kỹ thuật, mô hình CNN để huấn luyện và kiểm tra, mỗi hình ảnh đầu vào sẽ chuyển nó qua một loạt các lớp chập với các bộ lọc (kernels hoặc filters), tổng hợp lại các lớp được kết nối đầy đủ (Fully Connected) và áp dụng hàm Softmax để phân loại đối tượng có giá trị xác suất giữa 0 và 1. Hình 12 là toàn bộ luồng CNN để xử lý hình ảnh đầu vào và phân loại các đối tượng dựa trên giá trị.

#### 3.8.2.1 Lớp Convolution

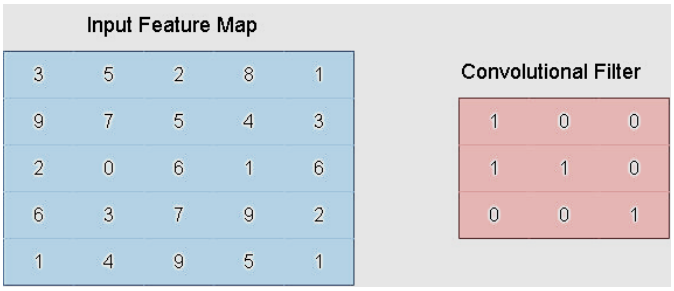
Lớp Convolution là lớp đầu tiên để trích xuất các đặc trưng từ hình ảnh đầu vào. Phép chập duy trì mối quan hệ giữa các pixel bằng cách tìm hiểu các đặc trưng hình ảnh bằng cách sử dụng các ô vuông nhỏ của dữ liệu đầu vào như minh họa ở Hình 12 và cụ thể hơn ở Hình 13.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh



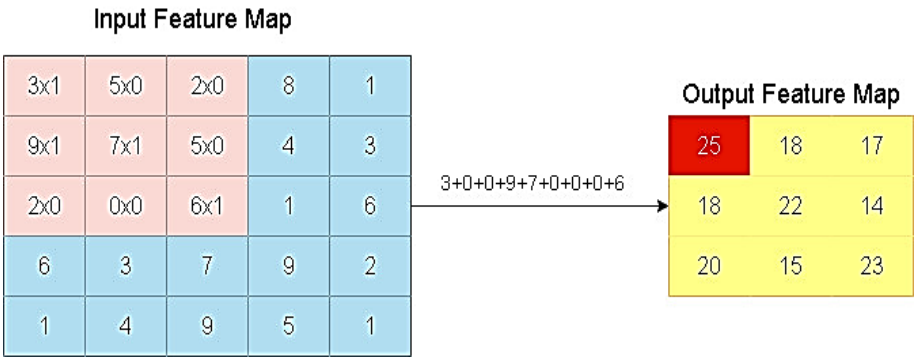
Hình 13: Minh họa phép chập [22]

Nó là một phép toán có 2 đầu vào như ma trận hình ảnh và 1 bộ lọc hoặc hạt nhân. Xem xét 1 ma trận 5 × 5 có giá trị pixel là 0 và 1. Ma trận bộ lọc 3 × 3 như hình 14.




Hình 14: Phép tích chập 5x5 và bộ lọc 3x3

Sau đó, lớp chập của ma trận hình ảnh 5 × 5 chập với ma trận bộ lọc 3 × 3 gọi là Feature Map như Hình 15 Sự kết hợp của 1 hình ảnh với các bộ lọc khác nhau có thể thực hiện các hoạt động như phát hiện cạnh, làm mờ và làm sắc nét bằng cách áp dụng các bộ lọc.

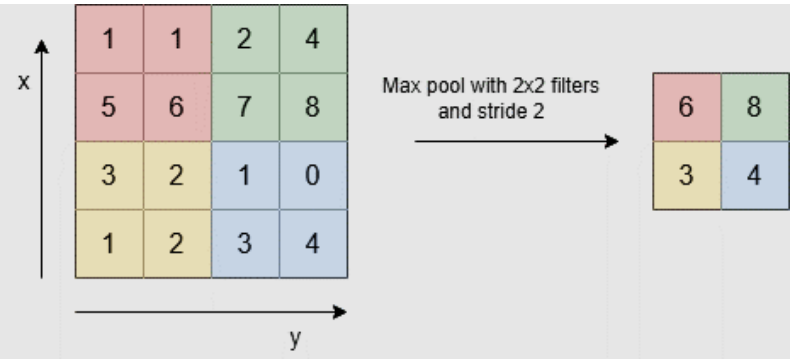



Hình 15: Kết quả tính feature map

### 3.8.2.2 Lớp Pooling

Lớp này sử dụng một cửa sổ trượt quét qua toàn bộ ảnh dữ liệu, mỗi lần trượt theo một bước trượt (stride) cho trước. Khác với lớp Convolution, lớp Pooling không tính xoắn mà tiến hành lấy mẫu (subsampling). Khi cửa sổ trượt trên ảnh, chỉ có một giá trị được xem là giá trị đại diện cho thông tin ảnh tại vùng đó (giá

trị mẫu) được giữ lại. Các phương thức lấy phổ biến trong lớp Pooling là Max Pooling (lấy giá trị lớn nhất), Sum Pooling (lấy giá trị tổng) và Average Pooling (lấy giá trị trung bình).
Hình 2.16 là một ví dụ về phương thức Max Pooling, sử dụng filter có kích thước $2 \times 2$, áp dụng vào ảnh kích thước $4 \times 4$ với bước trượt $\text{stride} = 2$. Filter sẽ lần lượt duyệt qua ảnh, với mỗi lần duyệt chỉ có giá trị lớn nhất trong 4 giá trị nằm trong vùng cửa sổ $2 \times 2$ của filter được giữ lại và đưa ra đầu ra. Như vậy sau khi qua lớp Pooling, ảnh sẽ giảm kích thước xuống còn $2 \times 2$ (kích thước mỗi chiều giảm 2 lần).
Lớp Pooling có vai trò giảm kích thước dữ liệu. Với một bức ảnh kích thước lớn qua nhiều lớp Pooling sẽ được thu nhỏ lại, tuy nhiên vẫn giữ được những đặc trưng cần cho việc nhận dạng thông qua cách lấy mẫu. Việc giảm kích thước dữ liệu sẽ làm giảm lượng tham số, tăng hiệu quả tính toán và góp phần kiểm soát hiện tượng quá khớp (overfitting).




Hình 16: Tính toán Max Pooling

### 3.8.2.3 Lớp Fully Connected

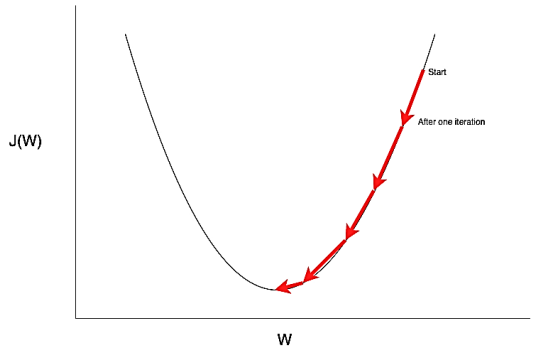
Lớp này tương tự với lớp trong mạng nơ ron truyền thẳng, các giá trị ảnh được liên kết đầy đủ vào nút trong lớp tiếp theo. Sau khi ảnh được xử lý và rút trích đặc trưng từ các lớp trước đó, dữ liệu ảnh sẽ không còn quá lớn so với mô hình truyền thẳng nên ta có thể sử dụng mô hình truyền thẳng để tiến hành nhận dạng. Tóm lại, lớp Fully Connected đóng vai trò như một mô hình phân lớp và tiến hành dựa trên dữ liệu đã được xử lý ở các lớp trước đó.

## 3.9 Kiến trúc mạng ResNet

### 3.9.1 Hiện tượng biến mất và bùng nổ radients

Như đã nêu ở mục trên, giải thuật cho thuật toán lan truyền ngược là một kĩ thuật thường được sử dụng trong trong quá trình huấn luyện mạng nơ ron sâu (Deep Neural Networks – DNN). Ý tưởng chung của thuật toán là sẽ đi từ output layer đến input layer và tính toán gradient của hàm cost tương ứng cho từng tham số weight của mạng. Sau đó, thuật toán Gradient Descent sẽ được sử dụng để cập nhật các tham số đó.
Quá trình trên sẽ được lặp lại cho tới khi các tham số của mạng hội tụ. Thông thường chúng ta sẽ có một siêu tham số định nghĩa cho số lượng vòng lặp để thực hiện quá trình trên. Siêu tham số đó thường được gọi là số epoch (hay số lần mà toàn bộ tập huấn luyện được duyệt qua một lần và weights được cập nhật). Nếu số lượng vòng lặp quá nhỏ, DNN có thể sẽ không cho ra kết quả tốt, và ngược lại thì thời gian huấn luyện sẽ quá dài nếu số lượng vòng lặp quá lớn. Ở đây, ta có một sự đánh đổi giữa độ chính xác và thời gian huấn luyện.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

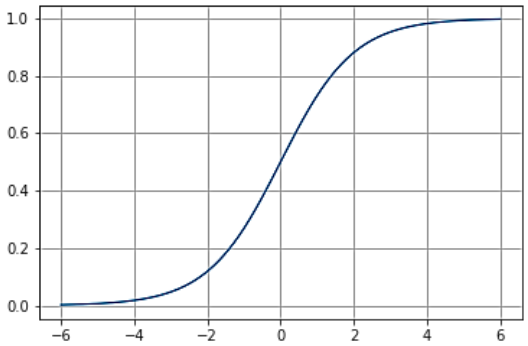



Hình 17: Minh họa đơn giản thuật toán gradient descent

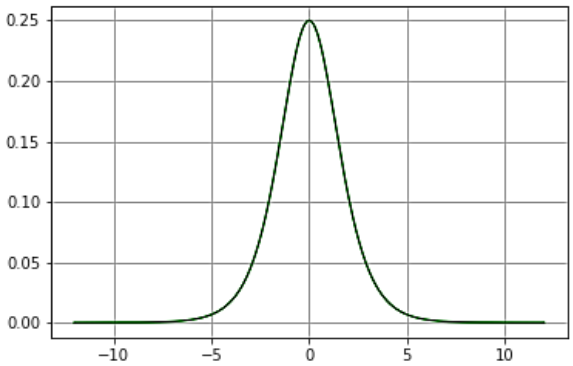
Tuy nhiên, trên thực tế, gradients thường sẽ có giá trị nhỏ dần khi đi xuống các lớp thấp hơn. Kết quả là các cập nhật thực hiện bởi Gradient Descent không làm thay đổi nhiều weights của các lớp đó, khiến chúng không thể hội tụ và DNN sẽ không thu được kết quả tốt. Hiện tượng này được gọi là mất mát gradients (Vanishing Gradients). Trong Hình 17, hàm cost có dạng đường cong dẹt, chúng ta sẽ cần khá nhiều lần cập nhật để tìm được điểm cực tiểu toàn cục (global minimum). Trong nhiều trường hợp khác, gradients có thể có giá trị lớn hơn trong quá trình lan truyền ngược, khiến một số lớp có giá trị cập nhật cho weights quá lớn khiến chúng phân kỳ (không hội tụ), tất nhiên, DNN cũng sẽ không có kết quả như mong muốn. Hiện tượng này được gọi là bùng nổ gradients (Exploding Gradients) và thường gặp khi sử dụng Recurrent Neural Networks (RNNs). Chung quy lại, trong quá trình huấn luyện DNN chúng ta có thể gặp phải các vấn đề liên quan đến việc gradients không ổn định khiến cho tốc độ học của các lớp khác nhau chênh lệch khá nhiều. Một trong những hàm kích hoạt phi tuyến khá phổ biến trong những giai đoạn đầu của mạng nơ ron là **logistic sigmoid activation function**; tuy nhiên hàm này có một số nhược điểm khiến quá trình huấn luyện mạng nơ ron gặp nhiều khó khăn. Về kỹ thuật **weight initialization, random initialization** sử dụng phân phối chuẩn (normal distribution) với kỳ vọng (mean) là 0 và độ lệch chuẩn (standard deviation) là 1. Chung quy lại, việc sử dụng hàm kích hoạt Sigmoid cùng với khởi tạo ngẫu nhiên (random initialization) khiến cho phương sai của các đầu ra của mỗi lớp lớn hơn khá nhiều so với phương sai của inputs cho lớp đó. Trong chiều đi xuôi của mạng (forward), các giá trị phương sai sẽ tăng dần và hàm kích hoạt sẽ trở nên bão hòa ở những lớp phía trên. Ta đi phân tích cụ thể với hàm Sigmoid – một hàm khá phổ biến với ANN hay DNN.

$$Sigmoid(z) = f(z) = \frac{1}{1 + e^{-z}} \tag{25}$$

$$f'(z) = f(z)(1 - f(z)) = \frac{1}{1 + e^{-z}}\left(1 - \frac{1}{1 + e^{-z}}\right) \tag{26}$$




Hình 18: Đồ thị hàm số Sigmoid

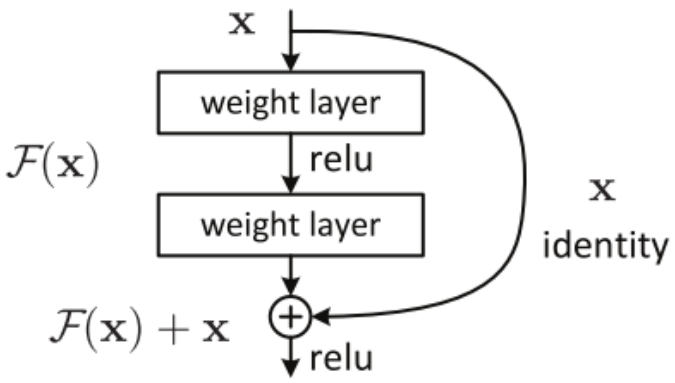



Hình 19: Đồ thị đạo hàm của hàm số Sigmoid

Quan sát hàm Sigmoid với đồ thị đạo hàm của hàm Sigmoid tại Hình 18 và Hình 19 tương ứng, chúng ta thấy hàm sẽ chuyển đổi các đầu vào thành các giá trị nằm trong khoảng $(0, 1)$, khá phù hợp khi chúng ta muốn tính toán xác suất hay thực hiện các bài toán phân lớp. Tuy nhiên khi các đầu vào khá lớn (âm hoặc dương), hàm này sẽ bão hòa tại 0 và 1, và đạo hàm sẽ rất gần với 0. Gradients nhỏ sẽ khiến cho quá trình lan truyền ngược trở nên khó khăn hơn. Quan sát đồ thị đạo hàm của hàm Sigmoid chúng ta thấy cực đại toàn cục (global maximum) có giá trị 0.25 và đồ thị có tiệm cận tại 0. Nói cách khác output của đạo hàm của hàm sigmoid sẽ có giá trị nằm trong khoảng nửa đóng $(0, 0.25]$. Trong quá trình tính toán gradients lỗi khi thực hiện lan truyền ngược, giá trị của các gradients sẽ phụ thuộc vào output của hàm sigmoid. Mặt khác nếu chúng ta sử dụng kỹ thuật random initialization nói trên, giá trị của weights sẽ nằm trong đoạn $[-1, 1]$. Nếu sử dụng quy tắc đạo hàm hàm hợp để tính toán gradient cho một lớp dựa vào gradients của các lớp trước đó thì giá trị gradient sẽ giảm đi rất nhanh khi chúng ta di chuyển xuống các lớp thấp hơn. Có thể thấy nhân một loạt các số nhỏ hơn 1 với nhau thì kết quả sẽ giảm đi rất nhanh. Đối với vấn đề vanishing gradient có thể có nhiều cách giải quyết như Batch Normalization hay sử dụng khối Residual trong ResNet hoặc thay đổi thuật toán khởi tạo tham số ban đầu.

### 3.9.2 Khối Residual

Giải pháp mà ResNet [7] đưa ra là sử dụng kết nối "tắt" đồng nhất để xuyên qua một hay nhiều lớp. Một khối như vậy được gọi là một khối Residual, như trong Hình 20.




Hình 20: Khối Residual

ResNet gần như tương tự với các mạng gồm có Convolution, Pooling, Activation và lớp Fully Connected. Hình 20 mô tả khối Residual được sử dụng trong mạng. Xuất hiện một mũi tên cong xuất phát từ đầu và kết thúc tại cuối khối Residual. Với H(x) là giá trị dự đoán, F(x) là giá trị thật (nhãn), chúng ta muốn H(x) bằng hoặc xấp xỉ F(x). Việc F(x) có được từ x như sau:

$$X \rightarrow weight_1 \rightarrow ReLU \rightarrow weight_2$$

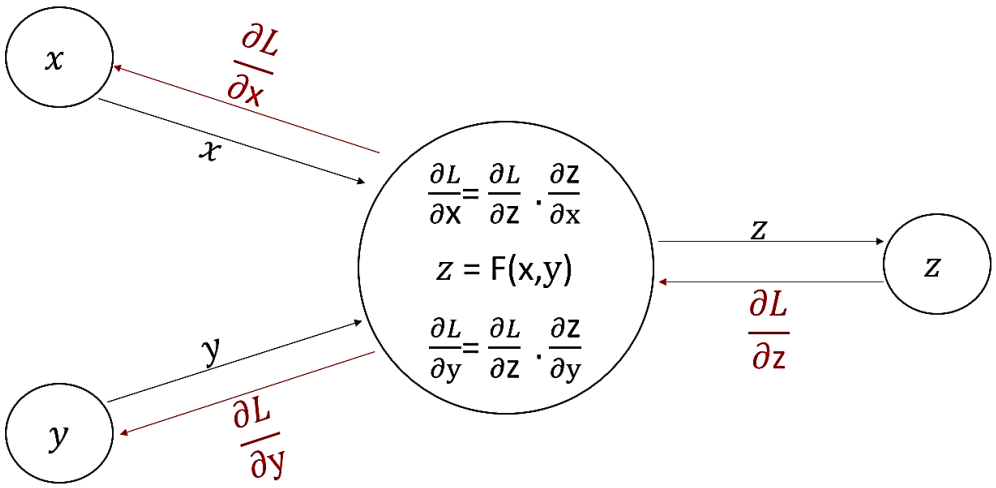
Giá trị H(x) có được bằng cách:

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

$$F(x) + x \rightarrow ReLU$$

Ta phân tích để lý giải cách xử lý vấn đề vanishing gradients với việc sử dụng khối Residual của mạng ResNet:

- Trọng số của mạng nơ ron được cập nhật bằng cách sử dụng thuật toán lan truyền ngược. Thuật toán lan truyền ngược tạo ra một thay đổi nhỏ đối với từng trọng số theo cách giảm tổn thất của mô hình. Nó cập nhật từng trọng lượng để nó thực hiện một bước theo hướng mà sự mất mát giảm xuống. Hướng này chính là gradient của trọng lượng này (đối với sự mất mát) [17].




Hình 21: Dòng chảy của gradient qua nơ ron [11]

- Sử dụng quy tắc chuỗi, chúng ta có thể tìm thấy gradient này cho mỗi trọng lượng. Nó được tính bằng (gradient cục bộ) x (gradient chảy từ phía trước).

Vấn đề là khi gradient này tiếp tục chảy ngược trở lại các lớp ban đầu, giá trị này tiếp tục được nhân với mỗi gradient cục bộ. Do đó, gradient trở nên nhỏ hơn và nhỏ hơn, làm cho các cập nhật cho các lớp ban đầu rất nhỏ, làm tăng thời gian huấn luyện đáng kể. Chúng ta có thể giải quyết vấn đề này nếu "gradient cục bộ" bằng cách nào đó trở thành 1 với kiến trúc tính toán được minh họa ở Hình 21.

$$z = f(x,y) = x + g(y) \tag{27}$$

$$\frac{\partial z}{\partial x} = 1(\text{Local gradient always equals 1}) \tag{28}$$

$$\frac{\partial z}{\partial y} = \frac{\partial g(y)}{\partial y} \tag{29}$$

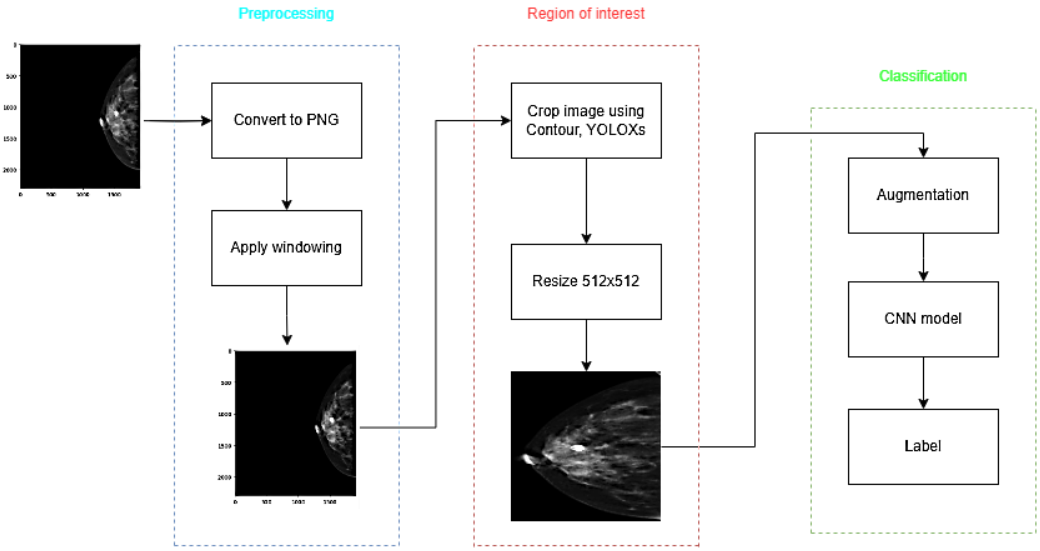
Gradient này được sao chép ngược, nó không giảm giá trị vì gradient cục bộ là 1.

## 4 MÔ HÌNH CHO BÀI TOÁN PHÁT HIỆN UNG THƯ VÚ

### 4.1 Kiến trúc mô hình

Mô hình được chia làm ba giai đoạn là: tiền xử lý (preprocessing), phân tích vùng quan tâm (Region of interest), phân loại (classification). Trong giai đoạn tiền xử lý, ảnh đầu vào được chuyển đổi từ ảnh Digital Image and Communications in Medicine (DICOM) sang dạng PNG. Đầu tiên, chúng tôi đọc ảnh và chuyển đổi ảnh thành dạng mảng pixel. Tiếp đến, sử dụng Windowing, ảnh DICOM thường chứa các giá trị lớn 255 (8bit) việc chuẩn hóa các giá trị pixel dẫn đến việc mất thông tin ảnh. Windowing cho phép chúng ta chọn dải pixel đặc biệt từ ảnh gốc trước khi chuẩn hóa. Điều này có tác dụng làm tăng độ tương phản cao hơn giữa các mô mềm và mô đặc biệt tiếp theo là sử dụng thuật toán nén ảnh cơ bản để chuyển về ảnh dạng

PNG. Ở giai đoạn Region of interest chúng tôi thử nghiệm 2 phương pháp, đầu tiên là áp dụng thuật toán Contour (hay còn gọi là thuật toán tìm viền) để tìm đường viền sau đó sẽ dựa vào đường viền đó xác định được vùng quan tâm. Tuy nhiên sẽ có các trường hợp hình ảnh đặc biệt mà thuật toán không thể tối ưu hết các trường hợp nên chúng tôi tiếp cận đến phương pháp thứ hai là sử dụng mô hình YOLOX và kết quả rất khả quan sẽ được chúng tôi trình bày ở phần kết quả và thực nghiệm. Đối với giai đoạn phân loại, dữ liệu sau khi được tiền xử lý và áp dụng một số phương pháp tăng cường dữ liệu sẽ được huấn luyện trên các mô hình học sâu CNN khác nhau như: EfficientNets, HRNet và đánh giá chúng để chọn ra mô hình dự đoán tốt nhất.




Hình 22: Các giai đoạn xử lý mô hình

## 4.2 Tiền xử lý

### 4.2.1 Chuyển ảnh DICOM sang ảnh PNG

Đầu tiên chúng tôi đọc ảnh và chuyển ảnh thành dạng mảng pixel. Tiếp đến chúng tôi sử dụng công thức sau để nén ảnh giảm dung lượng ảnh xuống.

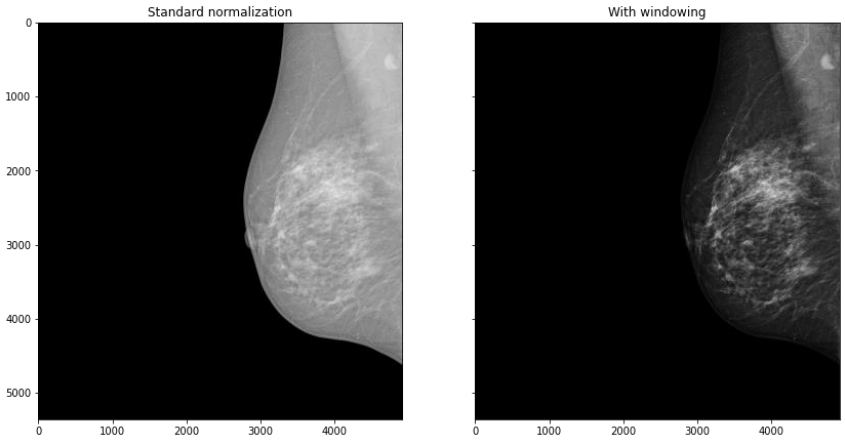
$$IMG = \frac{(IMG - \min(IMG))}{\max(IMG) - \min(IMG)} \tag{30}$$

Dữ liệu dicom có 2 dạng là MONOCHROME1 và MONOCHROME2 nó dùng để xác định cách hiển thị dữ liệu pixel ở dạng ảnh đơn sắc. Hai thông số này được lưu vào giá trị PhotometricInterpretation. Nếu là MONOCHROME1 nghĩa là giá trị pixel cao được hiển thị dưới dạng đen hơn. Còn MONOCHROME2 nghĩa là giá trị pixel cao được hiển thị dưới dạng trắng hơn. Chính vì vậy chúng tôi sẽ chọn MONOCHROME2 làm dạng chính nếu ảnh đang ở dạng MONOCHROME1 thì chúng tôi sẽ lấy 1 - các pixel ảnh đã được nén chuyển đổi thành dạng MONOCHROME2. Sau khi nén, chúng tôi sẽ nhân ngược 255 vào các pixel đã được nén giúp tăng độ sáng của ảnh lên lúc này chúng ta đã được ảnh dạng png với dung lượng ít hơn rất nhiều so với dicom.

### 4.2.2 Sử dụng windowing cho DICOM

Chúng tôi sử dụng Windowing còn được gọi là ánh xạ mức xám cho phép chọn các dãy pixel đặc biệt từ ảnh góc trước khi chuẩn hóa, điều này có tác dụng làm tăng độ tương phản cao hơn giữa các mô mềm và các phầm mô đặc biệt ngoài ra còn cho phép phạm vi lớn hơn khi điều chỉnh độ sáng/độ tương phản theo cách thủ công sau này. Kỹ thuật này nên được áp dụng cho việc xuất hình ảnh dưới dạng JPG/PNG.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

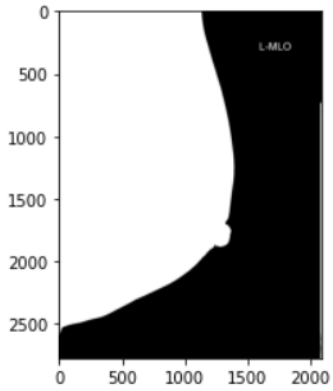



Hình 23: So sánh ảnh bình thường và ảnh khi sử dụng windowing

## 4.3 Phân tích vùng quan tâm (Region interest)

### 4.3.1 Cắt ảnh bằng thuật toán Contour

- Đầu tiên, ảnh được chuyển sang dạng ảnh xám để giảm kích thước dữ liệu cần xử lý.
- Sử dụng bộ lọc để loại bỏ nhiễu và làm mịn ảnh.
- Ngưỡng hóa ảnh xám thành ảnh nhị phân. Quá trình ngưỡng hóa này sẽ chuyển các điểm ảnh thành các giá trị 0 hoặc 255, tùy thuộc vào giá trị ngưỡng được thiết lập. Các điểm ảnh có giá trị lớn hơn ngưỡng sẽ được gán giá trị 255, các điểm ảnh còn lại sẽ được gán giá trị 0.




Hình 24: Ảnh nhị phân có chứa đường viền

Kết quả trả về của thuật toán Contour là danh sách các đường viền tìm thấy trên ảnh. Các đường viền này có thể được sử dụng để tính toán diện tích và vị trí của các vật thể trong ảnh, hay để phát hiện vật thể trong ảnh theo các tiêu chí cụ thể.
Đường viền to nhất sẽ là đường viền của nhũ, nên sẽ cắt phần có đường viền to nhất ra sẽ thu được vùng nhũ ảnh.

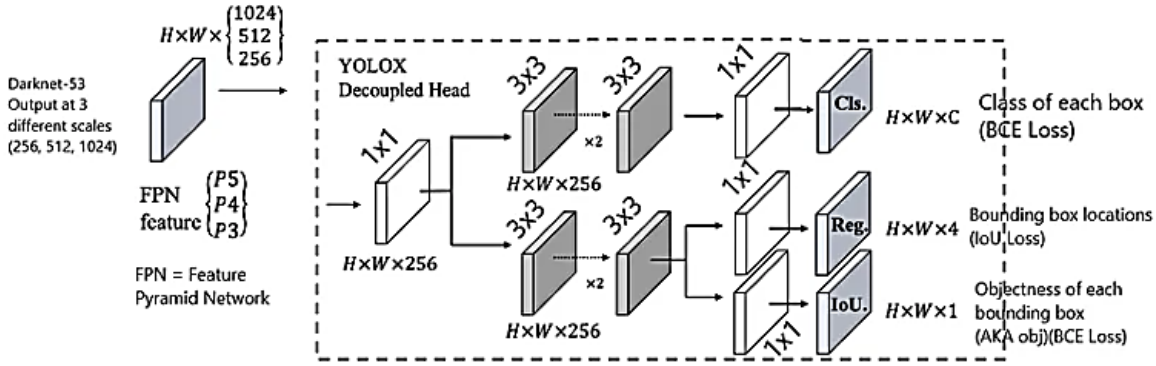
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 



Hình 25: Vùng nhũ ảnh sử dụng Contour để cắt

### 4.3.2 Cắt ảnh bằng mô hình YOLOX

YOLOX một mô hình mạng CNN cho việc phát hiện, nhận dạng và phân loại đối tượng, được tạo ra từ việc kết hợp giữa các convolutional layers và connected layers. Trong đó, các convolutional layers sẽ trích xuất ra các feature của ảnh, còn full-connected layers sẽ dự đoán ra xác suất đó và tọa độ của đối tượng.
YOLOX là một mô hình để phát hiện đối tượng trong ảnh nó được cải tiến rất nhiều về tốc độ và độ chính xác. Song với đó, nó sử dụng backbone là Darknet-53 với heads là Decoupled Head đây là điểm độc đáo của YOLOX so với các mô hình khác.




Hình 26: Mô tả cấu trúc YOLOX [18]

Ở sơ đồ này chúng ta có thể thấy rằng đầu vào của head YOLOX là đầu ra từ backbone Feature pyramid network (FPN) ở 3 tỉ lệ khác nhau – 1024, 512, 256 kênh [18]. Cùng với đó ta cũng thấy rằng đầu ra của YOLOX thực sự là 3 tensors chứa các thông tin khác nhau thay vì 1 tensors lớn chứa tất cả thông tin với các thông tin như sau:

- Cls: lớp của mỗi bounding box.
- Reg: 4 phần của bounding box (x, y, w, h).
- IoU: Chỉ ra mức độ tin cậy của đối tượng trong bounding box.

Một trong những thay đổi quan trọng của YOLOX là không sử dụng anchor trong phát hiện đối tượng. Anchor là tập hợp một lượng lớn các bounding box dự đoán, được trãi đều trên 1 hình ảnh. Và chính vì vậy nó sẽ phụ thuộc vào miền dữ liệu và thiết tính tổng quát. Song với đó nó còn làm tăng độ phức tạp của đầu phát hiện cũng như số lượng dự đoán mỗi ảnh. Nhưng phương pháp Anchor free giải quyết được gần hết những điều đó. Phương pháp này cố gắng bản địa hóa các đối tượng trực tiếp không sử dụng các hộp thay vào đó là sử dụng centers hoặc key points. Điều này làm cho mô hình trở nên đơn giản, linh hoạt hơn. Trong

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

mô hình phát hiện đối tượng YOLOX sử phương pháp tiếp cận centers để phát hiện đối tượng. Từ đó giá trị dự đoán chỉ là một vector 4D mã hóa vị trí của hộp ở mỗi pixel nền trước thành $T = \{left, top, right, bottom\}$. Trong đó left, top, right, bottom là khóa các từ vị trí của 4 cạnh của bouding box. Song với đó YOLOX còn sử dụng phương pháp simOTA. Việc sử dụng OTA giúp cải thiện độ chính xác của mô hình được huấn luyện. Nhưng đồng thời cũng tăng 25 % thời gian huấn luyện. Chính vì những lý do trên YOLOX được chúng tôi lựa chọn để sử dụng cho việc phát hiện vùng nhũ ảnh quan tâm.
Cắt ảnh bằng YOLOX được chúng tôi thực hiện gồm 3 bước:

- Gán nhãn hình ảnh cần huấn luyện.
- Huấn luyện mô hình YOLOX.
- Cắt ảnh.

### 4.4 Phân loại tổn thương

Dữ liệu sau khi Region of interest sẽ làm đầu vào cho giai đoạn phân loại tổn thương. Ở giai đoạn này chúng tôi không sử dụng những phương pháp xử lý ảnh thông thường mà thay vào đó sẽ sử dụng các mạng học sâu để phân loại tổn thương.
Chúng tôi sẽ sử dụng các mô hình CNN bao gồm là EfficientNets và HRNet và chọn ra mô hình tối ưu nhất về thời gian huấn luyện cũng như độ chính xác thu được. Tất cả các kiến trúc này được hiện thực bằng Framework Pytorch được phát triển bởi Facebook và được huấn luyện với hàm mất mát là Cross entropy, hàm tối ưu là Adam và các thông số giống nhau là: batch size bằng 64, learning rate bằng 0.001, và số lượng epoch bằng 100. Chúng tôi đã sử dụng Kaggle để huấn luyện với bộ nhớ RAM là 13GB, dung lượng ổ đĩa là 73.1GB và miễn phí sử dụng GPU Tesla P100 RAM 16GB.

#### 4.4.1 Mô hình EfficientNets

Mạng nơ-ron tích chập (Convolutional Neural Networks - ConvNets) thường được xây dựng với giới hạn tài nguyên cố định, sau đó có thể được tăng cường độ chính xác bằng cách sử dụng nhiều tài nguyên hơn. Tuy nhiên, việc cân bằng độ sâu, chiều rộng và độ phân giải mạng (network depth, width, and resolution) có thể đưa đến hiệu suất tốt hơn. Điều này đã được nghiên cứu một cách có hệ thống bởi nhóm tác giả Mingxing Tan và Quoc V. Le.
Trong ConvNet, có ba kích thước tỷ lệ quan trọng: depth, width và resolution. Depth tương ứng với số lớp trong mạng. Width được định nghĩa là số kênh trong lớp Conv. Resolution là độ phân giải của hình ảnh được đưa vào ConvNet.

- Depth là độ sâu của mạng tương đương với số lớp trong đó.
- Width là độ rộng của mạng. Ví dụ: một thước đo chiều rộng là số kênh trong lớp Conv.
- Resolution là độ phân giải hình ảnh được chuyển đến CNN.

Lớp ConvNet thứ i có thể được định nghĩa là một hàm $Y_i = F_i(X_i)$ trong đó $F_i$ là toán tử, $Y_i$ là tensor đầu ra, $X_i$ là tensor đầu vào, kích thước tensor sẽ là: $(H_i, Wi, C_i)$ trong đó $H_i$ và $W_i$ là spatial dimension, $C_i$ là channel dimension.
Một ConvNets N có thể được biểu diễn bằng danh sách các layer gồm:

$$N = F_k \odot F_{\{k-1\}} \odot F_1(X_1) = \bigodot_{\{j=1..k\}} F_j(X_1) \quad (31)$$

Trong thực tế, các lớp ConvNet thường được phân chia thành nhiều stage và tất cả các lớp trong mỗi stage đều có chung một kiến trúc:
Ví dụ: ResNet [7] có năm stage và tất cả các lớp trong mỗi stage có cùng một kiểu phức hợp ngoại trừ lớp đầu tiên thực hiện down sampling Do đó, chúng ta có thể định nghĩa Mạng Conv là:

$$N = \bigodot_{j=1..s} F_i^{L_i} X_{(H_i, W_i, C_i)} \quad (32)$$

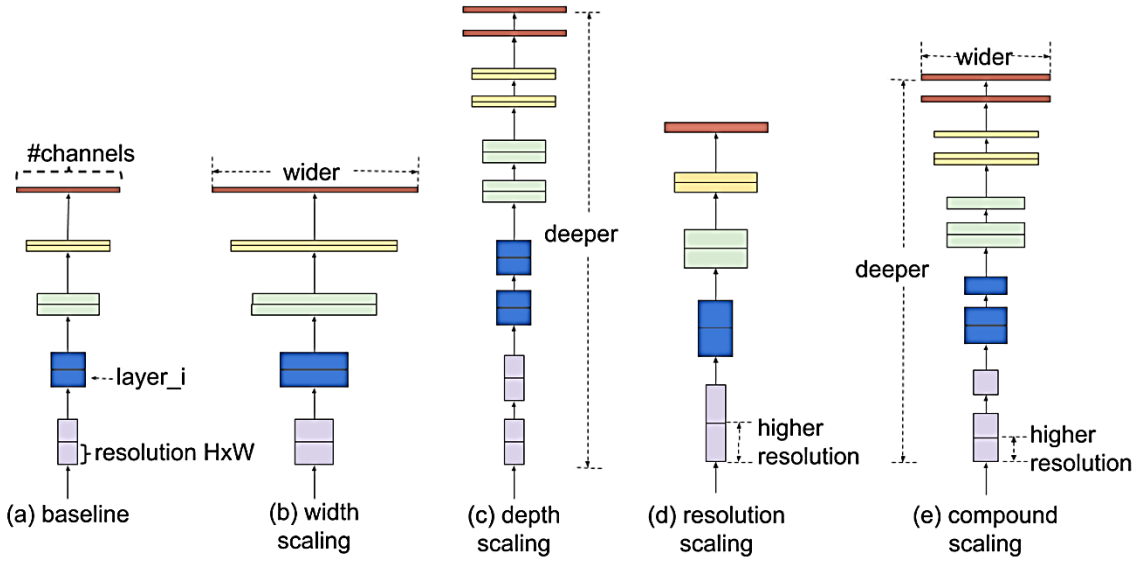
trong đó $F_i^{L_i}$ biểu thị lớp được lặp lại $L_i$ lần trong stage I $(H_i, W_i, C_i)$ biểu thị kích thước của tensor đầu vào X của lớp i. Không giống như các thiết kế ConvNet thông thường chủ yếu tập trung vào việc tìm kiếm kiến trúc lớp tốt nhất $F_i$ việc thu phóng mô hình cố gắng thu phóng chiều dài mạng $(L_i)$, chiều rộng $(C_i)$ hoặc độ phân giải $(H_i, W_i)$ mà không thay đổi được xác định trước trong mạng cơ sở bằng cách giữ nguyên

hàm $F_i$, việc thu phóng mô hình đơn giản hóa vấn đề về thiết kế đối với tài nguyên hạn chế. Tuy nhiên vẫn có rất nhiều khả năng có thể xảy ra bởi chúng ta có thể thay đổi cả 3 chiều của mỗi lớp với mỗi mức độ khác nhau. Nhằm thu hẹp không gian tìm kiếm, nhóm tác giả đã hạn chế rằng tất cả các lớp phải được thu phóng đồng nhất với tỷ lệ không đổi. Mục tiêu của họ là tối đa hóa độ chính xác của mô hình cho bất kỳ lượng hạn chế tài nguyên nhất định nào, vấn đề này có thể được xem như một vấn đề tối ưu hóa.

$$\max_{d,w,r} Accuracy(N(d,w,r))$$

$$N(d,w,r) = \bigodot_{j=1..s} \hat{F}^{d\hat{L}_i}_i X_{(r\hat{H}_i, r\hat{W}_i, r\hat{C}_i)} \tag{33}$$

Trong đó với $w, d, r$ là các hệ số để chia tỷ lệ chiều rộng, chiều sâu và độ phân giải của mạng; $\widehat{H_i}, \widehat{W_i}, \widehat{C_i}$ là các tham số được xác định trước trong mạng cơ sở.

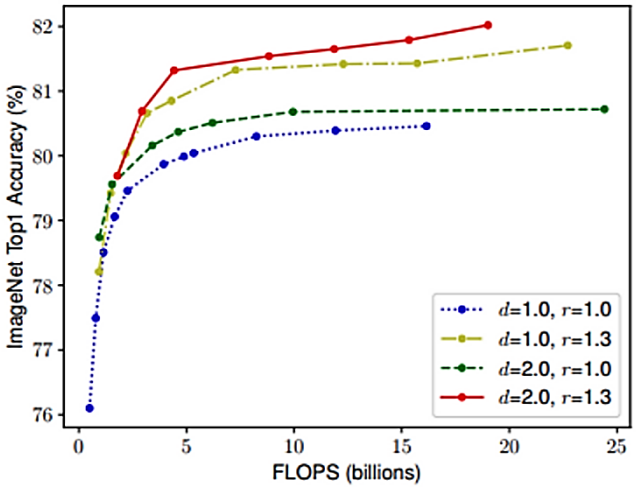



Hình 27: Model Scaling

*(a) là một ví dụ về mạng cơ sở; (b)-(d) là quy mô thông thường chỉ làm tăng một chiều của mạng chiều rộng, chiều sâu hoặc độ phân giải. (e) là phương pháp chia tỷ lệ hỗn hợp được đề xuất của tác giả, chia tỷ lệ đồng đều cả ba chiều với tỷ lệ cố định [19]*

Theo nhóm tác giả quan sát các chiều của các mô hình thường không thu nhỏ/thu phóng đọc lập với nhau. Bởi vậy nên, theo trực giác của bản thân, nhóm tác giả cho rằng chúng ta cần phối hợp và cân bằng các kích thước tỷ lệ khác nhau hơn là chia tỷ lệ một chiều thông thường. Để xác thực trực giác của mình, nhóm tác giả so sánh việc thu phóng theo chiều rộng và theo các độ sâu cũng như độ phân giải mạng ở các mức độ khác nhau.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 28: Mở rộng chiều rộng mạng cho các mạng cơ sở khác nhau. Mỗi dấu chấm trong một dòng biểu thị một mô hình có chiều rộng khác nhau hệ số (w) [19]

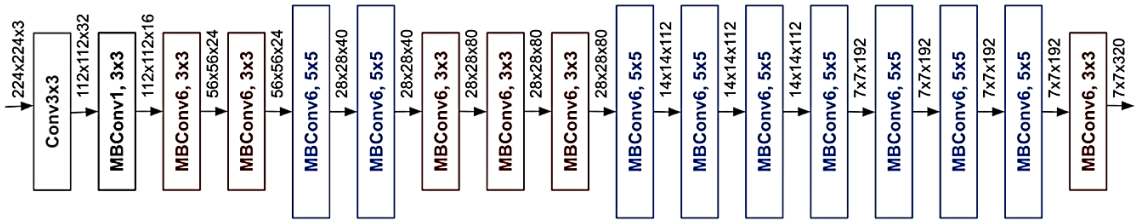
Từ kết quả được thể hiện trong hình trên, nhóm tác giả kết luận rằng: Để đạt được độ chính xác và hiệu quả tốt hơn, điều quan trọng là phải cân bằng tất cả các kích thước của chiều rộng, chiều sâu và độ phân giải mạng trong quá trình thu phóng quy mô ConvNet. Trong bài báo này, nhóm tác giả đề xuất một phương pháp thu phóng phức hợp mới, sử dụng hệ số kép để thu phóng đồng nhất chiều rộng, độ sâu và độ phân giải của mạng theo cách có nguyên tắc:

$$d = \alpha^{\phi}$$
$$w = \beta^{\phi}$$
$$r = \gamma^{\phi}$$
$$st\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$$
$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$$

Trong đó

- $d, w, r$ lần lượt là độ rộng, độ sâu và độ phân giải của mạng
- $\alpha, \beta, \gamma$ là các hằng số có thể được xác định bằng small grid search.

$\phi$ là hệ số do người dùng chỉ định để kiểm soát số lượng tài nguyên khác có sẵn để thu phóng mô hình trong khi $\alpha, \beta, \gamma$ chỉ định cách gán các tài nguyên bổ sung này cho độ rộng, độ sâu và độ phân giải của mạng tương ứng. Đáng chú ý, FLOPS của một op tích hợp thông thường tỷ lệ với $d, w^2, r^2$ tức là, độ sâu mạng tăng gấp đôi sẽ tăng gấp đôi FLOPS, nhưng tăng gấp đôi độ rộng hoặc độ phân giải của mạng sẽ tăng FLOPS lên bốn lần. Vì các hoạt động tích phân thường chiếm ưu thế trong chi phí tính toán trong ConvNets, nên việc thu phóng Mạng Conv với phương trình 3 sẽ làm tăng tổng FLOPS khoảng $(\alpha \cdot \beta^2 \cdot \gamma^2)^{\phi}$. Họ ràng buộc $\alpha \cdot \beta^2 \cdot \gamma^2 \approx 2$, sao cho với bất kì $\phi$ mới nào, tổng FLOPS sẽ tăng xấp xỉ lên $2^{\phi}$.
Hiệu quả của việc mở rộng mô hình cũng phụ thuộc rất nhiều vào mạng cơ sở. Vì vậy, để cải thiện hiệu suất hơn nữa, nhóm tác giả cũng đã phát triển một mạng cơ sở mới bằng cách thực hiện tìm kiếm kiến trúc nơ-ron bằng cách sử dụng khung AutoML MNAS, giúp tối ưu hóa cả độ chính xác và hiệu quả (FLOPS). Kiến trúc kết quả sử dụng tích chập cổ chai đảo ngược di động (MBConv), tương tự như MobileNetV2 và MnasNet, nhưng lớn hơn một chút do ngân sách FLOP tăng lên. Sau đó, chúng tôi mở rộng quy mô mạng cơ sở để có được một nhóm các mô hình, được gọi là EfficientNets.




Hình 29: Kiến trúc mạng EfficientNets [18]

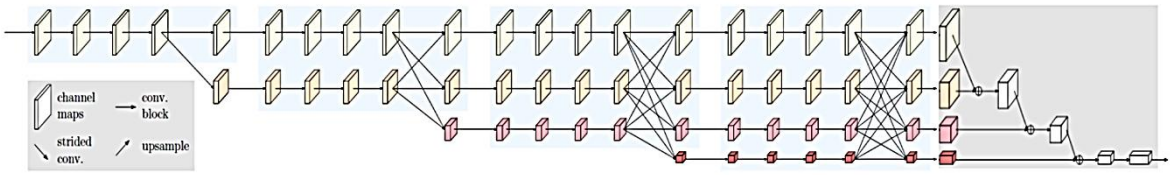
### 4.4.2 Mô hình HRNet

Trong lĩnh vực thị giác máy tính, các mạng CNN thường được thiết kế theo dạng bottom-up. Tức là, một ảnh đầu vào sau khi đi qua các lớp tích chập sẽ có độ phân giải (resolution) giảm, và độ sâu của mạng (depth) tăng lên. Tuy nhiên, đối với các bài toán xử lý ảnh y tế, như việc thu thập các đặc trưng sâu trong ảnh với độ phân giải cao, các kiến trúc mạng truyền thống này gặp nhiều khó khăn. Các kiến trúc mạng trước đây, như Hourglass, SegNet, DeconvNet, U-Net, SimpleBaseline, đã được ra đời nhằm giải quyết vấn đề này, tiêu biểu là học cách khôi phục độ phân giải cho đặc trưng sâu từ đặc trưng có độ phân giải thấp, tức là top-down. Tuy nhiên, cả bottom-up và top-down đều có khả năng sẽ mất mát thông tin. Để giải quyết vấn đề này, High-resolution netwwork (HRnet) ra đời. Khác với các mạng deep learning trước đây chỉ có thể xử lý ảnh với độ phân giải thấp hoặc trung bình, HRnet có khả năng xử lý hình ảnh với độ phân giải cao, thậm chí ảnh 4k hoặc cao hơn. HRnet nối các nhánh mạng high-to-low một cách song song chứ không hẳn là nối tiếp, giữ nguyên độ phân giải cao khi truyền ảnh chứ không phải là phục hồi như ý tưởng của các mạng trước đây. Để đạt được điều này, HRnet sử dụng cấu trúc mạng "kim tự tháp" với nhiều mô-đun nhỏ được kết hợp với nhau, mỗi mô-đun xử lý thông tin với độ phân giải khác nhau và có thể truyền thông tin giữa các lớp để cải thiện hiệu suất. Nhờ vậy, HRnet có khả năng trích xuất thông tin chi tiết và đa dạng từ các bức ảnh độ phân giải cao, đem lại hiệu quả tốt hơn cho các bài toán thị giác máy tính.

$$\begin{aligned}
&\mathcal{N}_{11} \rightarrow \mathcal{N}_{21} \rightarrow \mathcal{N}_{31} \rightarrow \mathcal{N}_{41} \\
&\searrow \mathcal{N}_{22} \rightarrow \mathcal{N}_{32} \rightarrow \mathcal{N}_{42} \\
&\searrow \mathcal{N}_{33} \rightarrow \mathcal{N}_{43} \\
&\searrow \mathcal{N}_{44}
\end{aligned}$$

Hình 30: Minh họa HRNet 4 giai đoạn theo cấu trúc "Kim tự tháp"

Trong quá trình kết hợp đặc trưng cấp thấp có độ phân giải cao và đặc trưng cấp cao có độ phân giải thấp, các cách tiếp cận trước đây thường sử dụng phép UpSampling đặc trưng độ phân giải thấp lên cho bằng đặc trưng độ phân giải cao và kết hợp lại bằng các phép cộng hoặc phép nối; HRNet sẽ thực hiện việc này nhiều lần, qua nhiều giai đoạn để các đặc trưng độ phân giải cao và đặc trưng sâu độ phân giải thấp sẽ bổ sung ngữ nghĩa cho nhau [20].




Hình 31: Kiến trúc mạng phân loại HRNet [19]

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Ngoài ra, HRnet cũng sử dụng các kỹ thuật mới nhất trong lĩnh vực mạng neural như residual learning và dilated convolution để cải thiện khả năng học và trích xuất đặc trưng của mạng. Điều đó khiến HRnet trở thành một mạng xương sống (backbone) mạnh trong các bài toán thị giác máy tính.
Trước tiên ảnh đầu vào sẽ được resized về 256x256, gồm 2 khối tích chập. mỗi khối gồm lớp tích chập 3x3 với bước trượt là 2. Theo sau là lớp chuẩn hóa (BatchNorrm2D) và hàm kích hoạt chỉnh sửa đơn vị tuyến tính (ReLU). Đầu ra của nó sau đó được xử lý bởi phần thân của HRnet, gồm 4 giai đoạnm mỗi giai đoạn có 3 thành phần bao gồm transition, mutil - resolution convolution và fusion. Mục tiêu chính của thành phần transition là thêm độ phân giải cao tới thấp (high-to-low).

$$R_{(s+1j)} = f_{(s+1j)}(R_{si}) \tag{3.4}$$

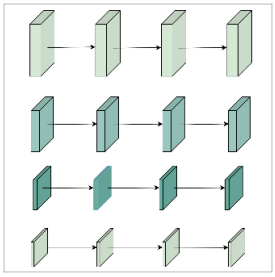
ở đây $f_{(s+1j)}$ là hàm chuyển đổi, thứ mà được dựa vào chỉ số của độ phân giải.

Nếu $i = j, f_{(s+1j)}(R) = R$, nếu $j > i, f_{(s+1j)}(R)$ là downsamples biểu diễn đầu vào R qua lớp tích chập 3x3 stride = 2

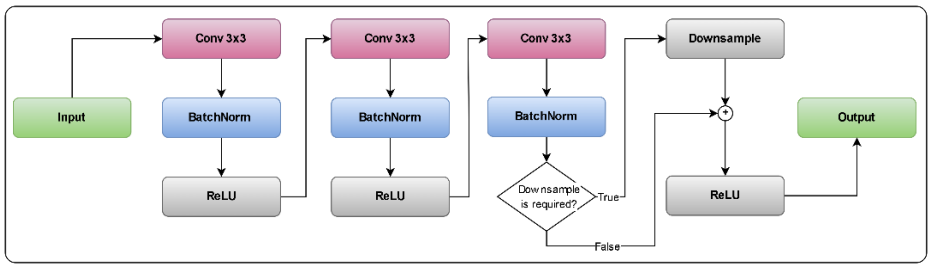



Hình 32: Thành phần chuyển đổi từ giai đoạn ba sang giai đoạn bốn [19]

Thành phần parallel-multi-resolution thêm từ cao xuống thấp độ phân giải từng luồng một để tạo ra các giai đoạn mới. Mỗi luồng trong thành phần đa-phân giải (multi-resolution) bao gồm 4 khối. Kiến trúc của mỗi khối được mô tả trong hình 33 bao gồm 3 lớp tích chập 3x3 và 1x1 theo sau đó sẽ là chuẩn hóa theo batch norm.



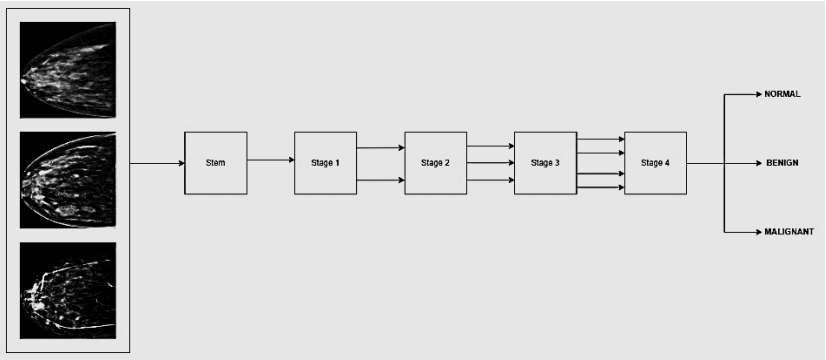
a) Bốn luồng tích chập song song




b) Cấu trúc của một khối trong luồng

Hình 33: Khối kiến trúc xử lý song song

Mô hình đề xuất cho việc nhận dạng tổn thương được thể hiện qua hình 34.

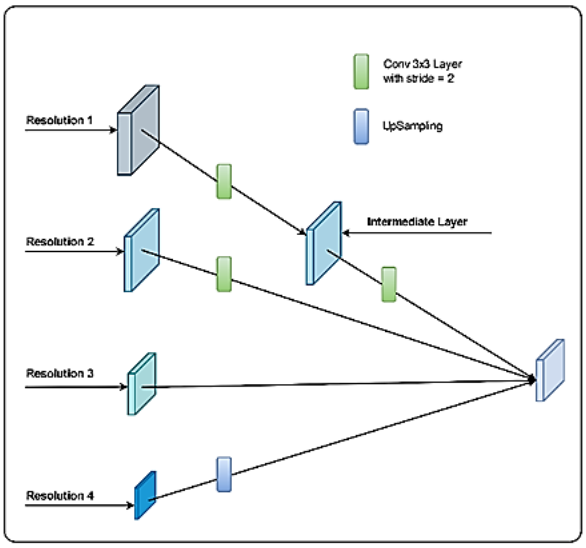



Hình 34: Tổng quan mô hình nhận dạng tổn thương

Thành phần hợp nhất (fusion component) có nhiệm vụ trao đổi thông tin giữa các thành phần đa phân giải. Hình 35 minh họa cho cách tổng hợp thông tin từ các độ phân giải khác nhau. Đầu ra $R_i^0$ tương ứng với chỉ số độ phân giải thứ i xác định bởi:

$$R_i^0 = \sum_j^N f_{ij}(R_j) \tag{35}$$

Theo (35) ở đây N là số lượng độ phân giải qua từng giai đoạn, $R_j$ là đầu vào thể hiện mới lớp phân giải thứ j và $f_{ij}$ chính là hàm kết hợp dựa trên chỉ số của đầu vào và đầu ra của chỉ số độ phân giải. Nếu $j = i, f_{ii}(R) = R.$ nếu $j < i, f_{ji}$, sẽ downsamples R bằng cách dùng $i - j$ lớp tích chập 3x3 với stride = 2, còn nếu $j > i, f_{ji}(R)$ sẽ upsamples đầu vào R bằng cách dùng bilinear interpolation theo sau là phép tích chập 1x1 để cân bằng số lượng kênh.




Hình 35: Minh họa cách tổng hợp thông tin từu các độ phân giải [19]

Do đó, hệ thống kết nối các luồng tích chập từ cao đến thấp song song với nhau, có thể không chỉ duy trì các biểu diễn độ phân giải cao mà còn tạo ra các độ phân giải khác nhau qua toàn bộ quá trình. HRNet đã cho thấy hiệu suất ưu việt trong một số ứng dụng như phân đoạn, phát hiện đối tượng, phân loại.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

# 5 KẾT QUẢ THỰC NGHIỆM

## 5.1 Dữ liệu

Ảnh Digital Imaging and Communications in Medicine (Dicom) là một tiêu chuẩn quốc tế được sử dụng trong lĩnh vực y tế để trao đổi và lưu trữ các hình ảnh y tế số như ảnh CT, MRI, siêu âm và đặc biệt là X-quang và bộ dữ liệu của chúng tôi sử dụng ở dạng X-quang có 7808 hình ảnh đây là bộ dữ liệu mini-DDSM được trích suất từ cơ sở dữ liệu kỹ thuật số chụp nhũ ảnh sàng lọc (DDSM) là một nguồn tài nguyên để cộng đồng nghiên cứu sử dụng phân tích hình ảnh chụp nhũ ảnh. DDSM đã được cộng đồng nghiên cứu sử dụng rộng rãi trong các nghiên cứu liên quan.

## 5.2 Kết quả giai đoạn region of interest

Hình ảnh sau khi qua giai đoạn tiền xử lý sẽ được đưa qua mô hình YOLOX với 2000 ảnh đã được gán nhãn. Chúng tôi sử dụng mAP (Mean Average Precision) là chỉ số đánh giá hiệu xuất của mô hình trong việc phát hiện đối tượng trên hình ảnh.
Được tính bằng công thức:

$$mAP = \frac{1}{n}\sum_{k=1}^{k=n} AP_k$$

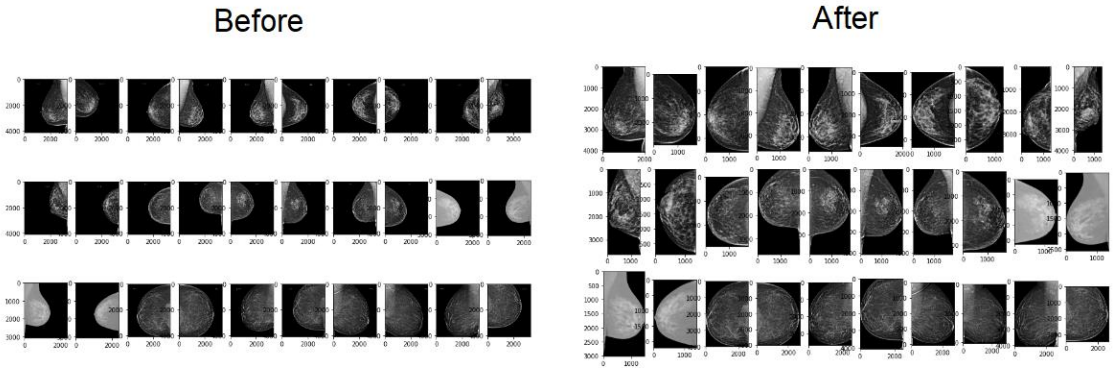
Trong đó:

$AP_k$ là Average Precision của lớp k

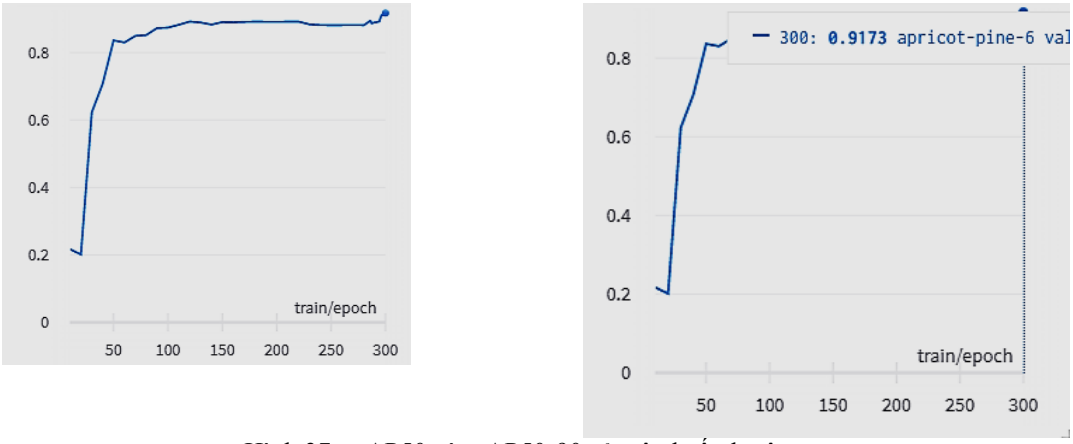
$AP = \sum_{k=0}^{k=n-1}[Recalls(k) - Recalls(k+1)$* Precisions(k)

- Recalls: là tỷ lệ đối tượng giữa số lượng đối tượng được phân loại đúng trên tổng số lượng đối tượng thực tế trong dữ liệu kiểm tra phân loại vào loại đúng.
- Precision: là tỷ lệ đối tượng giữa số lượng đối tượng được phân loại đúng trên tổng số lượng đối tượng được phân loại.
- n là số lượng các lớp

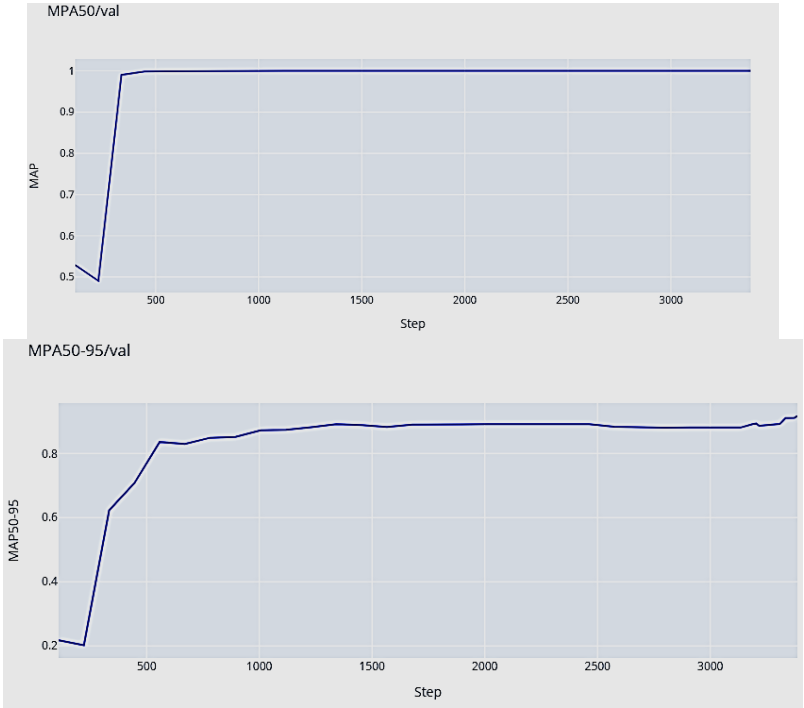
Từ đây cho ta thấy mAP50 của model YOLOX-s là 0.99 và mAP50-95 là 0.957 là khá cao cùng với 9 triệu tham số với tốc độ cắt ảnh trung bình xấp xỉ 0.05s một tấm hình. Thông số khá tốt nhưng cũng có một số tấm hình YOLOX-s không cắt được thì chúng tôi sẽ sử dụng contours để xử lý.




Hình 36: Ảnh trước và sau khi cắt




Hình 37: mAP50 và mAP50-90 của tập huấn luyện




Hình 38: mAP50 và mAP50-90 của tập validation

## 5.3 Kết quả giai đoạn phân lớp

Giai đoạn này, hình ảnh sau khi qua giai đoạn region of interest chúng tôi sẽ sử dụng các mô hình CNN bao gồm là EfficientNets và HRNet và chọn ra mô hình tối ưu nhất về thời gian huấn luyện cũng như độ chính xác thu được. Sau đây là confusion matrix cho từng mô hình đã sử dụng và các điểm như:

Precision: tỉ lệ số lượng tổn thương đúng được phân loại đúng trong số lượng ảnh được phân loại là tổn thương đó.

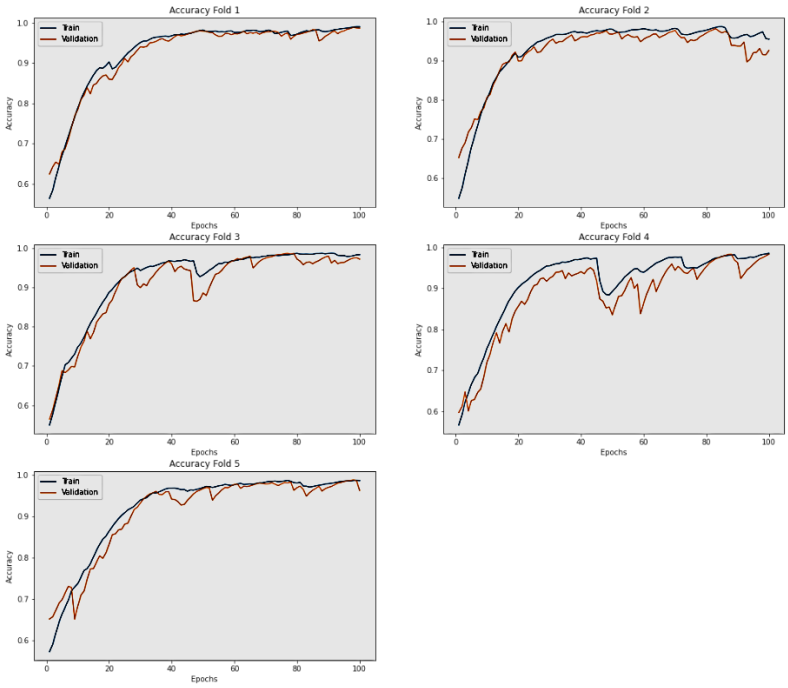
Recall: tỉ lệ số lượng tổn thương đúng được phân loại đúng trong số lượng ảnh thực sự đúng là tổn thương đó.

F1-score: là điểm trung bình hài hòa của precision và recall được tính bằng Công thức 36.

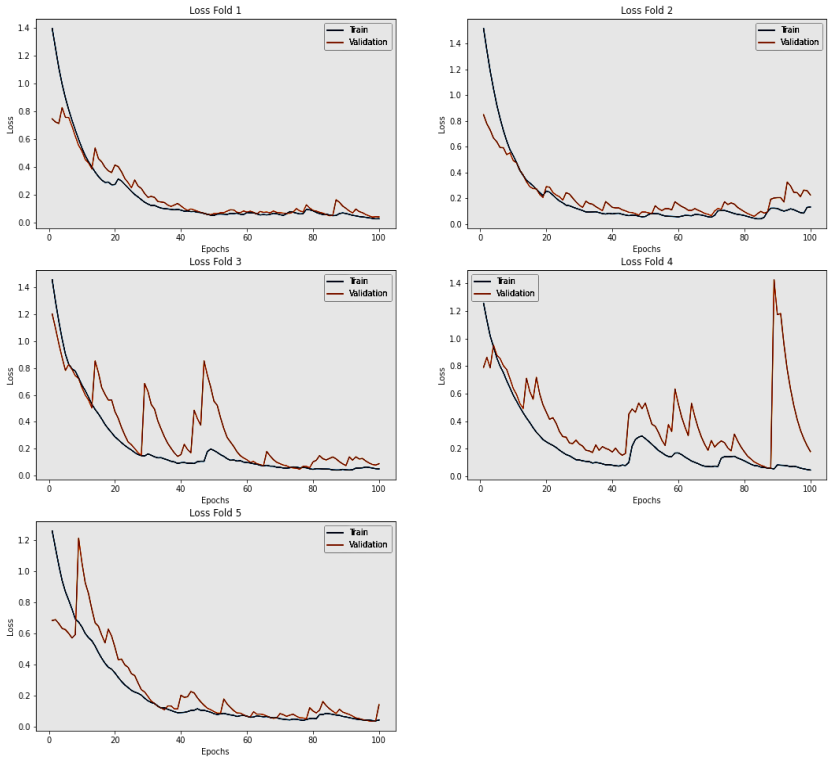
$$F1 = 2 * \left(\frac{precision * recall}{precision + recall}\right) \quad (36)$$


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 5.3.1 Kết quả mô hình EfficientNet




Hình 39: Biểu đồ độ chính xác trên dữ liệu DDSM với 5-Fold của EfficientNet




Hình 40: Biểu đồ mất mát của dữ liệu DDSM với 5-Fold của EfficientNet

| | Normal | Benign | Malignant | |
|---|---|---|---|---|
| Normal | 163 | 4 | 4 | 95% |
| Benign | 0 | 169 | 11 | 94% |
| Malignant | 4 | 8 | 197 | 94% |
| | 98% | 93% | 93% | |

Bảng 1: Confusion matrix trên dữ liệu kiểm thử DDSM của Fold tốt nhất

| | **Precision** | **Recall** | **F1-score** | **Support** |
|---|---|---|---|---|
| **Normal** | 98% | 95% | 96% | 171 |
| **Benign** | 93% | 94% | 94% | 180 |
| **Malignant** | 93% | 94% | 94% | 209 |

Bảng 2: Đánh giá EfficientNet trên dữ liệu DDSM qua precision, recall, f1-score

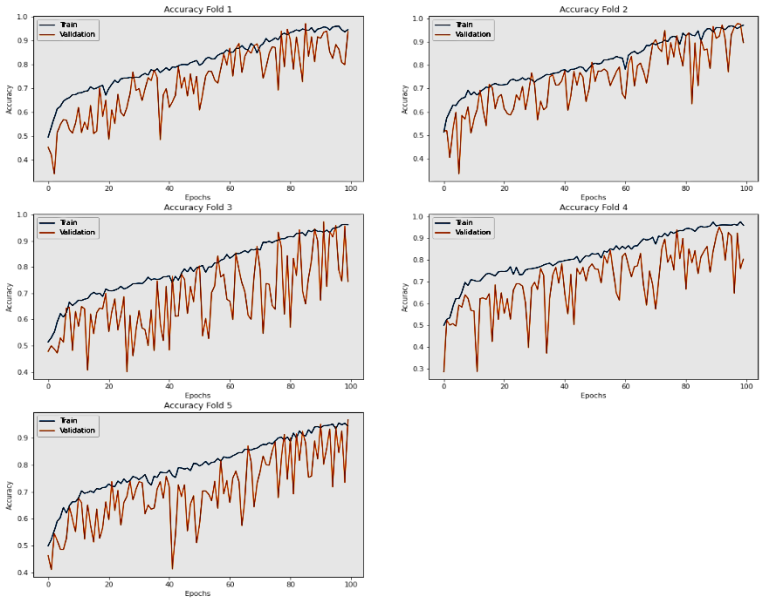
| **EfficientNet** | **Độ chính xác tập huấn luyện** | **Độ chính xác tập kiểm thử** | **Thời gian huấn luyện** |
|---|---|---|---|
| Fold 1 | 94,71% | 93,88% | 249 phút |
| **Fold 2** | **94,84%** | **93,48%** | **253 phút** |
| Fold 3 | 93,86% | 92,14% | 251 phút |
| Fold 4 | 93,16% | 90,12% | 257 phút |
| Fold 5 | 93,74% | 92,40% | 257 phút |

Bảng 3: So sánh độ chính xác và thời gian huấn luyện giữa các Fold EfficientNet

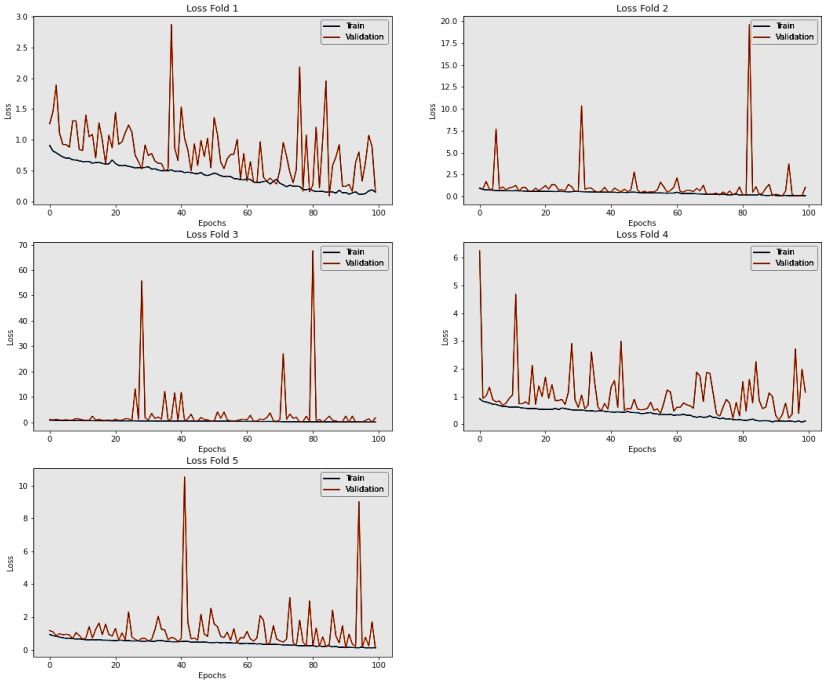
Từ Bảng 1 và Bảng 2 ta có thể thấy mô hình EfficientNet dự đoán rất tốt cho lớp Normal 98%. Tuy nhiên, hai lớp còn lại độ chính xác vẫn chưa đạt tới 94%. Và biểu đồ độ chính xác trên tập huấn luyện và kiểm thử Hình 39 đều khá tốt khi không có quá nhiều sự tụt dốc lớn về độ chính xác lúc huấn luyện mô hình.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 5.3.2 Kết quả mô hình HRNet




Hình 41: Biểu đồ độ chính xác trên dữ liệu DDSM với 5-Fold của HRNet




Hình 42: Biểu đồ mất mát của dữ liệu DDSM với 5-Fold của HRNet

Bảng 4: Confusion matrix trên dữ liệu DDSM của Fold tốt nhất HRNet

| | Normal | Benign | Malignant | |
|---|---|---|---|---|
| Normal | 160 | 8 | 3 | 94% |
| Benign | 7 | 163 | 10 | 91% |
| Malignant | 11 | 18 | 180 | 86% |
| | 90% | 86% | 93% | |

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Bảng 5: Đánh giá HRNet trên dữ liệu DDSM qua precision, recall, f1-score

| | Precision | Recall | F1-score | Support |
|---|---|---|---|---|
| Normal | 90% | 94% | 92% | 171 |
| Benign | 86% | 91% | 88% | 180 |
| Malignant | 93% | 86% | 90% | 109 |

Bảng 6: So sánh độ chính xác và thời gian huấn luyện giữa các Fold HRNet

| **HRNet** | **Độ chính xác tập huấn luyện** | **Độ chính xác tập kiểm thử** | **Thời gian huấn luyện** |
|---|---|---|---|
| Fold 1 | 80,95% | 72,54% | 300 phút |
| **Fold 2** | **80,90%** | **73,87%** | **304 phút** |
| Fold 3 | 79,90% | 67,86% | 300 phút |
| Fold 4 | 82,22% | 70,29% | 308 phút |
| Fold 5 | 79,97% | 70,42% | 298 phút |

Từ Bảng 5 và Bảng 6 ta có thể thấy là ở mô hình HRNet này có kết quả dự đoán lớp Malignant là khá tốt với độ chính xác là 93%. Tuy nhiên, hai lớp còn lại độ chính xác vẫn chưa đạt tới 93% và lớp Benign còn ở mức khá thấp là 86%.

### 5.3.3 So sánh độ chính xác trung bình và thời gian huấn luyện giữa các mô hình CNN

Bảng 7: So sánh độ chính xác trung bình và thời gian huấn luyện

| **Mô hình** | **Độ chính xác tập huấn luyện** | **Độ chính xác tập kiểm thử** | **Thời gian huấn luyện** |
|---|---|---|---|
| EfficientNet | **94,84%** | **93,48%** | **253 phút** |
| HRNet | 80,90% | 73,87% | 304 phút |

Bảng 8: So sánh với các nghiên cứu trước đó

| **TÁC GIẢ** | **DATASET** | **PHƯƠNG PHÁP** | **TÓM TẮT KẾT QUẢ** |
|---|---|---|---|
| M. Shalini and S. Radhika. [12] | Mammographic Mass of UCL | ANN, Decision Tree, SVM | Accuracy: 86%<br>Sensitivity: 83.75%<br>Specitivity: 87.61 % |
| MA. Al-Masni et al. [13] | 600 images of DDSM dataset | YOLO-based for two class Benign/Malignant | Accuracy: 85.52%<br>Sensitivity: 93.20%<br>Specitivity: 78.00% |
| S. Guan and M. Loew [14] | Mini- DDSM dataset | ANN, Decision Tree, SVM | Accuracy: 85.12% |
| N.S. Ismail and C. Sovuthy | IRMA dataset | VGG-16, ResNet50 | Precision: 89%, 88%<br>Recall: 99%, 94%<br>Accuracy: 94%, 91.7% |
| **Nghiên cứu này** | **Mini-DDSM** | **EfficientNet** | **Accuracy: 94,46%**<br>**Precision: 94,50%**<br>**Recall: 94,46%**<br>**F1 score: 94,47%** |


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Sau khi thực hiện các đánh giá về các mô hình ta có thể thấy hai mô hình đều có kết quả rất tốt khi độ chính xác trên dữ liệu kiểm huấn luyện đều trên 80%. Trong đó, mô hình EfficientNet vượt trội hơn so với HRNet với độ chính xác trên tập huấn luyện là 94,84% đồng thời độ chính xác trên tập kiểm thử là 93,48% bên cạnh đó thời gian huấn luyện của mô hình EfficientNet là 253 phút thấp hơn so với mô hình HRNet có thời gian huấn luyện là 304 phút. Ngoài ra, khi xem xét những biểu đồ khi huấn luyện mô hình ta cũng có thể thấy, biểu đồ của mô hình EfficientNet là khá tốt, khi không có quá nhiều sự lao dốc về độ chính xác trên tập kiểm thử khi huấn luyện. Do đó, mô hình EfficientNet mô hình tốt nhất sau khi huấn luyện.

## 6 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Sau các giai đoạn tiền xử lý ảnh DICOM, giai đoạn region of interest tiến hành các thuật toán cũng như mô hình đã cắt được vùng nhũ ảnh cần phân tích tổn thương qua đó tiến hành giai đoạn phân loại tổn thương. Ở đây các tổn thương trên nhũ ảnh sẽ được dự đoán bởi những mô hình CNN mà chúng tôi đã huấn luyện và cho ra kết quả loại tổn thương trên nhũ ảnh với độ chính xác cao nhất là hơn 94%. Qua đây, ta có thể kết luận là khi áp dụng các kỹ thuật xử lý ảnh và các mô hình học sâu vào trong bài toán phát hiện ung thư, tổn thương trên X-quang nhũ ảnh đã đem lại nhiều kết quả tốt và khả quan khi có thể hỗ trợ sàng lọc và phát hiện ung thư vú có thể thực hiện một cách nhanh chóng, độ chính xác cao, chi phí thấp, hỗ trợ các y bác sĩ trong việc có thể đưa ra các chẩn đoán bệnh kịp thời.

Bên trên là phần phân tích và xây dựng hệ thống phân tích vùng quan tâm và phân loại tổn thương trên X-quang nhũ ảnh. Cuối cùng, đồ án đã thu được một số kết quả sau:

Thực hiện các kỹ thuật xử lý ảnh để phân tích vùng quang tâm - Region of interest và xây dựng được mô hình phân loại tổn thương trên X-quang nhũ ảnh.

- Giai đoạn phân loại tổn thương vẫn chưa thực sự chính xác trên nhiều loại ảnh X-quang nhũ ảnh.
- Dữ liệu huấn luyện thiếu đa dạng.
- Tốc độ xử lý còn chậm.

Từ những nghiên cứu đã đạt được, có thể thấy việc sử dụng mô hình EfficientNet cho kết quả nhận diện rất ấn tượng. Việc tối ưu quá trình đào tạo cho mạng nơ ron tích chập cũng được nhóm lên kế hoạch để áp dụng với các kĩ thuật tối ưu tham số hiện đại như Expotentional Moving Average (EMA), học bán giám sát.

- Thu thập và mở rộng thêm bộ dữ liệu để có thể đa dạng.
- Nâng cấp giai đoạn phân loại để có độ chính xác cao hơn.
- Huấn luyện mô hình phân loại với nhiều dữ liệu hơn.

## TÀI LIỆU THAM KHẢO

[1] M. Shalini and S. Radhika, “Machine Learning techniques for Prediction from various Breast Cancer Datasets,” in *2020 Sixth International Conference on Bio Signals, Images, and Instrumentation (ICBSII)*, 2020, pp. 1–5, doi: 10.1109/ICBSII49132.2020.9167657.

[2] M. A. Al-Masni *et al.*, “Simultaneous detection and classification of breast masses in digital mammograms via a deep learning YOLO-based CAD system,” *Computer methods and programs in biomedicine*, vol. 157, pp. 85–94, 2018.

[3] S. Guan and M. Loew, “Breast cancer detection using synthetic mammograms from generative adversarial networks in convolutional neural networks.,” *Journal of medical imaging (Bellingham, Wash.)*, vol. 6, no. 3, p. 031411, 2019, doi: 10.1117/1.JMI.6.3.031411.

[4] I. Goodfellow *et al.*, “Generative adversarial networks,” *Communications of the ACM*, vol. 63, no. 11, pp. 139–144, 2020.

[5] N. S. Ismail and C. Sovuthy, “Breast Cancer Detection Based on Deep Learning Technique,” in *2019 International UNIMAS STEM 12th Engineering Conference (EnCon)*, 2019, pp. 89–92, doi: 10.1109/EnCon.2019.8861256.

[6] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv preprint arXiv:1409.1556*, 2014.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[8] K. Horsch, M. L. Giger, C. J. Vyborny, L. Lan, E. B. Mendelson, and R. E. Hendrick, "Classification of Breast Lesions with Multimodality Computer-aided Diagnosis: Observer Study Results on an Independent Clinical Data Set," *Radiology*, vol. 240, no. 2, pp. 357–368, 2006, doi: 10.1148/radiol.2401050208.

[9] R. M. Rangayyan, F. J. Ayres, and J. L. Desautels, "A review of computer-aided diagnosis of breast cancer: Toward the detection of subtle signs," *Journal of the Franklin Institute*, vol. 344, no. 3–4, pp. 312–348, 2007.

[10] A. Oliver *et al.*, "A review of automatic mass detection and segmentation in mammographic images," *Medical image analysis*, vol. 14, no. 2, pp. 87–110, 2010.

[11] H.-D. Cheng, X. Cai, X. Chen, L. Hu, and X. Lou, "Computer-aided detection and classification of microcalcifications in mammograms: a survey," *Pattern recognition*, vol. 36, no. 12, pp. 2967–2991, 2003.

[12] M. Kallergi, "Computer-aided detection, in its present form, is not an effective aid for screening mammography. Against the proposition.," *Medical Physics*, vol. 33, no. 4, pp. 812–814, 2006.

[13] S. Malur, S. Wurdinger, A. Moritz, W. Michels, and A. Schneider, "Comparison of written reports of mammography, sonography and magnetic resonance mammography for preoperative evaluation of breast lesions, with special emphasis on magnetic resonance mammography.," *Breast cancer research : BCR*, vol. 3, no. 1, pp. 55–60, 2001, doi: 10.1186/bcr271.

[14] L. Tab'ar, B. Vitak, H. H. Chen, M. F. Yen, S. W. Duffy, and R. A. Smith, "Beyond randomized controlled trials: organized mammographic screening substantially reduces breast carcinoma mortality.," *Cancer*, vol. 91, no. 9, pp. 1724–1731, 2001, doi: 10.1002/1097-0142(20010501)91:93.0.co;2-v.

[15] E. Honda, R. Nakayama, H. Koyama, and A. Yamashita, "Computer-Aided Diagnosis Scheme for Distinguishing Between Benign and Malignant Masses in Breast DCE-MRI," *Journal of Digital Imaging*, vol. 29, no. 3, pp. 388–393, 2016, doi: 10.1007/s10278-015-9856-7.

[16] E. Hunsberger and C. Eliasmith, "Training spiking deep networks for neuromorphic hardware," *arXiv preprint arXiv:1611.05141*, 2016.

[17] W. Viriyavit and V. Sornlertlamvanich, "Bed Position Classification by a Neural Network and Bayesian Network Using Noninvasive Sensors for Fall Prevention," *Journal of Sensors*, vol. 2020, pp. 1–14, 2020, doi: 10.1155/2020/5689860.

[18] Z. Ge, S. Liu, F. Wang, Z. Li, and J. Sun, "Yolox: Exceeding yolo series in 2021," *arXiv preprint arXiv:2107.08430*, 2021.

[19] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*, 2019, pp. 6105–6114.

[20] J. Wang *et al.*, "Deep high-resolution representation learning for visual recognition," *IEEE transactions on pattern analysis and machine intelligence*, vol. 43, no. 10, pp. 3349–3364, 2020.

[20] Website developers.google.com, Machine learning. [Online]. https://developers.google.com/machine earning/practica/image-classification/convolutional-neural-networks

[21] Website github.io, Neural-networks-1. [Online]. https://cs231n.github.io/neural-networks-1/

[22] Website github.io, Convolutional networks. [Online]. https://cs231n.github.io/convolutional-networks/

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F007**

# IUH ONE: CÔNG CỤ RÚT GỌN LIÊN KẾT, THEO DÕI LƯỢT TRUY CẬP VÀ ĐÁNH GIÁ TỈ LỆ TƯƠNG TÁC CỦA NGƯỜI DÙNG CHO CÁC WEBSITE

TON LONG PHUOC[1*], TOAN DUC VO[1], DINH NHAT PHAN[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*[*]tonlongphuoc@iuh.edu.vn*

**Tóm tắt.** Các liên kết (URL) rất cần thiết trong thế giới kỹ thuật số, vì chúng kết nối người dùng với nhiều nguồn tài nguyên trực tuyến khác nhau. Các liên kết này có thể xuất hiện dưới nhiều hình thức như mạng xã hội, tin nhắn hoặc bảng quảng cáo và tên của chúng cũng có vai trò thu hút người dùng truy cập nội dung bên trong. Tuy nhiên, một số liên kết quá dài, phức tạp hoặc không an toàn đối với người truy cập. Ngoài ra, theo dõi lượt truy cập và đánh giá tỉ lệ tương tác của người dùng khi truy cập các website là một điều cần quan tâm. Chủ sở hữu trang web có thể xây dựng các chiến lược quảng bá nội dung từ những thông số hiệu suất này để tiếp cận nhiều người dùng hơn, vì vậy công cụ IUH One được xây dựng để giải quyết các vấn đề trên. IUH One cho phép rút gọn liên kết nhằm nâng cao tính thân thiện và an toàn hơn cho người truy cập. Công cụ còn cung cấp cơ chế bảo mật cho các liên kết rút gọn; phân tích, trích xuất các tham số từ yêu cầu HTTP như: địa chỉ IP, User-Agent. Tất cả các chỉ số hiệu suất này được trực quan hoá dữ liệu trong thời gian thực trên nền tảng web. IUH One sử dụng các công nghệ như: Next.js, NestJS, PostgreSQL, Google Cloud Platform, Firebase, v.v. được triển khai với địa chỉ https://iuh.one với hàng nghìn lượt truy cập, hàng trăm liên kết đã đươc rút gọn. IUH One cung cấp các tính năng giống như các sản phẩm hiện có như TinyURL và Short.io. Trong tương lai, IUH One có kế hoạch triển khai các tính năng bổ sung dựa trên liên kết rút gọn, chẳng hạn như biểu mẫu khảo sát và biểu mẫu đăng ký sự kiện, đồng thời cho phép người dùng tạo và nhúng các hồ sơ mạng xã hội và thông tin liên hệ của họ vào phần tiểu sử. Sản phẩm hiện đang được triển khai cho các giảng viên, sinh viên và Trung tâm tư vấn hỗ trợ sinh viên trường Đại học Công nghiệp Thành phố Hồ Chí Minh.

**Từ khóa.** Công cụ rút gọn liên kết, Liên kết được rút gọn, Thống kê, Theo dõi, Phân tích, Bảo mật, Đo lường, Giám sát, Tương tác của người dùng.

# IUH ONE: LINK SHORTENER, NUMBER OF VISITS TRACKING TOOL, AND USER ENGAGEMENT RATE ASSESSMENT FOR WEBSITES

**Abstract.** Links (URLs) are essential in the digital world, connecting users to various online resources. These links can appear in many forms such as social networks, messages, or billboards, and their path names also play a role in attracting users to access the content within. However, some links are too long, complicated, or unsafe for visitors. In addition, tracking visits and evaluating user engagement rates when visiting websites is a matter of concern. Website owners can build content promotion strategies from these performance metrics to reach more users, so that's why IUH One comes to play. IUH One is a URL shortener that converts a long URL into a shorter one, friendlier and safer. IUH One also provides a security mechanism for shortened links; analyzes, and extracts data, parameters from HTTP requests such as IP address, or User-Agent. All these performance metrics are visualized in real-time automatically on our web-based platform. IUH One is built using these technologies: Next.js, NestJS, PostgreSQL, Google Cloud Platform, Firebase, etc. IUH One is deployed on https://iuh.one with thousands of hits, and hundreds of shortened links. IUH One offers the same features as existing products, such as TinyURL, and Short.io. In the future, IUH One plans to implement additional features based on the shortened link, such as survey forms, and event registration forms, and allow users to create and embed their social media profiles, and contact information, in their bio. IUH One are currently being rolled out to lecturers, students, and the Student Counseling and support center of the Industrial University of Ho Chi Minh City.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**Keywords.** Link shortener, Shortened link, Statistics, Tracking, Analytics, Security, Measuring, Monitoring, User engagement.

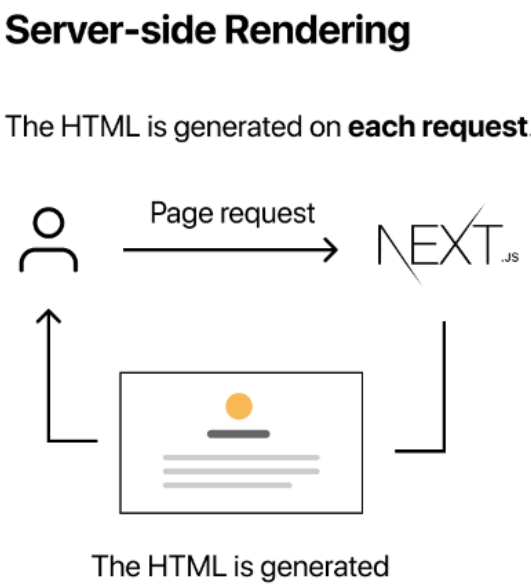
## 1 GIỚI THIỆU

Các liên kết (URL) rất cần thiết trong thế giới kỹ thuật số, vì chúng kết nối người dùng với nhiều nguồn tài nguyên trực tuyến khác nhau. Các liên kết này có thể xuất hiện dưới nhiều hình thức như mạng xã hội, tin nhắn hoặc bảng quảng cáo và tên của chúng cũng có vai trò thu hút người dùng truy cập nội dung bên trong. Tuy nhiên, một số liên kết quá dài, phức tạp hoặc không an toàn đối với người truy cập. Ngoài ra, theo dõi lượt truy cập và đánh giá tỉ lệ tương tác của người dùng khi truy cập các website là một điều cần quan tâm. Chủ sở hữu trang web có thể xây dựng các chiến lược quảng bá nội dung từ những thông số hiệu suất này để tiếp cận nhiều người dùng hơn, vì vậy công cụ IUH One được xây dựng để giải quyết các vấn đề trên.
IUH One cho phép rút gọn liên kết nhằm nâng cao tính thân thiện và an toàn hơn cho người truy cập. Công cụ còn cung cấp cơ chế bảo mật cho các liên kết rút gọn; phân tích, trích xuất các tham số từ yêu cầu HTTP như: địa chỉ IP, User-Agent. Tất cả các chỉ số hiệu suất này được trực quan hoá dữ liệu trong thời gian thực trên nền tảng web. IUH One sử dụng các công nghệ như: Next.js, NestJS, PostgreSQL, Google Cloud Platform, Firebase, v.v. được triển khai với địa chỉ https://iuh.one với hàng nghìn lượt truy cập, hàng trăm liên kết đã đươc rút gọn. IUH One có thể được tùy chỉnh để phù hợp với các nhu cầu và tình huống nghiệp vụ khác nhau.
Sản phẩm hiện đang được triển khai cho các giảng viên, sinh viên và Trung tâm Tư vấn hỗ trợ sinh viên trường Đại học Công nghiệp Thành phố Hồ Chí Minh. Ngoài ra, công cụ cũng sẽ được triển khai tại các doanh nghiệp muốn triển khai các dịch vụ SEO trên nền tảng Internet.
Bài báo của chúng tôi chia làm 5 phần: Phần 1 Giới thiệu tổng quan của hệ thống; Phần 2 Cơ sở lý thuyết; Phần 3 Công cụ IUH ONE; Phần 4 Kiểm nghiệm thực tế và Phần 5 Kết luận và hướng phát triển. Trong đó Phần 3 là phần đóng góp chính của bài báo.

## 2 CƠ SỞ LÝ THUYẾT

### 2.1 Next.js

Next.js là một framework mã nguồn mở được xây dựng trên nền tảng React và Node.js, cho phép phát triển ứng dụng web chạy trên phía người dùng và máy chủ. Next.js giúp cho việc xây dựng ứng dụng web trở nên đơn giản hơn bằng cách cung cấp cho nhà phát triển các tính năng như server-side rendering, static site generation (Hình 2.1), code splitting và pre-fetching, tất cả đều tích hợp sẵn trong framework.

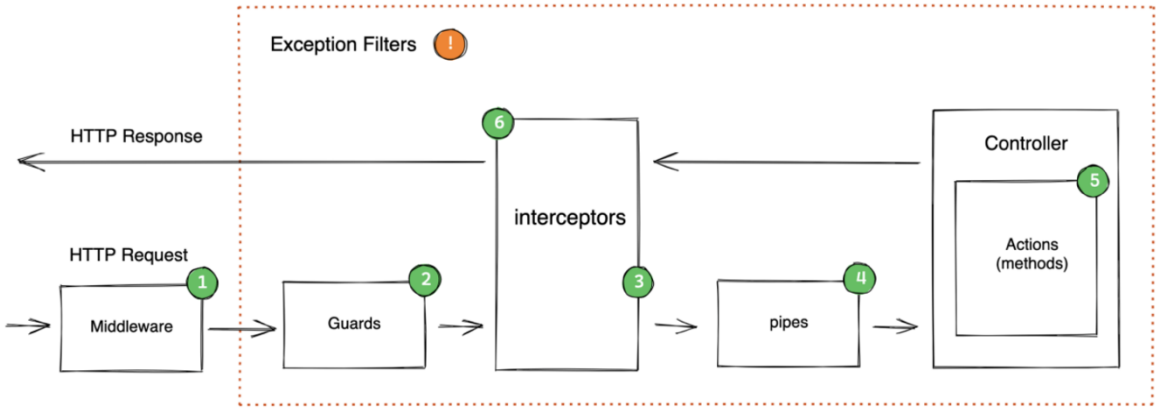



Hình 2.1: ServerSide Rendering trong NextJS

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 2.2 NestJS

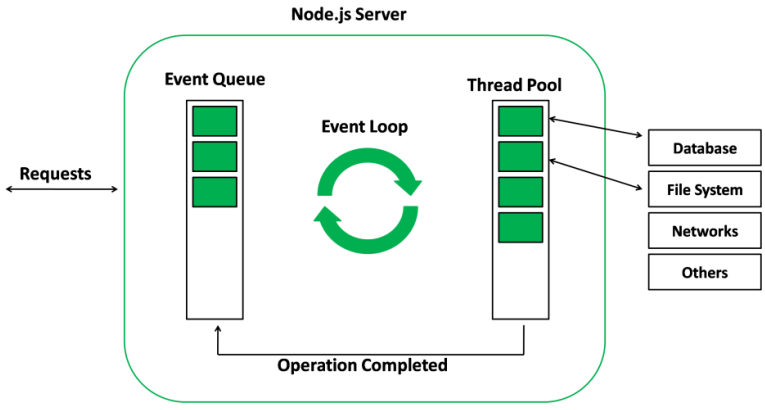
NestJS được phát triển dựa trên các khái niệm của Angular, bao gồm Dependency Injection, Decorators và Pipes, để tạo ra một kiến trúc ứng dụng rõ ràng và có cấu trúc. Nó hỗ trợ nhiều loại ứng dụng web, bao gồm các ứng dụng web SPA (Single Page Application), các ứng dụng web truyền thống và các API.
NestJS cung cấp nhiều tính năng hữu ích như Middleware, Exception Filters, Authentication và Authorization, WebSocket và nhiều hơn nữa. Nó cũng hỗ trợ nhiều loại cơ sở dữ liệu như MySQL, MongoDB, PostgreSQL, Redis, và Elasticsearch.




Hình 2.2. Vòng đời của một HTTP Request trong Nestjs

## 2.3 NodeJS

NodeJS là một nền tảng phát triển phía server dựa trên engine JavaScript V8 được phát triển bởi Google. Nó cho phép các nhà phát triển sử dụng JavaScript để viết các ứng dụng phía server bằng cách cung cấp các API và thư viện hỗ trợ cho việc xử lý các yêu cầu đến từ các client.
Với NodeJS, các nhà phát triển có thể xây dựng các ứng dụng web, ứng dụng di động, ứng dụng desktop và các ứng dụng khác. Nó có thể được sử dụng để xử lý các yêu cầu đa luồng, tạo các server WebSocket, xử lý dữ liệu realtime và nhiều tính năng khác.
NodeJS được phát triển và duy trì bởi Node.js Foundation và là một trong những công nghệ phát triển server phổ biến nhất hiện nay.



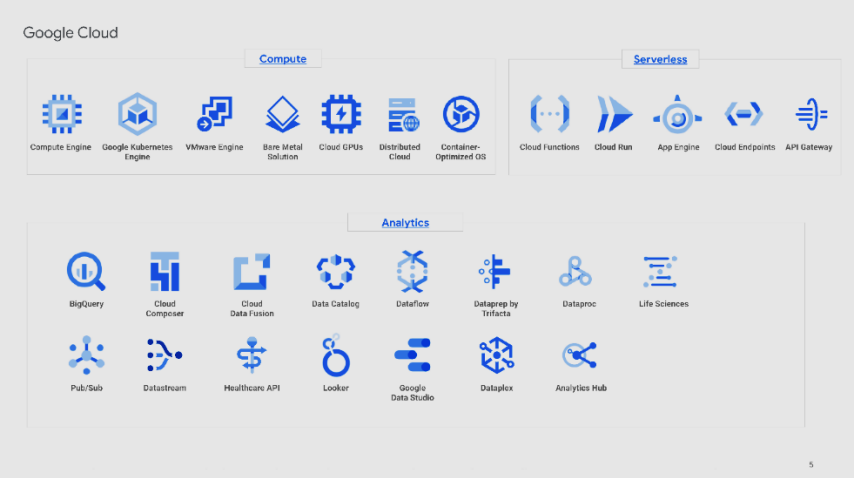

Hình 2.3: NodeJS Event Loop [8]

## 2.4 Google Cloud Platform

Google Cloud Platform (GCP) là một nền tảng đám mây được cung cấp bởi Google. GCP cung cấp một loạt các dịch vụ đám mây để giúp các doanh nghiệp và cá nhân xây dựng, triển khai và quản lý các ứng dụng và dịch vụ trên nền tảng đám mây của Google.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Các dịch vụ của Google Cloud Platform bao gồm máy ảo, lưu trữ, cơ sở dữ liệu, máy học (Machine Learning), Internet kết nối vạn vật (IoT), bảo mật và nhiều dịch vụ khác. GCP được sử dụng rộng rãi để xây dựng các ứng dụng web, game, dịch vụ và các sản phẩm khác. Nó cung cấp các tính năng như tự động mở rộng, bảo mật, khả năng tiết kiệm chi phí, độ tin cậy cao và nhiều tính năng khác giúp các khách hàng tối ưu hóa công việc của họ trên nền tảng đám mây.




Hình 2.4: Một số nhóm dịch vụ phổ biến trên Google Cloud Platform [9]

## 2.5 Firebase

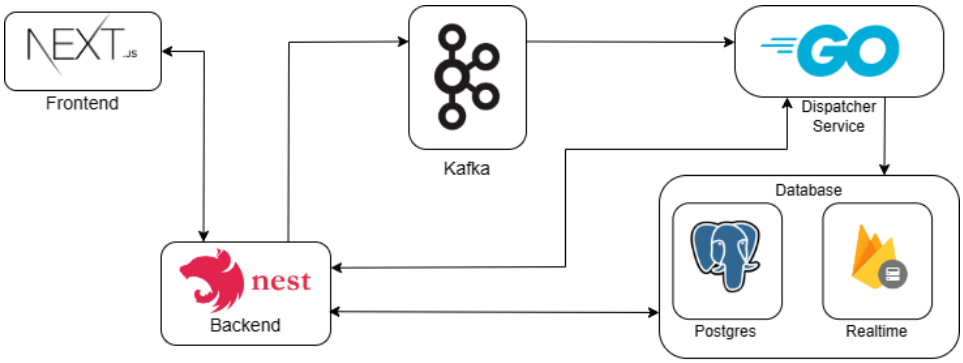
Firebase là một nền tảng của Google cung cấp các dịch vụ điện toán đám mây Backend-as-a-Service (Baas) để phát triển ứng dụng di động và web. Firebase cung cấp nhiều tính năng như lưu trữ dữ liệu, xác thực người dùng, phân tích và một số dịch vụ khác.

Với Firebase, nhà phát triển có thể xây dựng ứng dụng di động và web một cách nhanh chóng và hiệu quả hơn bằng cách sử dụng các tính năng có sẵn, giảm thiểu thời gian phát triển và chi phí. Firebase cũng cung cấp các công cụ để quản lý ứng dụng và theo dõi hoạt động của nó, giúp nhà phát triển nâng cao trải nghiệm người dùng và tối ưu hóa hiệu suất ứng dụng của họ [10].

# 3 IUH ONE: CÔNG CỤ RÚT GỌN LIÊN KẾT, THEO DÕI LƯỢT TRUY CẬP VÀ ĐÁNH GIÁ TỈ LỆ TƯƠNG TÁC CỦA NGƯỜI DÙNG CHO CÁC WEBSITE

## 3.1 Các thành phần trong hệ thống

Hệ thống IUH One được thiết kế theo mô hình Client - Server, trong đó, phía Server được thiết kế theo kiến trúc Microservices gồm một hệ thống Backend Monolithic và một Microservice (*Hình 3.1*).



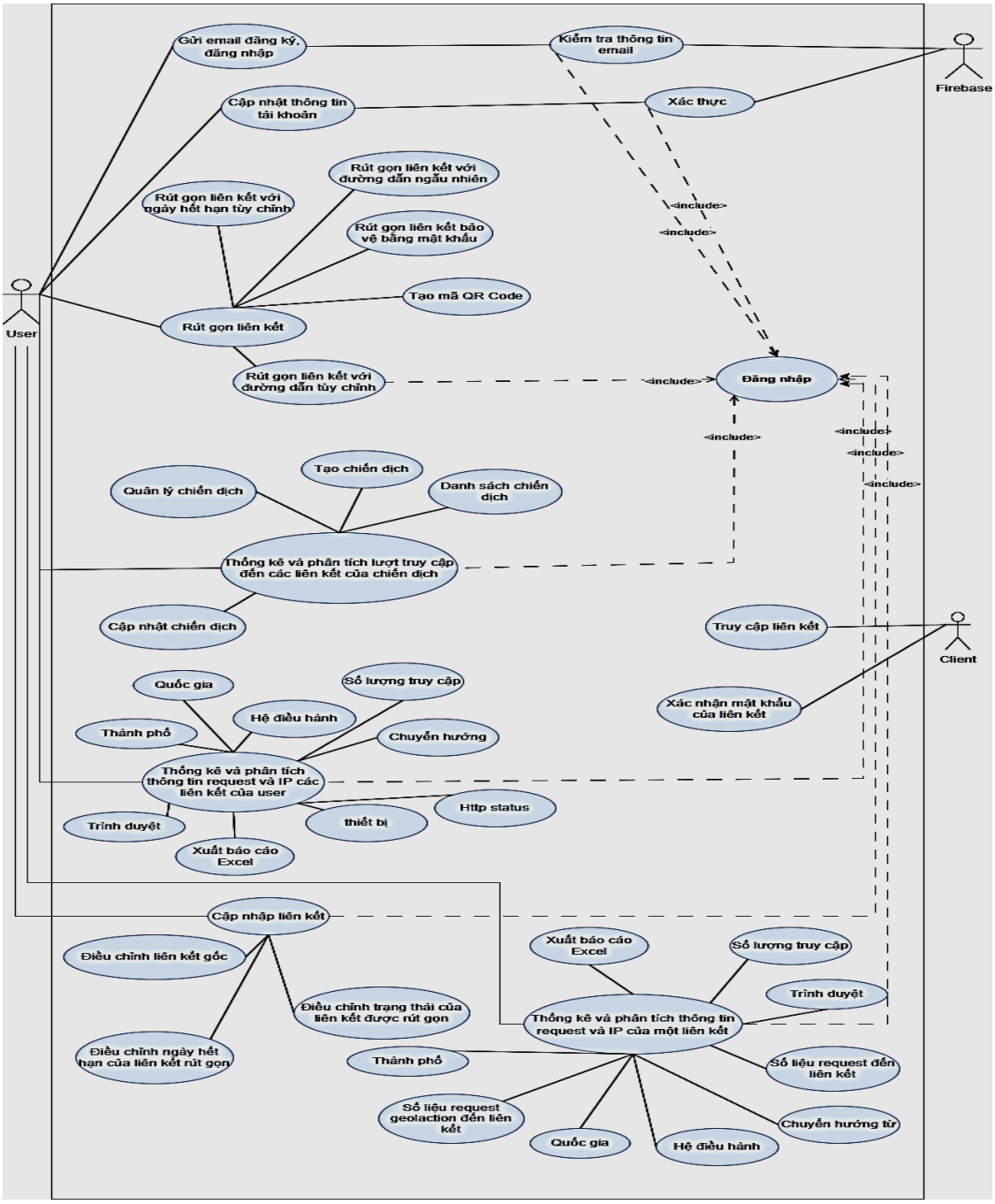

Hình 3.1: Kiến trúc hệ thống IUH ONE

## 3.2 Sơ lược công nghệ sử dụng

Về phía Frontend chúng tôi sử dụng NextJS một framework sử dụng ngôn ngữ Javascipt, các tính năng như server-side rendering, static site generation, code splitting và pre-fetching, tất cả đều tích

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

hợp sẵn trong framework giúp xây dựng hệ thống một cách nhanh chóng. Về Backend, chúng tôi sử dụng framework NestJS hỗ trợ một số tính năng hữu ích như Middleware, Exception Filters, Authentication và Authorization, WebSocket. Bên cạnh đó chúng tôi còn sử dụng một Microservice là Dispatcher Service được viết bằng ngôn ngữ Go có thể đáp ứng các yêu cầu của các hệ thống phân tán và có khả năng xử lý tốt các tác vụ đồng thời.

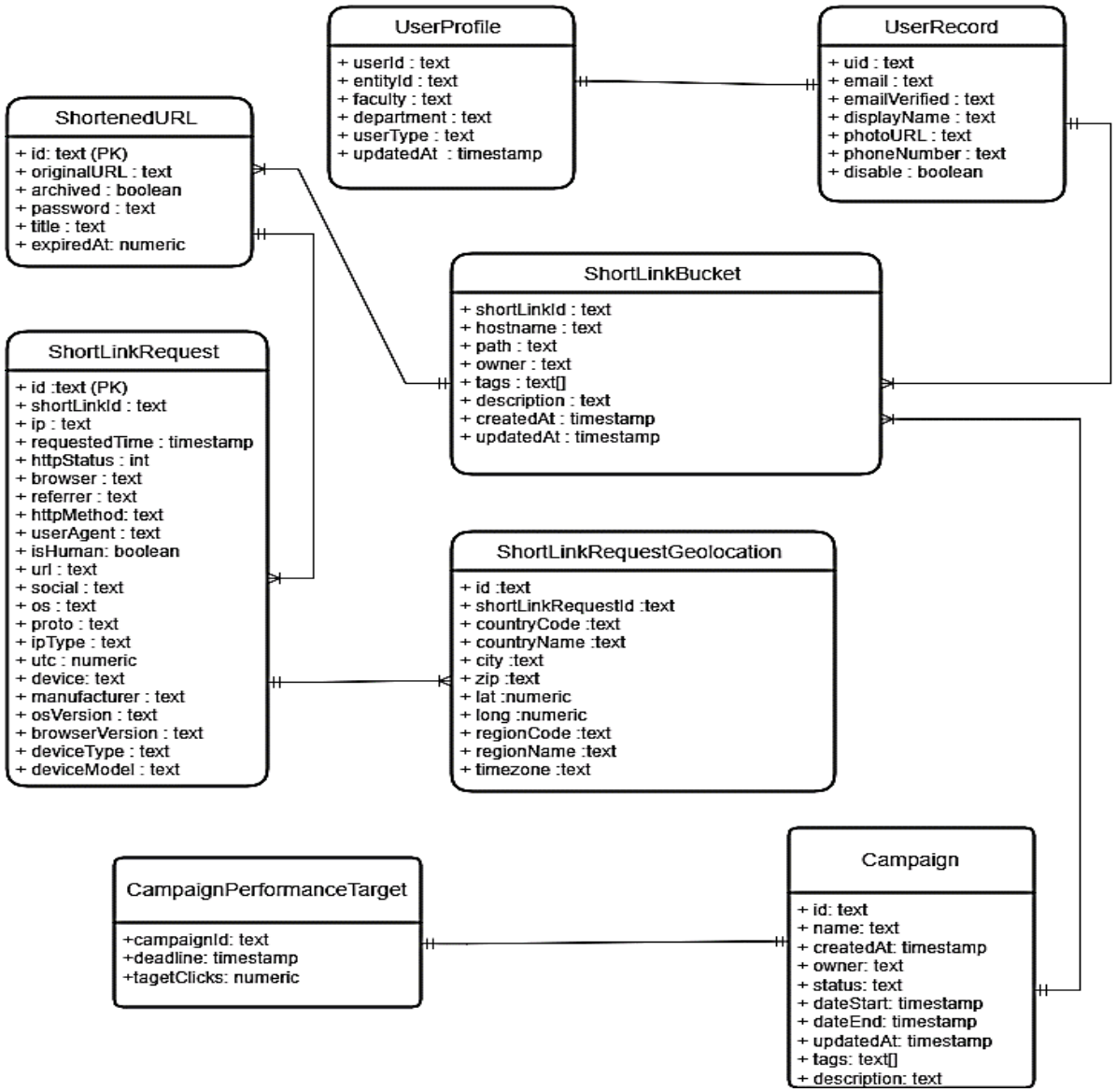



Hình 3.2: Sơ đồ Use case hệ thống IUH ONE

## 3.3 Phân tích thiết kế hệ thống

Hệ thống hỗ trợ cho hai đối tượng chính gồm: Người dùng là người sử dụng có đăng ký tài khoản trên hệ thống và có quyền sử dụng các tính năng như: rút gọn các liên kết và theo dõi các số liệu phân tích cho

những liên kết được rút gọn, tạo chiến dịch và theo dõi các chiến dịch; Người truy cập (Khách truy cập Website) là người truy cập vào các liên kết được rút gọn (*Hình 3.2*).

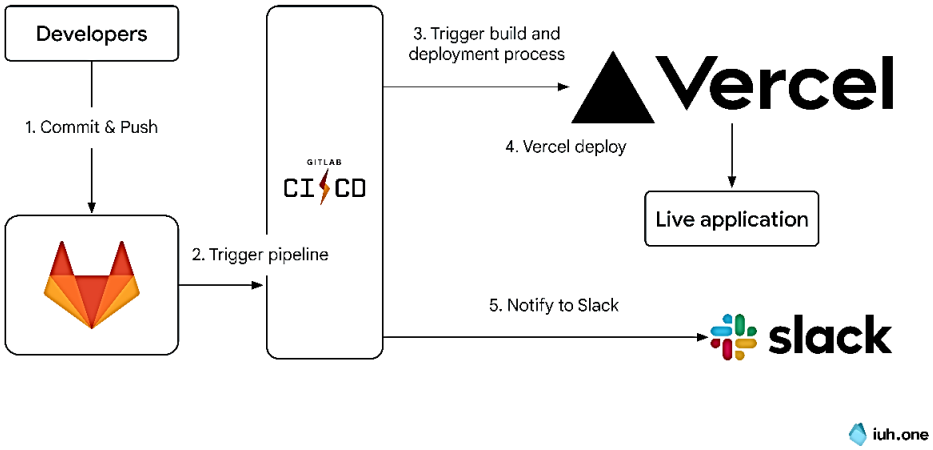



Hình 3.3 Sơ đồ cơ sở dữ liệu

## 3.4 Mô hình triển khai

Mô hình triển khai Frontend (Frontend Deployment Diagram)

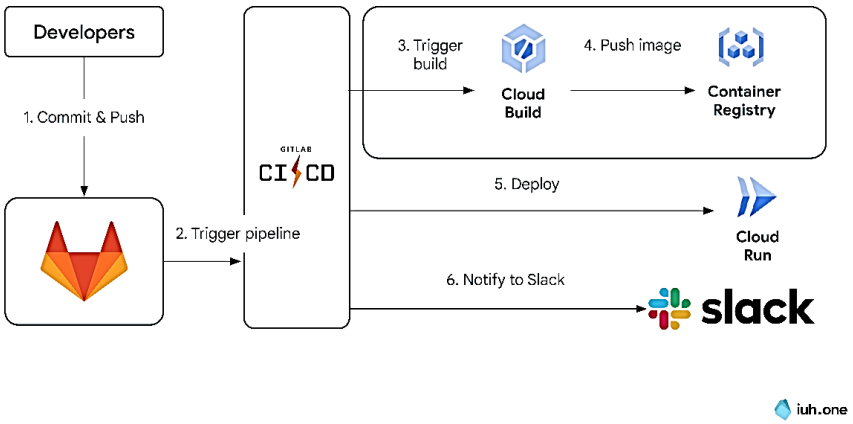
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 3.4: Sơ đồ triển khai Frontend

Vercel được thiết kế để cung cấp tốc độ và hiệu suất cao nhất cho các ứng dụng web và trang web tĩnh. Nó sử dụng một mạng lưới các máy chủ toàn cầu để đưa nội dung gần hơn với người dùng cuối và tối ưu hóa tốc độ tải trang web. Trong mô hình triển khai cho IUH ONE, chúng tôi đề xuất dùng Vercel để hỗ trợ tăng tốc truy xuất link rút gọn (*Hình 3.4*)

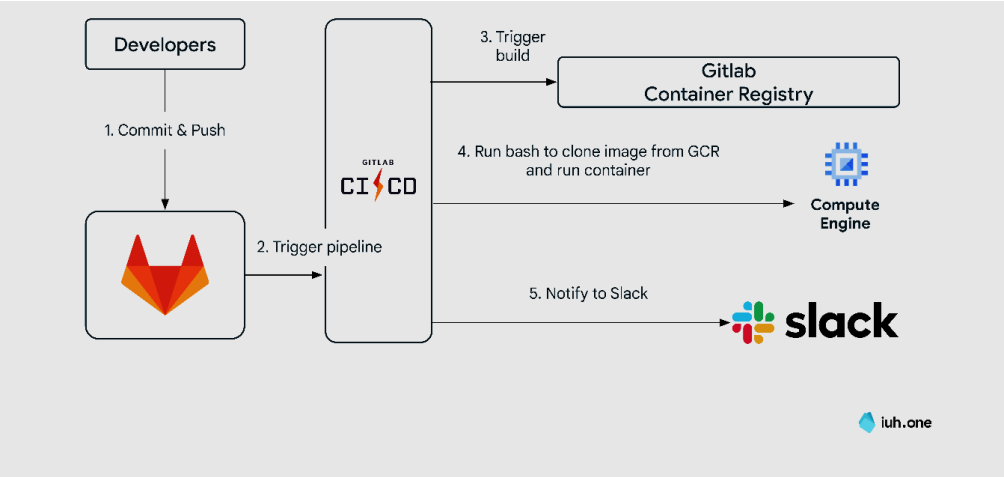
Mô hình triển khai Backend (Backend Deployment Diagram)




Hình 3.5: Sơ đồ triển khai Backend

GitLab là một nền tảng quản lý mã nguồn được sử dụng để quản lý các dự án phần mềm. Nó cung cấp nhiều tính năng, bao gồm quản lý mã nguồn, theo dõi vấn đề, phát hành, wiki, hệ thống hỗ trợ liên lạc và nhiều tính năng khác để hỗ trợ việc phát triển phần mềm.  Trong hệ thống IUH ONE chúng tôi cũng dùng GitLab để triển khai cho việc quản lý mã nguồn (*Hình 3.5*).

Dispatcher Service Deployment Diagram

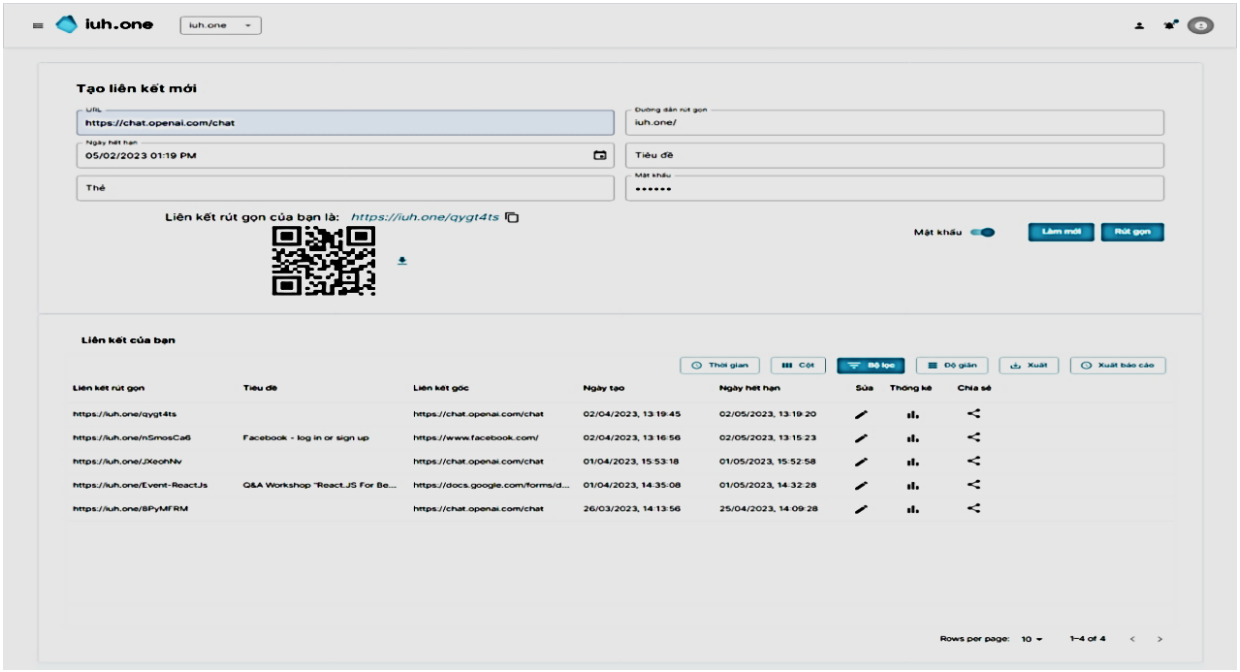



Hình 3.6: Sơ đồ triển khai Dispatcher Service

Slack là một nền tảng trò chuyện trực tuyến và cộng tác được sử dụng rộng rãi trong môi trường làm việc và nhóm làm việc. Nó cho phép người dùng gửi tin nhắn trực tiếp hoặc trò chuyện nhóm, chia sẻ tệp và tài liệu, và tích hợp với nhiều ứng dụng khác nhau để tăng cường khả năng cộng tác. Slack cũng cung cấp tính năng gọi video và âm thanh để giúp các thành viên trong nhóm giao tiếp với nhau từ xa. Slack đã trở thành một công cụ quan trọng trong việc quản lý và phát triển dự án (*Hình 3.6*).

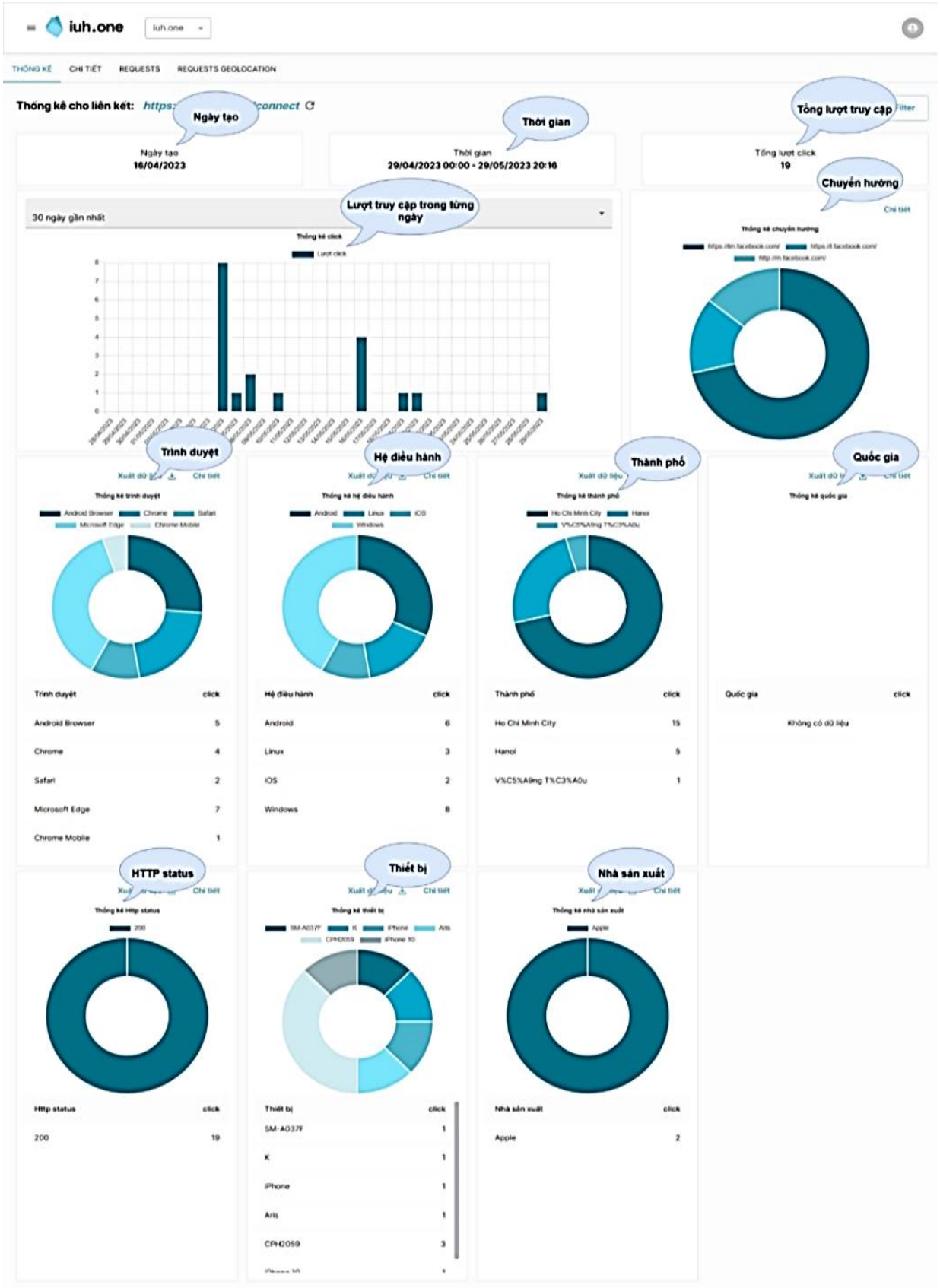
## 3.5 Giao diện của hệ thống IUH ONE

Người dùng có thể nhận được các kết quả thống kê cho các truy xuất vào link rút gọn mà hệ thống IUH ONE cung cấp. Với chức năng này, hệ thống sẽ tự động thu thập và thống kê các thông số như số lượt truy cập, thời gian, địa điểm truy cập vào link rút gọn và một số thông số khác. Đồng thời, hệ thống IUH ONE sẽ biểu diễn chúng dưới dạng các đồ thị để phân tích, so sánh với các phiên bản rút gọn hay các thông số như thời gian, địa điểm, thiết bị,…Từ đó, người dùng có thể có chiến lược để triển khai link rút gọn một cách hiệu quả theo một chiến dịch SEO nào đó (*Hình 3.7, Hình 3.8*).




Hình 3.7: Giao diện chính cho phép người dùng tạo liên kết rút gọn


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

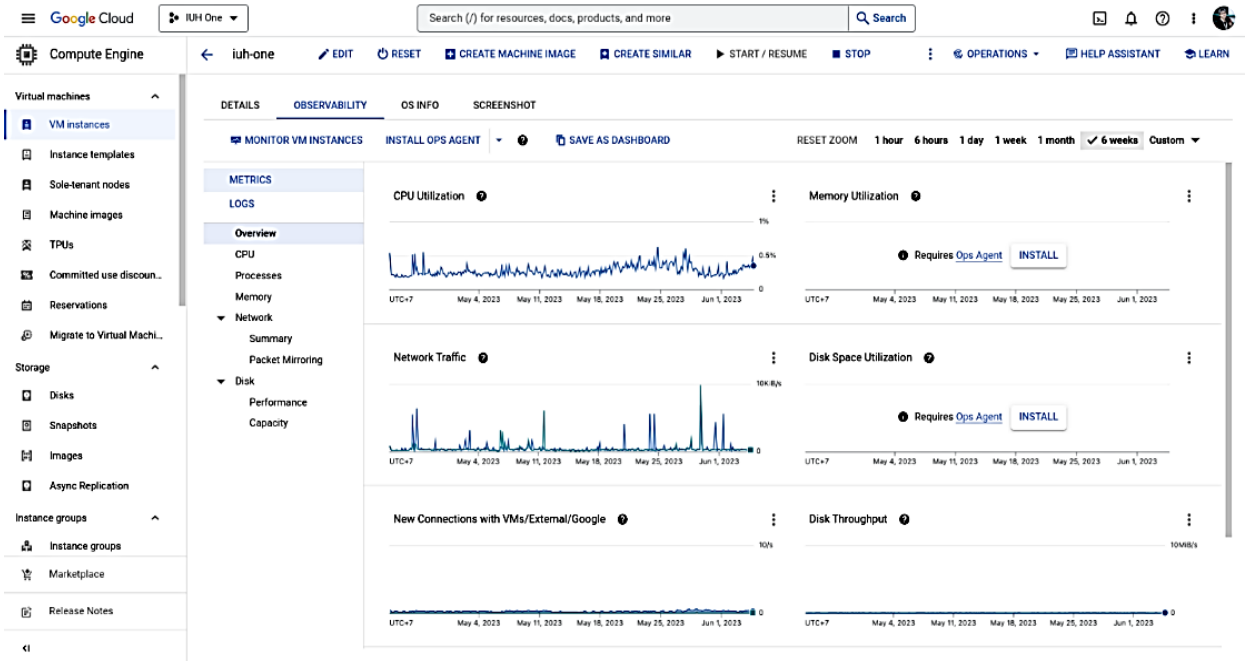



Hình 3.8: Thống kê các đại lượng đo lượng của liên kết rút gọn từ IUH ONE

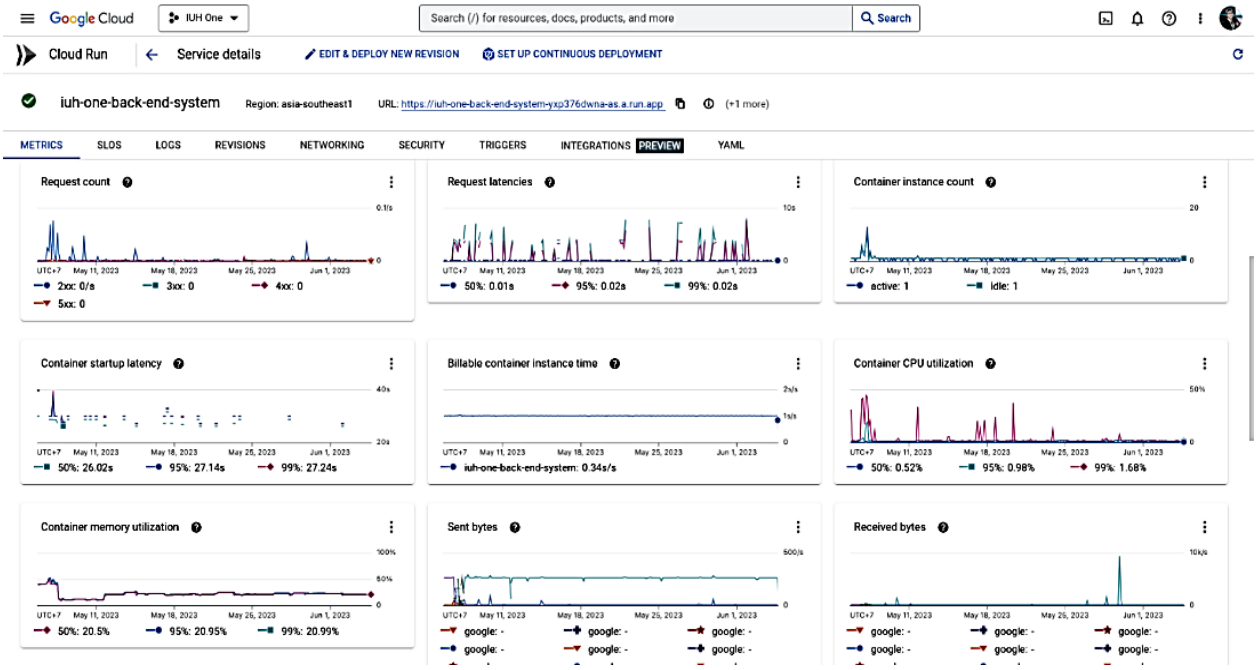
## 4 TRIỂN KHAI THỰC TẾ

Chúng tôi đã triển khai hệ thống máy chủ cho IUH One trên hệ thống Google Cloud Platfom. Trong giai đoạn này, chúng tôi sử dụng dịch vụ Compute Engine và Cloud Run để vận hành các dịch vụ của hệ thống (*Hình 4.1*, *Hình 4.2*).

Những ưu điểm của hệ thống này cũng đã được kiểm chứng bao gồm: chuỗi rút gọn an toàn và việc truy xuất với hiệu năng cao hơn so với các công cụ rút gọn khác. Dữ liệu thu thập đa dạng và được biểu diễn một cách trực quan, giúp cho các nhà phân tích, hoạch định chiến lược dễ dàng kiểm soát và đánh giá kết quả.




Hình 4.1: Compute Engine cho hệ thống IUH ONE trên nền tảng Google Cloud




Hình 4.2: Cloud Run cho hệ thống IUH ONE trên nền tảng Google Cloud


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

*Hình 4.1* và *Hình 4.2* thông tin các kết quả về hiệu năng của hệ thống IUH ONE triển khai trên môi trường Internet thật với dịch vụ GoogleCloud.

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Bài báo đã trình bày giải pháp rút gọn link người dùng và xây dựng công cụ hỗ trợ việc đánh giá hiệu suất truy xuất các link rút gọn. Để hiện thực giải pháp trên, chúng tôi đã xây dựng công cụ IUH ONE với những công nghệ phù hợp và cung cấp nhiều chức năng cho người dùng sử dụng. Trong đó, các chức năng chính được chúng tôi liệt kê như rút gọn link liên kết, phân tích và đánh giá hiện trạng của việc theo dõi và đo lường sự tương tác của người dùng đối với các liên kết đó. IUH One cho phép rút gọn liên kết với nhiều chọn lựa nhằm nâng cao tính thân thiện và an toàn hơn cho người truy cập. Công cụ còn cung cấp cơ chế bảo mật cho các liên kết rút gọn; phân tích, trích xuất các tham số từ yêu cầu HTTP như: địa chỉ IP, User-Agent. Tất cả các chỉ số hiệu suất này được trực quan hoá dữ liệu trong thời gian thực trên nền tảng web. IUH One sử dụng các công nghệ như: Next.js, NestJS, PostgreSQL, Google Cloud Platform, Firebase, v.v. Dự án của chúng tôi đã được triển khai thực tế tại Trung tâm tư vấn và hỗ trợ sinh viên trường Đại học Công nghiệp Thành phố Hồ Chí Minh, bắt đầu tháng 04/2023 trong các bài đăng tuyển dụng được chia sẻ trên mạng xã hội. Kết quả mang lại từ số lượng người dùng và số lượng link rút gọn là rất tốt (dữ liệu được triển khai với địa chỉ https://iuh.one với hàng nghìn lượt truy cập, hàng trăm liên kết đã đươc rút gọn). Những kết quả này được biểu diễn lên các biểu đồ của hệ thống Web nhằm trực quan hóa dữ liệu. Tuy nhiên, hệ thống vẫn còn một số chức năng cần mở rộng. Trong tương lai, chúng tôi sẽ liên hệ các tổ chức trong và ngoài trường để tìm kiếm các cơ hội hợp tác và mở rộng sản phẩm cải thiện các tính năng đã được triển khai, phát triển thêm một số tính năng mới như: hệ thống cho phép tạo biểu mẫu đăng ký sự kiện dựa trên liên kết rút gọn. Về mặt kỹ thuật, chúng tôi sẽ sử dụng đa dạng các nhà cung cấp dịch vụ điện toán đám mây khác nhau và áp dụng kiến trúc Microservice để tăng cường tính ổn định và hiệu quả của hệ thống.

## TÀI LIỆU THAM KHẢO

[1]. K. Faaborg and S. Pasquali *Mastering Node. js: build robust and scalable real-time server-side web applications efficiently*, Second Edition Packt Publishing Ltd, 2017.

[2]. Casciaro, Mario, and Luciano Mammino. *Node. js Design Patterns: Design and implement production-grade Node. js applications using proven patterns and techniques*. Packt Publishing Ltd, 2020.

[3]. C. Doxsey. *An Introduction to Programming in Go*. CreateSpace Independent Publishing Platform, 2012

[4]. Richardson, Chris. *Microservices patterns: with examples in Java*. Simon and Schuster, 2018.

[5]. Stigler, M. *Beginning Serverless Computing: Developing with Amazon Web Services*. Microsoft Azure, and Google Cloud 2018.

[6]. Next.js - https://nextjs.org/docs accessed 04/06/2023

[7]. NestJS - https://docs.nestjs.com accessed 04/06/2023

[8]. NodeJS - https://nodejs.org accessed 04/06/2023

[9]. Google Cloud Platform - https://cloud.google.com/docs accessed 04/06/2023

[10]. Firebase - https://firebase.google.com accessed 04/06/2023

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC.F008**

# XÂY DỰNG MỘT ỨNG DỤNG TRỰC TUYẾN VỚI MICROSERVICES VÀ CLUSTERING

TẠ DUY CÔNG CHIẾN[1], NGUYỄN VĂN TOÀN[2]

*[1]Information Technology Department, Industrial University of Ho Chi Minh City,*
*[2]Industrial University of Ho Chi Minh City;*
*taduycongchien@iuh.edu.vn*

**Tóm tắt.** Ngày nay, khi mà điện toán đám mây ngày càng phát triển thì việc các ứng dụng hoạt động trên nền tảng đám mây ngày càng nhiều. Bài báo này nhằm mục đích xây dựng một ứng dụng trực tuyến "Hệ thống bán sách", sử dụng công nghệ Vi dịch vụ (Microservices) và triển khai trên nền tảng đám mây của Google. Hệ thống này sẽ cung cấp cho người dùng các chức năng như tìm kiếm, xem chi tiết, đặt hàng, thanh toán và đánh giá sách. Hệ thống này cũng sẽ quản lý các thông tin về sách, khách hàng, đơn hàng và doanh thu. Để xây dựng hệ thống này, cúng tôi đã kết hợp công nghệ Nestjs để phát triển các Vi dịch vụ và React Native để phát triển ứng dụng di động phía giao diện người dùng (frontend). Chúng tôi cũng sử dụng nền tảng đám mây của Google với công nghệ công nghệ gom cụm (clustering) để triển khai hệ thống. Hệ thống bán sách trực tuyến phát triển trên công nghệ Vi dịch vụ (Microservices) triển khai trên nền tảng đám mây của Google với sự hỗ trợ của công nghệ gom cụm là một đề tài có tính thực tiễn cao, phù hợp với xu hướng hiện nay của các ứng dụng thương mại điện tử và đáp ứng được nhu cầu của người dùng.

**Từ khóa.** Microservices, Back-end, React Native.

## BUILDING AN ONLINE APPLICATION USING MICROSERVICES AND CLUSTERING

**Abstract.** Currently, as cloud computing is increasingly advancing, the number of applications operating on cloud-based platforms is also increasing. This paper proposes an approach to build an online application, "Book-selling system", using Microservices technology and running on cloud, such as Google Cloud. This system will provide users with functions such as searching, viewing details, ordering, paying, and rating books. This system will also manage information about books, customers, orders, and revenue. To build this system, we combine Nestjs technology to develop Microservices and React Native to develop mobile frontend applications. We also use Google Cloud with clustering technology to deploy this application. The online application "Book-selling system" with Microservices technology deployed on Google Cloud using clustering is a current trend with high practicality, suitable for the e-commerce applications and meeting the needs of end-users.

**Keywords.** Microservices, Clustering, Online application.

## 1 GIỚI THIỆU TỔNG QUAN

Trong bối cảnh hiện nay, điện toán đám mây đang trở thành một phần quan trọng của nền tảng công nghệ thông tin. Việc sử dụng điện toán đám mây mang lại nhiều lợi ích, bao gồm sự linh động, tiết kiệm chi phí, và khả năng mở rộng dễ dàng. Trong lĩnh vực ứng dụng trực tuyến, việc triển khai các ứng dụng trên nền tảng đám mây ngày càng phổ biến và mang lại nhiều lợi ích cho người dùng. Trong trường hợp cụ thể của hệ thống bán sách trực tuyến, việc sử dụng công nghệ đám mây có thể đem lại nhiều tiện ích cho cả người quản lý hệ thống và người dùng đầu cuối.

Trong bài báo này, chúng tôi đã xây dựng một ứng dụng có tên "Hệ thống bán sách trực tuyến" sử dụng công nghệ Vi dịch vụ và triển khai trên nền tảng đám mây của Google chỉ là một ứng dụng nhằm khai thác

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

những lợi ích mà công nghệ đám mây và kiến trúc Vi dịch vụ mang lại. Mục tiêu của chúng tôi là xây dựng một ứng dụng có khả năng tìm kiếm, xem chi tiết, đặt hàng, thanh toán và đánh giá, cung cấp trải nghiệm mua sắm trực tuyến tốt nhất cho người dùng để từ đó xây dựng các ứng dụng thiết thực hơn phục vụ cho xã hội và nhân loại.

Các lợi ích của việc sử dụng công nghệ Vi dịch vụ và triển khai trên nền tảng đám mây bao gồm:

- Tính mở rộng: Kiến trúc Vi dịch vụ cho phép chia nhỏ ứng dụng thành các dịch vụ nhỏ hơn, dễ dàng mở rộng theo nhu cầu người dùng và tải trọng của hệ thống.
- Tính sẵn sàng cao: sử dụng công nghệ đám mây giúp đảm bảo khả năng sẵn sàng và khả năng chịu lỗi cao, nhờ có sự phân tán và sao lưu dữ liệu.
- Tiết kiệm chi phí: Việc sử dụng dịch vụ đám mây cho phép giảm thiểu việc đầu tư vào cơ sở hạ tầng phần cứng và phần mềm, chỉ trả phí dựa trên việc sử dụng thực tế.
- Dễ dàng quản lý và mở rộng: Việc sử dụng công nghệ Vi dịch vụ và triển khai trên nền tảng đám mây giúp quản lý và mở rộng hệ thống dễ dàng hơn. Các dịch vụ độc lập trong kiến trúc Vi dịch vụ cho phép phát triển và cập nhật chúng độc lập với nhau, giúp khả năng mở rộng của hệ thống dễ dàng hơn.
- Tích hợp hệ thống: Kiến trúc Vi dịch vụ cho phép tích hợp các dịch vụ khác nhau dễ dàng. Việc triển khai trên nền tảng đám mây cung cấp các công cụ và dịch vụ hỗ trợ tích hợp, giúp kết nối các thành phần của hệ thống một cách nhanh hơn.
- Cung cấp trải nghiệm người dùng tốt hơn: Bằng cách sử dụng công nghệ Vi dịch vụ và triển khai trên nền tảng đám mây, chúng tôi có thể cung cấp một ứng dụng trực tuyến nhanh chóng và ổn định. Người dùng có thể tìm kiếm, xem chi tiết, đặt hàng và thanh toán một cách thuận tiện, đồng thời có khả năng đánh giá để chia sẻ thông tin và trải nghiệm của mình với cộng đồng.

Với những lợi ích trên, việc xây dựng hệ thống bán sách trực tuyến sử dụng công nghệ Vi dịch vụ và triển khai trên nền tảng đám mây phù hợp với xu hướng phát triển của ứng dụng thương mại điện tử hiện nay. Bài báo nhằm tìm hiểu và đánh giá khả năng thực tiễn và tính ứng dụng của việc kết hợp các công nghệ này trong việc xây dựng một ứng dụng trực tuyến một cách linh hoạt.

Với các mục tiêu này, chúng tôi hy vọng rằng bài báo này sẽ mang lại cái nhìn tổng quan về việc xây dựng một ứng dụng trực tuyến sử dụng công nghệ Vi dịch vụ và triển khai trên nền tảng đám mây.

## 1.1 Giới thiệu Vi dịch vụ (Microservices)

Trong quá trình xây dựng hệ thống bán sách trực tuyến, chúng tôi sử dụng công nghệ Vi dịch vụ để thiết kế kiến trúc của ứng dụng. Kiến trúc Vi dịch vụ là một phương pháp phát triển phần mềm mà trong đó ứng dụng được chia thành các dịch vụ nhỏ, độc lập và phân tán, mỗi dịch vụ đảm nhận một chức năng cụ thể [1] .

Các Vi dịch vụ trong hệ thống bán sách được xây dựng bằng cách sử dụng công nghệ Nestjs. Nestjs là một ngôn ngữ phát triển ứng dụng back-end được xây dựng trên nền tảng Nodejs. Nó cung cấp các thành phần và công cụ cần thiết để phát triển các Vi dịch vụ, bao gồm quản lý route, middleware, dependency injection và giao tiếp với cơ sở dữ liệu [2].

Mỗi Vi dịch vụ trong hệ thống bán sách có thể được triển khai độc lập và có khả năng mở rộng riêng. Việc chia nhỏ ứng dụng thành các dịch vụ độc lập giúp chúng tôi dễ dàng phát triển, bảo trì và mở rộng từng phần của hệ thống mà không ảnh hưởng đến các phần khác .

Kiến trúc Vi dịch vụ còn mang lại các lợi ích như khả năng tận dụng tài nguyên tốt, cải thiện sự phân tán và chịu lỗi, tăng khả năng tích hợp của hệ thống. Với kiến trúc này, chúng tôi có thể phát triển và triển khai các dịch vụ độc lập như Dịch vụ quản lý sách, Dịch vụ quản lý khách hàng, Dịch vụ quản lý đơn hàng và Dịch vụ thanh toán. Mỗi dịch vụ đảm nhận một chức năng cụ thể và có khả năng hoạt động độc lập với các dịch vụ khác.

Sử dụng kiến trúc Vi dịch vụ giúp chúng tôi xây dựng hệ thống trực tuyến dễ mở rộng và quản lý. Mỗi dịch vụ có thể được phát triển và triển khai độc lập, và chúng có khả năng hoạt động cùng nhau để cung cấp các chức năng đầy đủ cho người dùng.

### 1.2 Giới thiệu React Native

Trong phần giao diện người dùng của hệ thống bán sách trực tuyến, chúng tôi sử dụng công nghệ React Native để phát triển ứng dụng di động. React Native là một ngôn ngữ dùng phát triển ứng dụng di động đa nền tảng, cho phép viết mã một lần và triển khai trên cả hai nền tảng iOS và Android.

Việc sử dụng React Native giúp chúng tôi tiết kiệm thời gian và công sức trong việc phát triển ứng dụng di động, vì chúng tôi chỉ cần phát triển một phiên bản duy nhất của ứng dụng và có thể triển khai trên cả hai hệ điều hành di động chính. Điều này giúp tiết kiệm tài nguyên phát triển và bảo trì.

Với React Native, chúng tôi có thể sử dụng các thành phần UI đã được xây dựng sẵn và có thể tái sử dụng được, như TextInput, Button, hay ScrollView. Điều này giúp tăng tốc độ phát triển và đảm bảo tính nhất quán của giao diện người dùng trên các nền tảng khác nhau.

React Native cũng hỗ trợ các thư viện mạnh mẽ để tương tác với API và dữ liệu từ phía phụ trợ (backend). Chúng tôi có thể sử dụng thư viện Axios để gửi và nhận dữ liệu từ các restfullAPI của các dịch vụ Microservices trong hệ thống. Sử dụng Axios giúp chúng tôi xây dựng các yêu cầu HTTP và xử lý dữ liệu trả về một cách dễ dàng.

Với sự hỗ trợ của React Native, chúng tôi có thể tạo ra một giao diện người dùng di động thu hút và dễ sử dụng cho ứng dụng trực tuyến. Người dùng có thể tìm kiếm, xem thông tin chi tiết, đặt hàng và thanh toán một cách thuận tiện và trực quan trên các thiết bị di động của họ.

### 1.3 Giới thiệu Google Cloud và công nghệ gom cụm

Trong quá trình triển khai hệ thống bán sách trực tuyến, chúng tôi chọn sử dụng nền tảng đám mây của Google, được gọi là Google Cloud. Google Cloud cung cấp một loạt các dịch vụ đám mây mạnh mẽ và dễ dàng cho việc triển khai và quản lý ứng dụng trực tuyến.

Một trong những công nghệ quan trọng mà chúng tôi sử dụng trong việc triển khai hệ thống là công nghệ gom cụm. Công nghệ gom cụm trong Google Cloud cho phép chúng tôi tổ chức các tài nguyên và ứng dụng của hệ thống vào nhóm nhỏ, giúp tăng tính sẵn sàng, khả năng mở rộng và quản lý dễ dàng.

Việc sử dụng công nghệ gom cụm trong hệ thống bán sách cho phép chúng tôi phân chia các dịch vụ và tài nguyên vào các nhóm khác nhau, mỗi nhóm có thể chịu tải khác nhau và được điều phối một cách thích hợp. Điều này giúp chúng tôi xử lý tải lớn và đảm bảo hiệu suất ứng dụng trong quá trình hoạt động.

Ngoài ra, Google Cloud cũng cung cấp các dịch vụ liên quan đến quản lý cụ thể của hệ thống. Ví dụ, chúng tôi có thể sử dụng dịch vụ Auto Scaling để tự động mở rộng số lượng các máy ảo khi tải cao và thu hẹp khi tải thấp, giúp tiết kiệm tài nguyên và tối ưu hiệu suất. Chúng tôi cũng sử dụng dịch vụ cân bằng tải (Load Balancing) để phân phối lưu lượng truy cập đến các Vi dịch vụ khác nhau, đảm bảo rằng không có dịch vụ nào bị quá tải.

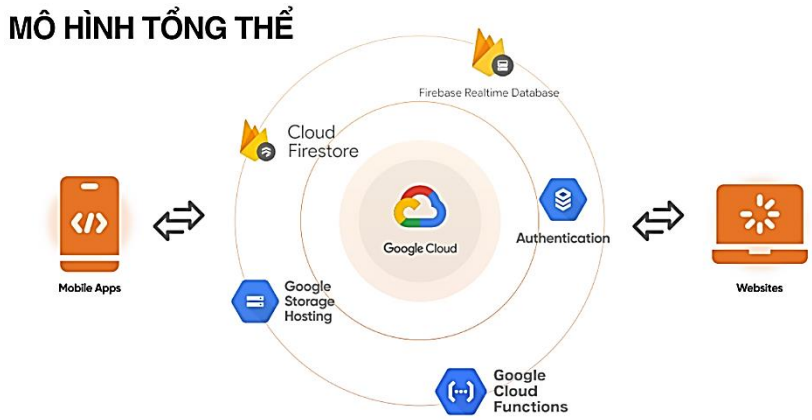
Sử dụng Google Cloud và công nghệ gom cụm giúp chúng tôi triển khai hệ thống bán sách trực tuyến một cách tin cậy và có khả năng mở rộng. Chúng tôi có thể tận dụng các dịch vụ mạnh mẽ của Google Cloud để quản lý và điều phối các Vi dịch vụ của hệ thống, đồng thời đảm bảo hiệu suất và khả năng phục vụ cho người dùng.

## 2 XÂY DỰNG ỨNG DỤNG VÀ TRIỂN KHAI HỆ THỐNG

### 2.1 Kiến trúc tổng quan của hệ thống

Hệ thống bán sách trực tuyến được thiết kế dựa trên kiến trúc Vi dịch vụ và triển khai trên nền tảng đám mây của Google. Kiến trúc tổng quan của hệ thống bao gồm các thành phần chính như: các Vi dịch vụ, cơ sở dữ liệu, giao diện người dùng, tất cả sau khi xây dựng xong sẽ được triển khai nền tảng đám mây của Google. Mô hình tổng thể của hệ thống được mô tả như hình 1.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 1: Mô hình tổng quan của hệ thống trực tuyến

## 2.2 Các thành phần chính của hệ thống trực tuyến

Tổng thể, hệ thống bán sách trực tuyến được thiết kế với một giao diện người dùng giao diện người dùng, các Vi dịch vụ xử lý tuần tự các nghiệp vụ, cơ sở dữ liệu lưu trữ thông tin, điện toán đám mây của Google để triển khai hệ thống và công nghệ gom cụm để quản lý tài nguyên. Sự kết hợp của các thành phần này đảm bảo tính mở rộng và độ tin cậy cao của hệ thống bán sách trực tuyến. Hệ thống bán sách trực tuyến được thiết kế với các thành phần chính như sau:

### 2.2.1 Giao diện người dùng

- Ứng dụng di động: Giao diện người dùng được xây dựng bằng React Native, hỗ trợ các thiết bị di động như điện thoại thông minh và máy tính bảng. Người dùng có thể tìm kiếm sách, xem thông tin chi tiết, đặt hàng và thanh toán thông qua ứng dụng này.
- Giao diện quản trị (Web sites): Giao diện dành cho quản trị viên để quản lý các thông tin hệ thống, bao gồm quản lý sách, khách hàng, đơn hàng và doanh thu. Giao diện này cung cấp các chức năng quản lý, xem thống kê và quản lý tài khoản quản trị.

### 2.2.2 Phụ trợ

- Vi Dịch vụ (Microservices): Hệ thống sử dụng kiến trúc Vi dịch vụ với các dịch vụ độc lập để quản lý các chức năng cụ thể của hệ thống. Điều này giúp tách biệt và quản lý các thành phần khác nhau của hệ thống. Các Vi dịch vụ có thể được triển khai và mở rộng độc lập, giúp tối ưu hiệu suất và khả năng mở rộng. Các dịch vụ Microservices bao gồm:
    - Dịch vụ Quản lý Sách: Xử lý việc tìm kiếm, xem chi tiết và đánh giá sách.
    - Dịch vụ Quản lý Khách hàng: Quản lý thông tin khách hàng, đăng ký, đăng nhập và quản lý tài khoản người dùng.
    - Dịch vụ Quản lý Đơn hàng: Xử lý quá trình đặt hàng, thanh toán và quản lý thông tin đơn hàng.
    - Dịch vụ Thanh toán: Liên kết với các cổng thanh toán để xử lý giao dịch thanh toán.

### 2.2.3 Cơ sở dữ liệu

Dữ liệu trong hệ thống được lưu trữ theo các đối tượng riêng biệt, bao gồm dữ liệu sách, khách hàng, đơn hàng và doanh thu. Các cơ sở dữ liệu này đảm bảo tính nhất quán và đáng tin cậy của dữ liệu trong quá trình hoạt động của hệ thống. Cụ thể bao gồm:

- Dữ liệu sách: Lưu trữ thông tin về sách bao gồm tiêu đề, tác giả, thể loại và mô tả.
- Dữ liệu khách hàng: Lưu trữ thông tin về khách hàng bao gồm tên, địa chỉ, số điện thoại và thông tin đăng nhập.
- Dữ liệu đơn hàng: Lưu trữ thông tin về đơn hàng bao gồm sách được đặt, số lượng, giá trị và trạng thái đơn hàng.
- Dữ liệu doanh thu: Ghi lại thông tin về doanh thu từ các giao dịch mua sách.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

### 2.2.4 Google Cloud và công nghệ gom cụm

- Sử dụng nền tảng đám mây của Google để triển khai hệ thống. Điều này cung cấp cho hệ thống một môi trường đáng tin cậy, mở rộng và dễ quản lý.
- Công nghệ gom cụm được sử dụng để tổ chức và quản lý các tài nguyên của hệ thống vào các nhóm nhỏ, gọi là cụm (cluster). Các cụm này có thể được quản lý độc lập và có thể mở rộng khi có nhu cầu. Việc sử dụng công nghệ gom cụm giúp tăng tính sẵn sàng và khả năng mở rộng của hệ thống, đồng thời đảm bảo hiệu suất cao và khả năng chịu tải.

### 2.2.5 Các giao tiếp và giao thức

- Các Vi dịch vụ giao tiếp với nhau thông qua các giao thức truyền thông như HTTP hoặc gRPC. Điều này cho phép các dịch vụ liên lạc và trao đổi dữ liệu để hoàn thành các nhiệm vụ cụ thể của hệ thống.
- Giao diện người dùng giao tiếp với phụ trợ thông qua các restfull API được cung cấp bởi các Vi dịch vụ. Người dùng có thể gửi yêu cầu và nhận phản hồi từ hệ thống thông qua các API này.

## 2.3 Giao tiếp giữa các Vi dịch vụ

Trong hệ thống bán sách trực tuyến, việc giao tiếp giữa các Vi dịch vụ là rất quan trọng để chia sẻ dữ liệu và thực hiện các nhiệm vụ cụ thể. Một trong những công cụ phổ biến được sử dụng để hỗ trợ giao tiếp giữa các Vi dịch vụ ở đây chúng tôi sử dụng là Redis.

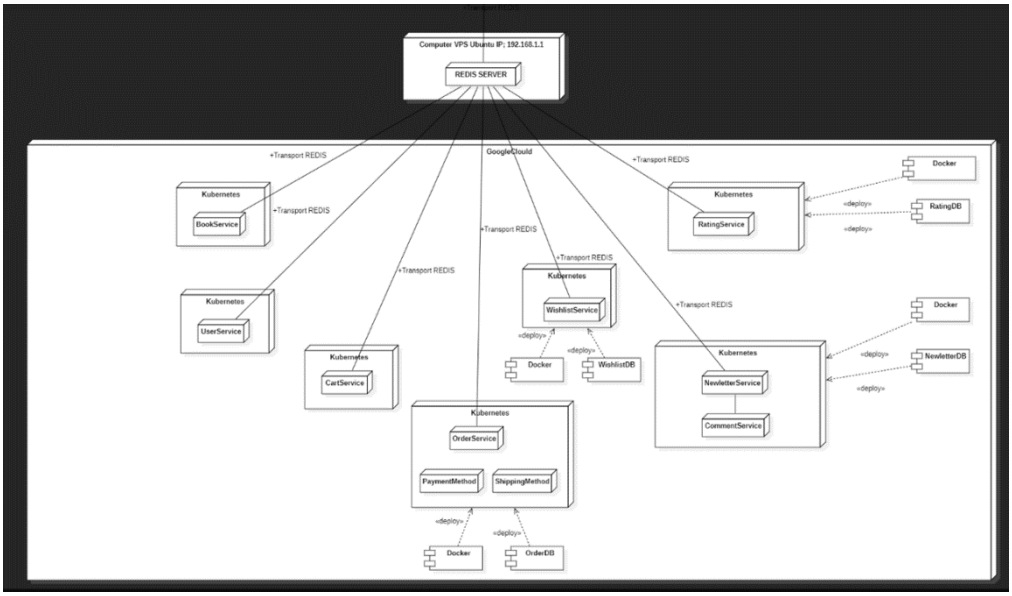
Redis là một hệ thống cơ sở dữ liệu lưu trữ dữ liệu trên bộ nhớ (in-memory database) với hiệu năng cao. Nó cung cấp một cấu trúc dữ liệu khóa giá trị đơn giản, và hỗ trợ các phương thức và thao tác nhanh chóng cho việc lưu trữ và truy xuất dữ liệu.

Trong kiến trúc Vi dịch vụ của hệ thống bán sách trực tuyến, Redis được sử dụng như một cơ chế trung gian để giao tiếp giữa các dịch vụ Vi dịch vụ. Các tác vụ chính mà Redis hỗ trợ bao gồm:

- Caching dữ liệu: Redis được sử dụng để lưu trữ các dữ liệu phổ biến và thường xuyên được truy cập, như thông tin sách phổ biến, danh sách khách hàng, và thông tin đơn hàng mới nhất. Bằng cách lưu trữ dữ liệu này trong Redis, các Vi dịch vụ có thể nhanh chóng truy xuất dữ liệu mà không cần truy vấn cơ sở dữ liệu chính, giảm thiểu thời gian truy cập và tăng tốc độ phản hồi.
- Message Queue: Redis hỗ trợ việc triển khai mô hình hàng đợi (queue) thông qua tính năng Pub/Sub (publisher/subscriber). Các Vi dịch vụ có thể gửi và nhận các thông điệp thông qua Redis để thông báo, đồng bộ hoặc thực hiện các tác vụ liên quan. Ví dụ, khi một đơn hàng mới được đặt, Vi dịch vụ Quản lý Đơn hàng có thể gửi thông điệp thông qua Redis để cập nhật thông tin đơn hàng và thông báo cho Vi dịch vụ Thanh toán thực hiện thanh toán.
- Locking và Semaphore: Redis cung cấp khả năng khóa và giữ khóa (locking) trên các tài nguyên chia sẻ. Điều này cho phép các Vi dịch vụ thực hiện các thao tác đồng bộ hoặc đảm bảo tính nhất quán trong việc truy cập và cập nhật dữ liệu. Ví dụ, khi hai Vi dịch vụ cần truy cập cùng một tài nguyên đồng thời, Redis có thể được sử dụng để khóa tài nguyên và đảm bảo rằng chỉ có một Vi dịch vụ được phép truy cập vào tài nguyên đó trong một thời điểm khi muốn thay đổi giá trị của tài nguyên đó.

Nói chung, việc sử dụng Redis trong hệ thống bán sách trực tuyến giúp cải thiện hiệu suất và khả năng mở rộng của các Vi dịch vụ. Nó cung cấp cơ chế caching để giảm thiểu truy cập vào cơ sở dữ liệu chính, hỗ trợ giao tiếp bằng cách sử dụng hàng đợi thông qua tính năng Pub/Sub và cung cấp khả năng locking và semaphore để đồng bộ hóa các thao tác trên tài nguyên chia sẻ. Sử dụng Redis là một trong các lựa chọn hợp lý trong kiến trúc Vi dịch vụ để đảm bảo tính nhất quán và hiệu suất của hệ thống. Sơ đồ giao tiếp giữa Redis với các Vi dịch vụ sau khi triển khai hệ thống trực tuyến được mô tả như hình 2.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh



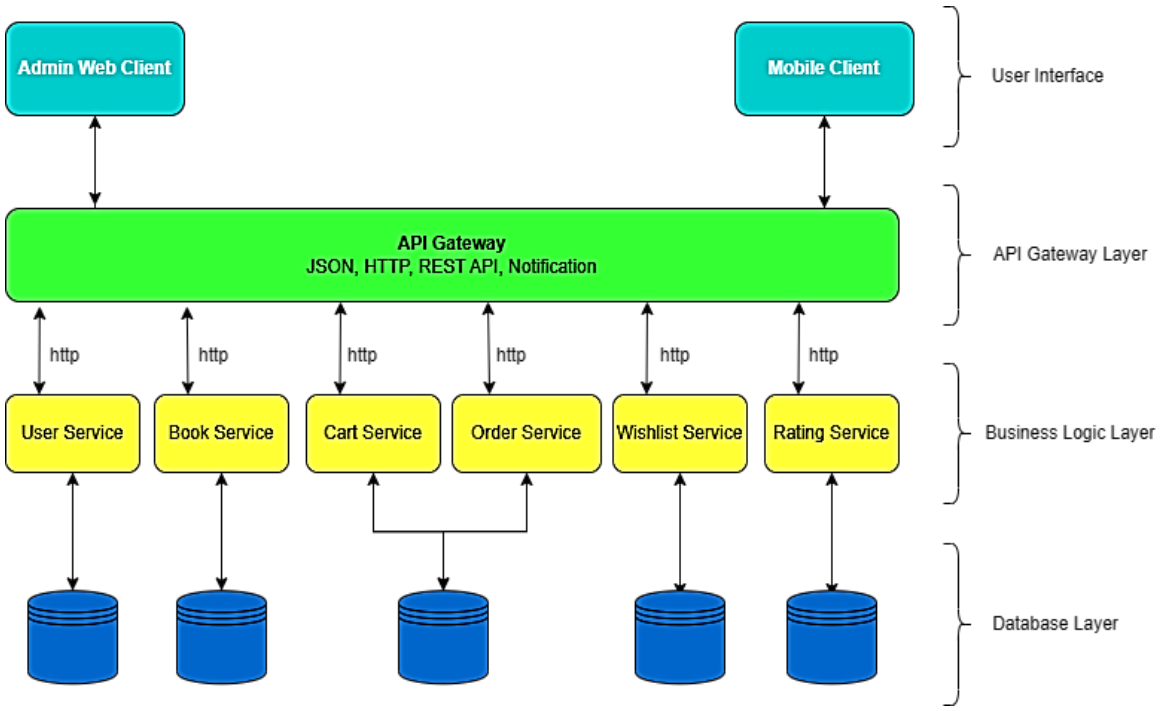

Hình 2. Giao tiếp giữa Redis với các Vi dịch vụ

## 3 TRIỂN KHAI HỆ THỐNG

Một trong những cách để xây dựng hệ thống bán hàng trực tuyến phục vụ nhiều người dùng và có thể mở rộng là triển khai kiến trúc Vi dịch vụ. Điều này liên quan đến việc chia nhỏ ứng dụng thành các dịch vụ độc lập, nhỏ hơn có thể được phát triển, triển khai và mở rộng một cách độc lập. Bước đầu tiên trong việc triển khai Vi dịch vụ cho hệ thống bán sách trực tuyến là thiết kế kiến trúc. Điều này liên quan đến việc xác định các dịch vụ khác nhau cần thiết cho hệ thống và xác định các hợp đồng dịch vụ [3]. Bằng cách làm như vậy, mỗi dịch vụ có thể được phát triển và triển khai độc lập, mang lại tính linh hoạt và khả năng mở rộng cao hơn. Khi kiến trúc vi dịch vụ đã được thiết kế, bước tiếp theo là triển khai các dịch vụ bằng cách sử dụng nền tảng đám mây của Google (Google Cloud Platform - GCP) [4]. GCP cung cấp một loạt các công cụ và dịch vụ có thể được sử dụng để triển khai và quản lý các vi dịch vụ, bao gồm Kubernetes, một nền tảng điều phối vùng chứa có thể được sử dụng để quản lý và mở rộng quy mô các dịch vụ siêu nhỏ [5]. Ngoài ra, GCP cung cấp một loạt các dịch vụ khác, chẳng hạn như Cloud Pub/Sub và Cloud Function, có thể được sử dụng để xây dựng và triển khai các Vi dịch vụ.

Bước cuối cùng trong việc xây dựng hệ thống bán sách trực tuyến với Vi dịch vụ là tích hợp các dịch vụ khác nhau thành một hệ thống gắn kết. Điều này có thể được thực hiện bằng cách sử dụng Cổng API, hoạt động như một điểm vào duy nhất cho tất cả các dịch vụ siêu nhỏ và cung cấp các tính năng như xác thực và thực thi chính sách bảo mật [6]. Ngoài ra, cổng API có thể được sử dụng để quản lý các phiên bản khác nhau của API và cung cấp giao diện nhất quán để khách hàng tương tác với hệ thống. Bằng cách làm theo các bước này, hệ thống bán sách trực tuyến có thể được xây dựng bằng Vi dịch vụ và triển khai trên GCP. Sơ đồ triển khai toàn bộ hệ thống được mô tả như hình 3.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 3: Sơ đồ triển khai toàn bộ hệ thống

Trong hình 3, phần Mobile client là ứng dụng di động phục vụ cho các khách hàng, phần Admin web client là ứng dụng web phục vụ cho việc quản lý ứng dụng di động, báo cáo, quản trị cho hệ thống trực tuyến. Hai phần này được liên kết với tầng "Business Logic" thông qua các hàm restfull API. Tầng "Business Logic" cũng kết nối với hệ thống cơ sở dữ liệu để phục vụ cho việc lưu trữ, truy vấn.

## 4 KẾT QUẢ ĐẠT ĐƯỢC

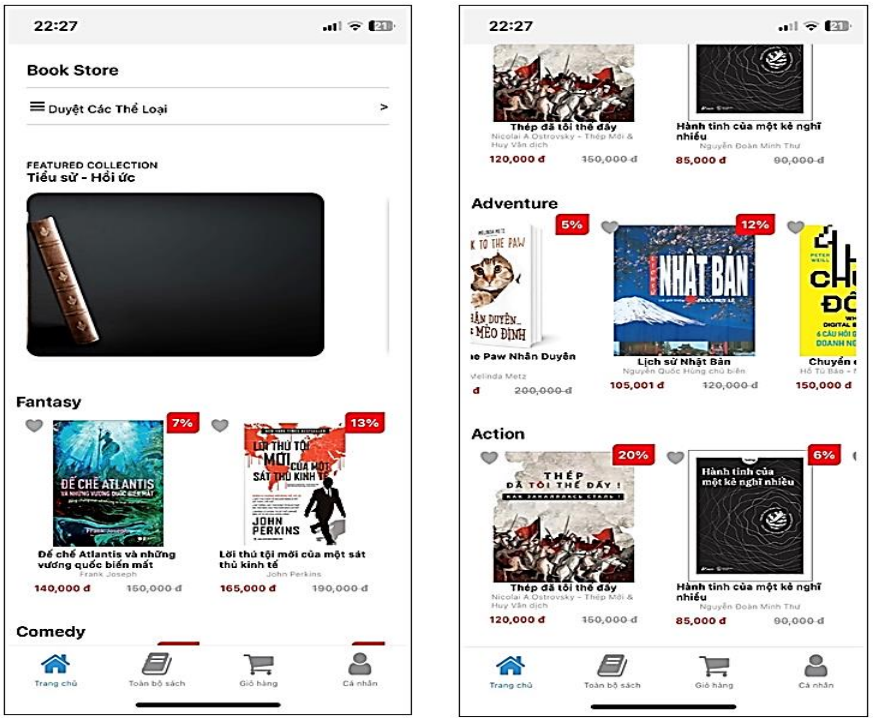
Qua quá trình nghiên cứu và thực hiện, chúng tôi đã đạt được những kết quả quan trọng sau đây:

- Thiết kế hệ thống bán sách dựa trên kiến trúc Vi dịch vụ: Chúng tôi đã thiết kế và triển khai một hệ thống bán sách linh hoạt và mở rộng, dựa trên kiến trúc Vi dịch vụ. Việc phân tách các thành phần chức năng thành các Vi dịch vụ độc lập đã giúp chúng tôi tăng tính mở rộng và quản lý của hệ thống.
- Triển khai trên nền tảng đám mây của Google với công nghệ gom cụm: Chúng tôi đã triển khai hệ thống trên nền tảng đám mây của Google, sử dụng công nghệ gom cụm để cân bằng tải và tăng khả năng chịu tải của hệ thống. Sự sử dụng của containerization và orchestration đã giúp chúng tôi quản lý và triển khai các Vi dịch vụ một cách tốt hơn.
- Các chức năng và tính năng hệ thống hoàn chỉnh: Hệ thống bán sách của chúng tôi cung cấp các chức năng quan trọng như tìm kiếm, xem chi tiết và đánh giá, quản lý giỏ hàng và đặt hàng. Người dùng có thể tìm kiếm, lọc và xem thông tin sách một cách thuận tiện. Họ có thể thêm sách vào giỏ hàng, chỉnh sửa và thanh toán trực tuyến. Đồng thời, hệ thống cũng cung cấp khả năng quản lý đơn hàng và theo dõi trạng thái của chúng.
- Hiệu suất và khả năng mở rộng: Thiết kế hệ thống dựa trên kiến trúc Vi dịch vụ và triển khai trên nền tảng đám mây của Google đã giúp chúng tôi đạt được hiệu suất tốt và khả năng mở rộng cao. Hệ thống có thể xử lý lượng lớn người dùng và giao dịch đồng thời, đảm bảo trải nghiệm tốt cho người dùng.

Nói chung, thông qua quá trình nghiên cứu và thực hiện nghiên cứu, chúng tôi đã đạt được những kết quả quan trọng và đáng chú ý. Bằng việc kết hợp kiến trúc Vi dịch vụ, triển khai trên nền tảng nền tảng đám mây của Google và sử dụng công nghệ React Native, chúng tôi đã xây dựng thành công một hệ thống ứng dụng trực tuyến mở rộng.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Kiến trúc Vi dịch vụ đã cho phép chúng tôi phân tách chức năng thành các thành phần độc lập nhau, tăng tính mở rộng của hệ thống. Việc sử dụng nền tảng đám mây của Google và công nghệ gom cụm đã giúp chúng tôi triển khai và quản lý hệ thống trên môi trường đám mây đươc thuận lợi hơn, đồng thời tận dụng các lợi ích của việc tự động hóa và mở rộng khả năng chịu tải.
Bên cạnh đó, việc sử dụng công nghệ React Native đã giúp chúng tôi xây dựng một giao diện di động tương thích trên nhiều thiết bị và hệ điều hành, mang lại trải nghiệm tốt và nhất quán cho người dùng như hình 4.



Hình 4: Giao diện người dùng

Kết quả đạt được từ dự án không chỉ giới hạn ở việc xây dựng một ứng dụng bán sách trực tuyến, mà còn đóng góp vào sự phát triển và ứng dụng của các công nghệ tiên tiến như Vi dịch vụ, đám mây và di động. Nghiên cứu này đã mang lại những kiến thức, kinh nghiệm và giải pháp thực tế cho việc xây dựng các ứng dụng thương mại điện tử linh hoạt, và mở rộng.
Trong tương lai, kết quả đạt được từ dự án này có thể được mở rộng và áp dụng cho các lĩnh vực khác như bán hàng trực tuyến, dịch vụ khách hàng và quản lý doanh nghiệp. Chúng tôi tin rằng kinh nghiệm và kiến thức thu thập được từ việc xây dựng hệ thống bán sách sẽ tiếp tục góp phần trong việc phát triển các ứng dụng đáng tin cậy trên nền tảng công nghệ tiên tiến trong tương lai.

## 5 HƯỚNG PHÁT TRIỂN

Kết quả của bài báo đã đạt được những kết quả đáng kể và mang lại lợi ích cho người dùng và nhà phát triển. Tuy nhiên, để nâng cao sự hoàn thiện và khả năng thích ứng với yêu cầu thị trường ngày càng khắt khe, chúng tôi xin trình bày một số hướng phát triển tiềm năng cho dự án trong tương lai:

- Mở rộng tính năng: Để tăng cường trải nghiệm người dùng, chúng tôi có thể mở rộng tính năng của hệ thống bằng cách thêm các chức năng như giới thiệu sản phẩm, gợi ý sách dựa trên sở thích của người dùng, tích hợp các phương thức thanh toán mới và cải tiến giao diện người dùng để tạo ra trải nghiệm tốt hơn.
- Tối ưu hóa hiệu suất: Để đáp ứng nhu cầu của người dùng trong môi trường tải cao, chúng tôi có thể tối ưu hóa hiệu suất hệ thống bằng cách áp dụng các kỹ thuật caching, tăng cường xử lý đa luồng, tối ưu hóa cơ sở dữ liệu và tăng cường khả năng mở rộng ngang để đảm bảo hệ thống hoạt động một cách ổn định và nhanh chóng.

- Mở rộng nền tảng đám mây: Chúng tôi có thể khai thác và mở rộng việc triển khai hệ thống trên nền tảng đám mây bằng cách tích hợp với các dịch vụ khác trong Google Cloud như BigQuery để phân tích dữ liệu, Cloud Functions để xử lý sự kiện tự động, hay AI Platform để áp dụng trí tuệ nhân tạo trong việc gợi ý sách và phân loại nội dung.
- Mở rộng quy mô hệ thống: Nếu nhu cầu mở rộng cho các ứng dụng trực tuyến ngày càng tăng, chúng tôi có thể xem xét mở rộng quy mô hệ thống bằng cách tăng số lượng các Vi dịch vụ, triển khai các cụm đám mây phân tán và sử dụng các công nghệ đám mây tiên tiến khác như Kubernetes để quản lý và mở rộng quy mô hệ thống một cách tốt hơn.
- Tăng cường bảo mật: Bảo mật là một yếu tố quan trọng trong hệ thống bán sách trực tuyến. Chúng tôi có thể tiếp tục nâng cao bảo mật bằng cách áp dụng các biện pháp bảo mật mạnh hơn như mã hóa dữ liệu, xác thực hai yếu tố, kiểm tra và giám sát liên tục để đảm bảo an toàn và bảo mật thông tin khách hàng và giao dịch.
- Nghiên cứu và áp dụng công nghệ mới: Để tiếp tục nắm bắt xu hướng công nghệ mới và nâng cao đáng kể trải nghiệm người dùng, chúng tôi sẽ tiếp tục nghiên cứu và áp dụng các công nghệ mới như trí tuệ nhân tạo, học máy, thực tế ảo và thực tế tăng cường để mang lại trải nghiệm mua sắm độc đáo và tương tác tốt hơn với sách và tác giả.

## TÀI LIỆU THAM KHẢO

[1] N. T. Linh, "Microservices thực tiễn: Từ thiết kế đến triển khai.," techmaster, [Trực tuyến]. Available: https://techmaster.vn/posts/34410/microservices-thuc-tien-tu-thiet-ke-den-trien-khai. [Đã truy cập 20-05-2023].

[2] Aptech, "Microservice là gì? Những hiểu biết cơ bản về Microservices.," Aptech, [Trực tuyến]. Available: https://aptech.fpt.edu.vn/microservice.html. [Đã truy cập 20-05-2023].

[3] Đ. T. Thành, "Khóa học microservice với Spring Webflux & Apache Kafka.," udemy, [Trực tuyến]. Available: https://www.udemy.com/course/khoa-hoc-microservice-voi-spring-webflux-apache-kafka/. [Đã truy cập 20-05-2023].

[4] CLOUD4C, "Hiện đại hóa trung tâm dữ liệu trên GCP - Cloud4C.," CLOUD4C, [Trực tuyến]. Available: https://www.cloud4c.com/vn/vi/google-cloud-services/gcp-dc-modernization. [Đã truy cập 21-05-2023]

[5] S. T. Team, "Triển khai ứng dụng theo kiến trúc microservices thật đơn giản.," VIBLO, [Trực tuyến]. Available: https://viblo.asia/p/trien-khai-ung-dung-theo-kien-truc-microservices-that-don-gian-EoW4oxzzJml. [Đã truy cập 21-05-2023]

[6] C. Ace, "Mercari - Triển khai Microservices trên Google Cloud Platform.," Cloud Ace, [Trực tuyến]. Available: https://cloud-ace.vn/tin-tuc/customer/mercari-trien-khai-kien-truc-microservices-tren-nen-tang-google-cloud/. [Đã truy cập 20-05-2023]

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F009**

# LINK: ỨNG DỤNG CHAT NỘI BỘ VÀ ĐỒNG BỘ HÓA TIẾN ĐỘ CÔNG VIỆC TRÊN ANDROID

TRẦN THỊ ANH THI[1], HOÀNG HUY VŨ[1], TRẦN THÀNH NAM[1]

[1]*Faculty of Information Technology, Industrial University of Ho Chi Minh City*
*tranthianhthi@iuh.edu.vn, vhong10092001@gmail.com, nam01697826435nam@gmail.com*

**Tóm tắt.** Hệ thống hỗ trợ giao việc trong các tổ chức hiện nay được triển khai ở nhiều mức độ, phạm vi khác nhau. Những hệ thống này ngoài các chức năng cơ bản vẫn còn một số hạn chế như cơ chế giao tiếp theo quy trình được định nghĩa bởi tổ chức doanh nghiệp. Ngoài ra, khả năng tích hợp với các hệ thống quản lý nhân sự vẫn còn hạn chế. Công cụ LINK chúng tôi đề xuất sẽ hỗ trợ những vấn đề này. Công cụ cho phép quản lý quy trình làm việc, giám sát quá trình thực thi của người được phân công trong hệ thống. Ngoài ra, công cụ cũng tích hợp được vào hệ thống quản lý nhân sự cho các tổ chức vừa và nhỏ. LINK được triển khai trên các nền tảng công nghệ như NodeJS và Java chỉ dành cho phiên bản di động. LINK cũng sử dụng các dịch vụ như AWS DynamoDB, Firebase OTP để triển khai và xác thực người dùng. Hiện công cụ LINK được triển khai thử nghiệm cho các nhóm làm việc vừa và nhỏ tại khoa công nghệ thông tin trường Đại học Công nghiệp Thành phố Hồ Chí Minh. Trong tương lai, chúng tôi sẽ triển khai mở rộng đối tượng người dùng cũng như các chức năng tiện ích khác.

**Từ khóa.** Ứng dụng di động, Giao tiếp nội bộ, Điện toán đám mây, Quản lý công việc.

## LINK: INTERNAL CHAT AND TASK SYNCHRONIZATION APPLICATION ON ANDROID

**Abstract.** The task management support system in organizations today is implemented at various levels and scopes. These systems, besides their basic functions, still have some limitations, such as communication mechanisms according to the processes defined by the enterprise. In addition, the ability to integrate with human resource management systems is still limited. Our proposed tool, LINK, will address these issues. The tool allows for the management of work processes and the monitoring of the execution process of the assigned personnel in the system. Additionally, the tool can be integrated into human resource management systems for both small and medium-sized organizations. LINK is implemented on technology platforms such as NodeJS and Java specifically for mobile versions. LINK also uses services such as AWS DynamoDB and Firebase OTP for user authentication and implementation. Currently, the LINK tool is being tested for small and medium-sized workgroups in the IT department at the Ho Chi Minh City University of Industry. In the future, we will expand the user base and add other utility functions.

**Keywords.** Mobile Application, Internal Communication, Cloud Computing, Task Management.

## 1 GIỚI THIỆU

Sau khi dịch bệnh Covid bùng phát, việc tiếp tục làm việc và duy trì giao tiếp trong tổ chức trở nên thách thức. Đó là lý do tại sao LINK ra đời một ứng dụng di động đáng tin cậy và tiện ích. LINK giúp các nhóm làm việc kết nối và giao tiếp dễ dàng thông qua tính năng chat nội bộ và đồng bộ tiến độ công việc trên nền tảng Android. Bằng cách sử dụng LINK, việc trao đổi thông tin, chia sẻ tệp tin và phối hợp công việc trở nên hiệu quả hơn.

Với sự bảo mật nội bộ mạnh mẽ, LINK đảm bảo rằng thông tin và dữ liệu của bạn được bảo vệ an toàn. Mọi dữ liệu truyền đi và lưu trữ trên LINK đều được mã hóa, ngăn chặn truy cập trái phép. Bạn cũng có

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

quyền kiểm soát quyền truy cập và phân quyền cho các thành viên trong nhóm, đảm bảo chỉ những người được phép mới có thể truy cập vào thông tin mật của công ty.
Điều này giúp duy trì kết nối và tiến triển công việc một cách suôn sẻ trong thời gian khó khăn do dịch bệnh. LINK là công cụ lý tưởng để tăng cường sự phối hợp trong công việc và đảm bảo thành công trong môi trường thay đổi do dịch bệnh.

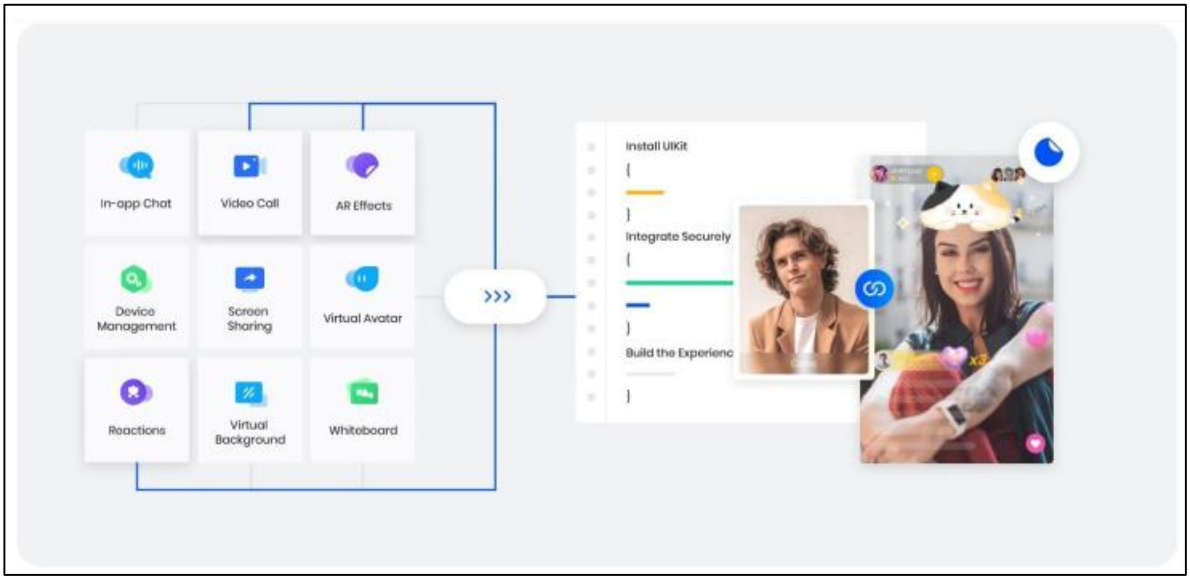
# 2 CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 Cloud Computing với AWS

a) Amazon Simple Storage Service (S3)
AWS S3 là một dịch vụ lưu trữ đối tượng do AWS cung cấp. Nó được ưa chuộng bởi nhiều người dùng AWS nhờ tính sẵn có, bảo mật cao và khả năng kết nối dễ dàng với các dịch vụ khác của AWS. AWS S3 phù hợp cho mọi trường hợp sử dụng, bao gồm ứng dụng di động/web, lưu trữ dữ liệu lớn, máy học và nhiều hơn nữa [1].
b) Amazon DynamoDB
AWS DynamoDB là một dịch vụ cơ sở dữ liệu quản lý hoàn toàn dựa trên mô hình NoSQL (Non-relational Database). Được xây dựng trên cơ sở hạ tầng phân tán của AWS, DynamoDB cung cấp tính linh hoạt, mở rộng và đáng tin cậy cho việc lưu trữ và truy vấn dữ liệu. [2]

## 2.2 Zegoclould

Zegocloud là một dịch vụ đám mây đơn giản và tiện lợi cho việc lưu trữ và truy cập dữ liệu từ xa thông qua internet. Người dùng có thể tải lên, lưu trữ và chia sẻ tệp tin, hình ảnh, video và các dữ liệu khác trên Zegocloud. Điều này cho phép họ truy cập dữ liệu từ bất kỳ thiết bị nào có kết nối internet, đồng thời đảm bảo tính an toàn và bảo mật của dữ liệu. Ngoài ra, Zegocloud cũng cung cấp các tính năng bổ sung như sao lưu tự động và chia sẻ dữ liệu với người dùng khác. Tóm lại, Zegocloud giúp người dùng lưu trữ và truy xuất dữ liệu một cách dễ dàng và an toàn từ bất kỳ đâu thông qua internet [3].



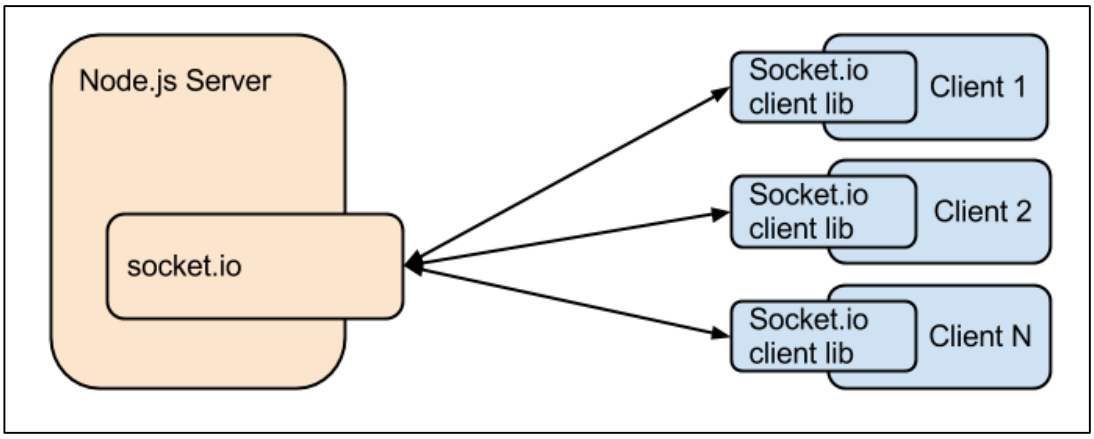

Hình 1: Các tác vụ của Zelocloud [3]

## 2.3 Socket io

Socket.io được phát triển để sử dụng các kết nối mở tạo điều kiện giao tiếp thời gian thực, trả về giá trị thực ở tại thời điểm đó. Socket.io cho phép giao tiếp hai chiều giữa máy khách và máy chủ. Giao tiếp hai chiều được bật khi có máy khách có Socket.io trong trình duyệt và máy chủ cũng đã tích hợp gói Socket.io
Socket.io không phải là ngôn ngữ, vì vậy nó phải được sử dụng kết hợp với những ngôn ngữ khác như: php, .net, nodejs …

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Socket.io thường được sử dụng trong các ứng dụng: Ứng dụng trò chơi đa người chơi, ứng dụng trò chuyện thời gian thực, ứng dụng cập nhật thời gian thực, ứng dụng đa phương tiện, ứng dụng theo dõi.
Một số ưu điểm của: Bảo mật cao, kết nối tự động server, mã hóa nhị phân, cho phép tạo kênh và phòng … [4]

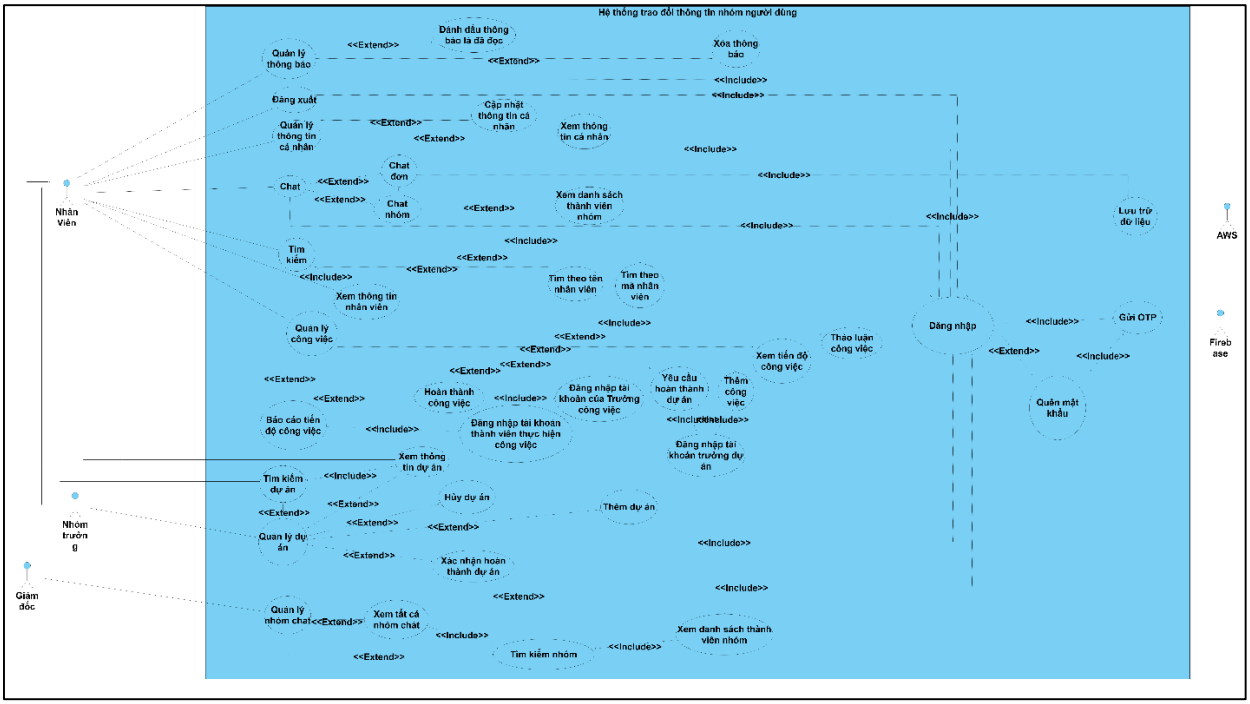



Hình 2: Mô hình sử dụng kết nối Client-Server của SocketIO [4]

# 3 PHÂN TÍCH HỆ THỐNG

## 3.1 Usecase tổng quát

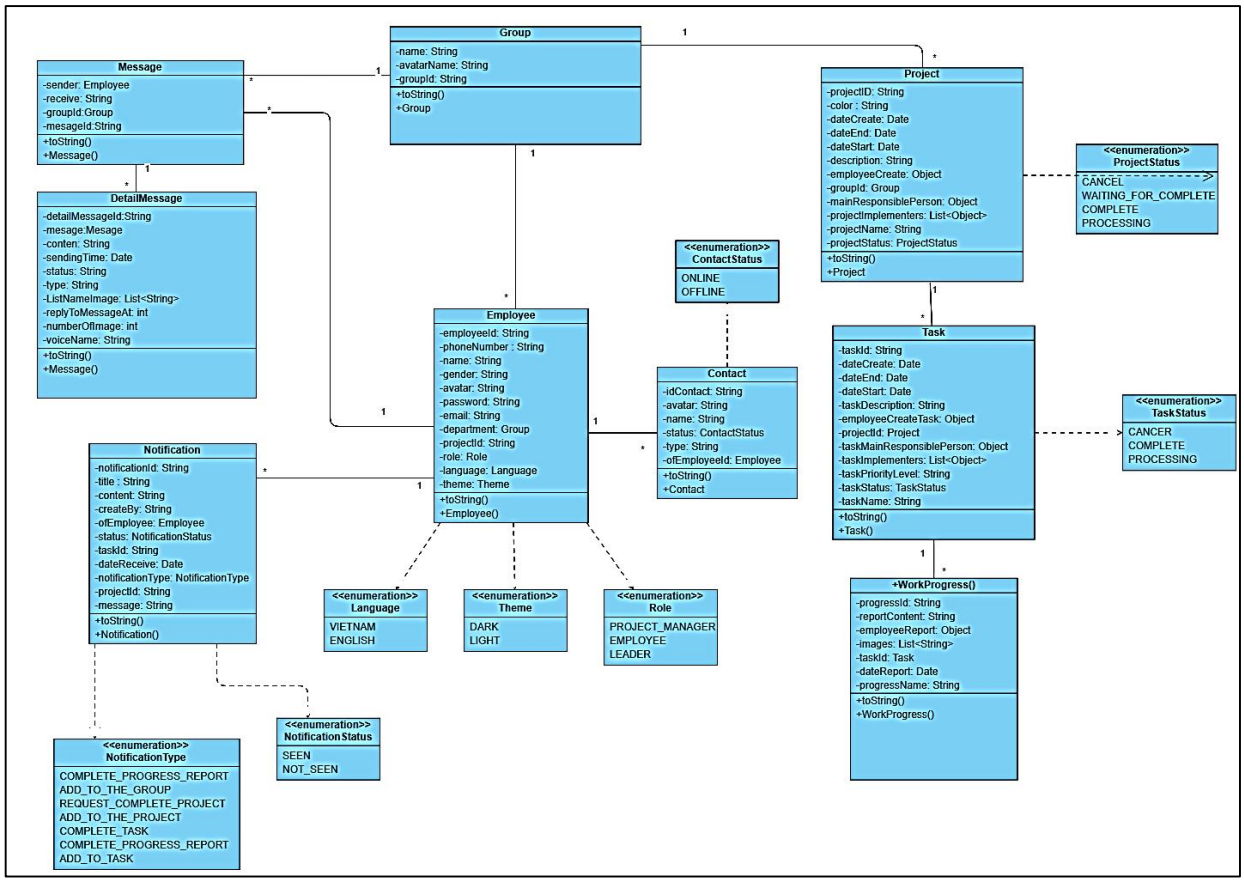
Đối tượng người dùng cho hệ thống LINK gồm có người nhóm trưởng, nhân viên và giám đốc. Trong đó, chúng tôi tập trung vào các chức năng của hệ thống là hỗ trợ giao tiếp như nhắn tin, gửi hình, gọi điện, quản lý công việc trong công ty gồm có: tạo dự án, công việc, phân chia công việc, báo cáo, thống kê, thông báo. Những chức năng này được chúng tôi trình bày ở *Hình 3*.




Hình 3: Sơ đồ Use Case tổng quát của hệ thống LINK

### 3.2 Danh sách Sơ đồ lớp

Từ các usecase đã phân tích ở trên thì chúng tôi tạo ra được lược đồ sơ đồ lớp của hệ thống LINK. Trong đó các lớp được chú trọng nhất là: Project, Task, WorkProgress, Notification. Trong đó chúng tôi mở rộng lớp Employee để phân quyền người dùng cho hệ thống. Để biết thêm chi tiết chúng tôi trình bày sơ đồ lớp được thiết kế ở *Hình 4.*




Hình 4: Sơ đồ Class Diagram

## 4 KẾT QUẢ HIỆN THỰC

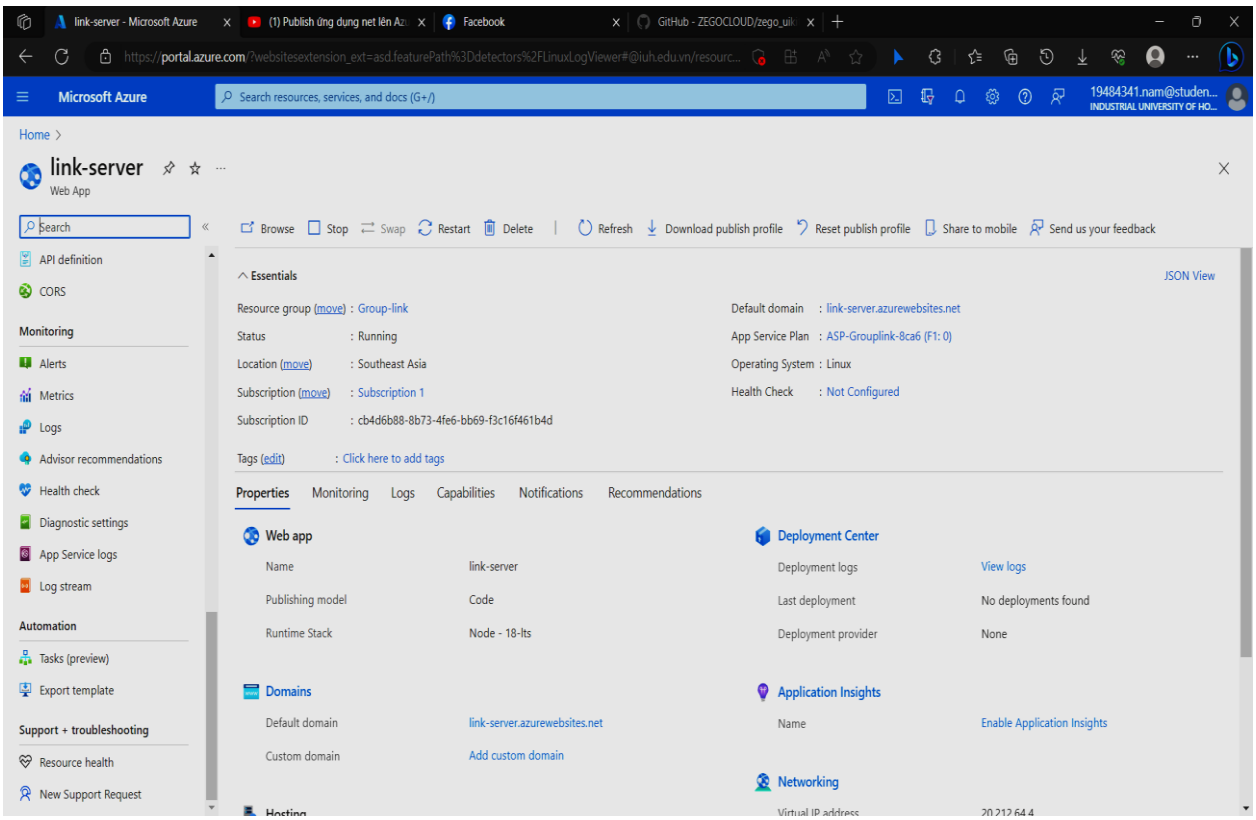
### 4.1 Cấu hình phần cứng, phần mềm

a) Cấu hình server

Dịch vụ Microsoft Azure là một nền tảng điện toán đám mây cho phép xây dựng, triển khai và quản lý các ứng dụng web và cơ sở dữ liệu trên Internet. Để triển khai một ứng dụng web trên Azure, chúng tôi đã sử dụng tính năng Ứng dụng web của Dịch vụ ứng dụng Azure như *Hình 5*.

Triển khai server Nodejs với domain: link-server.azurewebsites.net. Máy chủ được đặt tại Southeast Asia và chạy trên môi trường Linux.

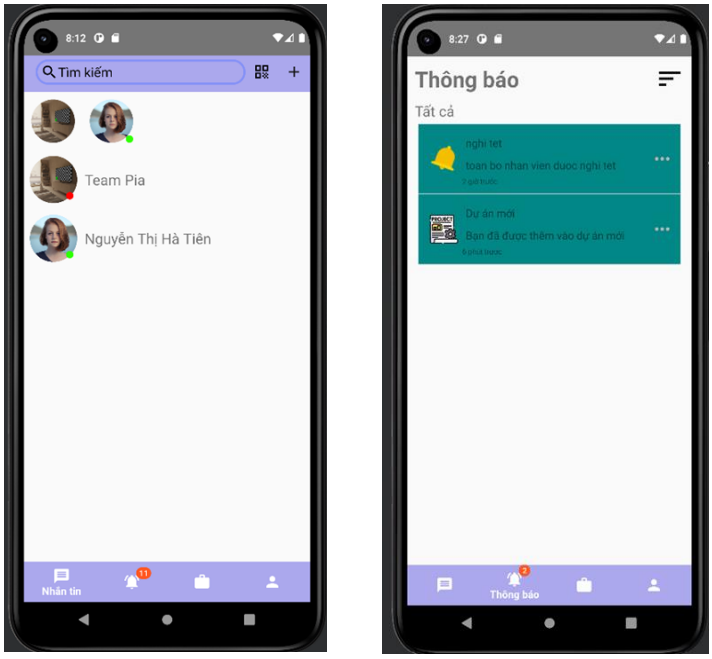
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

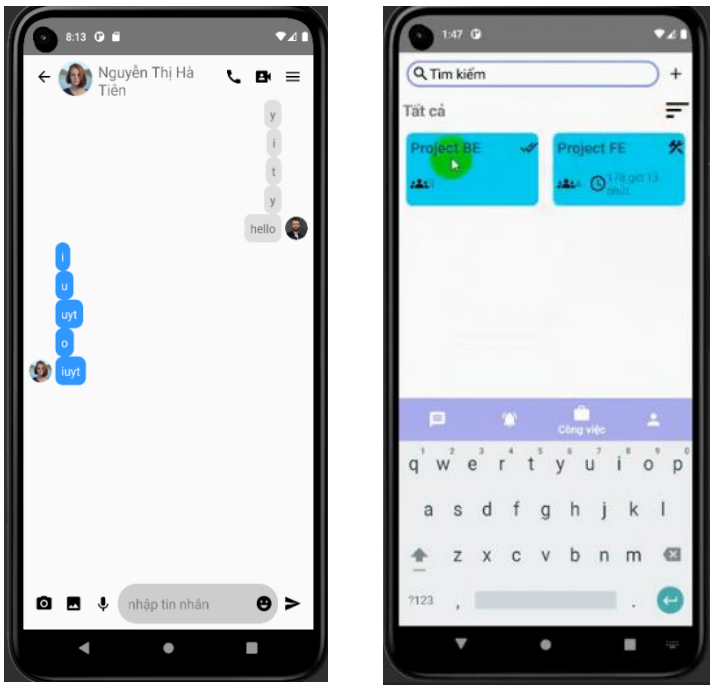



Hình 5: Trạng thái server đang hoạt động

## 4.2 Giao diện hiện thực màn hình

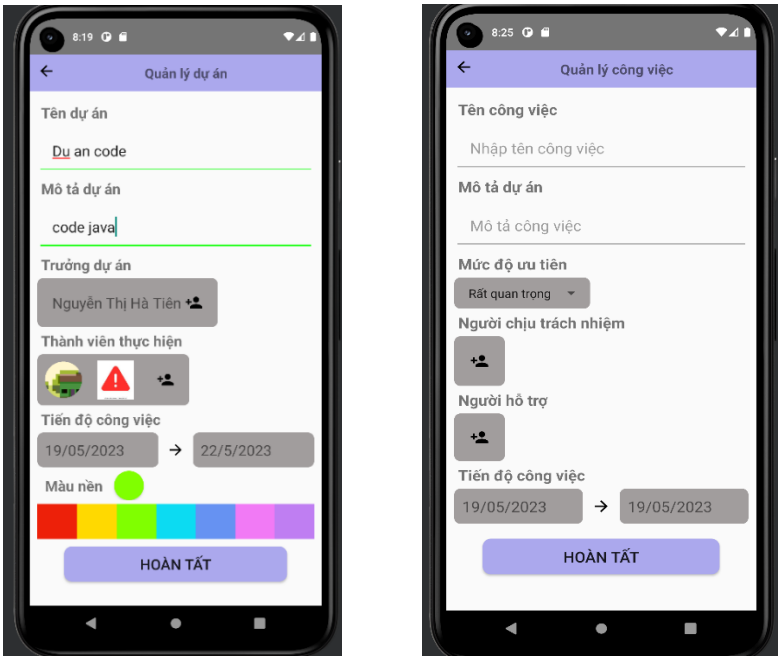
Để thuận tiện cho việc quản lý, nhận thông báo và nhắn tin, chúng tôi đã thiết kế màn hình chức năng hiển thị chung để dễ dàng quan sát và thao tác khi nhận được thông báo (*Hình 6*).





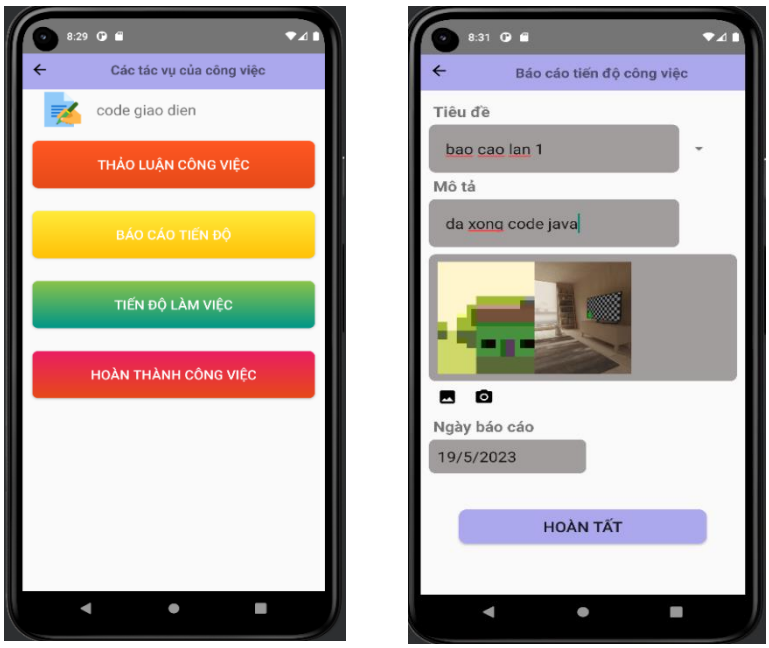

Hình 6: Màn hình các tác vụ chính

Tiếp đến chúng tôi thiết kế các quy trình quản lý công việc là tạo dự án dành cho nhóm trưởng, tạo công việc dành cho trưởng dự án, tham gia công việc báo cáo và hoàn thành tiến độ bởi người chịu trách nhiệm chính của công việc đó (*Hình 7*).





© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 7: Màn hình nhận công việc và thực hiện báo cáo

## 5 KẾT LUẬN

Khi viết bài báo này chúng tôi đưa ra giải pháp hỗ trợ giao tiếp, quản lý công việc nội bộ trong công ty một cách tối ưu dễ xử lý, tránh việc mất nhiều thời gian và chi phí để thực hiện. Cụ thể giải pháp của chúng tôi được thực hiện trên nền tảng Android là một thiết bị công nghệ mà đại đa số người dùng sử dụng và phù hợp với nhiều người. Phần mềm được phát triển bởi Java Android, một ngôn ngữ khá phổ biến dành cho các thiết bị di động Android có một lượng lớn cộng đồng sử dụng. Việc quản lý dữ liệu trong ứng dụng Android sử dụng Java có thể được thực hiện bằng cách sử dụng Java SDK của AWS để truy vấn dữ liệu từ cơ sở dữ liệu DynamoDB. Chúng ta có thể sử dụng các phương thức như Scan hoặc Query để truy xuất dữ liệu theo yêu cầu. Kết quả sẽ được trả về dưới dạng các đối tượng Java để xử lý trong ứng dụng Android. Tuy nhiên do sự hạn chế về thời gian, cũng như sự khảo sát chưa thật sự tốt và đầy đủ nên phần mềm còn gặp nhiều hạn chế chưa thực sự bám sát vào quy trình của một công ty thực tế. Cụ thể là một số quy trình nghiệp vụ còn nhiều khuyết điểm chưa thực sự mượt mà với nhiều trường hợp khác nhau. Đó là những mặt hạn chế của phần mềm LINK của chúng tôi, trong tương lai chúng tôi sẽ dành nhiều thời gian tìm hiểu phát triển LINK thành một phần mềm được áp dụng hỗ trợ trực tiếp cho công ty.

## TÀI LIỆU THAM KHẢO

[1] What is Amazon S3 ? (2023) [online],

https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html [Accessed 08/01/2023]

[2] What is Amazon DynamoDB ? (2023) [online],

https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html[Accessed 09/02/2023]

[3] Build Powerful Interaction Apps with Voice & Video Chat API (2023) [online],

http://surl.li/hlsve[Accessed13/02/2023]

[4] Bước đầu làm quen với NodeJS và Socket.io (Aug-2016) [online],

https://viblo.asia/p/buoc-dau-lam-quen-voi-nodejs-va-socketio-MJyGjQrWvPB[Accessed 18/02/2023]

**YSC5.F010**

# PHÁT TRIỂN GAME "KHU VƯỜN MINI" TRÊN NỀN TẢNG ANDROID KẾT HỢP CÔNG NGHỆ UNITY-2D

TRẦN THỊ ANH THI[1], PHAN HỒNG HÀ[1]

[1]*Faculty of Information Technology, Industrial University of Ho Chi Minh City*
*tranthianhthi@iuh.edu.vn, phanhongha96@gmail.com*

**Tóm tắt.** Với sự phát triển nhanh chóng của công nghệ, hàng loạt các ứng dụng được tạo ra nhằm phục vụ cho công việc, học tập, giao tiếp, giải trí. Nhưng trong đó các phần mềm phục vụ nhu cầu giải trí luôn xuất hiện với giao diện vô cùng bắt mắt. Các trò chơi giải trí cũng được phát triển nhiều hơn. Cùng với sự phát triển đó, thiết bị di động đang là thiết bị không thể thiếu trong đời sống chúng ta. Vì vậy, các trò chơi trên các thiết bị di động như: điện thoại thông minh, máy tính bảng… đang là hướng phát triển mạnh trong ngành công nghệ game. Nắm bắt được xu hướng đó, trò chơi "Khu vườn Mini" được phát triển nhằm đáp ứng nhu cầu giải trí của người dùng. Trò chơi này được phát triển trên nền tảng Android kết hợp các công nghệ và công cụ hỗ trợ tối ưu nhất cho việc xây dựng trò chơi như: Unity, Photoshop, Figma, Unity-2d, … Trò chơi "Khu vườn Mini" được định hướng là trò chơi có tính giải trí cao nhằm giúp người dùng có được trải nghiệm làm vườn thú vị, kích thích cảm xúc tích cực, giảm căng thẳng, nhưng không mất quá nhiều thời gian để chơi game. Trong tương lai trò chơi có thể phát triển và sử dụng được trên những nền tảng khác nhau, như: IOS, Window, …

**Từ khóa.** Trò chơi, Ứng dụng di động, Unity-2d.

## DEVELOPING THE GAME "MINI GARDEN" ON ANDROID FOUNDATION IN CONJUNCTION WITH UNITY-2D TECHNOLOGY

**Abstract.** As technological progress increases, more and more applications are developed to facilitate various human activities, including labor, education, communication, and recreation. But in it, the entertainment application always has a stunning graphical interface to it. Entertainment games are also developed more. Along with that development, mobile devices are becoming indispensable in our lives. Therefore, games on mobile devices, such as smartphones, tablets, ... are a strong development direction in the game technology industry. Grasping that trend, the game "Mini Garden" was developed to meet the entertainment needs of users. The "Mini Garden" game was made in response to this need for a relaxing pastime. The Android mobile version of the game was created using top-tier game development tools and technologies like Unity, Photoshop, Figma, and Unity-2d, … The "Mini Garden" game is fun and will show a user a lot about gardening while also making the user feel positive and relieving some tension, but do not take too much time to play the game. The game has potential for future development and uses across various platforms, such as IOS, Windows, …

**Keywords.** Game, Mobile Application, Unity-2d.

## 1 GIỚI THIỆU

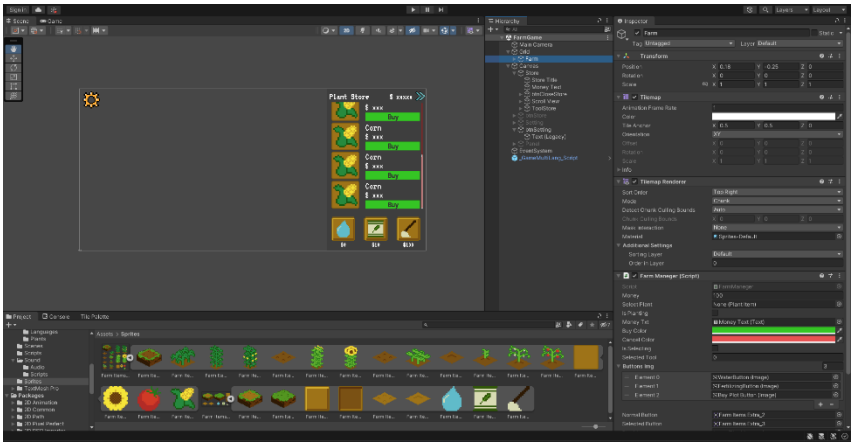
Cuộc sống ngày nay căng thẳng, nên người ta cần sự giải trí. Trò chơi di động đang trở thành xu hướng giải trí phổ biến. Tuy nhiên, trò chơi trực tuyến tốn nhiều thời gian và ảnh hưởng đến tâm lý giới trẻ. Các trò chơi giải trí đa phần phát hành trên desktop, nhưng xu hướng chuyển sang điện thoại và máy tính bảng.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Mọi người cần trò chơi không bạo lực, tiết kiệm thời gian, và có thể chơi bất cứ lúc nào, nơi đâu. Trò chơi làm vườn di động là lựa chọn giải trí tốt nhất.
Hiện nay các công nghệ lập trình game đang dần đa dạng và phổ biến hơn với các nhà phát triển, như công nghệ thực tế ảo (Virtual Reality - VR), Công nghệ đa luồng (Multithreading), Công nghệ Machine Learning và AI. Nhưng với các yêu cầu đã đặt ra thì công nghệ Unity – 2D là thích hợp nhất. Với các thư viện đã được hỗ trợ từ Unity cho phép nhà phát triển dễ dàng thiết kế và cài đặt trò chơi trên đa nền tảng.
Trò chơi "Khu vườn Mini" phát triển để giải trí mọi người. Được phát triển trên nền tảng Android và sử dụng công nghệ như Unity-2D. Dành cho nhân viên văn phòng, sinh viên, học sinh,... để có trải nghiệm làm vườn thú vị, giúp giảm căng thẳng và kích thích tích cực mà không tốn nhiều thời gian.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 Unity là gì?



Hình 1: Công cụ lập trình game Unity

Unity là một game engine đa nền tảng được phát triển bởi Unity Technologies, được sử dụng chủ yếu để phát triển video game trên 21 nền tảng khác nhau, bao gồm máy tính, các hệ máy chơi game (như PlayStation) và điện thoại di động. Hơn 50% số lượng game trên thị trường được tạo ra bằng Unity. Có nhiều tựa game nổi tiếng như Pokémon Go, Hearthstone, Ori And The Blind Forest, Monument Valley, Axie Infinity được tạo ra bằng Unity. Unity có khả năng áp dụng rộng rãi trong các thể loại game khác nhau, từ game AAA đỉnh cao đến game giáo dục cho trẻ em [1].

### 2.2 Các thành phần trong Unity

**Assets:** trong Unity là tài nguyên (hình ảnh, mô hình 3D, âm thanh, hiệu ứng...) sử dụng trong việc phát triển game. Chúng có thể được tải miễn phí hoặc trả phí từ Unity Asset Store, giúp tiết kiệm thời gian thiết kế và lập trình game [2].
**GameObject:** trong Unity là đối tượng đại diện cho các Assets trong game như nhân vật, cây cỏ, công cụ, đạo cụ, camera, hiệu ứng và nhiều thứ khác. Mỗi GameObject đều có thuộc tính cơ bản là Transform, dùng để xác định vị trí, hướng và kích thước của đối tượng [2].
**Components:** là các thuộc tính được gắn vào GameObject để xác định hình thái, hành vi và hiệu ứng của đối tượng trong game [2].
**Scene:** trong Unity là một cảnh game hoặc không gian game, nơi sắp xếp và định vị các GameObject, hoặc chứa các thiết lập giao diện như menu trong game. Việc tạo nhiều Scenes giúp quản lý tài nguyên và phân đoạn game một cách linh hoạt và độc lập [2].
**Scripts:** là một Component quan trọng trong Unity, được sử dụng để tương tác với hành động của người chơi và quản lý các sự kiện để thay đổi hướng của game theo kịch bản. Lập trình viên có thể viết Scripts bằng ngôn ngữ C# được cung cấp bởi Unity [2].

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**Prefabs:** trong Unity là các GameObject đã được hoàn chỉnh và lưu trữ để tái sử dụng. Việc nhân bản prefab tạo ra các GameObject giống nhau. Tạo prefab chỉ cần kéo GameObject từ Hierarchy vào Project [2].
**Camera:** trong Unity thể hiện khung hình và góc nhìn trong game. Có thể sử dụng nhiều camera để chia nhỏ màn hình, tạo hiệu ứng và tùy chỉnh như xem từ phía sau, bản đồ thu nhỏ, ... [2].

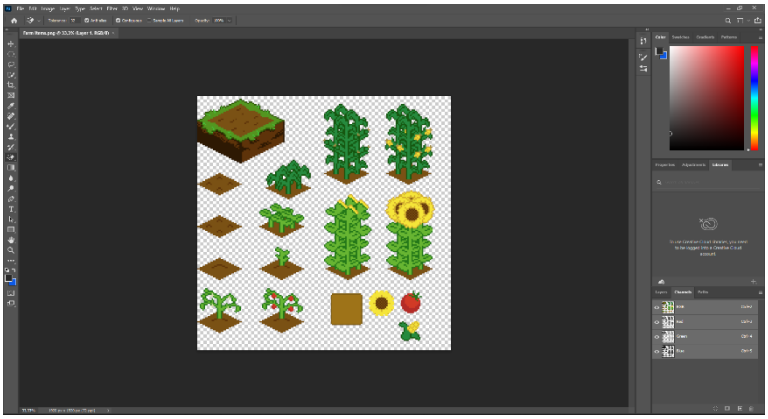
## 2.3 Spite Editor



Hình 2: Sprite Editor

Sprite là đối tượng đồ họa 2D trong trò chơi, bao gồm nhân vật, đạo cụ và các thành phần khác. Chúng được tạo từ hình ảnh bitmap (Texture2D). Có hai loại Sprite trong Unity 2D: single Sprite và multiple Sprite. Single Sprite được sử dụng khi muốn sử dụng toàn bộ nội dung của một hình ảnh cho một đối tượng. Multiple Sprite được sử dụng khi một file ảnh chứa nhiều hình ảnh, mỗi hình ảnh được sử dụng cho một đối tượng riêng biệt. Multiple Sprite giúp tiết kiệm dung lượng và dễ quản lý [2, 3].

## 2.4 Photoshop



Hình 3: Công cụ Photoshop

Trong một ứng dụng game cần rất nhiều yếu tố để cấu thành lên như: Scenes, Script, Audio, … Trong đó Scenes là đối tượng đưa ra hình ảnh, giao diện cho người chơi. Nhưng những hình ảnh đó không vừa vặn với các khung hình của game, từ đó các công cụ chỉnh sửa hình ảnh được áp dụng vào lập trình game, photoshop là một trong số đó.
Photoshop, còn được gọi là Adobe Photoshop, là phần mềm thiết kế và chỉnh sửa hình ảnh trên hệ điều hành Windows và MacOS. Với Photoshop, bạn có thể tạo và chỉnh sửa hình ảnh, thiết kế các tác phẩm nghệ thuật, và mô phỏng tranh ảnh. Nó là công cụ chỉnh sửa ảnh phổ biến nhất hiện nay, cung cấp các công cụ đơn giản và hỗ trợ nhiều định dạng ảnh khác nhau. Bạn có thể chỉnh sửa từng hình ảnh cụ thể hoặc hàng

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

loạt tùy theo nhu cầu sử dụng. Viết tắt của Photoshop thường là PTS, và khi người ta sử dụng cụm từ này, thường ám chỉ đến phần mềm Adobe Photoshop cho việc thiết kế và chỉnh sửa ảnh [4 ,5].

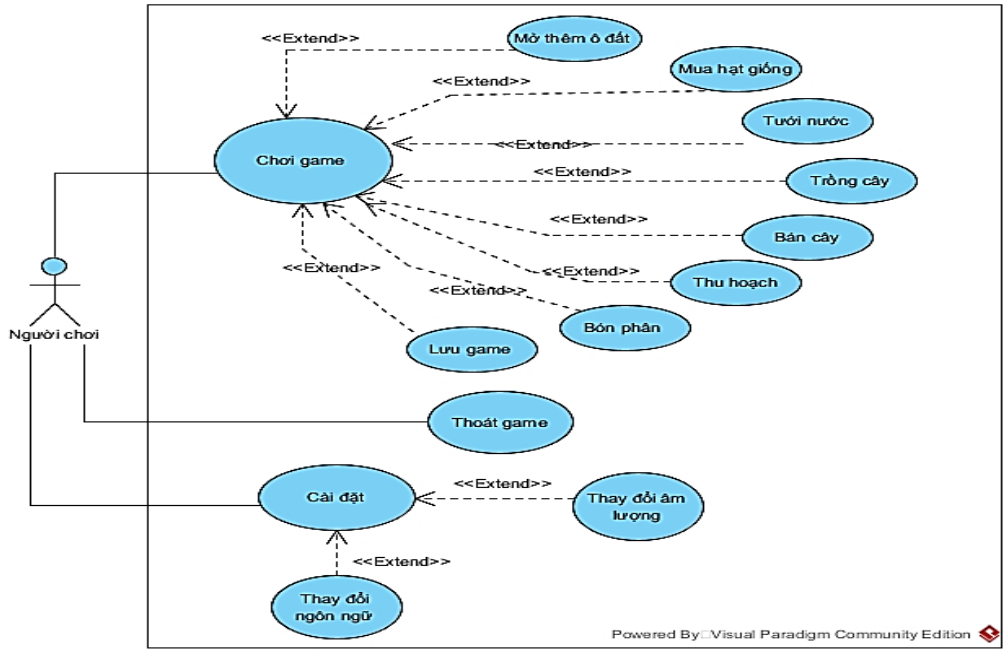
### 2.5 PlayerPrefs

PlayerPrefs là một lớp tĩnh (static class) được Unity hỗ trợ để lập trình viên có thể lưu trữ và truy xuất thông tin ở những lần chơi khác nhau trong game ở dạng key/value. PlayerPrefs cho phép bạn lưu trữ dữ liệu như điểm số, cấp độ hoặc cài đặt người chơi, và truy xuất chúng sau này trong các phiên chơi tiếp theo. Khi lưu trữ dữ liệu, bạn sử dụng một key duy nhất để xác định dữ liệu đó. Dữ liệu được lưu trữ bằng cách sử dụng một loại dữ liệu nguyên thủy như số nguyên, số thực hoặc chuỗi. Khi cần truy xuất dữ liệu, bạn chỉ cần sử dụng key tương ứng để nhận giá trị lưu trữ tương ứng. PlayerPrefs cung cấp một cách tiện lợi để lưu trữ dữ liệu game và tiết kiệm thời gian và công sức của lập trình viên trong việc quản lý dữ liệu trò chơi [6].

## 3 PHÂN TÍCH HỆ THỐNG

### 3.1 Sơ đồ tổng quát

Game "Khu Vườn Mini" được thiết kế với lối chơi đơn giản. Vì được thiết kế đơn giản nên trò chơi chỉ có một actor chính đó là người chơi. Các chức năng chính nhằm cho người chơi cảm nhận được việc làm vườn khi chơi game. Ngoài ra, còn có các chức năng phục vụ cho việc cấu hình game như: thay đổi ngôn ngữ, thay đổi âm lượng. Các chức năng của trò chơi chúng tôi đã trình bày trong Hình 3-1.

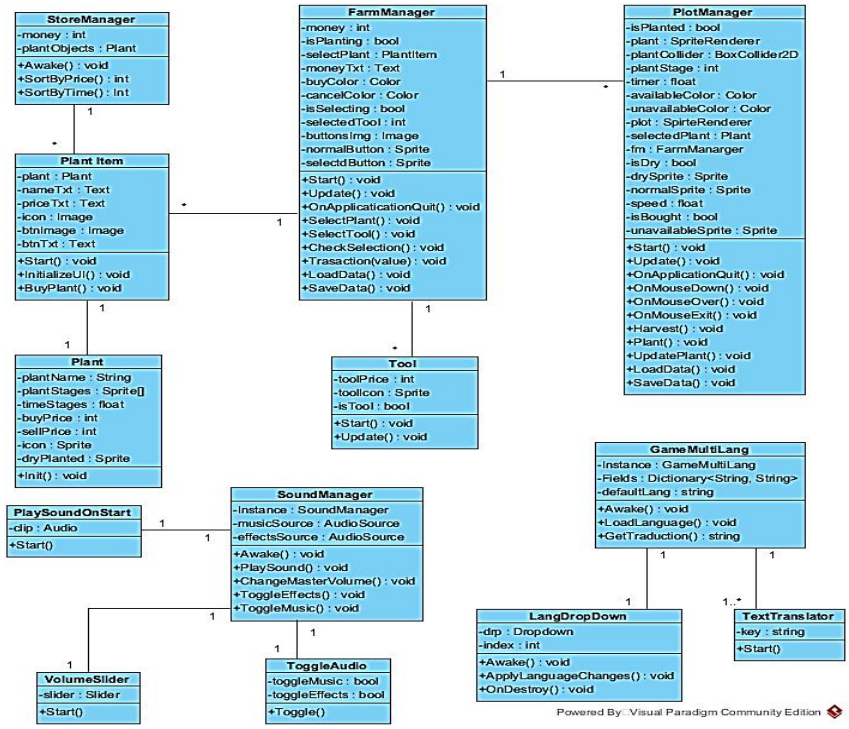



Hình 4: Usecase tổng quát

### 3.2 Sơ đồ lớp

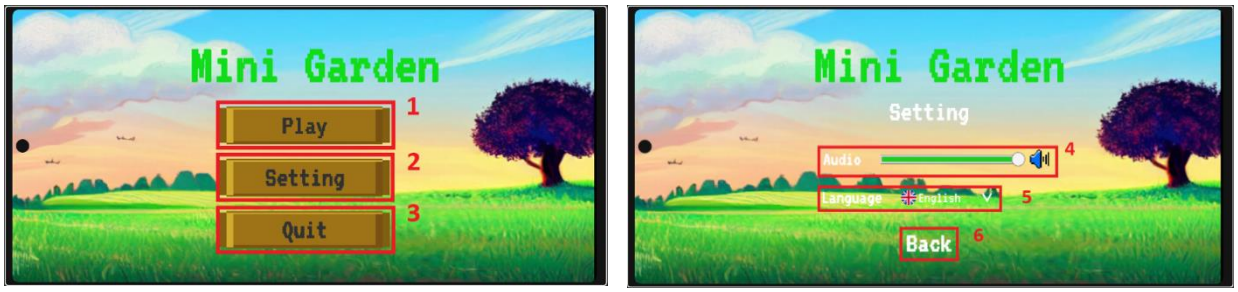
Từ các phân tích thiết kế của sơ đồ tổng quát chúng tôi đã đưa ra được lược đồ lớp của trò chơi. Dựa trên các chức năng chính của trò chơi chúng tôi chú trọng đến các lớp như: FarmManager, PlotManager, PlantItem, StoreManager, Tool, Plant. Ngoài ra các đối tượng như âm thanh và ngôn ngữ cũng tạo ra. Chi tiết của các đối tượng đã được chúng tôi trình bày trong sơ đồ lớp được thiết kế ở Hình 5.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 5: Sơ đồ lớp

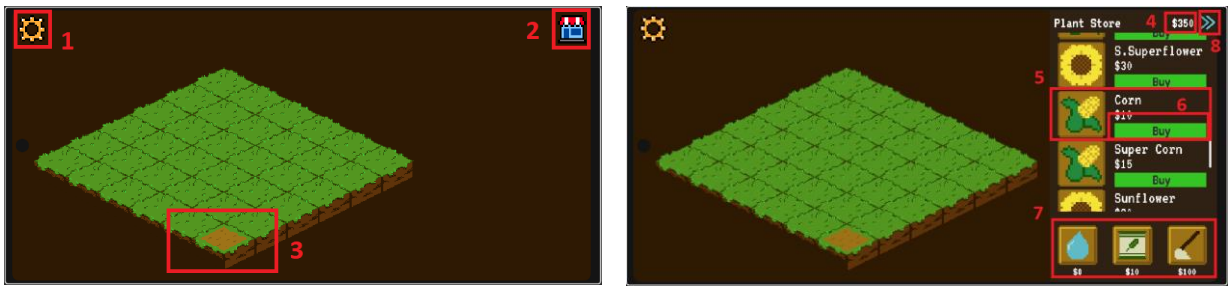
## 4 HIỆN THỰC TRÒ CHƠI




Hình 6: Màn hình khởi động trong trò chơi "Khu vườn Mini"

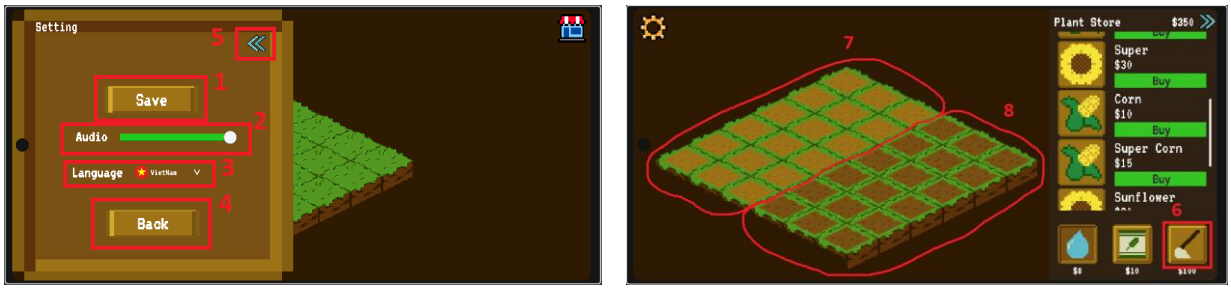
● Ở màn hình Menu sẽ hiển thị hình ảnh background, tiêu đề trò chơi và các nút xử lý.
1. Nút "Play" có chức năng cho phép người chơi vào màn hình chơi game.
2. Nút "Setting" có chức năng cho phép người chơi vào màn hình cài đặt.
3. Nút "Quit" có chức năng cho phép người chơi thoát khỏi ứng dụng game.
● Tiếp đến là màn hình của chức năng cài đặt và cấu hình game. Ở màn hình hiển thị các cấu hình như âm lượng và ngôn ngữ của game.
4. Thanh trượt "Audio" cho phép người chơi có thể thay đổi âm lượng to hoặc nhỏ theo ý muốn.
5. Nút thả xuống "Language" cho phép người chơi có thể thay đổi ngôn ngữ mà ứng dụng có sẵn.
6. Nút "Back" cho phép người chơi có thể quay lại màn hình Menu.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 7: Màn hình bắt đầu trò chơi "Khu vườn Mini"

● Màn hình Game được thiết kế đơn giản, có các chức năng chính của game.
1. Nút " " giúp hiển thị giao diện cài đặt và cho phép người chơi cài đặt và cấu hình khi đã vào game.
2. Nút " " giúp hiển thị giao diện cửa hàng và cho phép người chơi các chức năng của cửa hàng.
3. Giao diện nông trại mới bắt đầu hiển thị các ô đất và có 1 ô đất có sẵn cho người chơi sử dụng.
● Giao diện cửa hàng được hiển thị khi người chơi nhấn nút " ", giao diện nay hiển thị các chức năng chính của game và các thông số của cây trồng và nông trại.
4. Tiêu đề "$350" hiển thị số tiền mà người chơi có trong trò chơi.
5. Hiển thị hình ảnh, tên, giá mua của hạt giống mà người chơi có thể mua để gieo trồng.
6. Nút "Buy" cho phép người chơi mua hạt giống đó để gieo trồng.
7. Hiển thị các chức năng là công cụ để chăm sóc cây trồng trong nông trại.
8. Nút " " cho phép người chơi đóng giao diện cửa hàng.




Hình 8: Màn hình các chức năng chính "Khu Vườn Mini"

● Giao diện cài đặt trong game được hiển thị khi người chơi nhấn nút " ", giao diện này hiển thị các cấu hình và chức năng của game trò chơi.
1. Nút "Save" cho phép người chơi lưu lại các dữ liệu trong game.
2. Thanh trượt "Audio" cho phép người chơi có thể thay đổi âm lượng to hoặc nhỏ theo ý muốn khi đang trong game.
3. Nút thả xuống "Language" cho phép người chơi có thể thay đổi ngôn ngữ mà ứng dụng có sẵn khi đang trong game.
4. Nút "Back" cho phép người chơi quay lại màn hình Menu.
5. Nút " " cho phép người chơi đóng giao diện cài đặt trong game.
● Màn hình Game 1 cho thấy các ô đất đã được mở hết và các ô đất đó có sự khác nhau về màu sắc.
6. Nút " " cho phép người chơi mở thêm ô đất khi có đủ tiền để thực hiện chức năng này.
7. Các ô đất có màu sáng hơn là các ô đất chưa được tưới nước. Lúc mới mở thì các ô đất sẽ có màu như vậy.
8. Các ô đất có màu tối hơn là các ô đất đã được tưới nước. Lúc này thì hạt giống mới có thể gieo trồng.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 9: Màn hình trò chơi hoạt động

● Màn hình Game 2 hiển thị các ô đất đã cũng màu với nhau hết.
1. Nút " 💧 " cho phép người chơi tưới các ô đất khi ô đất đó sáng màu.
2. Giao diện hiển thị các ô đất đã được tưới đều, lúc này các ô đất có thể gieo trồng hạt giống.
● Màn hình Game 3 hiển thị các cây trồng đã được gieo trồng và các giai đoạn phát triển của cây.
3. Nút " 🌱 " cho phép người dùng bón phân cho cây trồng, giúp cây trồng giảm thời giảm thu hoạch.
4. Giao diện hiển thị các loại cây trồng khác nhau được gieo trồng và các giai đoạn phát triển của từng loại cây trồng.

## 5 KẾT LUẬN

Trò chơi 'Khu vườn Mini' đã được phát triển để đáp ứng nhu cầu giải trí của mọi người. Do sự phát triển mạnh mẽ của ứng dụng di động, trò chơi này được phát triển trên nền tảng Android, sử dụng công nghệ như Unity-2D. Đối với nhân viên văn phòng, sinh viên và học sinh, thời gian giải trí của họ hạn chế, vì vậy trò chơi nhằm cung cấp trải nghiệm làm vườn thú vị, kích thích cảm xúc tích cực và giảm căng thẳng, mà không tốn nhiều thời gian chơi game. Ứng dụng đã đạt được các yêu cầu của đề tài. Trò chơi được thiết kế đơn giản, thân thiện với người dùng và mang tính giải trí cao, đồng thời giúp giảm căng thẳng. Áp dụng các công nghệ mới nhất, ứng dụng đã tạo ra trải nghiệm thú vị và hiệu quả, đáp ứng nhu cầu ngày càng cao của người dùng. Tuy nhiên, do kinh nghiệm lập trình game chưa nhiều, trò chơi hiện tại còn đơn giản, thiếu tính năng độc đáo và đồ họa chưa sắc nét. Công nghệ mới trong lập trình game chưa được áp dụng nhiều trong dự án. Mặc dù trò chơi hướng tới thiết bị di động, nhưng hiện chỉ được phát triển trên nền tảng Android. Từ những hạn chế này, chúng tôi sẽ dành thời gian để tìm hiểu thêm về các công nghệ mới để lập trình game, hoàn thiện các chức năng và phát triển đồ họa. Trò chơi sẽ được bổ sung thêm nhiều tính năng và triển khai trên nhiều nền tảng như iOS.

## TÀI LIỆU THAM KHẢO

1. Janine Suvak (2015). Lập trình Game với Unity. Nhà xuất bản Đại học quốc gia Hà Nội.
2. Unity Documentation (2023). 2D game development. Truy xuất từ: https://docs.unity3d.com/Manual/Unity2D.html
3. Unity Documentation (2023). Spite Editor. Truy xuất từ: https://docs.unity3d.com/Manual/SpriteEditor.html
4. Photoshop, A. (2021). Adobe Photoshop. Preuzeto, 29, 2021. Truy xuất từ: http://www.ebookbou.edu.bd/Books/Text/SST/DCSA/dcsa_3301/Unit-06.pdf
5. Adobe Support (2023). Adobe Photoshop Learn & Support. Truy xuất từ: https://helpx.adobe.com/support/photoshop.html
6. Unity Documentation (2023). PlayerPrefs. Truy xuất từ: https://docs.unity3d.com/ScriptReference/PlayerPrefs.html

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F011**

# XÂY DỰNG ỨNG DỤNG QUẢN LÝ ĐẶT MÓN ATFOOD - TẠI HỆ THỐNG NHÀ HÀNG MỘC QUÁN TRÊN THIẾT BỊ ĐI ĐỘNG

TRẦN THỊ ANH THI[1], NGUYỄN THÁI AN[1], ĐINH THỊ PHƯƠNG THẢO[1]

[1]*Faculty of Information Technology, Industrial University of Ho Chi Minh City*
*tranthianhthi@iuh.edu.vn, nguyenthaian31102001@gmail.com, diinhphuongthao@gmail.com*

**Tóm tắt.** Với sự phát triển của công nghệ cũng như trong giai đoạn chuyển đổi số, những nghiệp vụ quản lý sẽ được triển khai trên máy tính cũng như các thiết bị di động. Những hệ thống này hiện đã góp phần rất lớn trong quá trình phát triển của các doanh nghiệp. Chúng được biết đến là các chương trình ứng dụng chạy trên nền tảng Web hay di động. Công cụ ATFood mà chúng tôi đề xuất cũng là một ứng dụng hỗ trợ việc quản lý các nghiệp vụ tại nhà hàng Mộc quán. ATFood sẽ có đầy đủ các chức năng và dịch vụ của một nhà hàng truyền thống. ATFood cung cấp các báo cáo và thống kê tự động về doanh thu, mức độ sử dụng, và các chỉ số kinh doanh quan trọng khác. Người dùng có thể đặt hàng trực tuyến qua ứng dụng di động và hệ thống sẽ tự động ghi nhận và xử lý đơn hàng một cách chính xác. Điều này giúp giảm thiểu sai sót trong quá trình đặt hàng và tăng tính linh hoạt cho khách hàng, giao diện sẽ thiết kế đơn giản, dễ làm quen để nhắm đến tất cả đối tượng người dùng. ATFood được xây dựng trên nền tảng công nghệ như Visual Studio, Google Cloud, Figma sự kết hợp của các công nghệ này đảm bảo tính bảo mật, khả năng mở rộng và hiệu suất cao cho ứng dụng. Hiện công cụ đã triển khai trên nền tảng Desktop, Mobile và Web, giúp cho việc giao dịch của khách hàng sẽ dễ dàng hơn. Trong tương lai, công cụ sẽ mở rộng các tính năng cũng như bổ sung các mô hình dịch vụ như khách sạn, siêu thị để đáp ứng nhu cầu đa dạng của các doanh nghiệp trong ngành nhà hàng và du lịch.

**Từ khóa.** Business service, Cloud, Web application, Desktop.

## BUILDING AN ORDER MANAGEMENT APPLICATION FOR ATFOOD RESTAURANT SYSTEM ON MOBILE DEVICES - AT MOC QUAN RESTAURANT

**Abstract.** With the advancement of technology and the digital transformation era, management tasks are being implemented on computers as well as mobile devices. These systems have greatly contributed to the development of businesses. They are known as web-based or mobile applications. The proposed tool, ATFood, is an application designed to support the management operations of Moc Quan restaurant. ATFood will encompass all the functions and services of a traditional restaurant. Its interface will be designed to be simple and user-friendly, targeting a wide range of users. ATFood is built on technology platforms such as Visual Studio, Google Cloud, and Figma. The combination of these technologies ensures security, scalability, and high performance for the application. Currently, the tool has been deployed on desktop, mobile, and web platforms, making customer transactions more convenient. In the future, the tool will expand its features and incorporate additional service models such as hotels and supermarkets, to cater to the diverse needs of businesses in the restaurant and tourism industry. ATFood provides automatic reports and statistics on revenue, usage rates, and other important business indicators. Users can place orders online through the mobile application, and the system will automatically record and process orders accurately. This helps minimize errors in the ordering process and increases flexibility for customers. Overall, ATFood stands out with its comprehensive features, user-friendly interface, and the utilization of advanced technologies, creating a modern and efficient management tool for the restaurant industry.

**Keywords.** Business service, Cloud, Web application, Desktop.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

# 1 GIỚI THIỆU

Hiện nay, việc đặt món qua ứng dụng đã trở thành một xu hướng phổ biến, mang lại sự tiện lợi và tiết kiệm thời gian cho khách hàng. Các ứng dụng đặt món phổ biến như GrabFood, Now.vn, Foody, và OpenTable đã giúp người dùng dễ dàng đặt món từ nhiều nhà hàng và quán ăn khác nhau.

Tuy nhiên, các công cụ hiện tại cũng đối mặt với một số thách thức và khó khăn. Quản lý đơn hàng từ nhiều nguồn khác nhau và đảm bảo thực phẩm đến đúng thời gian và địa điểm là một vấn đề phức tạp. Đồng thời, việc đồng bộ thông tin giữa nhà hàng và khách hàng có thể gặp phải sai sót và hiểu lầm, ảnh hưởng đến trải nghiệm của khách hàng. Ngoài ra, khả năng đáp ứng nhu cầu đặt món đa dạng và tùy chỉnh của khách hàng cũng là một thách thức đối với các công cụ hiện tại.

Với những thách thức và khó khăn này, ATFOOD ra đời nhằm giải quyết những vấn đề trên. ATFOOD hứa hẹn cung cấp một giải pháp hiệu quả để quản lý quá trình đặt món, từ việc đồng bộ thông tin chính xác giữa nhà hàng và khách hàng, đến khả năng đáp ứng đa dạng và tùy chỉnh của nhu cầu đặt món. Với ATFOOD, việc quản lý đặt món sẽ trở nên thuận tiện, chính xác và mang lại trải nghiệm tốt hơn cho cả nhà hàng và khách hàng.

# 2 CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 React Native

React Native được phát triển bởi Facebook, là một framework di động mã nguồn mở được thiết kế đặc biệt để xây dựng ứng dụng di động native cho iOS và Android bằng cách sử dụng JavaScript như ngôn ngữ lập trình. Nó sử dụng thư viện ReactJS để tạo giao diện người dùng trên các nền tảng di động khác nhau. Với React Native, nhà phát triển có thể tích hợp trực tiếp framework vào các ứng dụng di động iOS hoặc Android hiện có hoặc xây dựng hoàn toàn các ứng dụng native mới từ đầu. Hiện nay, React Native được sử dụng rộng rãi trong các ứng dụng di động phổ biến như Facebook, Instagram, Pinterest, Skype và nhiều ứng dụng khác [1].

## 2.2 React JS

ReactJS là một thư viện JavaScript mã nguồn mở được phát triển bởi Facebook. Nó tập trung vào việc xây dựng giao diện người dùng (UI) tương tác và linh hoạt cho các ứng dụng web. ReactJS sử dụng mô hình phát triển UI dựa trên các thành phần, giúp dễ dàng quản lý và tái sử dụng mã nguồn. ReactJS cũng hưởng lợi từ cộng đồng phát triển mạnh mẽ và sôi nổi, cùng với nhiều công cụ hỗ trợ, giúp phát triển ứng dụng web nhanh chóng và hiệu quả. Với khả năng tái sử dụng thành phần và hiệu suất tốt, ReactJS đã trở thành một trong những công nghệ phổ biến nhất trong phát triển giao diện người dùng [2].

## 2.3 Firebase

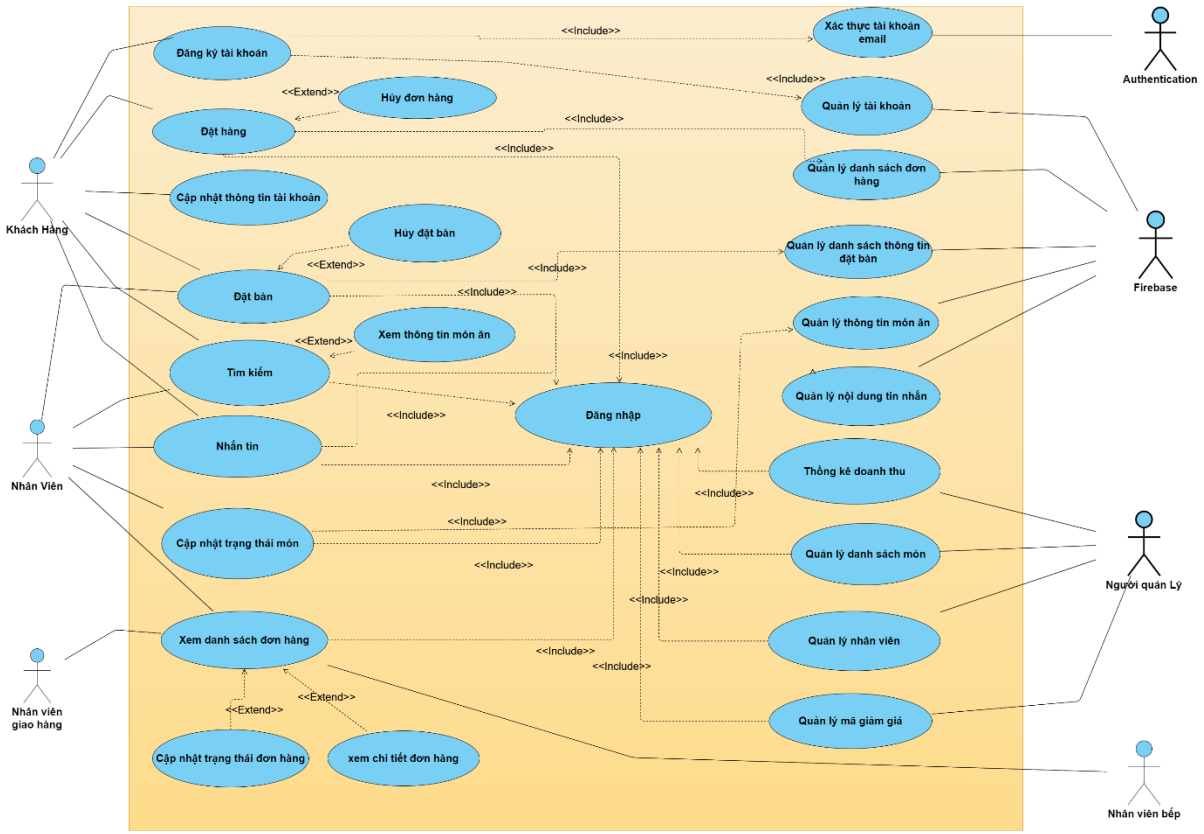
Firebase là một nền tảng dịch vụ dựa trên đám mây được thiết kế để hỗ trợ phát triển nhanh các ứng dụng web và di động. Nó cung cấp một loạt các tính năng cho phép nhà phát triển xây dựng và triển khai các ứng dụng chất lượng cao một cách nhanh chóng và dễ dàng. Firebase cung cấp một bộ công cụ và dịch vụ toàn diện giúp tối ưu quy trình phát triển, giúp nhà phát triển tập trung vào việc tạo ra các ứng dụng xuất sắc. Khả năng mở rộng linh hoạt và tích hợp mượt mà với các dịch vụ Google Cloud khác làm cho Firebase trở thành một lựa chọn phổ biến để xây dựng các ứng dụng web và di động mạnh mẽ và đa chức năng [3].

# 3 PHÂN TÍCH HỆ THỐNG

## 3.1 Sơ đồ tổng quát

Các vai trò người dùng sử dụng trong hệ thống ATFOOD bao gồm khách hàng, nhân viên, nhân viên giao hàng, nhân viên nhà bếp và quản lý nhà hàng. Một số chức năng chính dành cho khách hàng bao gồm đặt hàng, đặt chỗ và gửi tin nhắn. Ngoài ra, có các tính năng khác nhau để hỗ trợ quản lý nhà hàng, giúp việc giám sát và theo dõi tiến trình đặt hàng trở nên dễ dàng. Các chức năng của ứng dụng được trình bày trong Hình 1.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 1: Sơ đồ tổng quát

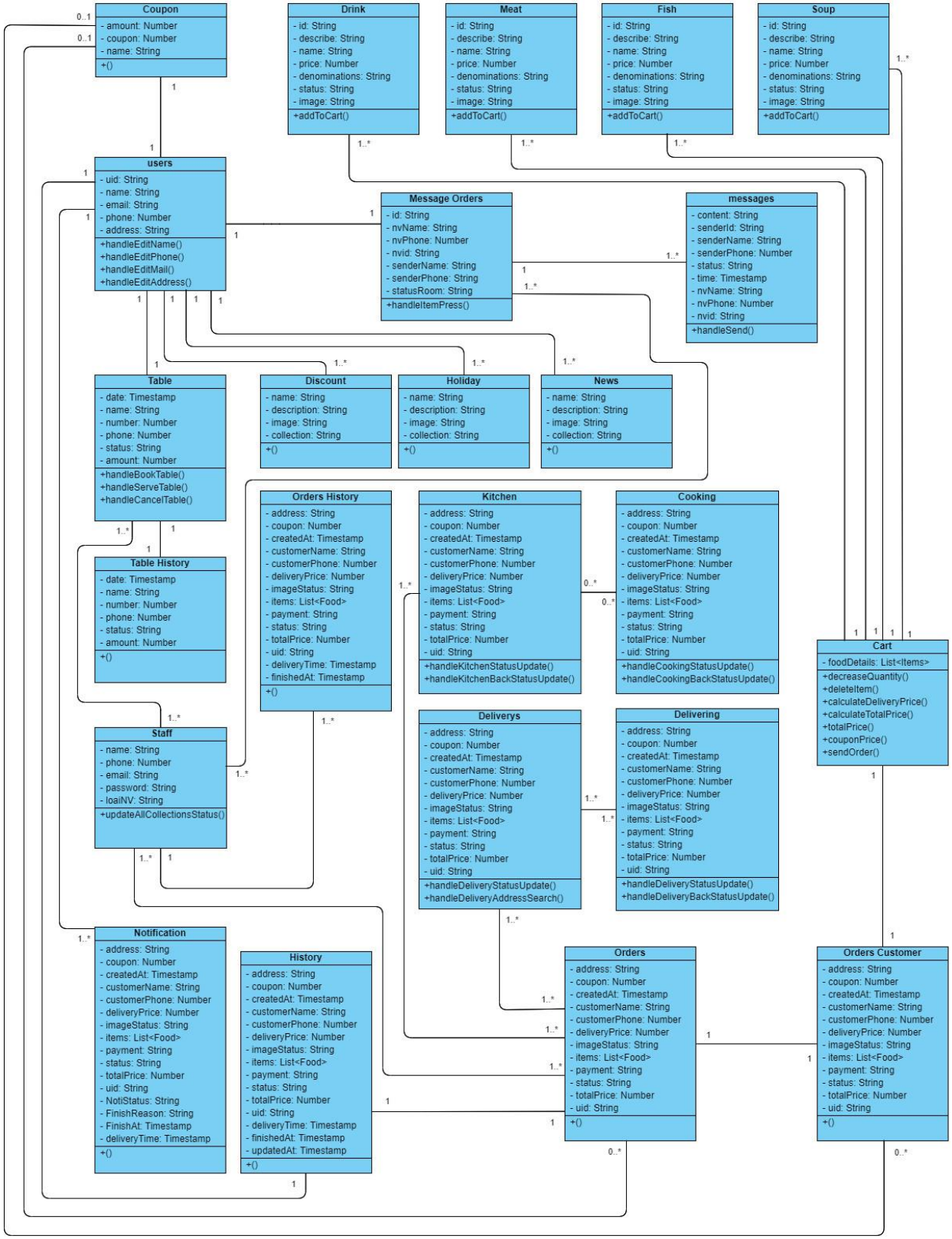
## 3.2 Sơ đồ lớp

Dựa trên việc phân tích và thiết kế tổng quan, chúng tôi đã tạo ra lược đồ lớp cho ứng dụng quản lý đặt món ATFOOD. Lược đồ lớp là một công cụ quan trọng để hiểu cấu trúc của hệ thống, bao gồm các lớp và mối quan hệ giữa các lớp. Ứng dụng tập trung vào các chức năng chính như đặt món, thanh toán và lịch sử đặt hàng của người dùng. Vì vậy, chúng tôi đã xác định các lớp quan trọng như sau: Table, Orders, messages, Cart, OrdersHistory,...

Sơ đồ lớp thiết kế của ứng dụng quản lý đặt món ATFOOD chi tiết các lớp và đối tượng được sử dụng trong hệ thống. Sơ đồ lớp này cung cấp một cấu trúc rõ ràng và tổ chức hiệu quả cho ứng dụng, giúp cho việc phát triển và bảo trì dễ dàng hơn.

Chi tiết về các lớp và đối tượng đã được chúng tôi trình bày chi tiết trong sơ đồ lớp thiết kế, có thể tham khảo trong Hình 2. Với sơ đồ lớp thiết kế này, chúng tôi đã xây dựng một cấu trúc rõ ràng và tổ chức hiệu quả để phát triển và bảo trì ứng dụng quản lý đặt món ATFOOD. Đồng thời, sơ đồ lớp cũng cung cấp khả năng mở rộng và tùy chỉnh trong tương lai, giúp cho ứng dụng có thể phát triển theo thời gian một cách linh hoạt và hiệu quả.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 2: Sơ đồ lớp

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

# 4 KẾT QUẢ HIỆN THỰC

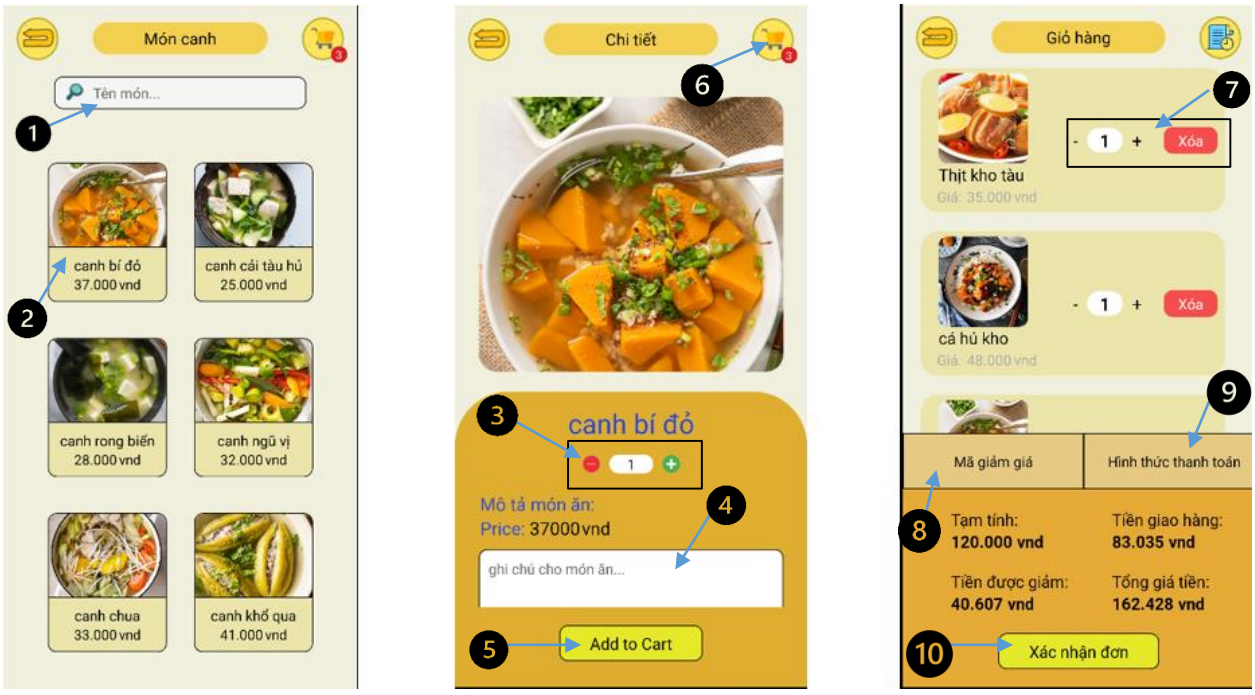
## 4.1 Cấu hình phần cứng, phần mềm

Để cấu hình server cho ứng dụng ATFOOD, chúng tôi sử dụng Firebase Hosting để cung cấp một môi trường phục vụ ứng dụng web (reactjs) và ứng dụng di động (reactnative). Chúng tôi tạo một dự án Firebase, sau đó tải lên mã nguồn ứng dụng lên Firebase Hosting. Việc này cho phép ứng dụng được phục vụ trực tiếp từ domain của chúng tôi, đồng thời tự động quản lý và triển khai các phiên bản mới.

Bên cạnh đó, Firebase Realtime Database được sử dụng để lưu trữ và đồng bộ dữ liệu của ứng dụng ATFOOD. Dữ liệu như danh sách món ăn, thông tin người dùng và tin nhắn sẽ được lưu trữ trong cơ sở dữ liệu này. Chúng tôi sử dụng API của Firebase để giao tiếp với cơ sở dữ liệu trực tiếp từ ứng dụng. Tất cả các cấu hình và tương tác với Firebase được thực hiện thông qua các phiên bản SDK của Firebase cho reactnative và reactjs. Điều này cho phép chúng tôi dễ dàng kết nối và tương tác với Firebase trong quá trình phát triển và triển khai ứng dụng ATFOOD.

## 4.2 Màn hình hiện thực

Để thuận tiện và dễ dàng sử dụng ứng dụng, giao diện đặt món trên ứng dụng ATFOOD được thiết kế đơn giản và thân thiện với người dùng. Với chỉ vài cú chạm, khách hàng có thể dễ dàng đặt hàng, xem danh sách món và tìm kiếm các món ăn yêu thích.



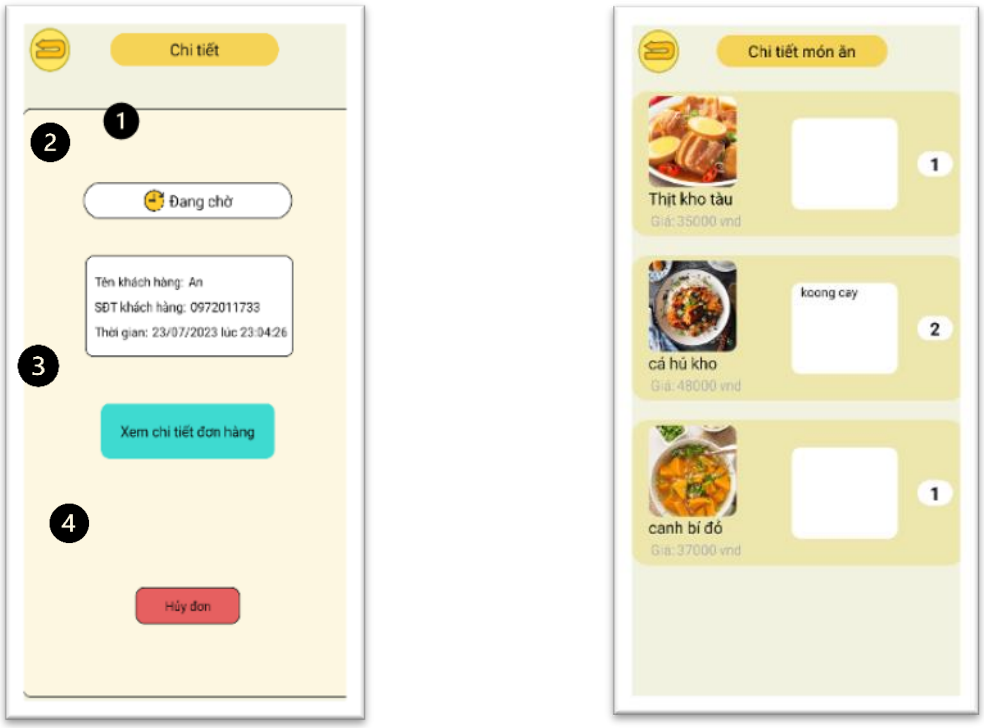

Hình 3 Giao diện thực hiện chức năng đặt hàng trên ứng dụng ATFOOD

- Ở màn hình danh sách món ăn cho phép khách hàng xem thông tin món và tìm kiếm món ăn:
  - Thanh tìm kiếm cho phép người dùng nhập và tìm kiếm món ăn yêu thích.
  - Người dùng bấm chọn để xem thông tin chi tiết món ăn.
- Khi người dùng chọn xem thông tin món ăn, hệ thống sẽ chuyển tới màn hình chi tiết món ăn đó:
  - Thanh điều chỉnh giúp người dùng có thể tăng giảm số lượng món.
  - Ô ghi chú cho phép người dùng nhập một số lưu ý đặc biệt cho món ăn trước khi cho vào giỏ hàng.
  - Nút "Add to Cart" có chức năng thêm thông tin món ăn vào giỏ hàng.
  - Nút biểu tượng giỏ hàng, người dùng bấm chọn để xem thông tin chi tiết giỏ hàng.
- Khi người dùng chọn xem thông tin giỏ hàng, hệ thống chuyển tới màn hình giỏ hàng:
  - Thanh điều chỉnh cho phép cho người dùng tăng giảm số lượng món hoặc xóa món ăn đó ra khỏi giỏ hàng.
  - Nút "Mã giảm giá" cho phép người dùng xem và chọn mã giảm giá áp dụng cho đơn hàng.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

- Nút "Hình thức thanh toán" cho phép người dùng xem và chọn phương thức thanh toán cho đơn hàng.
- Nút "Xác nhận đơn" có chức năng đặt hàng gửi thông tin đơn hàng tới cho nhà hàng.

Khi khách hàng bấm chọn "Xác nhận đơn", hệ thống sẽ chuyển đến màn hình chi tiết đơn hàng.




Hình 4: Giao diện thực hiện chức năng xem chi tiết đơn hàng trên ứng dụng ATFOOD

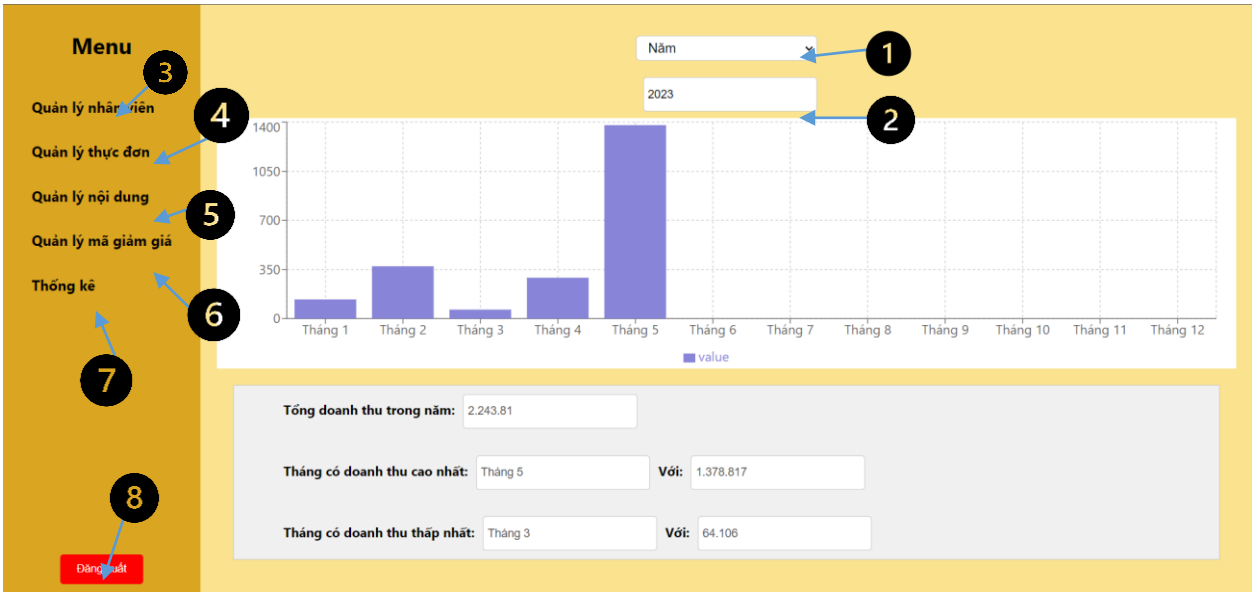
Tại đây, khách hàng có thể xem chi tiết được đơn hàng, trạng thái đơn hàng của mình hoặc sử dụng chức năng "Hủy đơn" khi có nhu cầu.

1. Click vào icon để quay lại màn hình trước đó.
2. Hiển thị trạng thái của đơn hàng "Đang chờ" biểu thị cho đơn hàng đang được đợi để xác nhận từ phía nhân viên của nhà hàng. Trạng thái sẽ được cập nhật khi có thao tác cập nhật trạng thái đơn hàng từ phía nhân viên của nhà hàng.
3. Bấm chọn nút "Xem chi tiết đơn hàng" để xem chi tiết đơn hàng đã được đặt.
4. Bấm chọn nút "Hủy đơn" để hủy đơn đặt hàng, chỉ có thể sử dụng chức năng này khi đơn hàng chưa chuyển trạng thái đơn hàng đang được chế biến.

Bên cạnh đó, hệ thống web của ATFOOD giúp người quản lý theo dõi tình hình kinh doanh một cách dễ dàng, cho phép người quản lý tình hình kinh doanh của nhà hàng, thực hiện các thao tác cập nhật thông tin về nhân viên, món ăn, nội dung, mã giảm giá và thống kê doanh thu. Điều này mang lại sự linh hoạt và tiện lợi cho công việc quản lý nhà hàng.

Hình 5 là hình ảnh giao diện thống kê dành riêng cho người quản lý, nơi mà người quản lý của nhà hàng có thể thực hiện các thao tác chức năng được kể ở trên.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 5: Giao diện chức năng thống kê dành cho người quản lý

Chức năng thống kê là chức năng được thiết kế dành riêng cho người quản lý của nhà hàng, tại đây người dùng có thể theo dõi được tình hình kinh doanh của cửa hàng theo tháng hoặc năm đưa ra so sánh từ đó dễ dàng đưa ra những phương án kinh doanh phù hợp.

1. Nút lọc, có chức năng giúp người dùng chọn loại thống kê theo năm hoặc theo tháng.
2. Nút chọn năm để thống kê, người dùng có thể chọn xem thống kê của một năm bất kỳ muốn xem.
3. Nút "Quản lý nhân viên", người dùng bấm chọn để chuyển tới cửa sổ quản lý nhân viên của nhà hàng.
4. Nút "Quản lý thực đơn", người dùng bấm chọn để chuyển tới cửa sổ quản lý danh sách món ăn.
5. Nút "Quản lý nội dung", người dùng bấm chọn để chuyển tới cửa sổ quản lý nội dung chương trình của nhà hàng.
6. Nút "Quản lý mã giảm giá", người dùng bấm chọn để chuyển tới cửa sổ quản lý mã giảm giá.
7. Nút "Thống kê" hiển thị thông tin thống kê doanh thu của cửa hàng.
8. Nút "Đăng xuất", có chức năng đăng xuất khỏi tài khoản hiện đang sử dụng.

## 5 THỰC NGHIỆM

Ứng dụng quản lý đặt món ATFOOD đã được triển khai thành công trên thiết bị di động. Ứng dụng này đem lại cho người dùng với nhiều tính năng hấp dẫn như đặt món, thanh toán, lựa chọn thực đơn, theo dõi trạng thái đơn hàng và theo dõi lịch sử đặt món. Quá trình triển khai và phát triển ứng dụng được bắt đầu từ tháng 12 năm 2022 và hoàn thành vào tháng 5 năm 2023, sử dụng mã nguồn mở và triển khai trên nền tảng Android.

Phản hồi của người dùng về ứng dụng rất tích cực, rằng ứng dụng đã cung cấp cho người sử dụng sự tiện lợi, giúp tiết kiệm thời gian và tối đa hóa trải nghiệm người dùng trên ứng dụng của nhà hàng. Tuy nhiên, một số người dùng cũng đã đề xuất một số cải tiến như nâng cao tính năng và tăng cường tính bảo mật.

Sau tất cả, việc triển khai công cụ quản lý đặt món ATFOOD đã mang lại nhiều lợi ích cho người dùng và hệ thống nhà hàng. Phản hồi từ người dùng cho thấy công cụ này đáp ứng được nhu cầu của họ, mặc dù còn cần được cải tiến để nâng cao tính năng và tăng cường bảo mật trong tương lai.

## 6 KẾT LUẬN

Bài báo này trình bày về giải pháp sử dụng công nghệ để xây dựng ứng dụng quản lý đặt món ATFOOD tại hệ thống nhà hàng Mộc Quán. Ứng dụng này đã được triển khai và mang lại nhiều tính năng hấp dẫn cho người dùng như đặt món, thanh toán, lựa chọn thực đơn, đánh giá và theo dõi lịch sử đặt món. Ngoài ra, việc triển khai một ứng dụng quản lý đặt món ăn cũng giúp cho các nhân viên của nhà hàng có thể quản

lý các đơn hàng một cách hiệu quả, đồng thời cũng giúp cho nhà hàng có thể tăng tính linh hoạt trong việc quản lý nhân viên và tài nguyên kinh doanh.
Kết quả thực tế cho thấy ứng dụng đã mang lại cho người dùng nhiều lợi ích như giúp tiết kiệm thời gian, đơn giản hóa quá trình đặt món và thanh toán, tối ưu hóa trải nghiệm người dùng tại nhà hàng. Công nghệ đưa vào ứng dụng cũng giúp cho quá trình vận hành và quản lý trong hệ thống nhà hàng trở nên dễ dàng hơn, hiệu quả hơn với việc tối ưu hóa quy trình hoạt động và quản lý thông tin.
Tuy nhiên, ứng dụng còn một số hạn chế như chỉ phù hợp với những cửa hàng kinh doanh nhỏ và chỉ dành cho bán hàng online. Điều này có thể giới hạn khả năng mở rộng và sử dụng của ứng dụng, giới hạn hệ sinh thái. Vì thế, hướng phát triển mới cho ứng dụng được đề xuất phát triển thêm hướng bán hàng tại cửa hàng và làm cho ứng dụng phù hợp hơn với những mô hình kinh doanh lớn hơn. Điều này sẽ giúp mở rộng phạm vi sử dụng của ứng dụng và thu hút nhiều đối tượng khách hàng khác nhau.

## TÀI LIỆU THAM KHẢO

[1]. Bonnie Eisenman. (2015). Learning React Native. O'Reilly Media, United States.

[2]. Morgan, J. (2021). How To Code in React.js. New York City, NY: DigitalOcean.

[3]. Smyth, N. (2017). Firebase Essentials – Android Edition. Payload Media.

[4]. Brown, B. (2017). Learning React Native: Building Native Mobile Apps with JavaScript. O'Reilly Media, Sebastopol, CA.

[5]. Ormrod, P. (2020). The Definitive Guide to Firebase: Build Android, iOS, and web apps using the Firebase platform. Apress, Berkeley, CA.

[6]. Banks, A., & Porcello, E. (2019). Learning React: Functional Web Development with React and Redux. O'Reilly Media, Sebastopol, CA.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F012**

# FARMING SIMULATOR 3D TT: PHÁT TRIỂN TRÒ CHƠI LÀM VƯỜN OFFLINE TRÊN THIẾT BỊ DI ĐỘNG SỬ DỤNG CÔNG NGHỆ UNITY

TRẦN THỊ ANH THI[1], LÊ VÕ MINH THUẬN[1], VÕ TRUNG TÍN[1]

*[1]Khoa Công nghệ thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*tranthianhthi@iuh.edu.vn, thuan22022001@gmail.com, votrungtinoppo@gmail.com*

**Tóm tắt.** Farming Simulator 3D TT là một trò chơi làm vườn trên di động được phát triển bởi engine Unity. Trò chơi được phát hành cho nền tảng Android vào năm 2023. Farming Simulator 3D TT cho phép người chơi trồng trọt, chăn nuôi và thu hoạch các loại cây trồng khác nhau trên một trang trại của riêng mình. Trò chơi có đồ họa 3D đẹp mắt và lối chơi hấp dẫn, giúp người chơi thư giãn và giải tỏa căng thẳng sau những giờ làm việc mệt mỏi. Farming Simulator 3D TT là một trò chơi mô phỏng nông nghiệp tuyệt vời dành cho mọi lứa tuổi. Trò chơi mang đến trải nghiệm chân thực về cuộc sống nông thôn Việt Nam và giúp người chơi thư giãn và giải tỏa căng thẳng. Nếu bạn là một người yêu thích nông nghiệp hoặc đang tìm kiếm một trò chơi giải trí hấp dẫn, thì Farming Simulator 3D TT là một lựa chọn tuyệt vời dành cho bạn.

**Từ khóa.** Trò chơi di động, Engine Unity, Trò chơi làm vườn.

## FARMING SIMULATOR 3D TT: DEVELOP OFFLINE GARDENING GAMES ON MOBILE DEVICES USING UNITY TECHNOLOGY

**Abstract.** Farming Simulator 3D TT is a mobile gardening game developed by Unity engine. It was released for Android platform in 2023. Farming Simulator 3D TT allows players to plant, raise and harvest various crops on their own farm. The game has beautiful 3D graphics and an engaging gameplay, which helps players relax and relieve stress after a long day of work. Farming Simulator 3D TT is a great farming simulation game for all ages. It brings a realistic experience of rural life in Vietnam and helps players relax and relieve stress. If you are a fan of farming or looking for an exciting entertainment game, then Farming Simulator 3D TT is a great choice for you.

**Keywords.** Mobile game, Engine Unity, Gardening game.

## 1 GIỚI THIỆU

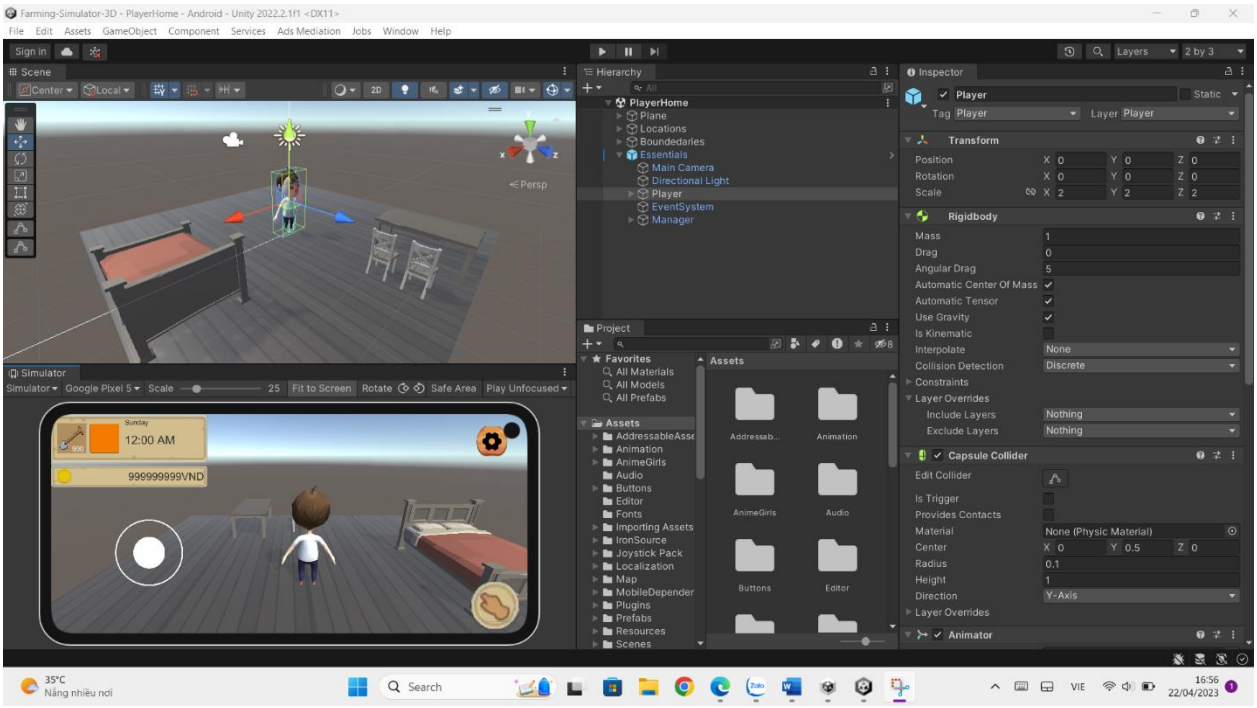
Công nghệ trong lĩnh vực Game hiện nay đang phát triển đáng kể, với sự tập trung vào trải nghiệm người dùng, nâng cao chất lượng hình ảnh và âm thanh, cũng như đa nền tảng. Người dùng đòi hỏi một trải nghiệm cao cấp và đơn giản hơn, vì vậy các Game Engine hiện nay hỗ trợ phát triển trò chơi một cách đơn giản và nhanh chóng, cung cấp nhiều tính năng đáp ứng các yêu cầu của người dùng. Các Game Engine đang hỗ trợ phát triển game bao gồm Unreal Engine, CryEngine, Unity và nhiều hơn nữa. Họ cung cấp một nền tảng phát triển chuyên nghiệp, hỗ trợ các tính năng đa dạng và đồng thời cung cấp khả năng tổ chức, quản lý tài nguyên của trò chơi. Unity Engine là một trong những Game Engine phổ biến và mạnh mẽ nhất hiện nay, với khả năng hỗ trợ phát triển trò chơi đa nền tảng và giúp người phát triển tiết kiệm thời gian và công sức trong việc phát triển các trò chơi khác. Nếu muốn phát triển một trò chơi làm vườn trên thiết bị di động, Unity là một lựa chọn tuyệt vời. Người dùng có thể tạo ra một trò chơi đồ họa đẹp mắt với nhiều tính năng độc đáo và chính xác, từ đó giúp người chơi có trải nghiệm thực tế và học hỏi được nhiều kiến thức về trồng trọt, mua sắm và chăm sóc cây trồng. Ngoài ra Unity cũng có nhiều tài liệu hướng dẫn và hỗ trợ từ cộng đồng để giúp người dùng phát triển trò chơi một cách dễ dàng và nhanh chóng [1].

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

# 2 CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 Unity

Unity là một công cụ phát triển phần mềm và trò chơi hoạt động trên nhiều nền tảng, bao gồm các hệ điều hành Windows, macOS, Linux, cũng như các thiết bị di động Android, iOS, Windows Phone và các hệ máy chơi trò chơi như Xbox và PlayStation. Unity được sử dụng phổ biến trong ngành công nghiệp game, giúp các nhà phát triển tạo ra các trò chơi với đồ họa và hiệu ứng chuyển động đẹp mắt, đồng thời cung cấp các tính năng đa nền tảng, dễ dàng tích hợp và phát triển các ứng dụng trên nhiều loại thiết bị khác nhau. Ngoài ra, Unity cũng được sử dụng trong nhiều lĩnh vực khác như tạo ứng dụng thực tế ảo, giáo dục, y tế, kiến trúc, xây dựng, và phát triển các ứng dụng trải nghiệm người dùng khác… Hơn 50% số lượng game trên thị trường được sản xuất bởi Unity. Một vài tựa game vô cùng nổi tiếng được tạo ra bởi Unity có thể được kể đến như Pokémon Go, Hearthstone, Ori And The Blind Forest, Monument Valley, Axie Infinity,… Độ "phủ sóng" của Unity rất rộng, có thể được áp dụng phổ biến trong nhiều dòng game khác nhau từ game "hạng nặng" Triple A (AAA) cho đến game giáo dục đơn giản cho các em nhỏ [2].

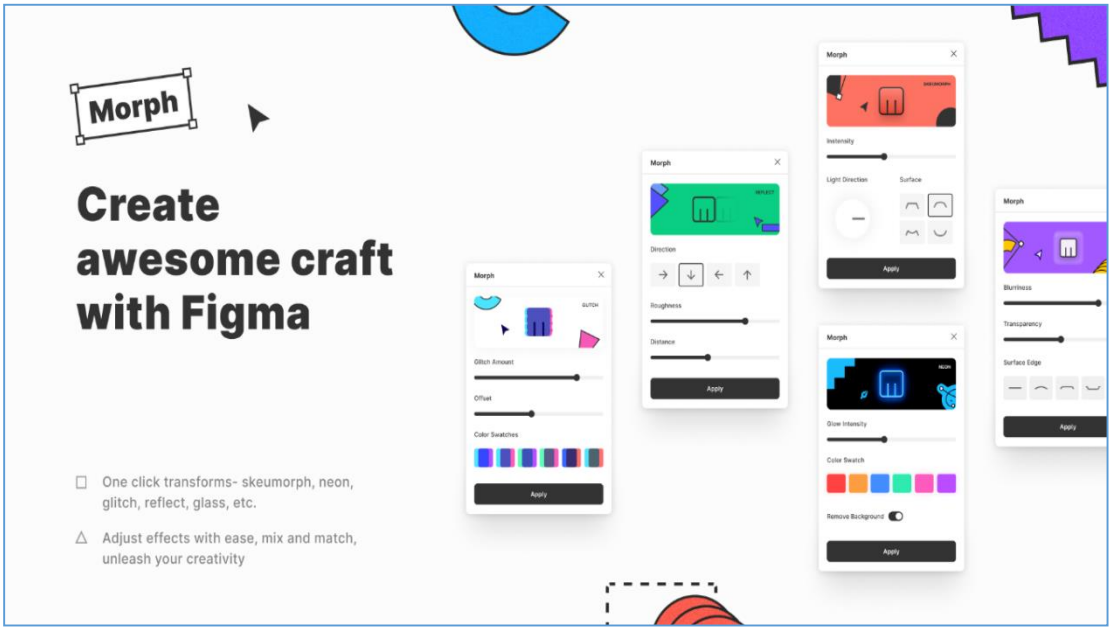



Hình 1: Màn hình thiết kế trên Unity Editor

## 2.2 Figma

Figma là một phần mềm biên tập đồ họa vector và dựng prototype. Figma chủ yếu hoạt động trên nền web, dù có một số tính năng ngoại tuyến trên các phiên bản ứng dụng desktop cho hệ điều hành macOS và Windows [3].

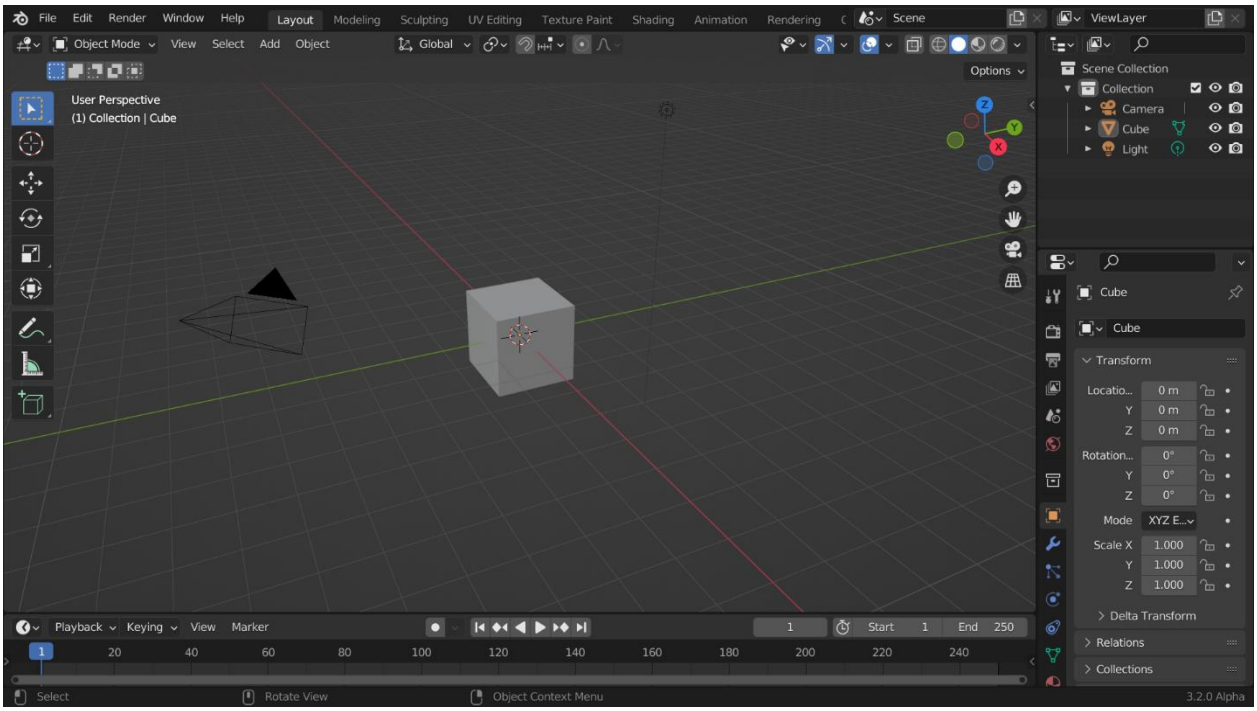

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 2: Giao diện đang nhập của Figma [3]

## 2.3 Phần mềm đồ họa Blender

Blender là một phần mềm miễn phí và mã nguồn mở được sử dụng để tạo và chỉnh sửa hình ảnh đồ họa, video và phim hoạt hình 3D. Nó được sử dụng rộng rãi trong các công việc liên quan đến thiết kế đồ họa, hoạt hình, trò chơi điện tử, quảng cáo và các lĩnh vực khác. Blender cung cấp một loạt các công cụ và tính năng cho việc mô hình hóa, ánh sáng, vật liệu và kết xuất, cùng với một số tính năng đặc biệt như tạo và chỉnh sửa rigging và animation [4].




Hình 3: Giao diện phần mềm đồ họa Figma

## 2.4 C Sharp

C Sharp là một ngôn ngữ lập trình phổ biến ra đời vào năm 2000 được phát triển bởi Microsoft, được sử dụng trong nhiều lĩnh vực, trong đó bao gồm cả game development. Trên nền tảng game development, C Sharp có thể được sử dụng để viết các Script cho game engine Unity, một trong những game engine phổ biến nhất trên thị trường hiện nay. Unity hỗ trợ CSharp với một số tính năng đáng chú ý như:

- Scripting API: Unity cung cấp một bộ thư viện API dành cho CSharp để giao tiếp với các đối tượng trong game, giúp phát triển từng thành phần của game.
- MonoDevelop: Unity đi kèm với trình biên dịch MonoDevelop để phát triển và quản lý các Script C Sharp.
- Visual Studio: Người dùng cũng có thể sử dụng Visual Studio để phát triển Script C Sharp, nhờ vào plugin hoặc sử dụng các công cụ có sẵn.

Với C Sharp trên Unity, người lập trình có thể xây dựng các tính năng gameplay, các phương thức kiểm soát game logic, sự kiện, cảnh, vật liệu và nhiều hơn nữa. C Sharp cũng cung cấp tính linh hoạt cao trong việc tùy chỉnh và mở rộng game engine Unity. Vì vậy, C Sharp là một trong những lựa chọn hàng đầu cho lập trình game trên Unity [5].

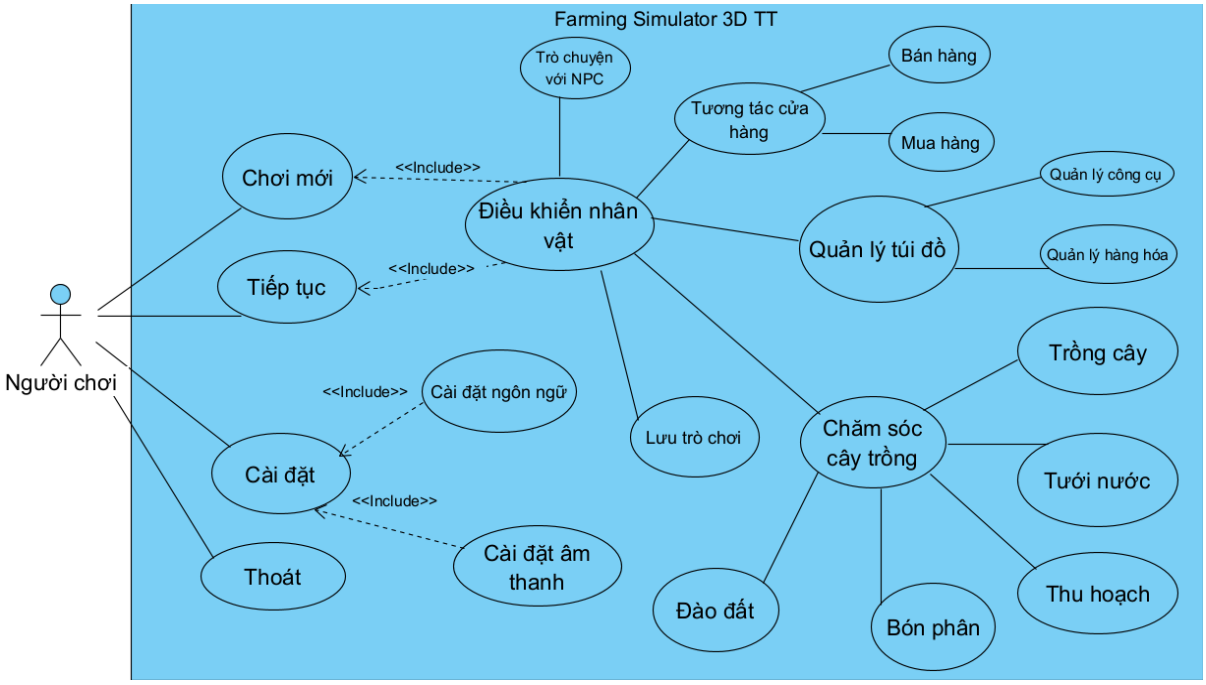
# 3 THIẾT KẾ VÀ HIỆN THỰC

Danh sách các tác nhân và mô tả:

Bảng 1: Bảng danh sách tác nhân và mô tả

| Tác nhân | Mô tả |
|---|---|
| Người chơi | Người chơi truy cập vào trò chơi, thực hiện các thao tác trong trò chơi như: trồng cây, chăm sóc cây trồng, quản lý túi đồ, mua bán vật phẩm... |

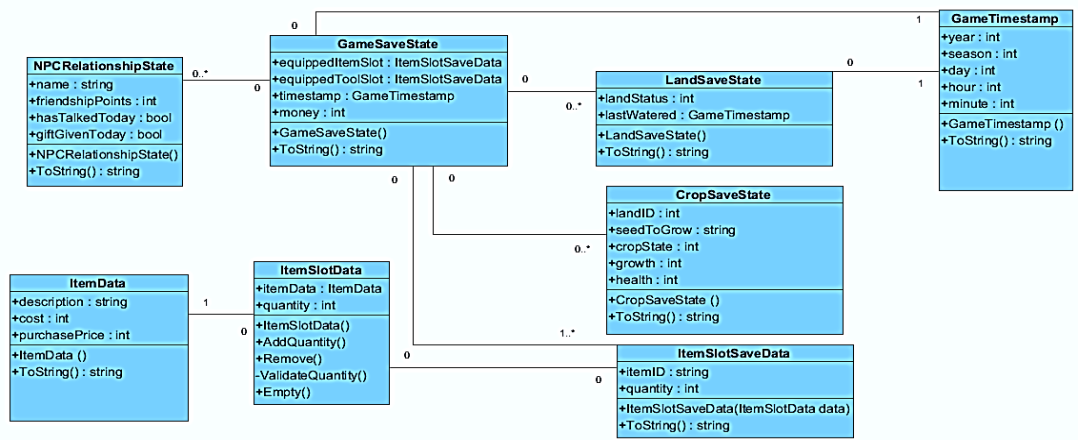
## 3.1 Sơ đồ Usecase tổng quát




Hình 4: Sơ đồ Usecase tổng quát của Farming Simulator 3D TT

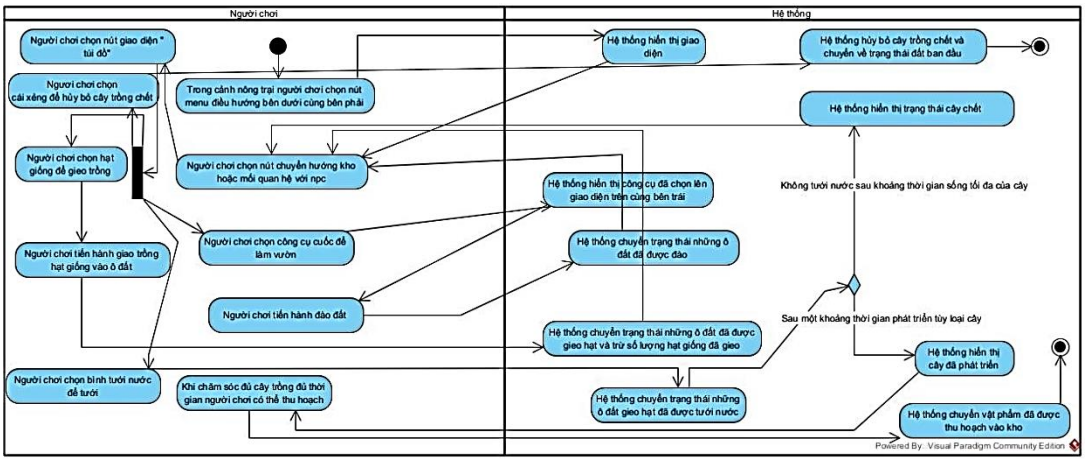

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 3.2 Sơ đồ lớp



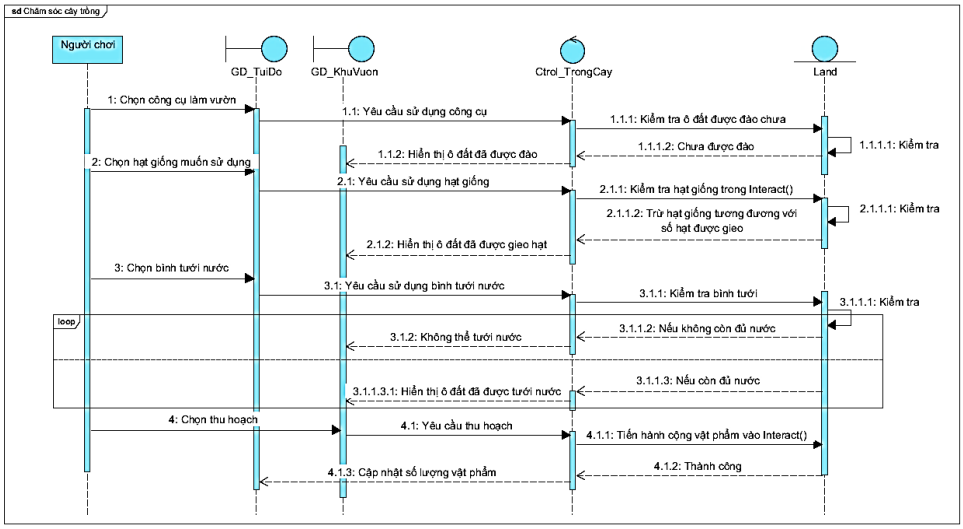

Hình 5: Sơ đồ lớp trong Farming Simulator 3D TT

## 3.3 Sơ đồ hoạt động (Activity Diagram) cho chức năng chăm sóc cây trồng




Hình 6: Sơ đồ hoạt động cho chức năng chăm sóc cây trồng

## 3.4 Sơ đồ tuần tự (Sequence Diagram) cho chức năng chăm sóc cây trồng




Hình 7: Sơ đồ tuần tự cho chức năng chăm sóc cây trồng

# 4 KẾT QUẢ HIỆN THỰC

## 4.1 Cấu hình phần cứng, phần mềm

Cấu hình phần cứng khuyến nghị để cài đặt trò chơi:

Bảng 2: Cấu hình phần cứng

| | |
|---|---|
| RAM | 4 GB |
| ROM | 64 GB |
| Hệ điều hành | Android 5.0 |

Cấu hình phần mềm các công cụ xây dựng trò chơi:

Bảng 3: Cấu hình phần mềm

| **Tên phần mềm** | **Phiên bản** | **Loại** |
|---|---|---|
| Unity Hub | 3.4.1 | Công cụ lập trình |
| Unity Editor | 2022.2.1f1 | Công cụ lập trình |
| Visual Studio 2022 | 17.3.5 | Công cụ lập trình |

## 4.2 Giao diện hiện thực màn hình

Quá trình bắt đầu chơi game khá đơn giản, cụ thể như sau: Sau khi vào trò chơi, người chơi được đưa ngay đến màn hình menu (hình 4.1) với các tùy chọn: Chơi mới (số 1), chơi tiếp tục (số 2 và sẽ không thể chơi tiếp tục khi người chơi lần đầu cài đặt), cài đặt (số 3), thoát (4). Khi người chơi chọn vào chơi mới, một panel hiện lên và hỏi tên người chơi. Sau khi điền các thông tin, người chơi được đưa đến màn hình nhà của mình, khởi đầu trò chơi thì người chơi được cung cấp số tiền (coin) là 10.000, các vật dụng cơ bản để làm vườn,... Có nhiều thứ để làm trong trò chơi, vì vậy hãy giành thời gian khám phá thế giới và bắt đầu trồng trọt.




Hình 8: Màn hình menu bắt đầu trò chơi


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Tại màn hình khu vườn (hình 9), người chơi tiến hành đào đất để có thể gieo hạt giống thông qua sử dụng công cụ là cuốc chim bên trong túi đồ (hình 10), công cụ này sẽ có sẵn khi người chơi vừa bắt đầu trò chơi. Sau khi đã đào đất xong người chơi sẽ chọn hạt giống từ túi đồ và tiến hành gieo trên những ô đã được đào. Để cây trồng phát triển nhanh hơn, người chơi tưới nước bằng cách sử dụng bình tưới nước có sẵn trong túi đồ.



Hình 9: Màn hình khu vườn




Hình 10: Màn hình túi đồ

Sau một khoảng thời gian, tùy theo từng loại cây trồng mà hạt giống mà người chơi đã gieo sẽ phát triển thành cây con (hình 11). Sau khi cây đến giai đoạn trưởng thành và có thể thu hoạch (hình 12) người chơi thu hoạch bằng cách di chuyển nhân vật đến cây cần thu hoạch và nhất vào nút "Thu hoạch"



Hình 11: Cây trồng trưởng thành



Hình 12: Cây trồng có thể thu hoạch

## 4.3 Kiểm thử

- Phạm vi kiểm thử: kiểm thử về các chức năng, kiểm thử API, kiểm thử cơ sở dữ liệu, phần cứng, kiểm thử giao diện.
- Mục tiêu kiểm thử:
  - Đảm bảo rằng trò chơi đáp ứng đúng các yêu cầu của người chơi.
  - Phát hiện và sửa lỗi trong trò chơi.
  - Cải thiện chất lượng của trò chơi.
- Phương pháp kiểm thử: Unit Test, Kiểm thử API
- Tài nguyên kiểm thử:
  - Nguồn nhân lực: Các thành viên nhóm
  - Tài nguyên hệ thống:

Bảng 4: Bảng tài nguyên hệ thống

| TT | Tài nguyên | Mô tả |
|---|---|---|
| 1 | Test tool | Công cụ kiểm thử là tự động hóa kiểm thử, mô phỏng hoạt động của người dùng, tạo kết quả kiểm thử |

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

- Kết quả kiểm thử: Nhóm đã tiến hành kiểm thử và thu được bảng kết quả như sau:

Bảng 5: Bảng kết quả kiểm thử

| Testcase | Số lần pass | Số lần fail |
|---|---|---|
| Testcase chăm sóc cây trồng | 5 | 4 |
| Testcase mua vật phẩm | 5 | 1 |
| Testcase bán vật phẩm | 3 | 1 |

## 5 KẾT LUẬN

Bài báo này đã trình bày ý tưởng triển khai các công nghệ Unity, Blender, Figma,...vào việc xây dựng trò chơi Farming Simulator 3D TT với các chức năng cơ bản của việc làm vườn như đào đất, trồng cây, gieo hạt, tưới nước, thu hoạch, mua bán vật phẩm trong trò chơi.

Trò chơi của chúng tôi triển khai trên các thiết bị di động Android, một thiết bị công nghệ phổ biến và phù hợp với mọi người. Để phát triển trò chơi làm vườn chúng tôi sử dụng C Sharp, một ngôn ngữ lập trình phù hợp với công nghệ Unity hỗ trợ làm game chạy trên thiết bị di động, Unity hỗ trợ đa nền tảng và có tiềm năng, cộng đồng hỗ trợ to lớn. Để quản lý dữ liệu chúng tôi sử dụng tập tin nhị phân. Chúng tôi cũng đã tiến hành kiểm chứng phần mềm qua các qui trình kiểm thử phần mềm và đều các kết quả đạt theo yêu cầu của một ứng dụng chạy trên di động. Tuy nhiên trò chơi vẫn còn hạn chế một số tính năng hỗ trợ các nghiệp vụ làm vườn khác. Cụ thể những quy trình, nghiệp vụ của trò chơi chưa diễn ra thực sự mượt mà với nhiều kịch bản cho tình huống khác nhau, hay còn chưa triển khai trò chơi trên nhiều nền tảng thiết bị di động khác nhau (iOS). Đây là một hạn chế của trò chơi làm vườn FARMING SIMULATOR 3D TT. Trong tương lai chúng tôi sẽ dành nhiều thời gian, cũng như tìm kiếm sự hợp tác để phát triển trò chơi hoàn thiện hơn nhằm phục vụ nhu cầu giải trí của mọi người.

## TÀI LIỆU THAM KHẢO

[1]. A. A. H. &. F. H. Motallebinejad, “Evaluation of three mobile game-based learning games and proposing a suitable evaluation model,” International Journal of Learning spaces Studies, pp. 23-41, 2023.

[2]. S. L. S. H. J. K. J. H. J. J. M. L. T. H. &. W. J. Kim, “Using Unity 3D to facilitate mobile augmented reality game development,” In 2014 IEEE World Forum on Internet of Things (WF-IoT), pp. 21-26, 2014.

[3]. “Figma,” 27 9 2016. [Trực tuyến]. Available: https://www.figma.com/developers/api.

[4]. “Blender Documentation - blender.org,” 25 4 2023. [Trực tuyến]. Available: https://docs.blender.org/.

[5]. “C# documentation,” 8 11 2022. [Trực tuyến]. Available: https://learn.microsoft.com/en-us/dotnet/csharp/.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F013**

# DỰ ĐOÁN GIÁ CỔ PHIẾU SỬ DỤNG THUẬT TOÁN DI TRUYỀN

NGUYỄN NĂNG ANH[1*], NGUYỄN TẤN ĐẮT[1], NGUYỄN HUỲNH ĐAN KHÁNH[1],
ĐỖ THÀNH MINH PHÚ[1], CAO HUỲNH VÕ THANH[1], LÊ PHÚC LỮ[1], NGUYỄN HỮU TÌNH[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
**nanganha10@gmail.com*

**Tóm tắt.** Dự đoán giá cổ phiếu là một trong những bài toán thực tế thu hút được sự quan tâm của cả các nhà đầu tư và nhà nghiên cứu về kinh tế. Tuy nhiên, bài toán này gặp nhiều khó khăn do giá cổ phiếu có tính ngẫu nhiên và biến động cao. Trong nghiên cứu này, chúng tôi giới thiệu phương pháp GA-LSTM, sử dụng thuật toán di truyền (GA) để lựa chọn các đặc trưng đầu vào cho mô hình bộ nhớ dài - ngắn hạn (LSTM). Phương pháp GA-LSTM được áp dụng để phân tích và dự đoán giá cổ phiếu, giúp cho việc đưa ra các quyết định đầu tư trở nên hiệu quả hơn. Các kết quả thực nghiệm trên dữ liệu thực tế cho thấy rằng phương pháp đề xuất đạt được kết quả tốt hơn so với các phương pháp thống kê truyền thống và mô hình học máy thông thường.

**Từ khoá.** Dự đoán giá cổ phiếu, Thuật toán di truyền, Bộ nhớ dài - ngắn hạn, dự báo tài chính, phân tích chuỗi thời gian.

## STOCK PRICE PREDICTION USING GENETIC ALGORITHM

**Abstract.** The problem of predicting stock prices is an interesting and important task that attracts the attention of investors and economic researchers. However, it is a complex problem, as stock prices are usually noisy and random. In this paper, we introduce the GA-LSTM method that combines genetic algorithm (GA) and long short-term memory (LSTM) network to select input features for stock price analysis and prediction. The GA-LSTM method is applied to analyze and predict stock prices, improving the effectiveness of investment decision-making. Experimental results on real data show that the proposed method achieves better performance than traditional statistical methods and conventional machine learning models.

**Keyword.** Stock price prediction, Genetic algorithm, LSTM, Financial forecasting, Time series analysis.

## 1 GIỚI THIỆU

Trong nền kinh tế phát triển, số lượng công ty được niêm yết trên sàn chứng khoán ngày càng tăng đã mở ra nhiều cơ hội sinh lời cho nhà đầu tư, làm cho thị trường cổ phiếu trở thành một chủ đề hấp dẫn trong lĩnh vực tài chính [1]. Sự biến đổi xu hướng của giá cổ phiếu phụ thuộc bởi nhiều yếu khác nhau bao gồm: sự kiện chính trị và địa chính trị; tỷ giá hối đoái; diễn biến của những thị trường cổ phiếu khác; tình hình nền kinh tế và tâm lý của các nhà đầu tư [2] [3] [4]. Phương pháp dự đoán giá cổ phiếu truyền thống thường xây dựng các mô hình dự đoán tuyến tính dựa trên dữ liệu lịch sử của cổ phiếu.

Tuy nhiên, do đặc thù của dữ liệu cổ phiếu nên Bowden và cộng sự [5] đã dùng phương pháp ARIMA (Autoregressive Integrated Moving Average) thường được sử dụng để xây dựng để xây dựng mô hình tự hồi quy dự đoán giá cổ phiếu. Mặc dù phương pháp này có một số ưu điểm về hiệu suất tính toán, giả định rằng phân phối thống kê và sự ổn định của dữ liệu nghiên cứu giới hạn khả năng mô hình hóa chuỗi thời gian vốn đã phi tuyến tính và không ổn định. Bên cạnh đó các giá trị ngoại lai trong dữ liệu ảnh hưởng đáng kể đến kết quả dự đoán của mô hình. Giá cổ phiếu có thể bị ảnh hưởng bởi nhiều yếu tố khác nhau.

Cùng với sự  phát triển của các kỹ thuật thống kê trong lĩnh vực tài chính, các nhà nghiên cứu đã khai thác nhiều yếu tố ảnh hưởng đến thị trường chứng khoán và định lượng những yếu tố đó thành dữ liệu cụ thể để

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

nghiên cứu xu hướng biến động của các cổ phiếu. Sự hỗ trợ của dữ liệu lớn đã hiện thực hoá khả năng của các thuật toán máy học. Ngày càng có nhiều nhà nghiên cứu bắt đầu sử dụng mô hình dự đoán phi tuyến tính của máy học để dự đoán giá cổ phiếu. Nair và đồng nghiệp [6] đã đề xuất một hệ thống cây quyết định trên tập mờ. Phương pháp kết hợp những ưu điểm của cây quyết định và tập mờ, nhưng dễ gặp hiện tượng overfitting khi xử lý tập dữ liệu có nhiều nhiễu dẫn đến ảnh hưởng đến kết quả dự đoán của mô hình. Theo lý thuyết, mạng nơ-ron nhân tạo (ANN) có thể học được bất kỳ mối quan hệ phi tuyến tính nào và ít bị ảnh hưởng hơn bởi dữ liệu nhiễu, chính vì vậy mà nó đã được sử dụng phổ biến trong lĩnh vực dự đoán chuỗi thời gian. Li và đồng nghiệp [7] đã tiến hành một loạt mô hình dự đoán sử dụng mạng nơ-ron và đạt được kết quả khả quan hơn so với cây quyết định. Tuy nhiên, các mạng nơ-ron dễ gặp vấn đề tối ưu hóa cục bộ trong quá trình thực nghiệm, và Support Vector Machine (SVM) dựa trên giảm thiểu hoá rủi ro cấu trúc giúp tỉ lệ rơi vào lỗi tối ưu cục bộ của mô hình giảm đi đáng kể. Cao và đồng nghiệp [8] đã xây dựng mô hình dự đoán cổ phiếu với SVM từ đó cải thiện hiệu suất tổng thể của mô hình.

Sự vượt bậc của trí tuệ nhân tạo, học máy, học sâu thu hút sự chú ý rộng rãi nhờ hiệu suất vượt trội trong các lĩnh vực như là: dịch máy [9] nhận diện cảm xúc giọng nói [10]; nhận dạng hình ảnh [11] và nhiều khía cạnh khác nữa. So với mô hình thống kê truyền thống, Deep Neural Network (DNN) có thể phân tích mối quan hệ phi tuyến phức tạp qua biểu diễn đặc trưng thành các lớp, điều này phù hợp với vấn đề phức tạp, bị ảnh hưởng bởi nhiều yếu tố, tính mất ổn định và phi tuyến tính của dữ liệu cổ phiếu [12]. Tsantekidis và đồng nghiệp [13] đề xuất một mô hình dự đoán cổ phiếu dựa trên mạng nơ-ron tích chập (CNN) và so sánh với các mô hình khác để xác minh tính hiệu quả của mô hình tích chập trong dự đoán cổ phiếu. Tuy nhiên, do tính thời điểm của dữ liệu cổ phiếu, mạng nơ-ron tích chấp không phải là mô hình mạng nơ-ron phù hợp nhất cho dự đoán cổ phiếu. Selvin và đồng nghiệp [14] đã đề xuất ba mô hình dự đoán cổ phiếu dựa trên CNN, mạng nơ-ron hồi quy (RNN) và LSTM. Kết quả cho thấy phương pháp LSTM là phù hợp nhất để dự báo thị trường chứng khoán và chuỗi thời gian là do khả năng ghi nhớ dài hạn của mạng. Đối với dự báo tài chính dữ liệu đa biến, việc lựa chọn được đặc trưng đóng vai trò rất là quan trọng. Lựa chọn đặc trưng mang lại nhiều lợi ích, chẳng hạn như:

(i) giảm thời gian huấn luyện mô hình;
(ii) đơn giản hoá sự mô hình bằng cách loại bỏ đặc trưng không cần thiết;
(iii) cải thiện độ chính xác của mô hình;
(iv) giảm việc overfitting nhờ loại bỏ đi những biến không cần thiết [15].

Yu và đồng nghiệp [16] đã thành công cải thiện độ chính xác dự đoán của mô hình bằng cách sử dụng PCA để giảm chiều dữ liệu kết hợp với mô hình SVM. Dựa trên sự thay đổi những thông tin ảnh hưởng qua các giai đoạn khác nhau, Zheng và đồng nghiệp [16] đã sử dụng Multistage Attention Network và học được thông tin ảnh hưởng của nhiều chuỗi thời gian không dự đoán tác động lên chuỗi mục tiêu qua các giai đoạn thời gian khác nhau.

Mặc dù những phương pháp này có thể có hiệu quả trong việc bắt lấy đặc trưng tạm thời nhưng chúng không thể xác định hiệu quả của một tổ hợp yếu tố. Khi mà số lượng yếu tố dữ liệu tăng lên, và các yếu tố có xu hướng tương quan hoặc ảnh hưởng lẫn nhau. Việc áp dụng thuật toán di truyền dựa trên quần thể có thể giải quyết hiệu quả các vấn đề về nhiễu và cộng tuyến của đặc trưng. Bằng việc ứng dụng thuật toán di truyền để chọn lọc đặc trưng và đưa những đặc trưng đó vào mô hình dự đoán giá cổ phiếu LSTM.

## 2 CƠ SỞ LÝ THUYẾT

### 2.1 Dự đoán thị trường chứng khoán

Thị trường chứng khoán phải đối mặt với nhiều dữ liệu chứng khoán đến từ các ngành công nghiệp khác nhau, bao gồm toàn bộ dữ liệu về thị trường tài chính. Dựa trên các dự đoán về thị trường, các nhà đầu tư điều chỉnh hành động của mình dựa trên việc mua và bán các cổ phiếu. Có một số yếu tố ảnh hưởng đến tình trạng thị trường bao gồm: thông báo về lợi nhuận; thông báo chia cổ tức; thay đổi đội ngũ quản lý; và nhiều yếu tố khác. Các chuyên gia xem xét rằng các vấn đề về giao dịch chứng khoán có ảnh hưởng đến giá của cổ phiếu trên thị trường chứng khoán [17].

Hiện tại, các công cụ khai phá dữ liệu đóng một vai trò không nhỏ trong việc hỗ trợ các nhà đầu tư, quỹ đầu tư, tổ chức để dự đoán xu hướng và hành vi của thị trường, chủ động đưa ra các giải pháp dựa trên dữ liệu có cơ sở kiến thức [18].

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

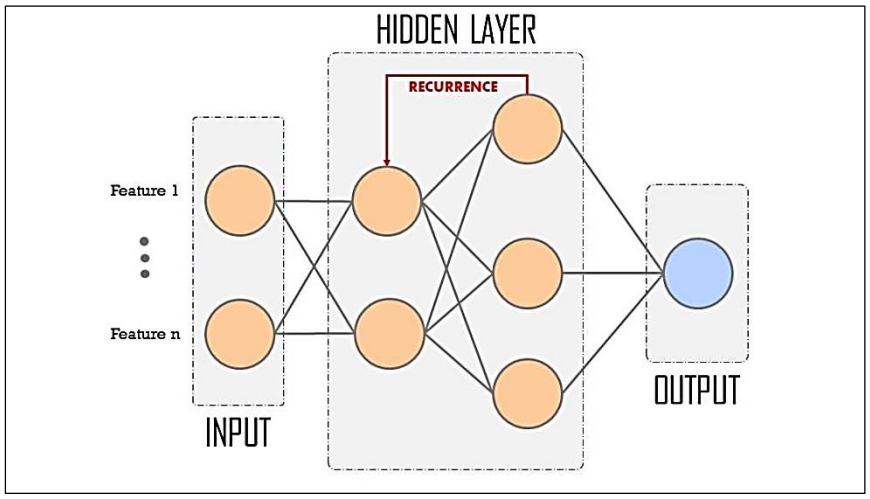
## 2.2 RNN – LSTM

### 2.2.1 Giới thiệu về Deep learning

Deep Learning là một phần của Machine learning, là phương pháp xây dựng một mạng nơ-ron có nhiều lớp. Những mạng lưới thần kinh này cố gắng mô phỏng hành vi của não người cho phép deep learning học được từ một lượng lớn dữ liệu. Trong khi mạng nơ-ron có một lớp vẫn có thể đưa ra các dự đoán gần đúng, các lớp ẩn bổ sung có thể giúp tối ưu hóa và tinh chỉnh để có độ chính xác.
Deep Learning thúc đẩy nhiều ứng dụng và dịch vụ trí tuệ nhân tạo (AI) nhằm cải thiện tự động hóa, thực hiện các tác vụ phân tích và vật lý mà không cần sự can thiệp của con người.
Công nghệ deep learning được ứng dụng rộng rãi trong các sản phẩm và dịch vụ hàng ngày chẳng hạn như trợ lý kỹ thuật số, điều khiển từ xa hỗ trợ giọng nói và phát hiện gian lận thẻ tín dụng cũng như các công nghệ mới nổi chẳng hạn như ô tô tự lái.
Các thuật toán Deep learning nổi bật: Convolution Neural Network (CNN) sử dụng rất phổ biến trong các bài toán về Computer Vision, Recurrent Neural Network (RNN) lại phù hợp với các loại dữ liệu có mối liên hệ về thời gian như time serial forecasting hay trong các bài toán về xử lý ngôn ngữ tự nhiên, bởi vì một phần output của nó ở thời điểm này được đưa trở lại thành input ở thời điểm tiếp theo, ...

### 2.2.2 Mạng nơ-ron hồi quy (RNN)

Mạng nơ-ron hồi quy là một mạng nơ-ron nhân tạo có khả năng tính toán mạnh mẽ và ổn định, được sử dụng hiệu quả trong việc giải quyết các vấn đề phức tạp trong nhiều lĩnh vực bao gồm xử lý ngôn ngữ tự nhiên, nhận dạng giọng nói và xử lý ảnh [16]. Khác với mạng nơ-ron nhân tạo truyền thống (ANN), không thể ghi nhớ được thông tin lịch sử trước đó của các chuỗi, RNN có hidden layer để duy trì các đặc điểm của toàn bộ chuỗi.

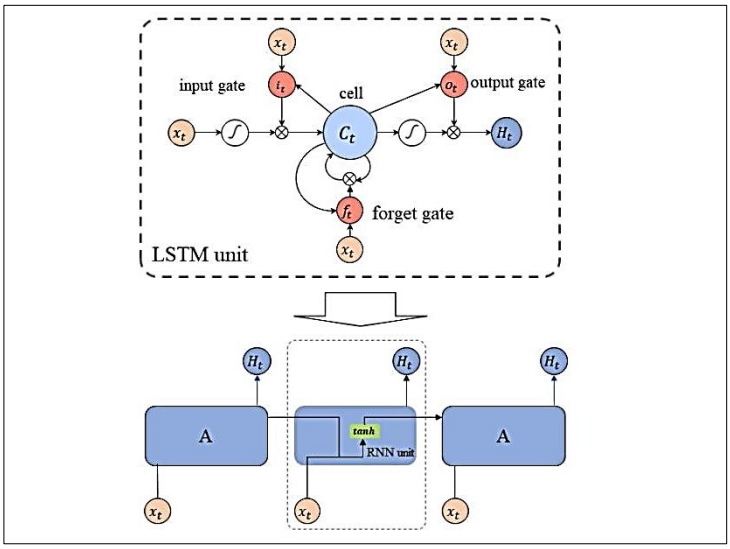



Hình 1: Mạng nơ-ron hồi quy truyền thống (RNN)

### 2.2.3 Long Short-Term Memory (LSTM) Network

Khi mà số lớp (layer) trong mạng ngày càng trở nên nhiều hơn, vấn đề của RNN là chỉ mang được thông tin qua một số bước nhất định do vấn đề triệt tiêu đạo hàm – vanishing gradient [19]. Bộ nhớ dài ngắn hạn (LSTM) được đề xuất bởi Hochreiter và Schmidhuber đã giải quyết được vấn đề này [20] [21].
Cấu trúc của mạng LSTM được thiết kế các ô nhớ cho việc lưu trữ thông tin dài hạn. Quá trình sử dụng và thay đổi thông tin của các thời điểm được quyết định qua ba cổng: cổng vào – input gate; cổng quên – forget gate và cổng ra - output gate.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 2: Tối ưu hoá LSTM cho một đơn vị RNN.

Cho h là ô nhớ LSTM, c là giá trị của ô nhớ và x là dữ liệu đầu vào. Quá trình cập nhật có thể được chia thành các bước sau đây:

(1) Đầu tiên chúng ta tính giá trị từng thành phần cho ô nhớ ở thời điểm hiện tại $\overline{c_t}$, $W_{xc}$ và $W_{hc}$ dựa trên công thức RNN truyền thống, lần lượt là dữ liệu đầu vào và ma trận trọng số của đầu ra của ô nhớ trước đó.

$$\overline{c_t} = tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c).$$

(2) Tính toán giá trị $i_t$ lượng dữ liệu được đi qua cổng đầu vào. Cổng vào dùng để điều khiển lượng giá trị ảnh hưởng đối với trạng thái ô nhớ hiện tại. Việc tính toán tất cả các cổng không chỉ bị ảnh hưởng bởi dữ liệu hiện tại $x_t$ và giá trị đầu ra của ô nhớ liền trước đó $h_{t-1}$ mà còn bởi giá trị của ô nhớ $c_{t-1}$ trước đó. Mô hình này gọi là peephole connections.

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i).$$

(3) Tính toán giá trị của cổng quên $f_t$. Khác với cổng vào, cổng quên điều chỉnh sự ảnh hưởng của thông tin lịch sử với trạng thái hiện tại của ô nhớ.

$$f_t = \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right).$$

(4) Tính giá trị trạng thái của ô nhớ hiện tại $c_t$.

$$c_t = f_t \odot c_{t-1} + i_t \odot \overline{c_t}.$$

$\odot$ được định nghĩa là phép nhân theo từng điểm. – product point by point

Công thức cho thấy trạng thái ô nhớ được cập nhật dựa trên lượng thông tin đưa vào ô nhớ trước đó sẽ loại bỏ đi thông tin không cần thiết và xác định lượng thông tin mới từ giá trị ô nhớ được đưa vào, lần lượt là $c_{t-1}$ và $\overline{c_t}$, và hai phần này được điều chỉnh bởi cổng quên và và cổng vào tương ứng $f_t$ và $i_t$.

(5) Tính giá đầu ra $o_t$, đây là đầu ra để điều chỉnh giá trị trạng thái của đơn vị ô nhớ.

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o).$$

(6) Đầu ra cuối cùng của một đơn vị LSTM.

$$h_t = o_t \odot tanh(c_t).$$

Hàm kích hoạt là hàm Sigmoid chung trên các công thức có giá trị từ khoảng 0 đến 1. Thiết kế của 3 cổng và ô nhớ riêng cho phép đơn vị LSTM lưu trữ, đọc, xóa và cập nhật dù là dài hay ngắn hạn.

## 2.3 Thuật toán di truyền (GA)

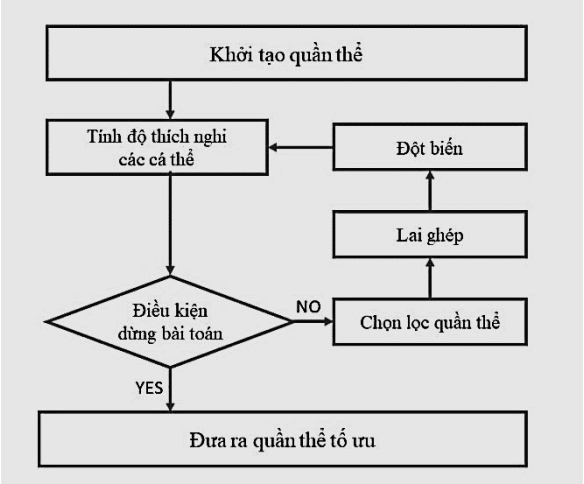
### 2.3.1 Khái niệm

Thuật toán di truyền (GA) là một thuật toán metaheuristic chỉ phương pháp toán nhằm tối ưu hoá vấn đề của bài toán được lấy cảm hứng từ quá trình tiến hoá và di truyền tự nhiên. Thuật toán được sử dụng rộng rãi như một thuật toán tìm kiếm lời giải tối ưu hoặc sấp sỉ tối ưu cho trong một khoảng tìm kiếm lớn. Quá

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

trình di truyền của thuật toán bao gồm những toán tử bắt chước theo quá trình di truyền tự nhiên và nguyên tắc tiến hoá, có liên quan đến hoạt động lai ghép và đột biến.
Ứng với mỗi cá thể tượng trưng cho một lời giải nhất định được mã hoá thành chuỗi nhị phân. Các cá thể sẽ được một cách ngẫu nhiên và dựa vào những toán tử như lai chéo, đột biến để có cơ hội tạo ra những cá thể tốt hơn hay chính là lời giải tốt hơn.

### 2.3.2 Các bước trong giải thuật di truyền




Hình 3: Các bước thực hiện cơ bản của giải thuật di truyền

#### 2.3.2.1 Xác định quần thể ban đầu

Việc khởi tạo quần thể hoàn toàn ngẫu nhiên các cá thể sẽ được mã hoá nhị phân và được khởi tạo ngẫu nhiên, cần xác định rõ quần thể có số lượng cá thể đủ lớn để đảm bảo độ đa dạng của quần thể đó. Và tính chất của mỗi cá thể sẽ được xác định.

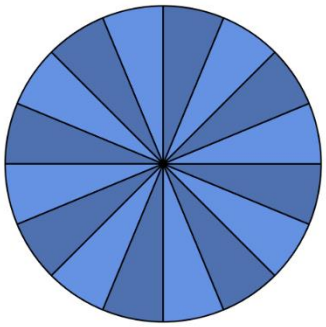
#### 2.3.2.2 Quá trình lai ghép các cá thể

Nguyên tắc chọn theo mô hình bánh xe roulette: Trong việc lựa chọn các cá thể để lai chéo, thuật toán mong muốn cá thể được sẽ đạt được những tiêu chí tốt và tạo ra thế hệ mới tốt hơn. Có nhiều phương pháp để lựa chọn cá thể tốt như là:

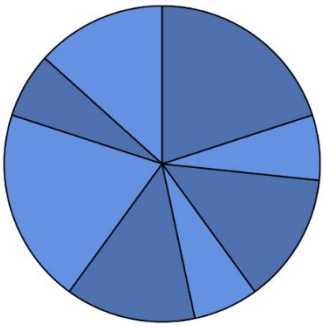
- Chọn theo thứ hạng (rank selection).
- Chọn theo độ ổn định (steady state selection).
- Chọn theo vòng xoay roulette (roulette wheel selection).

Trong đó phương pháp chọn theo bánh xe roulette có những đặc điểm như sau: Các cá thể được chọn bằng cách đánh giá trên độ thích nghi của cá thể đó; những cá thể tốt hơn, thì xác suất được chọn sẽ cao hơn. Với ý tưởng giống vòng xoay roulette, tuy nhiên phương pháp này có những điểm khác biệt quan trọng là các ô trên vòng xoay luôn có kích thước ngang nhau, đồng nghĩa với việc xác suất được chọn lựa như nhau.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 



Hình 4: Bánh xe roulette nguyên bản

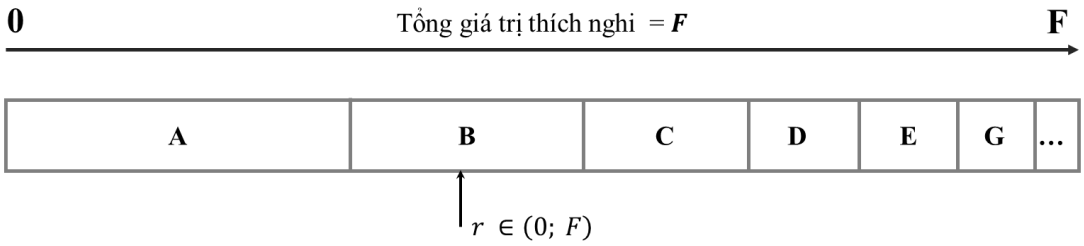


Hình 5: Bánh xe roulette có trọng số

Thay vào đó chung ở mỗi lần chọn cá thể được diễn ra, một phiên vòng xoay roulette có trọng số sẽ được tính toán. Trong đó trọng số là của mỗi cá thể, dựa trên giá trị thích nghi của cá thể đó. Xác suất để cá thể được lựa chọn là:

$$p_i = \frac{f_i}{\sum_{i=1}^{N} f_i}$$

Với $f_i$ là giá trị thích nghi của cá thể; N là số lượng cá thể có trong quần thể đó.




Hình 6: Dàn trải các giá trị trọng số của bánh xe roulette

Mô phỏng quá trình này có được giải thích như sau:
[Sum] tính toán tổng $F$ của tất cả các giá trị thích nghi của cá thể thứ $i$ tương ứng với $f_i$ *với* $i \in (0, N)$.
[Select] chọn ngẫu nhiên giá trị r trong giá trị từ $(0;\ F)$ hay hiểu là $r \in (0; F)$.
[Loop] tiến hành vòng lặp bắt đầu từ vị trí đầu tiên cho đến vị trí cuối cùng của quần thể và lần tính tổng các giá trị thích nghi các cá thể để được giá trị là tổng $s$.
Khi mà tổng $s$ lớn hơn $r;\ s > r$, dừng lại và trả về cá thể đang chọn.

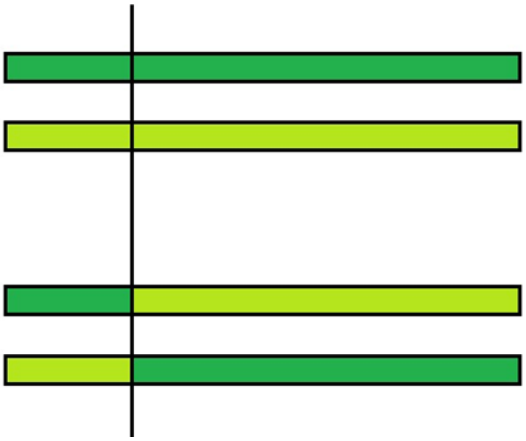
$totalSum := 0$
$pop$ ← the population
**for each** $chromosome \in pop$ **do**
    $fitnessValuation$ ← computeFitness($chromosome$)
    $totalSum := totalSum + fitnessValuation$
rand ← get random number between $(0;\ totalSum)$
$partialSum := 0$
**for each** $chromosome \in pop$ **do**
    $fitnessValuation$ ← computeFitness($chromosome$)
    $partialSum := partialSum + fitnessValuation$
    **if** $partialSum \geq rand$ **then**
        **return** $chromosome$.

Mã giả của mô phỏng quá trình chọn cá thể theo phương pháp vòng xoay roulette.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

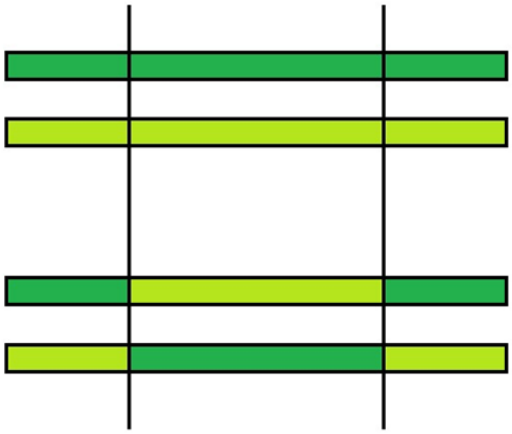
### 2.3.2.3 Lai ghép

Quá trình lai ghép diễn ra giữa 2 cá thể sẽ tiến hành trao đổi các đặc trưng nằm trên đoạn gene của cá thể đó. Có nhiều cách để tiến hành lai ghép, trong đó 3 cách phổ biến để lai ghép là: Ghép đơn điểm (single point crossover); Ghép hai điểm (two point crossover) và Ghép đa điểm ngẫu nhiên (multi-point crossover). Lai ghép đơn điểm, phép lai chọn một vị trí thứ k bất kì trên các cá thể và tiến hành bắt chéo để trao đổi đặc trưng sao cho tạo thành hai cá thể mới có cùng độ dài chuỗi.



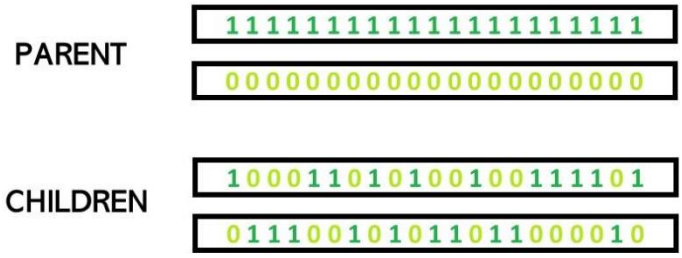
Hình 7**:** Lai ghép đơn điểm

Lai ghép hai điểm, chọn hai vị trí bất kì trên đoạn mã của hai cá thể và tiến hành và tiến hành trao đổi các đặc trưng nằm giữa hai điểm để tạo ra cá thể mới.



Hình 8: Lai ghép hai điểm

Lai ghép đa điểm, mỗi điểm trên cá thể đều có cơ hội ngẫu nhiên để trao đổi chéo với các thể khác.




Hình 9: Lai ghép đa điểm

#### 2.3.2.4 Đột biến thế hệ con

Đột biến thế hệ con giúp cho quần thể đa dạng hơn và giúp cho quần thể không bị trả về một giá trị liên tục ở một số điểm làm việc giải bài toán không có kết quả. Đột biến phải dựa vào tỷ lệ đột biến nhất định có thể thay đổi dựa vào từng bài toán khác nhau vì sự thay đổi mang hướng ngẫu nhiên của mỗi bài toán là khác nhau và phải có sự đột biến để bài toán không bị trả về một giá trị sai ở một vị trí gen liên tục làm cho toàn bộ gen đều sai cùng một vị trí và không thoát ra được.

#### 2.3.2.5 Đánh giá con

Đánh giá cá thể con bằng cách tạo một hàm thích nghi để cá thể phù hợp với mục đích giải thuật. Đánh giá giúp cho ta biết cá thể xem thế hệ mà ta tạo ra đã đúng với mục đích hay không để dừng hoặc tiếp tục chương trình. Và giúp cho ta xem cá thể mới tốt hay tệ hơn cá thể cũ để có thể đưa ra quyết định giữ, thêm hoặc xóa cá thể đó.

## 3 PHƯƠNG PHÁP TIẾP CẬN

### 3.1 Mô tả thuật toán thực hiện

Để xây dựng mô hình dự đoán giá cổ phiếu GA và LSTM quá trình sẽ được chia thành hai giai đoạn chính. Giai đoạn đầu tiên sử dụng GA để sắp xếp các yếu tố quan trọng theo các bước sau đây:
(i) Mã hoá nhị phân các cá thể là cơ sở để xây dựng một quần thể, quá trình khởi tạo cá thể được diễn ra một cách ngẫu nhiên, được định nghĩa như sau:

$$POP = \begin{bmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,k} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,k} \\ \cdots & \cdots & \cdots & \cdots \\ c_{m,1} & c_{m,2} & \cdots & c_{m,k} \end{bmatrix}$$

Trong đó POP là ma trận quần thể chứa các cá thể, mỗi dòng đại diện cho 01 cá thể tức là một bộ các đặc trưng được chọn. Độ dài của của một cá thể là k đại diện cho tổng số của các đặc trưng hiện có, số m đại diện cho kích thước quần thể. Trong đó ứng với mỗi ô giá trị $c_{i,j}$ nhận giá trị 0 hoặc 1, với 1 đại diện cho đặc trưng được tham gia, 0 đại diện cho đặc trưng không được lựa chọn.
(ii) Sử dụng phương pháp chọn vòng xoay roulette. Trong quá trình thực hiện thuật toán di truyền sẽ cập nhật tỉ lệ giá trị thích nghi của các cá thể. Và cập nhật liên tục các trọng số hay xác suất được chọn của toàn bộ quần thể trong suốt vòng đời của một thế hệ. Độ thích nghi của một bộ đặc trưng được chọn là hệ số xác định R-squared định nghĩa như sau:

$$r^2 = 1 - \frac{\Sigma(y - \hat{y})^2}{\Sigma(y - \overline{y})^2}$$

Trong đó, hệ số xác định được biểu diễn bởi $r^2$, y là giá trị thực tế, $\hat{y}$ là giá trị mà mô hình dự đoán, $\overline{y}$ là trung bình giá trị thực, và khoảng giá trị của $r^2$ là $[0,1]$. Hệ số xác định càng lớn đồng nghĩa với việc khả năng mà cá thể được di truyền càng cao, đóng vai trò quan trọng trong việc lai ghép và đột biến trong quá trình di truyền.
(iii) Phương pháp đa điểm ngẫu nhiên được lựa chọn cho quá trình lai ghép, các đặc trưng trên mỗi cặp cá thể được chọn sẽ đổi chỗ cho nhau. Trong quá trình thuật toán xảy ra sẽ sinh ra một giá trị xác suất ngẫu nhiên nếu như giá trị này nhỏ hơn xác suất lai ghép được đặt ra trước đó, thì việc trao đổi gen được diễn ra.
(iv) Đột biến nhị phân được áp dụng trong quá trình diễn ra thuật toán. Tuy nhiên mức độ đột biến sẽ được ước định bằng một xác suất nhỏ, để có đảm bảo quần thể không bị biến động nhiều. Để xác định xem đặc trưng nào trên quần thể sẽ đột biến, sinh ngẫu nhiên một giá trị xác suất nếu giá trị này nhỏ hơn xác suất đột biến được định sẵn thì đặc trưng đó sẽ thay đổi trạng thái được chọn từ 1 thành 0 hoặc ngược lại.
Tiến hành lặp lại các bước (ii) đến (iv) với số lần lặp được xác định sẵn. Kết thúc quá trình di truyền, ta sẽ nhận được quần thể có 100 cá thể chứa những đặc trưng tối ưu nhất. Sau đó xác định mức độ quan trọng của các đặc trưng bằng cách thống kê và xếp hạng lại số lần xuất hiện của đặc trưng đó trong quần thể. Đặc trưng càng xuất hiện nhiều lần được xem là càng quan trọng.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Giai đoạn thứ hai của tối ưu hoá bằng cách lựa chọn các đặc trưng quan trọng cho mô hình LSTM cho mô hình dự đoán cổ phiếu. Dựa vào xếp hạng các đặc trưng quan trọng đã được đánh giá, ta lựa chọn bộ 5, 10, 20, 30 và 40 đặc trưng để đưa vào mô hình. Bằng việc đánh giá kết quả dự đoán của mô hình, ta sẽ chọn được những đặc trưng quan trọng, và những đặc trưng tối ưu hoá được mô hình dựa trên việc so sánh dựa trên đường baseline giữa các mô hình với nhau để đề xuất được một mô hình tối ưu và cải thiện độ chính xác. Và để đánh giá độ chính xác của mô hình LSTM sẽ dựa trên 3 chỉ số đánh giá chính:
Sai số toàn phương trung bình (MSE), độ lỗi tuyệt đối trung bình (MAE) và độ lỗi phần trăm tuyệt đối trung bình (MAPE).
Sai số toàn phương trung bình (Mean Squared Error)

$$MSE = \frac{1}{n}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2$$

Sai số tuyệt đối trung bình (Mean Absolute Error)

$$MAE = \frac{\sum_{i=1}^{m}|y_i - \hat{y}_i|}{n}$$

Độ lỗi phần trăm trung bình tuyệt đối (Mean Absolute Percentage Error)

$$MAPE = \frac{\sum_{i=1}^{m}|(y_i - \hat{y}_i)/y_i|}{m}$$

Trong đó m là số lượng được lấy mẫu, $y_i$ là giá cổ phiếu thực tế, $\hat{y}_i$ là giá cổ phiếu do mô hình dự đoán. Dựa vào các chỉ số này ta có thể đánh giá được độ chính xác của mô hình sau khi lựa chọn được bộ các đặc trưng hợp lý.

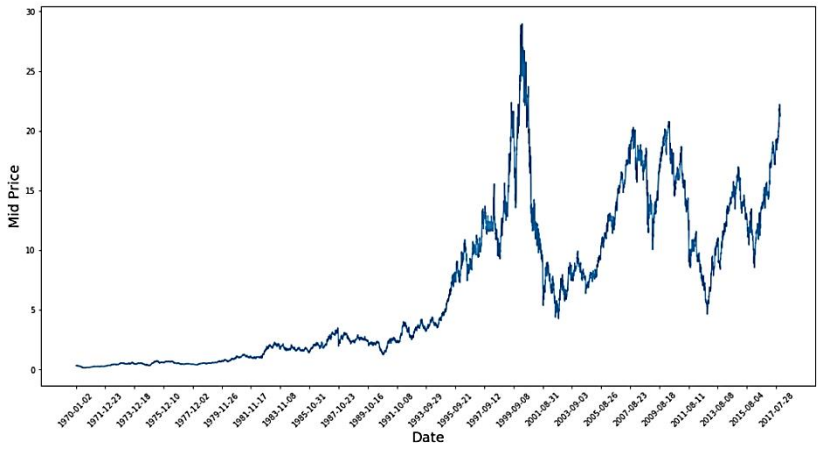
## 3.2 Thực nghiệm với dữ liệu cụ thể

Dữ liệu thực nghiệm được lấy từ dataset "Huge Stock Market Dataset" - Historical daily prices and volumes of all U.S. stocks and ETFs của Kaggle. Dữ liệu cung cấp đầy đủ về khối lượng và giá hàng ngày cho tất cả các cổ phiếu và quỹ ETF có trụ sở tại Hoa Kỳ đang giao dịch trên NYSE, NASDAQ và NYSE MKT (đây là một trong những bộ dữ liệu tốt nhất thuộc loại này). Dữ liệu cập nhật lần cuối vào ngày 10/11/2017 và được trình bày ở định dạng CSV gồm: Date, Open, High, Low, Close, Volume, OpenInt.
Dữ liệu sau khi sort theo ngày sẽ có dạng như sau:

| | Date | Open | High | Low | Close |
|---|---|---|---|---|---|
| 0 | 1970-01-02 | 0.30627 | 0.30627 | 0.30627 | 0.30627 |
| 1 | 1970-01-05 | 0.30627 | 0.31768 | 0.30627 | 0.31385 |
| 2 | 1970-01-06 | 0.31385 | 0.31385 | 0.30996 | 0.30996 |
| 3 | 1970-01-07 | 0.31385 | 0.31385 | 0.31385 | 0.31385 |
| 4 | 1970-01-08 | 0.31385 | 0.31768 | 0.31385 | 0.31385 |

Hình 10: Top 5 của dataset được sắp xếp tăng dần theo ngày

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

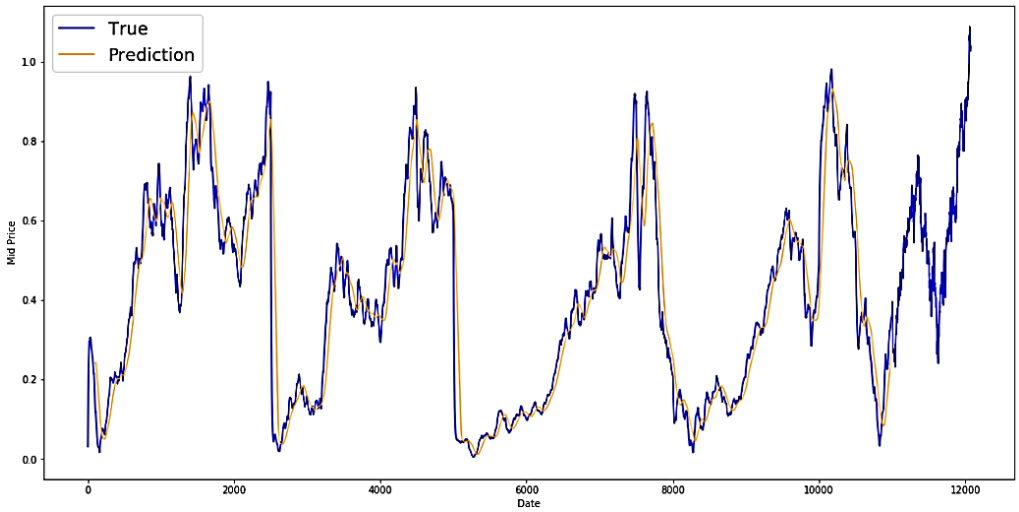



Hình 11: Dữ liệu được visualize dựa theo Mid Price

Tiếp theo, ta tính Mid price dựa trên giá thấp nhất và cao nhất của mỗi ngày rồi chia thành tập train (gồm 11 ngàn data point đầu tiên) và tập test (phần còn lại).
Ta chuẩn hóa lại dữ liệu bằng hàm MinMaxScalar() để đưa tất cả về khoảng (0;1). Ta có thể làm mịn dữ liệu bằng cách sử dụng exponential moving average. Điều này giúp hạn chế tính rời rạc vốn có của dữ liệu về giá cổ phiếu và tạo ra một đường cong mượt hơn.

```
EMA = 0.0
gamma = 0.1
for ti in range(11000):
  EMA = gamma*train_data[ti] + (1-gamma)*EMA
  train_data[ti] = EMA
all_mid_data = np.concatenate([train_data,test_data],axis=0)
```

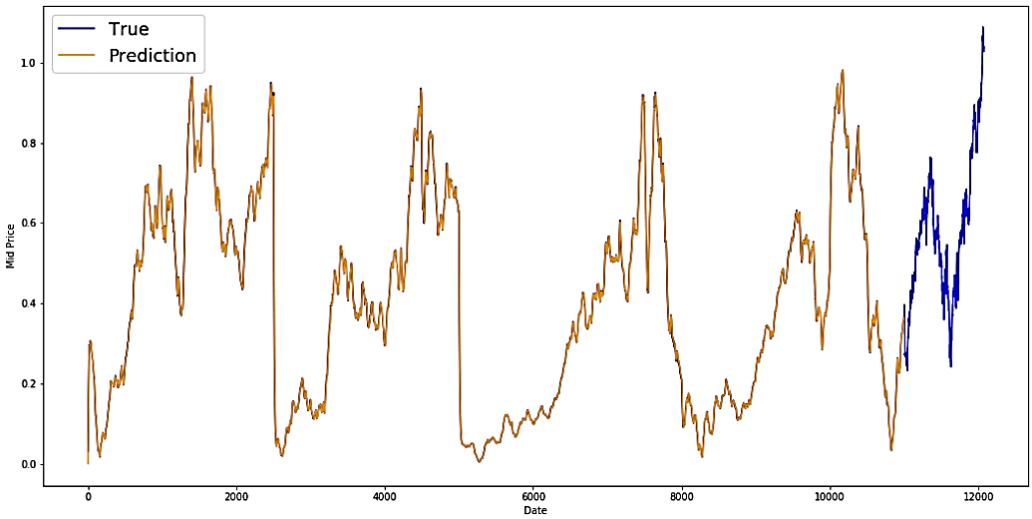
Tiếp theo, ta dùng MSE – sai số toàn phương đã trình bày ở mục trước để dự đoán giá cổ phiếu của ngày tiếp theo dựa trên 100 ngày trước đó. Do chỉ cần dự đoán thêm 1 ngày tiếp theo nên kết quả đạt được có độ chính xác cao (99%) và thu được kết quả như bên dưới:




Hình 12: Biểu đồ so sánh giữa việc dự đoán và thực tế khi dùng MSE

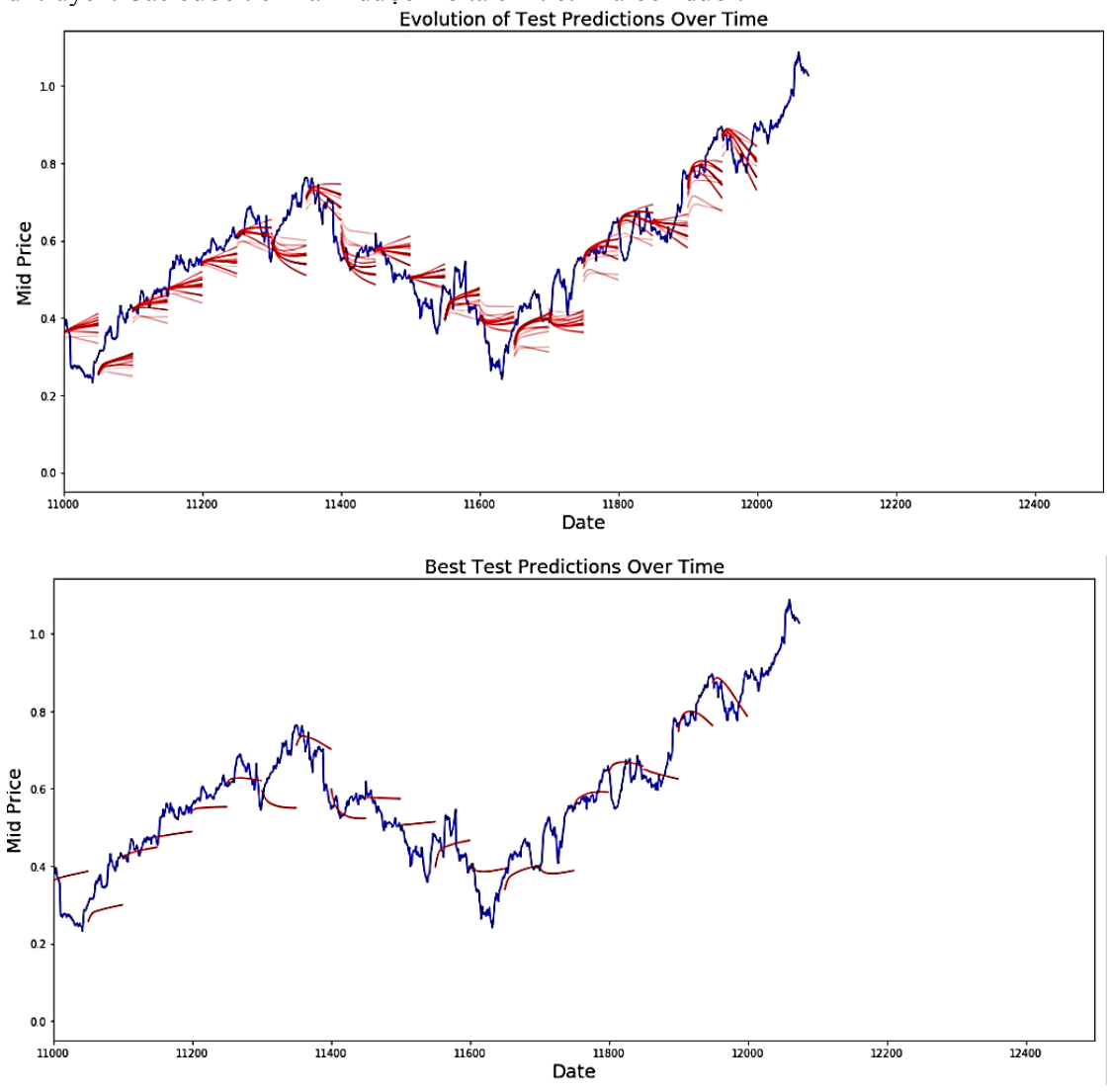
Ta tiếp tục có thể cải tiến bằng cách sử dụng EMA (Exponential Moving Average) đã đề cập ở trên và thu được kết quả chính xác cao hơn nữa:

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 13: Biểu đồ so sánh giữa việc dự đoán và thực tế khi dùng MSE

Tuy nhiên, nếu muốn dự đoán xa hơn thì cần dùng đến yếu tố time series của kỹ thuật LSTM kết hợp giải thuật di truyền. Các bước tiến hành được mô tả chi tiết như bên dưới.




Hình 14: Biểu đồ mô tả tính hiệu quả của mô hình dựa trên tập dữ liệu test

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

- Bước 1: sử dụng GA để chọn ra các đặc trưng phù hợp nhất cho mô hình như đã mô tả ở mục 3.1 (tuy có nhiều thuộc tính nhưng ta chỉ chọn ra thuộc tính quan trọng là giá cao nhất, thấp nhất).
- Bước 2: data generator, batch theo input data nhận tuần tự và output batch với số lượng tương ứng.
- Bước 3: data augmentation, chọn ngẫu nhiên một sample output với kích thước nhỏ.
- Bước 4: defining hyperparameters & input + output, đây cũng là các bước tiền xử lý dữ liệu.
- Bước 5: defining parameters of the LSTM and regression layer, ta có 3 layer của LSTM và 1 layer của mô hình hồi quy.
- Bước 6: calculating LSTM output and feeding it to the regression layer to get final prediction, ta tạo các biến TensorFlow để giữ các trạng thái ẩn của LSTM cell, từ input của tập train, ta tính ra output của hàm RNN trong mô hình (có chia output ra thành danh sách các tensors).
- Bước 7: loss calculation and optimizer, ta dùng MSE và tính tổng bình phương sai số, sau đó định nghĩa một optimizer dùng cho mô hình neural.
- Bước 8: prediction related calculations, sử dụng các hàm đã xây dựng để đưa các output từ RNN vào bước hồi quy.
- Bước 9: running LSTM & visualizing the predictions, chạy mô hình và hiển thị kết quả dự đoán được. Từ đó rút ra đánh giá kết luận.

# 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 4.1 Kết luận

Ngày nay, các ứng dụng của các mô hình học sâu đang được ứng dụng rộng rãi và thu được nhiều thành công, và GA như một sử dụng nhiều hơn trong việc tìm kiếm các bộ tham số xấp xỉ tối ưu. Sử dụng GA, chúng ta có thể xác định được các đặc trưng phù hợp nhất của từng mô hình, chọn lọc các dữ liệu tiềm năng từ tập dữ liệu để việc huấn luyện đạt hiệu quả. Để nâng cao khả năng dự đoán chính xác của mô hình. Các tham số của GA (số lượng cá thể, mức độ đột biến, mức độ lai ghép, số thế hệ di truyền, quá trình mã hoá và các định nghĩa một hàm mục tiêu tốt hơn), trọng số, sai số của LSTM cần được thiết lập chính xác và phù hợp với từng loại mô hình. Và không kém phần quan trọng là nguồn dữ liệu lớn và đáng tin cậy dùng để huấn luyện mô hình. Một số hạn chế của mô hình:

(i) Độ phức tạp tính toán lớn: Mô hình GA và LSTM đòi hỏi độ tính toán đáng kể, đặt biệt là khi số lượng cá thể trong quần thể tăng lên, quá trình tìm kiếm và tối ưu bằng GA sẽ mất rất nhiều thời gian và tài nguyên để tính toán. Song việc đánh đổi này là xứng đáng tìm ra được những bộ đặc trưng quan trọng cho việc đưa vào mô hình dự đoán.

(ii) Việc lựa chọn tham số phức tạp: Mô hình dự đoán cổ phiếu này có rất nhiều tham số để lựa chọn và cấu hình đúng để đạt được hiệu suất cao, song cần phải có những nghiên cứu sâu hơn để lựa chọn những tham số thích hợp mà không ảnh hưởng quá nhiều đến độ chính xác của mô hình.

(iii) Việc mô hình dự đoán giá cổ phiếu chỉ mang tính chất tham khảo.

## 4.2 Hướng phát triển

Qua các nghiên cứu trên hoàn toàn cho thấy tiềm năng to lớn của GA – LSTM mang lại trong các vấn đề thực tiễn nói chung là dự đoán giá cổ phiếu nói riêng. Tuy nhiên, để mô hình hoàn thiện và đạt hiệu quả cao cần có các nghiên cứu chuyên sâu, các giải pháp tối ưu hơn nữa trong tương lai.

# TÀI LIỆU THAM KHẢO

[1]. Eugene F. Fama, Kenneth R. French, "Common risk factors in the returns on stocks and bonds," Journal of Financial Economics, vol. 33, no. 1, pp. 3-56, 1993.

[2]. G. Gyözö, "Using News Articles to Predict Stock Price Movements," 2001.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

[3]. H. El-Chaarani, "The Impact of Oil Prices on Stocks Markets: New Evidence During and After the Arab Spring in Gulf Cooperation Council Economies," International Journal of Energy Economics and Policy, vol. 9, no. 4, pp. 1-26, 2019.

[4]. H. El-Chaarani, " Exploring the Impact of Emotional Intelligence on Portfolio Performance," Humanomics, vol. 32, no. 4, pp. 1-28, 2016.

[5]. Nicholas Bowden, James E. Payne, "Short term forecasting of electricity prices for MISO hubs: Evidence from ARIMA-EGARCH models," Energy Economics, vol. 30, no. 6, pp. 3186-3197, 2008.

[6]. Binoy.B.Nair, V.P Mohandas, N. R. Sakthive, "A decision tree-rough set hybrid system for stock market trend prediction," International Journal of Computer Applications, vol. 6, pp. 1-6, 2010.

[7]. Feng Li, Cheng Liu, "Application Study of BP Neural Network on Stock Market Prediction," 2009 Ninth International Conference on Hybrid Intelligent Systems, 2009.

[8]. L Cao, Franklin R Tay, "Support vector machine with adaptive parameters in financial time series forecasting," IEEE Transactions on Neural Networks, vol. 14, no. 6, pp. 1506-1518, 2003.

[9]. Marta R Costa-jussà, Alexandre Allauzen, Loïc Barrault, Kyunghun Cho, Holger Schwenk, "Introduction to the special issue on deep learning approaches for machine translation," Computer Speech & Language, vol. 46, pp. 367-373, 2017.

[10]. Haytham M Fayek, Margaret Lech, Lawrence Cavedon, "Evaluating deep learning architectures for Speech Emotion Recognition," Neural Networks, vol. 92, pp. 60-68, 2017.

[11]. Junliang Xing, Kai Li, Weiming Hu, Chunfeng Yuan, Haibin Ling, "Diagnosing deep learning models for high accuracy age estimation from a single image," Pattern Recognition, vol. 66, pp. 106-116, 2017.

[12]. Lv Dongdong, Dong Wang, Meizi Li, Yang Xiang, "DNN models based on dimensionality reduction for stock trading," Intelligent Data Analysis, vol. 24, pp. 19-45, 2020.

[13]. Avraam Tsantekidis, Nikolaos Passalis, Anastasios Tefas, Juho Kanniainen, Moncef Gabbouj, Alexandros Iosifidis, "Forecasting Stock Prices from the Limit Order Book Using Convolutional Neural Networks," IEEE 19th Conference on Business Informatics (CBI), pp. 10-15, 2017.

[14]. Sreelekshmy Selvin, R. Vinayakumar, E A Gopalakrishnan, Vijay Krishna Menon, K P Soman, "Stock price prediction using LSTM, RNN and CNN-sliding window model," International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 234-239, 2017.

[15]. Umair F Siddiqi, Murat Uysal, Okyay Kaynak, "Genetic Algorithm for the Mutual Information-Based Feature Selection in Univariate Time Series Data," IEEE Access, vol. 8, pp. 9597 - 9609, 2020.

[16]. Jun Hu, Wendong Zheng, "Multistage attention network for multivariate time series prediction," Neurocomputing, vol. 383, pp. 122-137, 2020.

[17]. Minal P. Bharambe, S.C.Dharmadhikari, "Stock Market Analysis Based on Artificial Neural Network with Big data," Proceedings of 8th post graduate conference for information technology., pp. 1-5, 2015.

[18]. Usama M. Fayyad and Gregory Piatetsky-Shapiro, Padhraic Smyth, "The KDD process for extracting useful knowledge from volumes of data," Commun. ACM, vol. 39, pp. 27-34, 1996.

[19]. D H Wang, Jifei Fan, Hanliang Fu, Bing Zhang, "Research on Optimization of Big Data Construction Engineering Quality Management Based on RNN-LSTM," Complexity Problems Handled by Big Data Technology, vol. 15, pp. 15-20, 2018.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

[20]. Guimin Jia, Yu-Jun Lu, Wei-Bing Lu, Yihua Shi, Jinfeng Yang, "Verification method for Chinese aviation radiotelephony readbacks based on LSTM-RNN," Electronics Letters, vol. 53, no. 6, pp. 401-403, 2017.
[21]. Zhijian Liu, Hao Li, Kejun Liu, Han-Cheng Yu, Kewei Cheng, "Design of high-performance water-in-glass evacuated tube solar water heaters by a high-throughput screening based on machine learning: A combined modeling and experimental study," Solar Energy, vol. 142, pp. 61-67, 2017.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F014**

# T&T HEALTHCARE: CÔNG CỤ HỖ TRỢ SỨC KHỎE THÔNG MINH CHO BỆNH NHÂN ĐƯỜNG HUYẾT DỰA VÀO TRÍ TUỆ NHÂN TẠO

TÔN LONG PHƯỚC[1*], LÊ TUẤN[1], TĂNG BẢO TRẤN[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
**tonlongphuoc@iuh.edu.vn*

**Tóm tắt.** Ngày nay, việc chăm sóc sức khỏe là ưu tiên hàng đầu của mọi người. Theo Bộ Y tế, các bệnh liên quan đến đường huyết, huyết áp là nguyên nhân gây tử vong hàng đầu và làm tăng nguy cơ mắc các bệnh khác. Cụ thể, tình trạng bệnh đái tháo đường đang có xu hướng gia tăng ở Việt Nam, hiện nay có khoảng 5 triệu người mắc bệnh đái tháo đường. Trong khi đó, tỷ lệ người mắc bệnh cao huyết áp ở Việt Nam cũng khá cao, ước tính khoảng 20-25% dân số. Do đó, việc phát triển một ứng dụng chăm sóc sức khỏe thông minh giúp người dùng quản lý các chỉ số sức khỏe quan trọng như BMI, đường huyết, huyết áp là điều cần thiết. T&T HEALTHCARE là công cụ được chúng tôi khuyên dùng với các chức năng hỗ trợ bệnh nhân và bác sĩ quản lý sức khỏe bệnh nhân đường huyết. Nó hỗ trợ nhận dạng giọng nói thành văn bản thay vì nhập dữ liệu, tự động đưa ra cảnh báo cho người dùng nếu sức khỏe có dấu hiệu xấu đi. Ngoài ra, người dùng sẽ được cung cấp lời khuyên cá nhân hóa thông qua dịch vụ chatbot GPT. Người dùng cũng có thể truy cập mạng lưới các chuyên gia Y tế để nhận hỗ trợ và tư vấn về sức khỏe. Qua đó, người dùng có thể biết được tình trạng sức khỏe của mình để điều trị kịp thời, tránh để quá muộn. Hiện tại công cụ đã dùng cho một số bệnh nhân ở Gò Vấp, Tp.HCM. Trong tương lai, ứng dụng sẽ phát triển thêm nhiều tính năng và dịch vụ mới nhằm mang đến cho người dùng nhiều tiện ích và hỗ trợ toàn diện hơn trong việc chăm sóc sức khỏe.

**Từ khóa.** Chăm sóc sức khỏe, Chat GPT, Đọc văn bản, ZegoCloud, MongoDB.

## T&T HEALTHCARE: An INTELLIGENT HEALTHCARE SUPPORT TOOL for BLOOD SUGAR PATIENTS BASED on ARTIFICIAL INTELLIGENCE

**Abstract.** Nowadays, taking care of health is everyone's top priority. According to the Ministry of Health, diseases related to blood sugar and blood pressure are the leading causes of death and increase the risk of other diseases. Specifically, the state of diabetes is on the rise, about 5 million people have diabetes in Vietnam today. Meanwhile, the rate of people with high blood pressure in Vietnam is also quite high, estimated at 20-25% of the population. Therefore, it is necessary to develop a smart healthcare application that will help users manage important health indicators such as BMI, blood sugar and blood pressure. T&T HEALTHCARE is our recommended tool with functions to support patients and doctors for managing blood sugar patients' health. It supports speech to text recognition instead of data input, automatically give warnings to the user if the health is showing signs of deterioration. In addition, users will be provided with personalized advice through the GPT chatbot service. Users can also access a network of Health professionals to receive health support and advice. Thereby, users can know their health status for timely treatment, avoiding too late. Currently, the tool has been used for some patients in Go Vap. In the future, the application will develop more new features and services to provide users with more convenience and comprehensive support in healthcare.

**Keywords.** Health care, chatbot GPT, Speed to text, ZegoCloud, MongoDB

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

# 1 GIỚI THIỆU TỔNG QUAN

Hiện nay, trong lĩnh vực chăm sóc sức khỏe, việc theo dõi và quản lý bệnh nhân đường huyết và huyết áp là vô cùng quan trọng để đảm bảo sự ổn định của các chỉ số sức khỏe này và ngăn ngừa các biến chứng có thể gây hại. Tuy nhiên, việc theo dõi thủ công và giám sát liên tục trở nên khó khăn và tốn thời gian đối với cả bệnh nhân và nhóm chăm sóc y tế. Để giải quyết vấn đề này, cần phát triển một hệ thống khám và cảnh báo thông minh, giúp tự động hóa quá trình theo dõi và cung cấp các cảnh báo kịp thời khi có sự thay đổi không bình thường về các chỉ số đường huyết và huyết áp của bệnh nhân. Vì vậy, chúng tôi đề xuất ứng dụng có tên T&T HEALTHCARE, với 2 nền tảng là website và ứng dụng di động nhằm khắc phục những vấn đề trên. T&T HEALTHCARE giúp hỗ trợ cho bệnh nhân và bác sĩ quản lý các chỉ số sức khỏe như đường huyết và huyết áp, được cung cấp lời khuyên cá nhân hóa thông qua dịch vụ chatbot GPT có hỗ trợ cả việc nhận dạng giọng nói thay vì nhập dữ liệu, người dùng có thể đăng ký khám bệnh trực tuyến, có thể chọn bác sĩ theo mong muốn cá nhân. Ngoài ra, ứng dụng này kết hợp các công nghệ và thuật toán để thu thập, phân tích, xử lý và theo dõi tự động các chỉ số đường huyết và huyết áp, có thể sử dụng các thiết bị y tế khác để đo lường các chỉ số sức khỏe như đường huyết và huyết áp. Dữ liệu này sẽ được truyền đến ứng dụng sau khi người dùng nhập vào các chỉ số và được xử lý bởi các thuật toán thông minh tích hợp. Ứng dụng cung cấp giao diện trực quan bằng các biểu đồ hình ảnh và thân thiện cho người dùng, cho phép bệnh nhân dễ dàng nhập liệu và theo dõi sự thay đổi của các chỉ số sức khỏe của mình theo thời gian.

Tiếp theo, chúng tôi đề xuất hướng giải quyết cho vấn đề trên là T&T HEALTHCARE không chỉ cung cấp khả năng theo dõi tự động, mà còn hỗ trợ cảnh báo thông minh. Dựa trên các nguyên tắc như: thu thập dữ liệu các chỉ số về loại bệnh, từ đó chúng tôi so sánh với các khoảng giá trị của bệnh đường huyết hay huyết áp, sau khi qua quá trình xử lý thì hệ thống lập tức đưa ra các cảnh báo sức khỏe mà bệnh nhân đang mắc phải. Khi phát hiện các giá trị đường huyết hoặc huyết áp bất thường, ứng dụng sẽ tự động cảnh báo cho bệnh nhân và bác sĩ. Điều này, đảm bảo rằng các biến chứng nguy hiểm như tăng đột ngột đường huyết hay huyết áp cao không bị bỏ qua và có thể được xử lý kịp thời. Hơn nữa, T&T HEALTHCARE còn hỗ trợ các tính năng như lưu trữ dữ liệu sức khỏe, theo dõi tiến trình điều trị, lịch sử khám. Điều này, giúp bệnh nhân và bác sĩ có cái nhìn toàn diện, chi tiết về tình trạng sức khỏe và cải thiện chất lượng cuộc sống của nhóm bệnh nhân đường huyết và huyết áp. Với khả năng theo dõi tự động và cảnh báo kịp thời, hệ thống này hứa hẹn giúp cải thiện quản lý và chăm sóc sức khỏe, đồng thời giảm thiểu các rủi ro và tăng cường sự an toàn cho bệnh nhân dựa trên bảng đánh giá các chỉ số.

Có thể thấy, hệ thống chăm sóc sức khỏe thông minh hiện nay là cần thiết và rất quan trọng. Nó giúp chúng ta theo dõi, chăm sóc sức khỏe và có thể điều trị ngay tại nhà, điều này sẽ giảm số lượng người đến bệnh viện, tiết kiệm được chi phí đi lại, thời gian, … nhưng vẫn đảm bảo được chất lượng và hiệu quả tốt. Hiện tại, chúng ta có thể thấy một số hệ thống chăm sóc sức khỏe thông minh như: VAIPE, Elfie, … VAIPE là một hệ thống theo dõi và hỗ trợ chăm sóc sức khỏe thông minh cho người Việt, hệ thống cũng tích hợp các công nghệ tiên tiến trong lĩnh vực AI, phân tích dữ liệu. Hệ thống sẽ thu thập, quản lý, phân tích dữ liệu về các chỉ số sức khỏe của bệnh nhân như: thói quen sử dụng thuốc, nhịp tim, huyết áp, BMI. Từ đó, hệ thống có thể đưa ra các khuyến cáo về sử dụng thuốc, cảnh báo uống nhầm thuốc, chuẩn đoán sớm các bệnh lý.

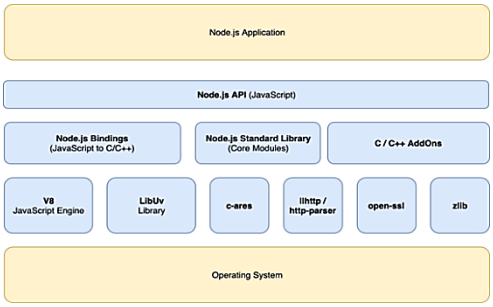
Bài báo của chúng tôi chia làm 5 phần: Phần 1 Giới thiệu tổng quan về hệ thống; Phần 2 Các công cụ liên quan; Phần 3 Công cụ T&T HEALTHCARE; Phần 4 Kiểm nghiệm thực tế và Phần 5 Kết luận và hướng phát triển. Trong đó Phần 3 là phần đóng góp chính của bài báo.

# 2 CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 NodeJS

Node.js: là một nền tảng lập trình được xây dựng trên động cơ JavaScript V8 của Google, cho phép viết mã JavaScript trên phía máy chủ, không chỉ trên trình duyệt [1]. Với Node.js, các lập trình viên có thể viết các ứng dụng back-end, bao gồm các ứng dụng Web, ứng dụng dòng lệnh và các công cụ hệ thống. Node.js được phát triển và duy trì bởi OpenJS Foundation, với mục tiêu tạo ra một nền tảng phát triển được xây dựng trên cơ sở mã nguồn mở, linh hoạt và hiệu quả cao cho các ứng dụng Web hiện đại. Một số ưu điểm của Node.js bao gồm khả năng xử lý đồng thời, hiệu suất cao và khả năng mở rộng dễ dàng. Node.js cũng được hỗ trợ bởi một số thư viện và framework như Express, Nest.js và Socket.IO, giúp cho việc phát triển ứng dụng Web trở nên dễ dàng hơn [2, 3].

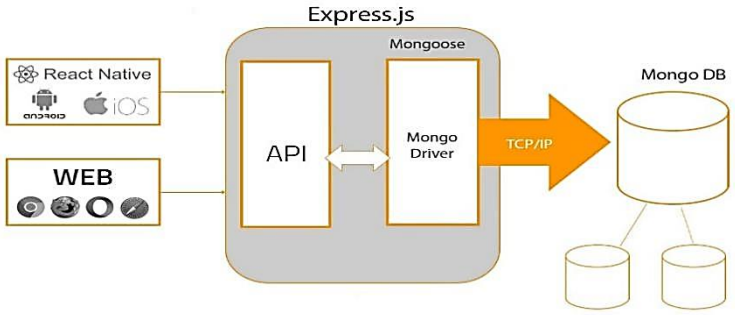
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 1: Kiến trúc của NodeJS [1]

## 2.2 ExpressJS Framework

ExpressJS: là một framework backend cho Node.js, giúp lập trình viên xây dựng các ứng dụng Web và API một cách nhanh chóng và dễ dàng. Đây là một trong những framework phổ biến nhất cho Node.js [4].

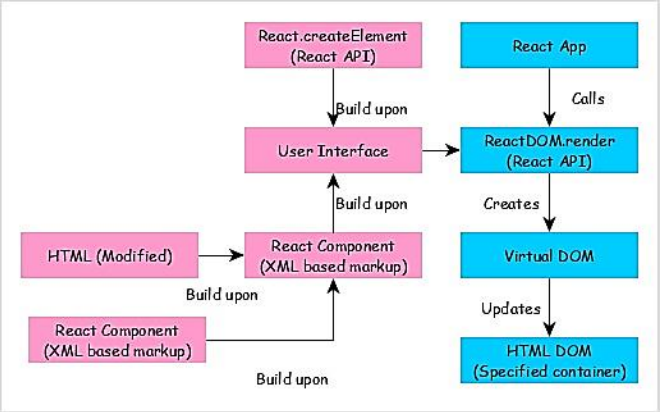



Hình 2: ExpressJS được dùng là backend cho hệ thống [4]

**Ưu điểm của ExpressJS:** Tính năng linh hoạt: ExpressJS cho phép lập trình viên tuỳ biến các tính năng và middleware để phù hợp với yêu cầu của ứng dụng; Tốc độ và hiệu suất: Với việc xây dựng trên Node.js, ExpressJS có thể xử lý các yêu cầu nhanh hơn so với các framework khác; Hỗ trợ routing: ExpressJS cung cấp router để quản lý các định tuyến trong ứng dụng; Cộng đồng lớn: ExpressJS có cộng đồng lớn, đầy đủ tài liệu hướng dẫn và hỗ trợ từ các lập trình viên khác [5].
**Nhược điểm của ExpressJS:** Thiếu tính năng mặc định: ExpressJS chỉ cung cấp các tính năng cơ bản, các tính năng phức tạp hơn phải được xây dựng hoặc cài đặt thông qua các middleware hoặc các module bên thứ ba**;** Có thể phức tạp cho phần quản lý: Với các ứng dụng lớn, ExpressJS có thể trở nên phức tạp và khó khăn trong việc quản lý middleware; Bảo mật: ExpressJS không cung cấp tính năng bảo mật nào mặc định và việc bảo mật ứng dụng phụ thuộc vào khả năng của lập trình viên [5].

## 2.3 ReactJS

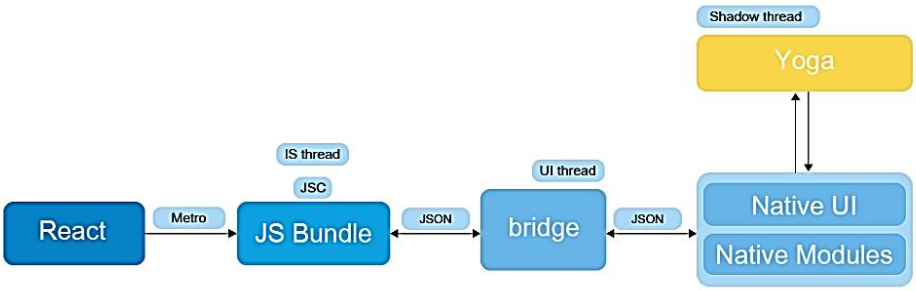
ReactJS: là một framework mã nguồn mở được phát triển bởi Facebook, nhưng được sử dụng để xây dựng các ứng dụng Web chạy trên trình duyệt [6].
**Ưu điểm của ReactJS:** Component-based: ReactJS sử dụng cấu trúc component, giúp tách biệt các thành phần trong ứng dụng và dễ dàng tái sử dụng code; Virtual DOM: ReactJS sử dụng Virtual DOM, giúp cải thiện hiệu suất bằng cách chỉ cập nhật các phần thay đổi trong DOM thay vì toàn bộ; Codebase chia sẻ: ReactJS cũng cho phép chia sẻ codebase giữa các ứng dụng Web và mobile, giảm thời gian và chi phí phát triển; Cộng đồng lớn: ReactJS có cộng đồng lớn, hỗ trợ và cập nhật liên tục.
**Nhược điểm của ReactJS:** Không đầy đủ tính năng: ReactJS chỉ là một phần trong một ứng dụng Web và không cung cấp đầy đủ tính năng như các framework khác như Angular hoặc Vue.js; Khó học: vì ReactJS sử dụng JSX để tạo giao diện, việc học ReactJS đòi hỏi sự hiểu biết về cả HTML và Javascript; Khó khăn trong việc quản lý state: với các ứng dụng lớn, việc quản lý state trong ReactJS có thể trở nên khó khăn và phức tạp.




Hình 3: Quy trình mà ReactJS hiện thực một phần tử [6]

## 2.4 React native

React Native: là một framework mã nguồn mở được phát triển bởi Facebook, cho phép người lập trình xây dựng các ứng dụng di động đa nền tảng bằng việc sử dụng ngôn ngữ lập trình JavaScript.



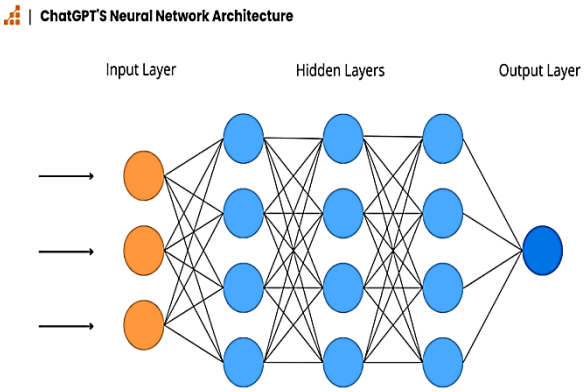

Hình 4: Kiến trúc của React Native [7]

**Ưu điểm của React native:** Cross-platform: cho phép phát triển ứng dụng đa nền tảng với một lần viết mã; Hiệu suất cao: React Native sử dụng native components để tạo giao diện người dùng, giúp ứng dụng có hiệu suất tốt hơn so với các ứng dụng hybrid; Codebase chia sẻ: React Native cho phép chia sẻ codebase giữa ứng dụng di động và Web, giảm thời gian và chi phí phát triển; Cộng đồng lớn: React Native có cộng đồng lớn, hỗ trợ và cập nhật liên tục [7].
**Nhược điểm của React native:** Thư viện bị giới hạn: một số thư viện và chức năng của React Native không được hỗ trợ hoặc cập nhật tốt như trên các nền tảng khác; Khó khăn trong debugging: vì ứng dụng được xây dựng trên JavaScript, debugging có thể trở nên khó khăn hơn so với các ứng dụng native; Độ phức tạp của ứng dụng: khi phát triển các ứng dụng lớn, React Native có thể trở nên phức tạp và khó khăn trong việc quản lý [7].

## 2.5 ChatGPT

ChatGPT là một mô hình ngôn ngữ tự nhiên lớn được đào tạo bởi OpenAI, dựa trên kiến trúc GPT-3.5. Nó được thiết kế để trả lời các câu hỏi và tương tác với con người thông qua các cuộc trò chuyện tự động trên các nền tảng trò chuyện trực tuyến. ChatGPT sử dụng trí tuệ nhân tạo và học sâu để phân tích và đưa ra câu trả lời chính xác và thông minh cho các câu hỏi và vấn đề khác nhau được đưa ra [8].

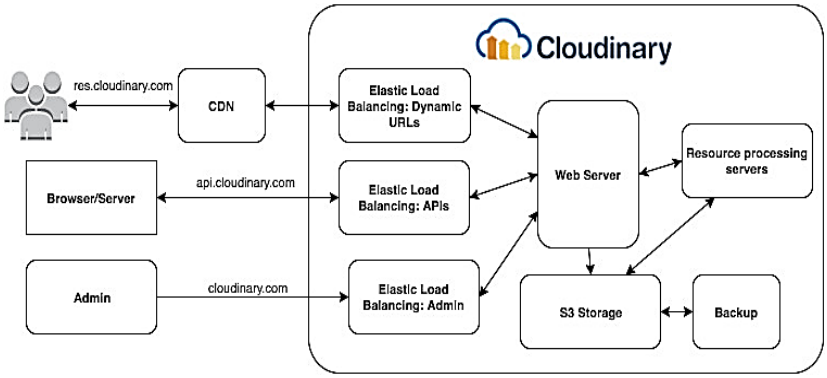
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 5: Kiến trúc mạng lưới neural của ChatGPT [8]

## 2.6 Cloudinary

Cloudinary: là một dịch vụ quản lý hình ảnh và video trực tuyến được cung cấp dưới dạng đám mây (cloud) cho các ứng dụng Web và mobile. Nó cung cấp các tính năng chuyên nghiệp để tối ưu hóa, xử lý và lưu trữ các tệp đa phương tiện của bạn trên đám mây [9].




Hình 6: Kiến trúc của Cloudinary [9]

Cloudinary cho phép bạn tải lên hình ảnh và video của mình và tự động chuyển đổi chúng thành các định dạng tối ưu hóa cho việc hiển thị trên Web, bao gồm cả định dạng ảnh động và ảnh động WebP. Nó cũng cung cấp các tính năng xử lý ảnh chuyên nghiệp như cắt, xoay, thu phóng và thay đổi kích thước ảnh, đồng thời hỗ trợ các tính năng đồng bộ hoá với các nền tảng CMS và e-commerce.

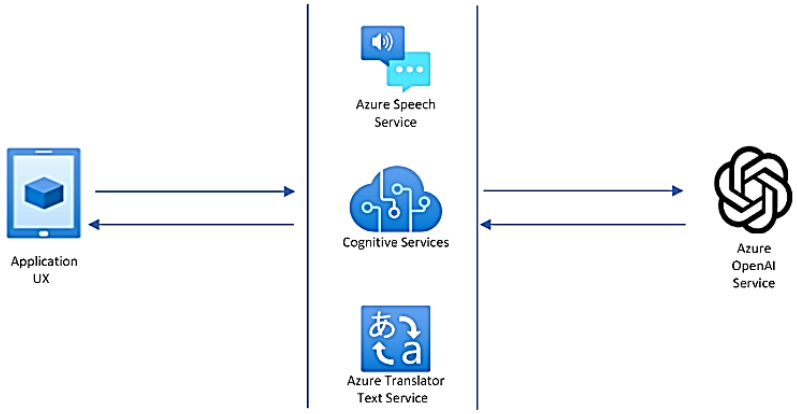
## 2.7 ZegoCloud

ZegoCloud là nhà cung cấp dịch vụ đám mây âm thanh và video chuyên nghiệp (Hình 7). Chúng ta có thể dễ dàng xây dựng các ứng dụng trò chuyện thoại và trò chuyện video trực tiếp với API thân thiện với nhà phát triển trong vòng vài phút [10].

## 2.8 OpenAI

Azure: không phải là một AI Model, mà thực chất là một dịch vụ đám mây của Microsoft. Azure cung cấp một loạt các dịch vụ điện toán đám mây, bao gồm cơ sở hạ tầng điện toán, lưu trữ dữ liệu, dịch vụ lưu trữ, máy ảo, dịch vụ dự đoán, và nhiều dịch vụ khác nhau. Điều đặc biệt là Azure cung cấp các dịch vụ liên quan đến Trí tuệ nhân tạo (Artificial Intelligence - AI) như Azure Cognitive Services và Azure Machine Learning [11].

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

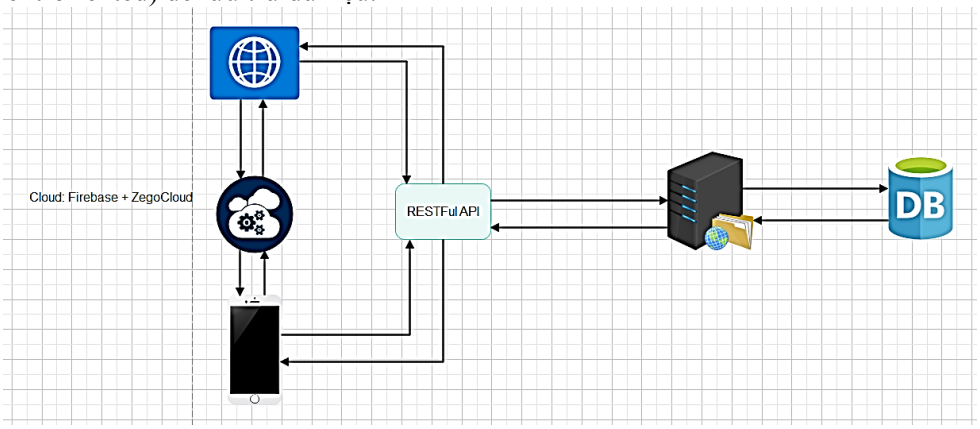



Hình 7: Các tùy chọn tích hợp Azure OpenAI [11]

## 3 T&T HEALTHCARE: CÔNG CỤ HỖ TRỢ SỨC KHỎE THÔNG MINH CHO BỆNH NHÂN ĐƯỜNG HUYẾT

### 3.1 Giới thiệu

Để đảm bảo các yêu cầu đề ra, chúng tôi sử dụng các công nghệ như NodeJS, ExpressJS, Javascript, ReactJS, React Native, SocketIO, ChartJS, ChatGPT, JSON Web Token (JWT). Và cũng để triển khai cho người dùng sử dụng, chúng tôi sử dụng các dịch vụ của điện toán đám mây (Cloud Computing) như Firebase Authentication, Cloudinary, ZegoCloud, MongoDB. Hệ thống của chúng tôi sử dụng trên 2 nền tảng là website và ứng dụng di động. Trong đó, ReactJS dùng để xây dựng nên website và các chức năng theo yêu cầu bài toán, React Native xây dựng ứng dụng trên điện thoại với hệ điều hành Android/iOS và các chức năng tương tự để tạo sự đồng bộ với nhau. SocketIO xây dựng website và app với thời gian thực (real-time), đồng bộ hóa dữ liệu trên nhiều thiết bị và ứng dụng tương tác trực tiếp giữa client và server. ChartJS để phác họa các biểu đồ thống kê, giúp người dùng có thể nhìn trực quan hơn các thông tin. ChatGPT dựa trên kiến trúc GPT-3.5, nó được thiết kế để tự động trả lời các câu hỏi và tương tác với con người thông qua các cuộc trò chuyện trên các nền tảng trực tuyến. JWT để mã hóa dữ liệu dựa trên JSON được sử dụng để xác thực và truyền dữ liệu an toàn giữa các bên. Về điện toán đám mây, đầu tiên là Firebase Authentication hỗ trợ việc đăng nhập, xác thực thông tin như email, số điện thoại giúp tăng tính bảo mật, ngăn chặn các cuộc tấn công hay lừa đảo. Tiếp theo là Cloudinary giúp quản lý các hình ảnh và video, xử lý các tệp đa phương tiện trên đám mây, ZegoCloud cung cấp các dịch vụ đám mây âm thanh và video chuyên nghiệp và được chúng tôi sử dụng để tạo phòng khám cho bệnh nhân. Cuối cùng là MongoDB sử dụng mô hình lưu trữ tài liệu (document-oriented) để lưu trữ dữ liệu.

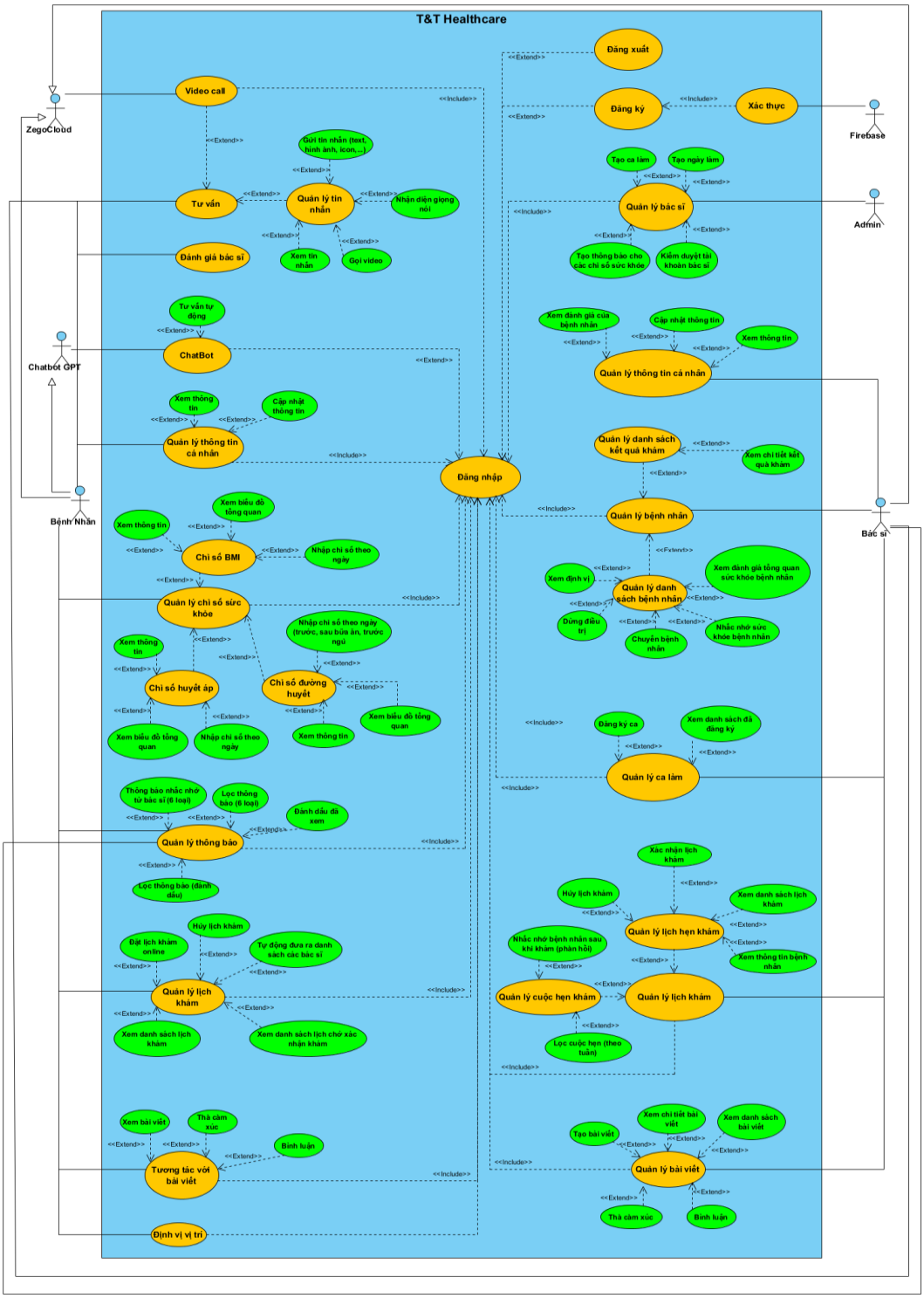



Hình 8: Mô hình kiến trúc hoạt động của ứng dụng T&T Healthcare


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 3.2 Kiến trúc hệ thống

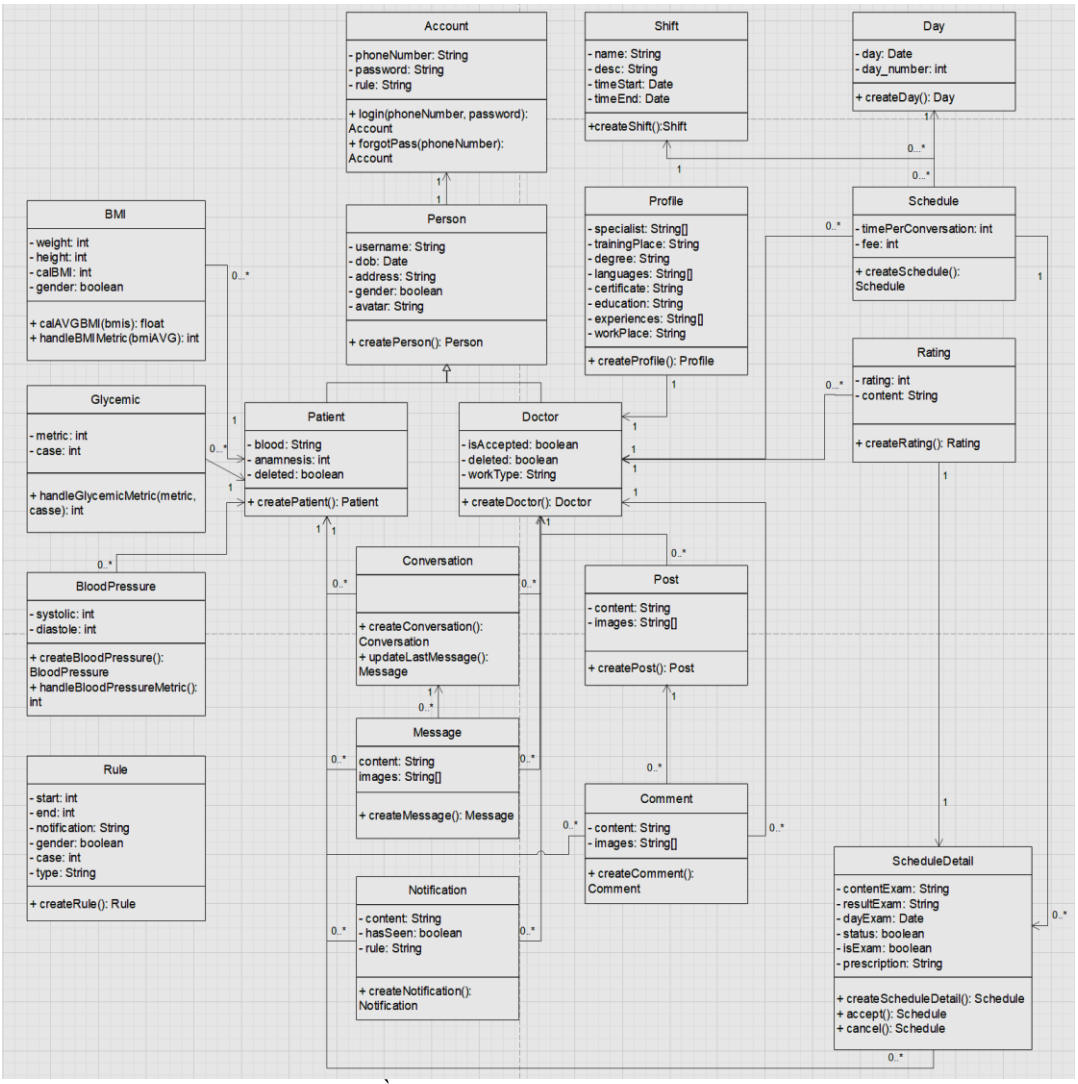
- Ứng dụng cả hai mô hình RESTful API và Client-Server nhằm mục đích thực hiện một kiến trúc ứng dụng hoàn chỉnh có thể đáp ứng được nhu cầu của bài toán đề ra (*Hình 8*), cụ thể là:
+ Client: website và ứng dụng di động thực hiện tương tác với server thông qua RESTful API.
+ Client tích hợp các dịch vụ của Firebase và ZegoCloud nhằm mục đích kế thừa các chức năng đã được dựng sẵn và có thể tích hợp vào hệ thống để sử dụng một cách nhanh chóng.




Hình 9: Sơ đồ usecase của T&T HEALTHCARE

## 3.3 Phân tích hệ thống T&T HEALTHCARE

Đối tượng dùng cho hệ thống T&T HEALTHCARE của chúng tôi gồm có: admin, bác sĩ và bệnh nhân. Hệ thống chủ yếu tập trung vào các chức năng chính để giải quyết các vấn đề đang còn tồn đọng đó là việc theo dõi thủ công và giám sát liên tục trở nên khó khăn và tốn thời gian đối với cả bệnh nhân và bác sĩ. Thêm vào đó, các chức năng của admin cũng được thêm vào để có thể quản lý bác sĩ dễ dàng hơn. Các chức năng chính của bác sĩ và bệnh nhân. Từ các chức năng trên, chúng tôi tập trung vào các chức năng chính của hệ thống T&T HEALTHCARE như: Quản lý bệnh nhân, Quản lý các chỉ số sức khỏe, Thông báo về tình trạng sức khỏe cho bệnh nhân. Sau đó, chúng tôi mở rộng thêm các chức năng như *(Hình 9)*. Trong sơ đồ usecase bên trên, người dùng chính mà chúng tôi hướng đến đó là các actor chính như bác sĩ và bệnh nhân. Đối với bác sĩ sẽ theo dõi sức khỏe của bệnh nhân hằng ngày, đưa ra các nhắc nhở, cảnh báo kịp thời cho bệnh nhân và tư vấn khám bệnh cho bệnh nhân. Đối với bệnh nhân có thể quản lý sức khỏe của mình hằng ngày bằng việc nhập các chỉ số như BMI, đường huyết và huyết áp, xem tình trạng sức khỏe của mình và nhận được các thông báo kịp thời về tình trạng sức khỏe từ bác sĩ. Ngoài ra, chúng tôi còn mở rộng các usecase cho các chức năng của hệ thống, thêm các dịch vụ khác nhằm tạo trải nghiệm tốt nhất đến người sử dụng. Tuy nhiên, các chức năng của dịch vụ đó chúng tôi chưa đề cập ở đây. Chúng tôi xây dựng các chức năng trên tích hợp cùng các bên thứ ba, đó cũng là các actor phụ trong hệ thống: Firebase Authentication, ZegoCloud, Chatbot GPT và Google Map API. Đồng thời, dựa vào các chức năng trên, chúng tôi tiến hành thiết kế mô hình quan hệ cho ứng dụng như *(Hình 10)*.




Hình 10: Sơ đồ Class Diagram của T&T HEALTHCARE

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.4 Giao diện minh họa

Hệ thống T&T HEALTHCARE được triển khai trên hai nền tảng là website và ứng dụng di động. Trong đó, đối với website được deploy lên host và có thể tìm kiếm trên thanh tìm kiếm của Google bằng từ khóa "suckhoethongminh iuh", còn đối với ứng dụng di động có thể download file apk có tên "HealthcareApp" và cài đặt nó để sử dụng nhằm mục đích tạo trải nghiệm tốt nhất cho người dùng. Chúng tôi cũng xin giới thiệu về một số giao diện cho các chức năng trên website cũng như trên ứng dụng di động. Ở đây, bác sĩ sẽ đăng nhập trên các trình duyệt Web còn bệnh nhân sẽ đăng nhập trên ứng dụng di động.

#### 3.4.1 Một số giao diện phía website của bác sĩ trên hệ thống T&T HEALTHCARE

Ở phần này, chúng tôi hiện thực một số giao diện với các chức năng thuộc phân quyền của bác sĩ trên website. Đầu tiên, sẽ là danh sách các thông báo của bác sĩ nhận được từ bệnh nhân thuộc sự theo dõi của bác sĩ đó bên dưới *(Hình 11)*.




Hình 11: Giao diện quản lý danh sách thông báo của bác sĩ

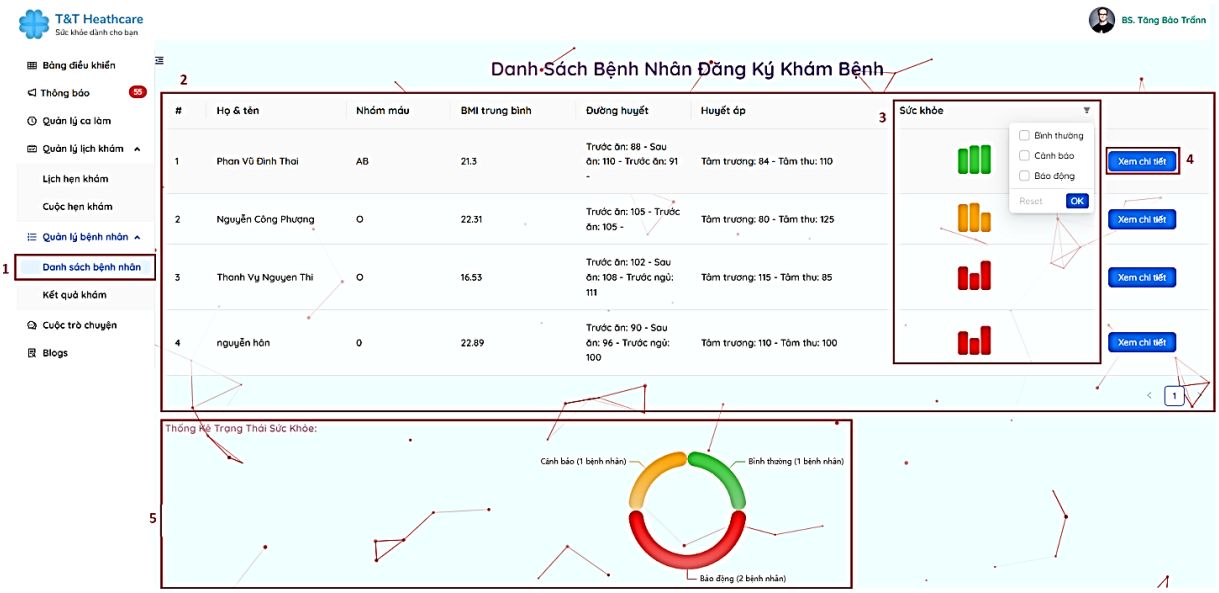
Trước hết, bác sĩ nhấn vào "Thông báo" *(vùng 1)* ở menu bên trái, hệ thống hiển thị danh sách các thông báo nhận được từ bệnh nhân mà mình quản lý *(vùng 2)*.

- Ngoài ra, các thông báo còn được chúng tôi phân loại ra gồm sáu loại tương ứng với sáu màu sắc khác nhau để giúp bác sĩ dễ dàng nhận biết, cũng như để quản lý bệnh nhân tốt và hiệu quả nhất *(vùng 3)*.
- Khi bệnh nhân đăng ký hay hủy lịch khám hoặc nhập các chỉ số về sức khỏe như BMI, đường huyết, huyết áp thì hệ thống sẽ tự động phân loại các thông báo và sau đó hiển thị thông báo đến bác sĩ ngay lập tức *(vùng 4)* để giúp bác sĩ có thể theo dõi các bệnh nhân và đưa ra các cảnh báo kịp thời nếu tình trạng sức khỏe của bệnh nhân đang có dấu hiệu xấu đi.
- Tất cả các bệnh nhân thuộc sự theo dõi và đảm nhận của bác sĩ sẽ được hiển thị ở *(Hình 12)* bên dưới.
- Bác sĩ nhấn vào "Danh sách bệnh nhân" *(vùng 1)* ở menu bên trái, hệ thống hiển thị danh sách bệnh nhân mà bác sĩ đó theo dõi và đảm nhận *(vùng 2)*.
- Chúng tôi còn phân loại tình trạng sức khỏe của bệnh nhân theo ba trạng thái tương ứng với ba màu sắc khác nhau nhằm giúp bác sĩ dễ dàng nhận biết tình trạng sức khỏe để đưa ra các nhắc nhở và cảnh báo kịp thời cho bệnh nhân. Phân loại tình trạng sức khỏe *(vùng 3)*:

+ Sức khỏe ở trạng thái "Bình thường" được thể hiện bằng màu xanh lá cây và chuyển động chậm rãi.
+ Sức khỏe ở trạng thái "Cảnh báo" được thể hiện bằng màu cam và chuyển động nhanh dần.
+ Sức khỏe ở trạng thái "Báo động" được thể hiện bằng màu đỏ và chuyển động rất nhanh.

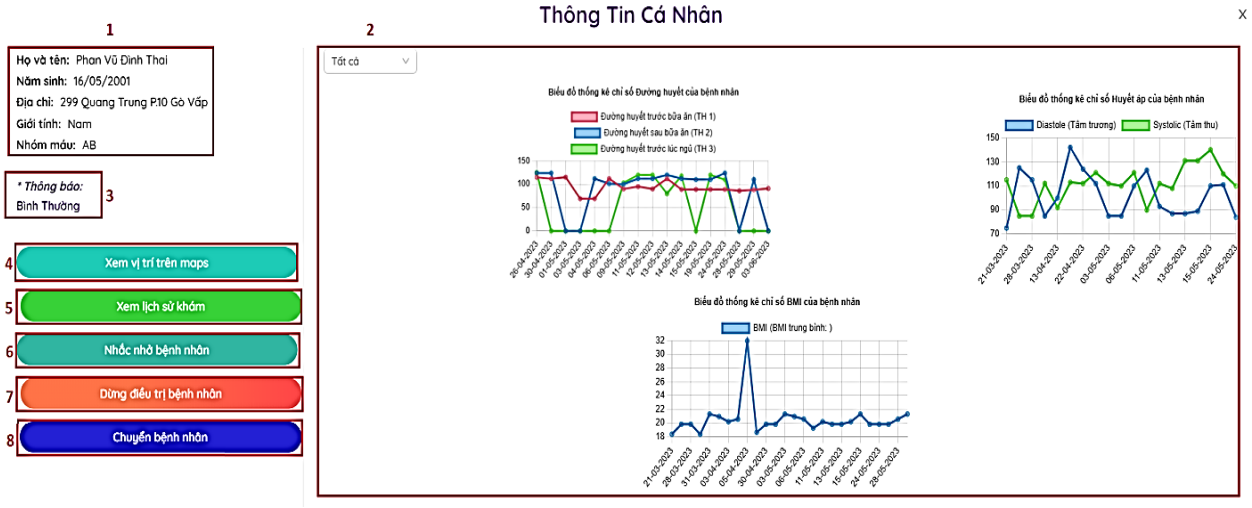
- Ngoài ra, chúng tôi còn phác họa thêm biểu đồ nhằm giúp cho bác sĩ có cái nhìn bao quát hơn về tình hình sức khỏe của bệnh nhân mà mình đảm nhận theo dõi. Biểu đồ này, cũng có ba trạng thái tương ứng theo phân loại tình trạng sức khỏe của bệnh nhân *(vùng 5)* và sẽ xem được có bao nhiêu bệnh nhân đang ở trạng thái nào.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 12: Giao diện quản lý danh sách bệnh nhân của bác sĩ

- Khi bác sĩ nhấn vào "Xem chi tiết" bệnh nhân đó *(vùng 4)*, hệ thống hiển thị modal như *(Hình 13)* bên dưới.




Hình 13: Giao diện xem chi tiết thông tin một bệnh nhân của bác sĩ

- Ở màn hình này, bác sĩ có thể xem được các thông tin của bệnh nhân như họ tên, năm sinh, địa chỉ, giới tính, nhóm máu *(vùng 1)*.
- Ở đây, chúng tôi còn hỗ trợ dựng các biểu đồ thống kê về các loại chỉ số như BMI, đường huyết và huyết áp *(vùng 2)*. Ngoài ra, còn hỗ trợ thêm cho bác sĩ về lọc biểu đồ theo tuần, theo tháng với mục đích giúp cho bác sĩ có cái nhìn tổng quan về tình hình sức khỏe của bệnh nhân đó. Đồng thời, xem tình trạng sức khỏe *(vùng 3)*, và đưa ra các cảnh báo kịp thời về tình hình sức khỏe của bệnh nhân *(vùng 6)*.
- Ngoài ra, bác sĩ có thể thực hiện một số chức năng khác như:
+ Xem vị trí trên maps: xem được vị trí của bệnh nhân (đường đi, khoảng cách và thời gian) *(vùng 4)*.
+ Xem được lịch sử khám cho bệnh nhân nếu trước đó bệnh nhân đã khám *(vùng 5)*.
+ Dừng điều trị cho bệnh nhân: bệnh nhân sẽ không còn được sự theo dõi, quản lý của bác sĩ đó *(vùng 7)*.
+ Chuyển bệnh nhân: bệnh nhân được chuyển sang một bác sĩ khác để theo dõi và điều trị (tất nhiên bệnh nhân không còn được sự theo dõi, quản lý bởi bác sĩ trước đó) *(vùng 8)*.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.4.2 Một số giao diện phía ứng dụng di động của bệnh nhân trên hệ thống T&T HEALTHCARE

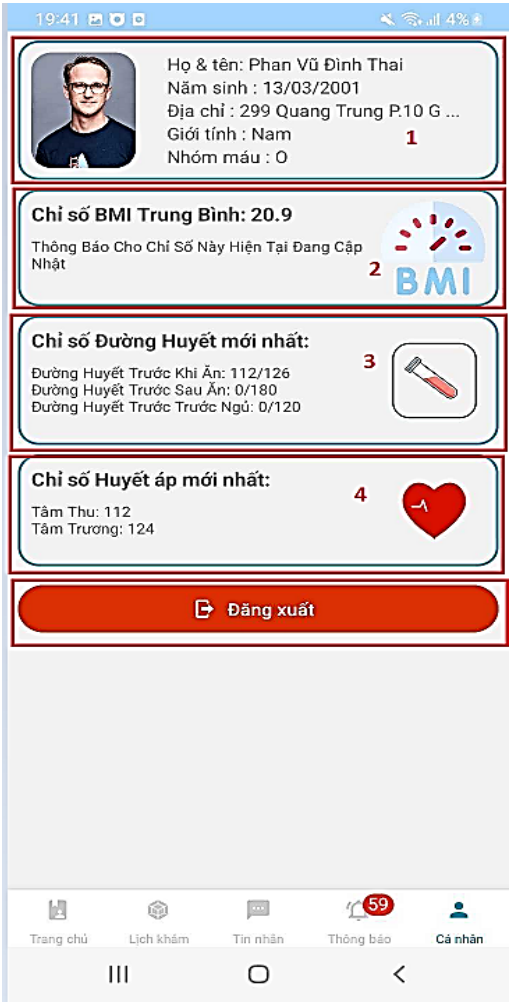
Ở đây, chúng tôi hiện thực một số giao diện với các chức năng thuộc phân quyền của bệnh nhân trên ứng dụng di động. Đầu tiên, sẽ là danh sách các thông báo của bệnh nhân nhận được từ bác sĩ (bệnh nhân được bác sĩ đó theo dõi và quản lý) ở dưới *(Hình 14, Hình 15)*.

Bệnh nhân chọn "Cá nhân", hệ thống chuyển đến màn hình cá nhân *(Hình 14)*. Hiển thị các phần như:

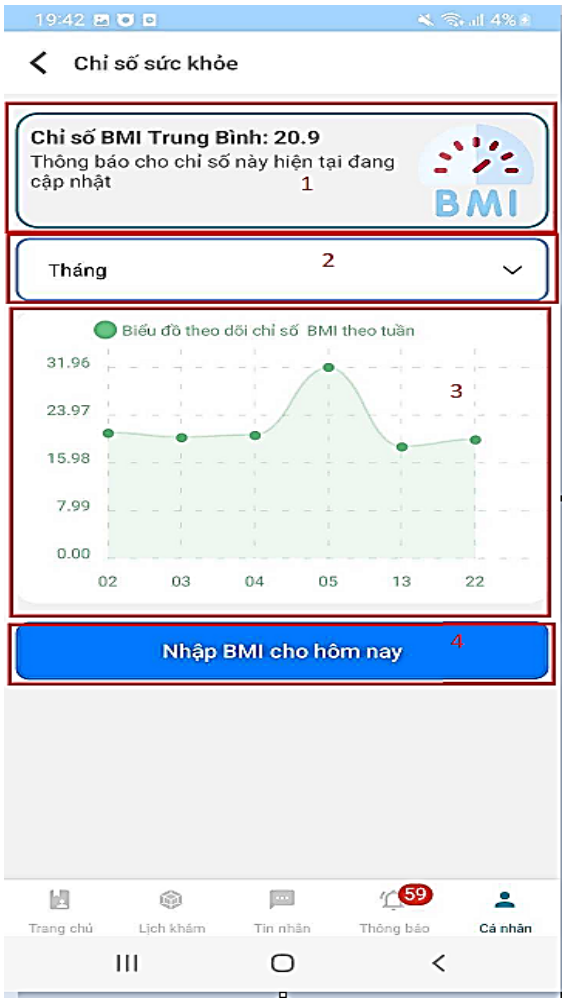
+ Thông tin cá nhân *(vùng 1)*.
+ Thông tin chỉ số BMI *(vùng 2)*.
+ Thông tin chỉ số đường huyết *(vùng 3)*.
+ Thông tin chỉ số huyết áp *(vùng 4)*.

- Khi chọn *(vùng 1)* thì hệ thống chuyển đến giao diện thông tin cá nhân và hiển thị các thông tin như: họ tên, năm sinh, địa chỉ, giới tính, nhóm máu, ...
- Khi chọn *(vùng 2)* thì hệ thống chuyển đến giao diện thông tin về chỉ số BMI *(Hình 15)*. Hiển thị các phần:

+ Chỉ số BMI trung bình và đánh giá của chỉ số được gửi lên từ hệ thống *(vùng 1)*.
+ Có thể lọc theo tháng và tuần khi chọn vào *(vùng 2)*.
+ Chỉ số BMI được thể hiện theo biểu đồ phụ thuộc vào bộ lọc *(vùng 3)*.
+ Bạn có thể nhập chỉ số BMI cho ngày hôm nay bằng cách chọn vào *(vùng 4)*.



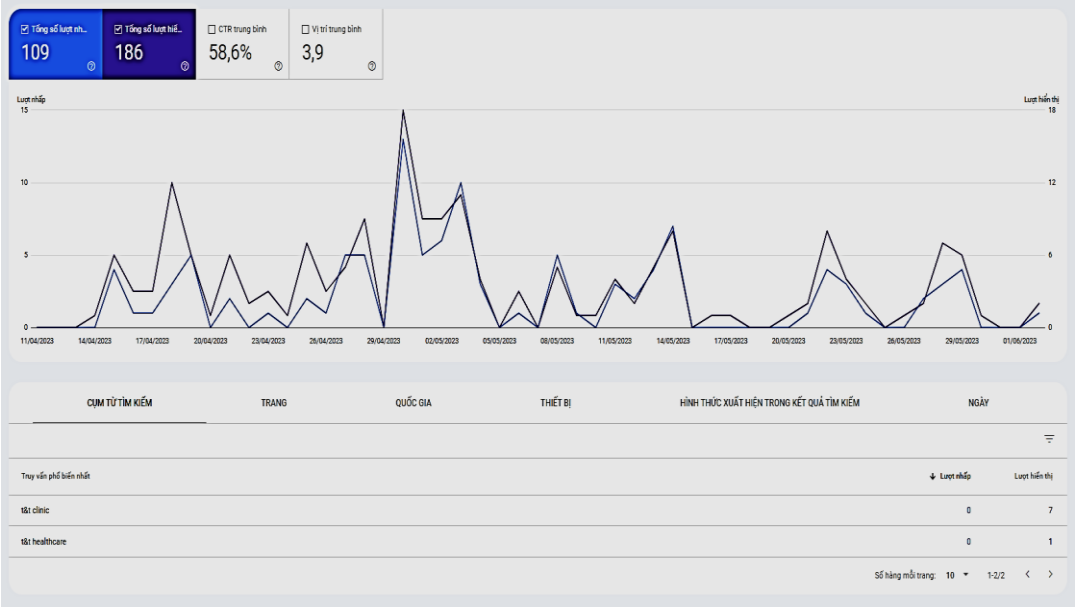

Hình 14: Giao diện thông tin bệnh nhân




Hình 15: Giao diện theo dõi BMI

## 4 KIỂM NGHIỆM THỰC TẾ

Chúng tôi sử dụng Vercel để deploy hệ thống T&T HEALTHCARE, sau đó chúng tôi dùng dịch vụ Google Search Console của Google để giúp người sử dụng có thể tìm kiếm được trang website trên thanh tìm kiếm của Google bằng từ khóa "suckhoethongminh iuh" với mục đích tạo sự tiện lợi cho người dùng và cũng để giúp T&T HEALTHCARE có thể được nhiều người biết đến và được sử dụng rộng rãi trên thị trường.
Sau khi sử dụng dịch vụ Google Search Console, chúng tôi có thể dễ dàng theo dõi, duy trì và khắc phục sự cố liên quan đến trang Web của chúng tôi trong kết quả tìm kiếm của Google (*Hình 16*). Ví dụ như:
+ Xem dữ liệu về lưu lượng truy cập từ Google Search Console đến trang Web của chúng tôi: tần suất trang Web xuất hiện trong Google Search Console, cụm từ tìm kiếm nào làm trang Web của chúng tôi hiển thị, tần suất người tìm kiếm nhấp vào trang đối với cụm từ đó.
+ Nhận được các thông báo khi Google gặp phải vấn đề chỉ mục, nội dung spam hoặc các vấn đề khác như lỗi, … trên trang Web của chúng tôi.
+ Khắc phục được vấn đề lập chỉ mục và yêu cầu lập chỉ mục lại nội dung mới hoặc nội dung cập nhật.




Hình 16: Các thông số về hiệu suất của hệ thống T&T HEALTHCARE

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này, chúng tôi đã đề ra và trình bày hai giải pháp cho bài toán xây dựng công cụ T&T HEALTHCARE nhằm hỗ trợ sức khỏe thông minh cho bệnh nhân đường huyết và huyết áp trên cả hai nền tảng là website và ứng dụng di động. Cả hai giải pháp chúng tôi đều sử dụng các công nghệ phù hợp để phát triển cho hệ thống và đảm bảo được các tiêu chí mà bài toán đặt ra. Chúng tôi sử dụng ReactJS dùng để xây dựng nên website và các chức năng theo yêu cầu bài toán, React Native xây dựng ứng dụng trên điện thoại với hệ điều hành Android/iOS và các chức năng tương tự để tạo sự đồng bộ với nhau. SocketIO để xây dựng website và app với thời gian thực (real-time), đồng bộ hóa dữ liệu trên nhiều thiết bị và ứng dụng tương tác trực tiếp giữa client và server. ChartJS để phác họa các biểu đồ thống kê, giúp người dùng có thể nhìn trực quan hơn các thông tin. ChatGPT dựa trên kiến trúc GPT-3.5, để tự động trả lời các câu hỏi và tương tác với con người thông qua các cuộc trò chuyện trên các nền tảng trực tuyến và một số các công nghệ khác. JWT để mã hóa dữ liệu dựa trên JSON được sử dụng để xác thực và truyền dữ liệu an toàn giữa các bên. Chúng tôi còn sử dụng một số dịch vụ dựa trên nền tảng đám mây (Cloud Computing), Firebase Authentication hỗ trợ việc đăng nhập, xác thực thông tin như email, số điện thoại giúp tăng tính bảo mật, ngăn chặn các cuộc tấn công hay lừa đảo. Cloudinary giúp quản lý các hình ảnh và video, xử lý các tệp đa phương tiện trên đám mây, ZegoCloud để cung cấp các dịch vụ đám mây âm thanh và video chuyên nghiệp và được chúng tôi sử dụng để tạo phòng khám cho bệnh nhân và MongoDB để lưu trữ dữ liệu. Chúng tôi còn sử dụng Vercel để deploy hệ thống T&T HEALTHCARE, sau đó dùng dịch vụ Google

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Search Console của Google để giúp người sử dụng có thể tìm kiếm nhanh trang website trên thanh tìm kiếm của Google với từ khóa "suckhoethongminh iuh". Ngoài ra, hệ thống dựa trên các chỉ số về sức khỏe của bệnh nhân hằng ngày, sau đó sẽ đưa ra các cảnh báo thông minh đến bệnh nhân một cách nhanh chóng và kịp thời. Nhờ các chức năng thông minh như hỗ trợ theo dõi, chăm sóc và điều trị tại nhà cho các bệnh nhân về đường huyết và huyết áp nên đã giảm tải được phần nào về số lượng bệnh nhân đến bệnh viện. Hệ thống của chúng tôi được thiết kế với tính thực tiễn cao, phù hợp với tình trạng ngày càng tăng của bệnh nhân đường huyết cần điều trị tại nhà. Hiện tại hệ thống đã được triển khai thực tế và được sử dụng bởi một số bệnh nhân ở khu vực Gò Vấp, Tp. HCM. Trong tương lai, chúng tôi sẽ mở rộng và phát triển thêm một số tính năng mới để đáp ứng nhu cầu sử dụng cũng như đem lại trải nghiệm tốt nhất cho người sử dụng: có thể quản lý thêm nhiều loại bệnh khác nhau, nhận các thông báo kể cả khi ứng dụng đang bị tắt.

## REFERENCES

[1]. NodeJS là gì? Những kiến thức cơ bản liên quan tới NodeJS (fpt.edu.vn) – [Accessed 16/04/2023]

[2]. Alex Young, Bradley Meck, and Mike Cantelon with Tim Oxley, Marc Harter, T.J. Holowaychuk, and Nathan Rajlich, Node.js in Action, Second Edition - August 2017 – [Accessed 16/04/2023]

[3]. Jonathan Wexler, Get Programming with NodeJS, February 2019. – [Accessed 16/04/2023]

[4]. Express - Node.js web application framework (expressjs.com) – [Accessed 16/04/2023]

[5]. https://expressjs.com -Quick Start – React – Hình ảnh: ReactJS - Architecture (tutorialspoint.com) – [Accessed 16/04/2023]

[6]. Rawat, P., Mahajan, A.N.: Reactjs: A modern web development framework. International Journal of Innovative Science and Research Technology 5(11) (2020) – [Accessed 16/04/2023]

[7]. https://reactnative.dev-React Native · Learn once, write anywhere – [Accessed 16/04/2023]

[8]. Introduction - OpenAI API – [Accessed 16/04/2023]

[9]. Cloudinary - Wikipedia – Hình ảnh: Service Architecture (wearecommunity.io) – [Accessed 16/04/2023]

[10]. https://www.zegocloud.com/docs – Hình ảnh: Mix the video streams - Video Call SDK - ZEGOCLOUD Doc - [Accessed 16/04/2023]

[11].https://www.linkedin.com/pulse/azure-openai-integration-options-ankit-saxena- [Accessed 29/07/2023]

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F015**

# AVTEDU - CÔNG CỤ HỖ TRỢ VÀ QUẢN LÝ HỌC VỤ TẠI TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH

TÔN LONG PHƯỚC[1*], NGUYỄN VIỆT ANH[1], PHẠM NGUYỄN VĂN TRƯỜNG[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
**tonlongphuoc@iuh.edu.vn*

**Tóm tắt.** Trường Đại học Công nghiệp TP.HCM có hơn 36.000 sinh viên. Qua đó, dễ dàng nhận thấy nhu cầu về một hệ thống hỗ trợ sinh viên học tập tại trường vận hành hiệu quả là rất cao. Tuy nhiên, hệ thống của trường chưa thể đáp ứng được các nhu cầu của người dùng tại một số thời điểm. Vấn đề đầu tiên là hệ thống luôn bị nghẽn vào một số giờ cao điểm, cụ thể là vào thời điểm đăng ký môn học trong mỗi học kỳ. Vấn đề thứ hai, hệ thống phần mềm triển khai trên thiết bị di động chưa có, còn phụ thuộc bên thứ ba và chưa hỗ trợ hết các chức năng cho sinh viên. Do đó, chúng tôi đề xuất một công cụ hỗ trợ và quản lý học tập có tên là AVTEdu. AVTEdu hỗ trợ nhiều phân hệ từ học viên đến giảng viên và giáo vụ khoa, với mục tiêu chính là giải quyết tình trạng tắc nghẽn hệ thống trong thời gian truy cập cao điểm bằng cách sử dụng Autoscaling và Load Balancer của EC2 AWS. Bên cạnh đó, AVTEdu cũng hỗ trợ học vụ cho sinh viên trên nền tảng ứng dụng di động. Các công nghệ mà chúng tôi đang sử dụng để xây dựng nên hệ sinh thái mang tên AVTEdu bao gồm Nodejs, React, AWS Cloud Computing. Chúng tôi đã triển khai hệ thống này cho sinh viên chuyên ngành Công nghệ phần mềm của Khoa Công nghệ thông tin. Trong thời gian tới, chúng tôi sẽ triển khai cho học sinh toàn trường và mở rộng ra các trường khác.

**Từ khóa.** Tắc nghẽn hệ thống, Ứng dụng di động, Auto Scaling, Load Balancing, AWS.

## AVTEDU - A SUPPORT and STUDY MANAGEMENT TOOL at INDUSTRIAL UNIVERSITY of HO CHI MINH CITY

**Abstract.** Ho Chi Minh the City University of Industry has more than 36,000 students. Thereby, it is easy to see that the need for a support system for students studying at the school to operate smoothly is very high. However, the school's system has not been able to respond well. The first problem, the system is always congested at some peak hours, specifically around the time of course registration. The second problem, the software system deployed on mobile devices is still depended of third parties and is not optimized for students. Therefore, we propose a learning support and management tool called AVTEdu. AVTEdu encompasses multiple subsystems for students, instructors, and department administrators, with the primary objective of resolving system congestion during peak access times. This is achieved through the utilization of Autoscaling and Load Balancer features provided by AWS EC2. Additionally, AVTEdu provides administrative support for student affairs through a mobile application platform. The technologies we employ in building the AVTEdu ecosystem include Node.js, React, and AWS Cloud Computing. We have already implemented this system for students majoring in Software Engineering at the Faculty of Information Technology. Moving forward, our plan is to deploy it for the entire student body and expand its usage to other universities.

**Keywords.** System Congestion, Mobile Apps, Auto Scaling, Load Balancing, AWS.

## 1 GIỚI THIỆU TỔNG QUAN

Với sự bùng nổ của cách mạng công nghiệp 4.0, nhiều lĩnh vực đã và đang được số hóa trên môi trường kỹ thuật, bao gồm ngành giáo dục. Cụ thể tại trường Đại học Công nghiệp Thành phố Hồ Chí Minh, với số lượng sinh viên học tập tại đây ngày càng tăng, nhu cầu về hệ thống hỗ trợ học tập cho sinh viên có tính đa

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

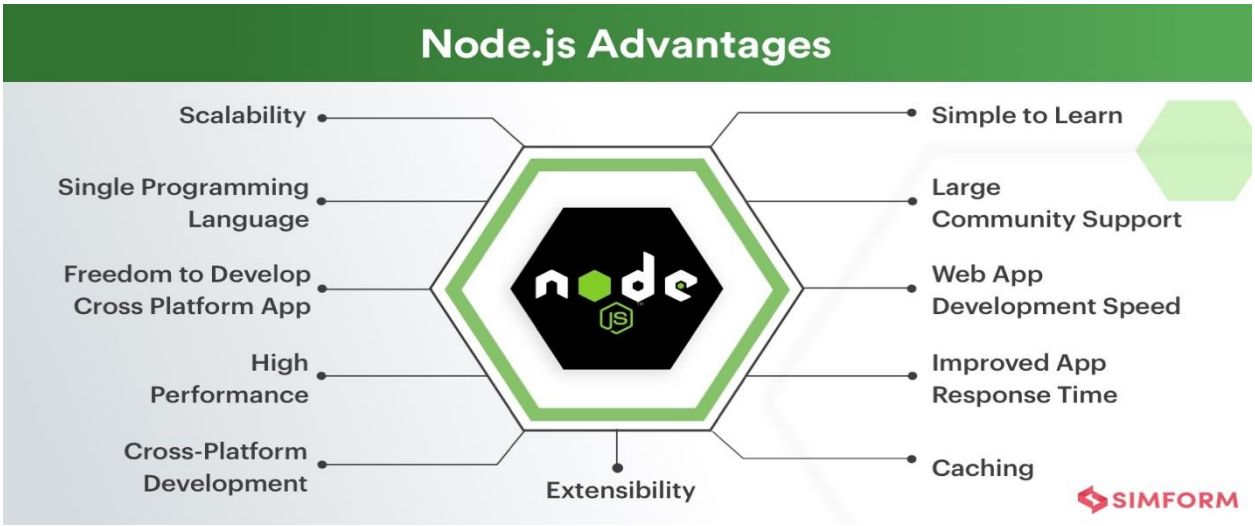
dụng, tính ổn định và sử dụng được trên hai môi trường phổ biến hiện nay là Web và Andrioid. Chúng tôi đã xây dựng chương trình hỗ trợ học tập sinh viên IUH với những tiêu chí trên. Đây là một chương trình có tính ứng dụng cao, bám sát được với nhu cầu về quản lý học tập của sinh viên IUH cũng như chương trình cũng tạo ra để hỗ trợ người quản lý chương trình đào tạo tại trường (sinh viên, giảng viên, lớp học phần,…) và giảng viên quản lý (xem lịch giảng dạy, xem thông tin lớp học,..) quan trọng là việc đơn giản hóa việc đăng ký học phần giúp giảm thiểu sự tắc nghẽn dữ liệu. Hệ thống cũng giải quyết các vấn đề còn tồn đọng của hệ thống website của trường Đại học Công nghiệp hiện tại như việc đăng ký học phần khi trường cố gắng mở đăng ký học phần từ nửa đêm để giảm thiểu việc truy cập đăng ký học phần của sinh viên nhưng hệ thống vẫn quá tải khiến việc đăng ký học phần của sinh viên khó khăn hơn khi phải thức khuya dậy sớm để đăng ký hoặc tệ nhất là không đăng ký được gì và đôi lúc việc truy cập vào website trường có thể không thực hiện được do số lượng truy cập hàng ngày của sinh viên là rất cao. Hiện tại hệ thống hạ tầng vật lý của trường không còn đáp ứng đủ nhu cầu ngày một tăng cao khi lượng sinh viên của trường mỗi năm ngày càng tăng. Việc nâng cấp hệ thống vật lý này rất khó khăn khi có nhiều vấn đề như: Rủi ro có cao không? Chi phí nâng cấp? Thời gian nâng cấp? ....

Để giải quyết vấn đề này, chúng tôi đề xuất sử dụng điện toán đám mây nhằm lưu trữ, xử lý các nghiệp vụ để giảm tải cho máy chủ của trường [1]. Scale là dịch vụ có khả năng tự động điều chỉnh số lượng máy chủ để đáp ứng nhu cầu truy cập của khách hàng (tăng khi yêu cầu nhiều và giảm khi yêu cầu ít). Do đó, nhà phát triển hệ thống không còn phải đầu tư quá nhiều chi phí chỉ để giải quyết nhu cầu cho việc sử dụng tài nguyên của hệ thống trong một quãng thời gian ngắn trong năm. Đó cũng là lý do chúng tôi chọn giải pháp này cho bài toán xây dựng hệ thống hỗ trợ đào tạo cho sinh viên.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 Nodejs

Node.js tạo ra ứng dụng có tốc độ xử lý nhanh với thời gian thực (real-time). Node.js cũng được áp dụng để tạo các sản phẩm có lượng truy cập lớn, mở rộng nhanh và đổi mới công nghệ… hoặc tạo dự án startup tiết kiệm thời gian nhất. Nodejs có nhiều đặc điểm nổi bật, vượt trội. Nắm được các đặc điểm này sẽ giúp chúng ta hiểu sâu hơn về Nodejs, đồng thời đưa ra quyết định sử dụng nền tảng này chính xác [2, 3, 4].




Hình 1: Những ưu điểm vượt trội của Nodejs [3]

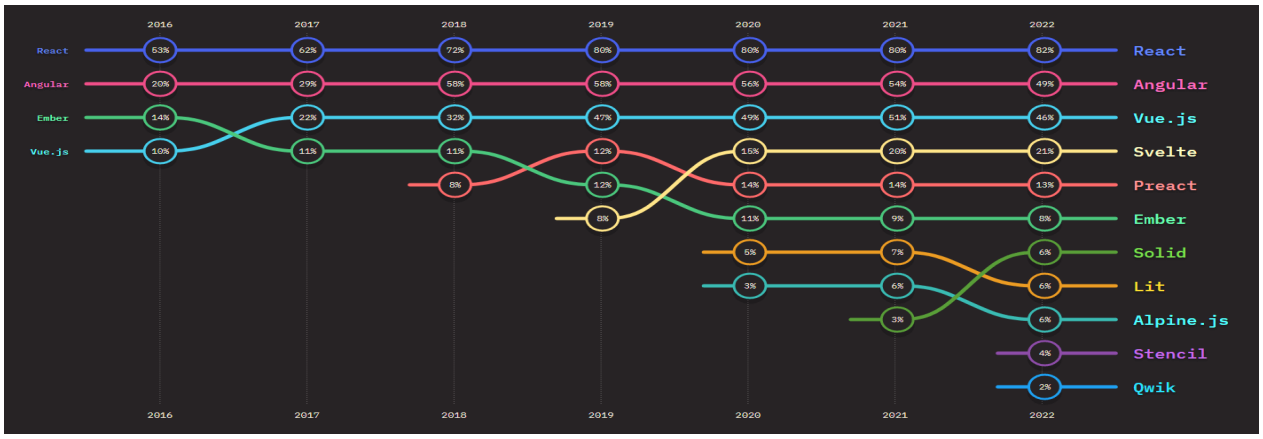
### 2.2 React Native

React Native là một framework do công ty công nghệ nổi tiếng Facebook phát triển nhằm mục đích giải quyết bài toán hiệu năng của Hybrid và bài toán chi phí khi mà phải viết nhiều loại ngôn ngữ native cho từng nền tảng di động. React Native giúp lập trình viên chỉ cần phát triển ứng dụng trên một mã nguồn duy nhất mà vẫn có thể tương thích với cả hai hệ điều hành Android và iOS (thậm chí là cả các hệ điều hành

cho Ipad và và các dòng tivi). React Native giúp tiết kiệm được nhiều thời gian khi muốn phát triển một ứng dụng nhanh chóng nhưng vẫn đảm bảo hiệu năng tương đối ổn định. Quy mô đội ngũ phát triển nhỏ nhưng vẫn dễ dàng bảo trì, quản lý [5].

## 2.3 Reactjs

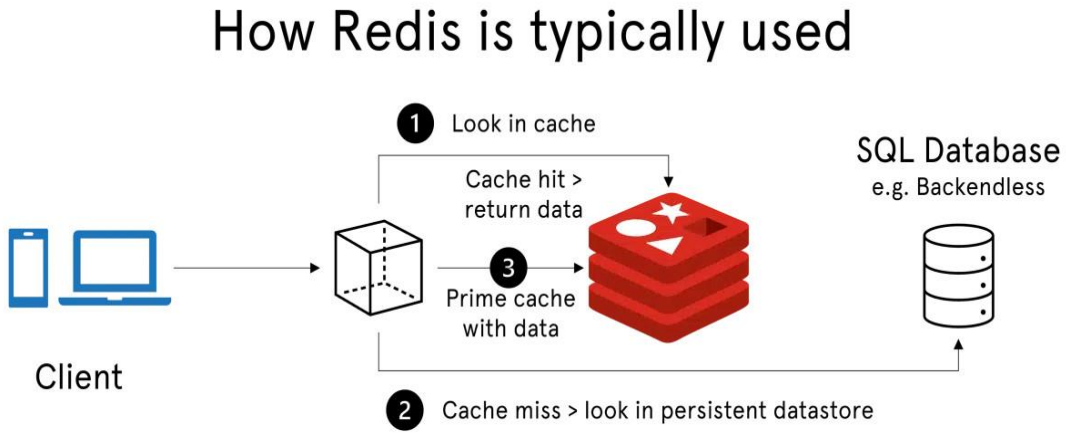
Vì sao chúng tôi lại lựa chọn Reactjs cho ứng dụng web? Nếu cần xây dựng một ứng dụng phức tạp, tải cao, bắt buộc phải xác định cấu trúc của ứng dụng ngay từ đầu vì nó có thể ảnh hưởng đến hiệu suất của ứng dụng của bạn. Mô hình DOM có cấu trúc dạng cây. Vì vậy, một sửa đổi nhỏ ở lớp cấp cao hơn có thể ảnh hưởng đáng kể đến giao diện người dùng của ứng dụng. Để giải quyết vấn đề này, Facebook đã giới thiệu tính năng DOM ảo. Virtual DOM, như cái tên đã cho thấy đây là đại diện ảo của DOM cho phép thử nghiệm tất cả các thay đổi đối với DOM ảo trước tiên để tính toán rủi ro với mỗi sửa đổi. Do đó, cách tiếp cận này giúp duy trì hiệu suất ứng dụng cao và đảm bảo trải nghiệm người dùng tốt hơn [6, 7].




Hình 2: Biểu đồ phát triển của các Frontend Framework qua từng giai đoạn [6]

## 2.4 Redis

Redis là một sự lựa chọn tuyệt vời khi ta cần đến một server lưu trữ dữ liệu đòi hỏi tính mở rộng cao (scaleable) và chia sẻ bởi nhiều tiến trình, nhiều ứng dụng và nhiều server khác nhau. Caching: Sử dụng làm bộ nhớ đệm. Chính tốc độ đọc ghi nhanh mà Redis có thể làm bộ nhớ đệm, nơi chia sẻ dữ liệu giữa các ứng dụng hoặc làm database tạm thời. Ngoài ra Redis có thể sử dụng để làm Full Page Cache cho website. Cũng vì tính nhất quán của Redis, cho dù restart Redis thì người dùng cũng không có cảm nhận chậm khi tải trang [8, 9].




Hình 3: Mô tả sơ lược về mô hình hoạt động của Redis [8]


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

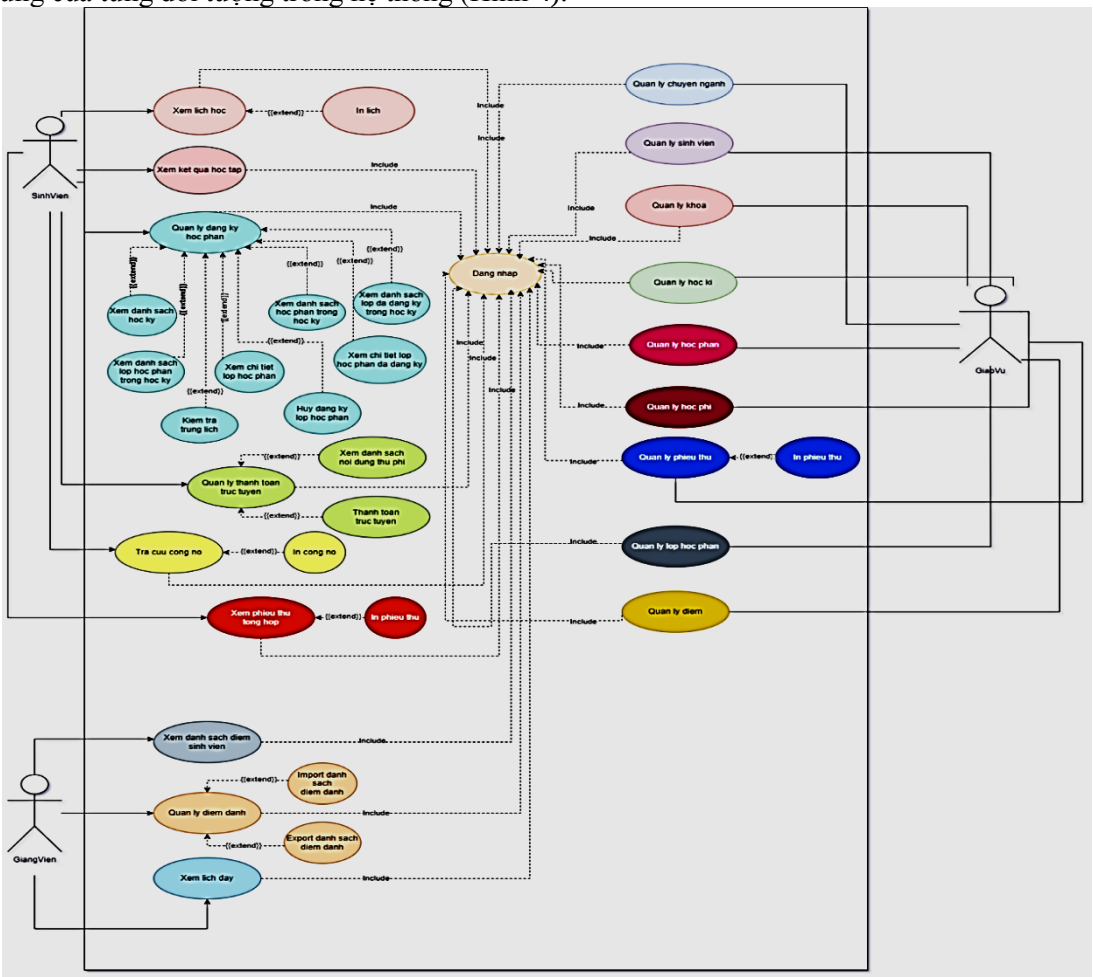
# 3 AVTEDU - CÔNG CỤ HỖ TRỢ VÀ QUẢN LÝ HỌC VỤ TẠI TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH

## 3.1 Giới thiệu

Để triển khai các giải pháp mà chúng tôi nêu ra trong phần giới thiệu, chúng tôi đề xuất xây dựng công cụ AVTEdu. Công cụ được triển khai dựa vào các công nghệ như NodeJS, ReactJS/React-Native. Đồng thời để triển khai công cụ đến người dùng, chúng tôi sử dụng các dịch vụ Cloud Computing của AWS và Vercel. Hệ thống AVTEdu là sự kết hợp ứng dụng di động (mobile app) và website (web app), tạo ra một trải nghiệm linh hoạt cho người dùng. Trong đó chúng tôi sử dụng NodeJS để triển khai các Restful-API, xử lý các yêu cầu người dùng. ReactJS chịu trách nhiệm xây dựng webstie, React-Native xây dựng ứng dụng mobile [10]. Để đảm bảo hiệu suất và độ tin cậy, AVTEDU sử dụng các dịch vụ điện toán đám mây trong AWS. Elastic Load Balancing (ELB) được sử dụng để cân bằng tải các yêu cầu từ người dùng, đảm bảo hệ thống hoạt động trơn tru và ổn định. EC2 được sử dụng để tạo máy ảo lưu trữ mã nguồn cho server. RDS để lưu trữ toàn bộ dữ liệu của hệ thống. Ngoài ra Docker tạo ra các container có thể triển khai AutoScaling của hệ thống nhanh hơn, Redis nhằm lưu trữ token của người dùng để có thể truy xuất nhanh hơn cũng như bảo mật hơn khi lưu trong cơ sở dữ liệu.

## 3.2 Phân tích và thiết kế hệ thống

Hệ thống phục vụ cho 3 đối tượng người dùng: Sinh viên, giảng viên và giáo vụ cùng với các chức năng tương ứng của từng đối tượng trong hệ thống (Hình 4).



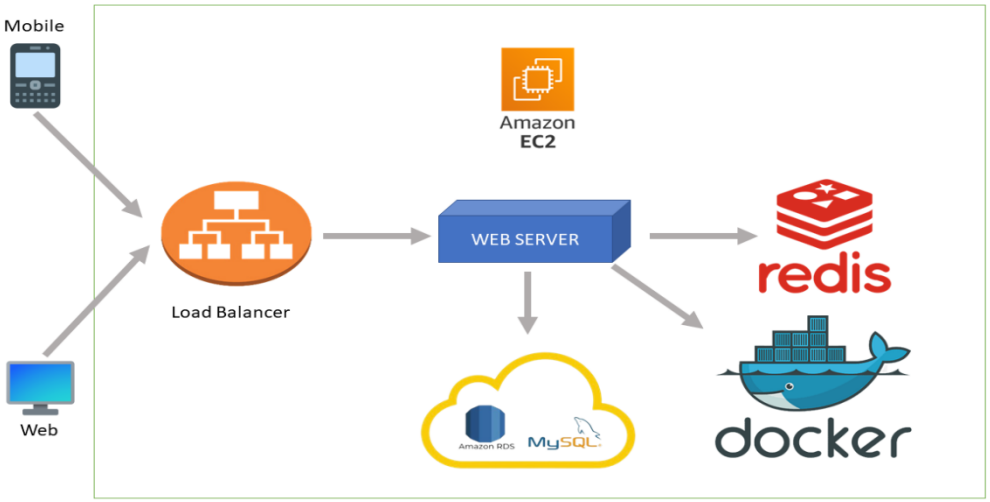

Hình 4: Lược đồ usecase của hệ thống AVTEdu

Lược đồ usecase tập trung vào ba tác nhân chính: sinh viên, giảng viên, giáo vụ khoa. Giáo vụ khoa thực hiện các chức năng quản lý. Sinh viên được phép đăng ký học phần, xem lịch học và thanh toán học phí trực tuyến. Giảng viên có thể import, export các danh sách như điểm danh sinh viên, xem lịch dạy.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

### 3.3 Kiến trúc hệ thống

Chúng tôi sử dụng kiến trúc Client-Server để xây dựng ứng dụng, bởi vì mô hình này hoạt động theo cách là khi mà máy client gửi yêu cầu đến server, lúc này server sẽ xử lý yêu cầu và trả kết quả về cho client. Client sẽ không chia sẻ bất kỳ tài nguyên nào của họ. Mô hình Client-Server có một số ưu điểm chính để lựa chọn như tính tập trung, tính bảo mật, có khả năng mở rộng và khả năng truy cập (Hình 5).




Hình 5: Mô hình kiến trúc của hệ thống AVTEdu

### 3.4 Giao diện đồ họa

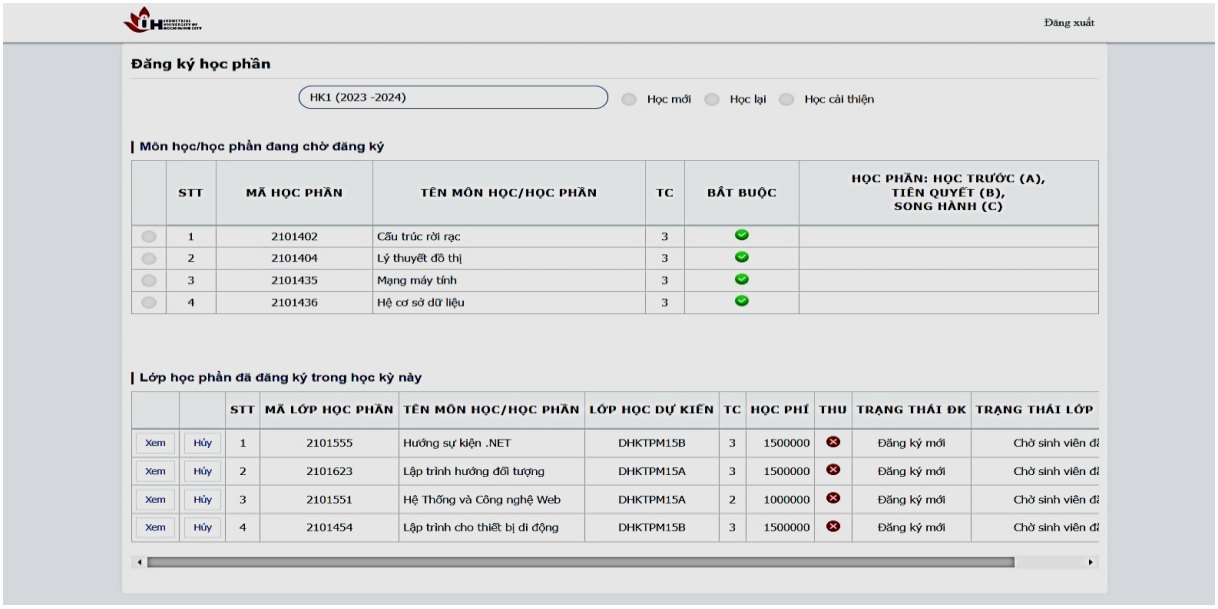
#### 3.4.1 Hiện thực hệ thống website AVTEdu

Công cụ AVTEdu đã có thể triển khai trên hai nền tảng web và mobile app. Chúng tôi xin được giới thiệu trước về nền tảng webapp. Đối với hệ thống web của AVTEdu, chúng tôi xây dựng giao diện, chức năng hướng đến 3 loại người dùng chính: sinh viên, giảng viên và giáo vụ. Sau đây là mô tả sơ lược về giao diện cho loại người dùng là sinh viên.

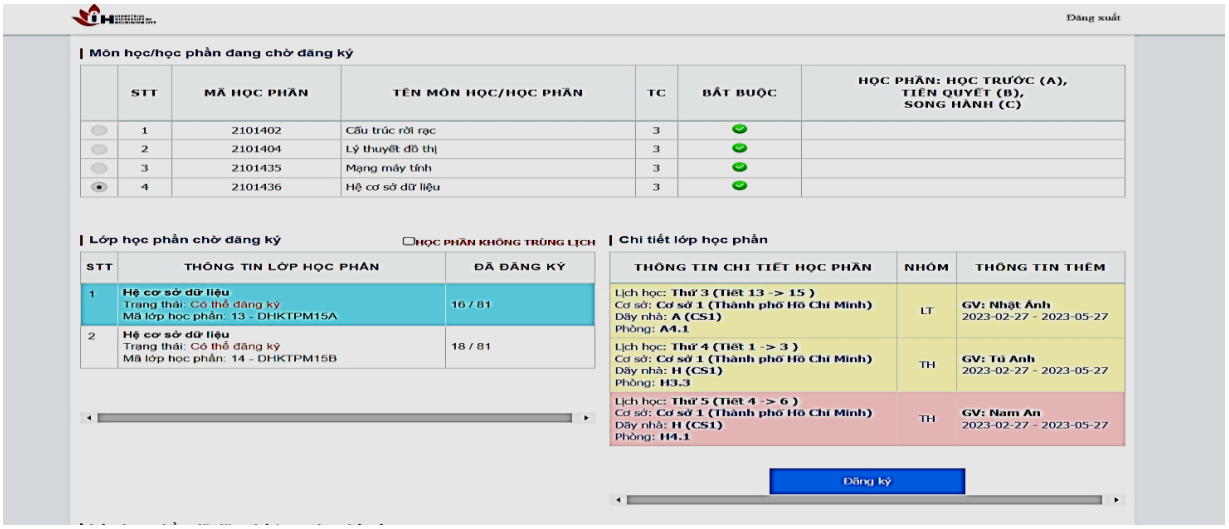
Sinh viên khi chọn vào option Đăng ký học phần để tiến hành đăng ký môn học. Đối với chức năng này, mặc định sẽ không hiển thị môn học. Sinh viên cần chọn đợt đăng ký, ở đây sinh viên sẽ được chọn theo học kỳ. Đối với học kỳ đã đăng ký và hoàn thành đầy đủ các môn học – tức học kỳ cũ, sinh viên sẽ thấy trang web hiển thị dòng thông báo "Học kỳ này không còn môn đã đăng ký" (*Hình 6*).

Sau khi chọn đợt đăng ký, tại đây sinh viên có thể thấy hệ thống đang hiển thị 2 mục chính: mục thứ nhất, là toàn bộ môn học/học phần chờ đăng ký, mục thứ hai là những lớp học phần đã đăng ký trong học kỳ tương ứng. Trên mỗi dòng thông tin lớp học phần đã đăng ký đều có tùy chọn xem chi tiết và hủy học phần. Ở bước tiếp theo, sinh viên chọn vào mục check ứng với môn/học phần muốn đăng ký. Hệ thống hiển thị toàn bộ lớp học phần thuộc học phần mà sinh viên đã chọn, đồng thời sinh viên nên tiếp tục tích chọn vào tùy chọn "Học phần không trùng lịch" để các lớp học phần chỉ hiển thị những học phần không bị trùng khớp lịch học với những môn đã đăng ký. Sinh viên tiếp tục chọn lớp học phần mong muốn, hệ thống tiếp tục hiển thị chi tiết lớp học phần. Đối với học phần có lớp thực hành, sinh viên bắt buộc chọn vào các lớp thực hành mong muốn trước khi đăng ký, nếu không hệ thống sẽ thông báo lỗi "Bạn chưa chọn lớp thực hành". Nếu đăng ký thành công, sinh viên sẽ nhận được thông báo thành công đồng thời tiếp tục theo dõi trang đăng ký sẽ thấy sự thay đổi: loại bỏ môn vừa đăng ký thành công ra khỏi danh sách môn học/học phần chờ đăng ký và thêm thông tin về học phần vừa đăng ký tại bảng lớp học phần đã đăng ký *(Hình 7).*


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh



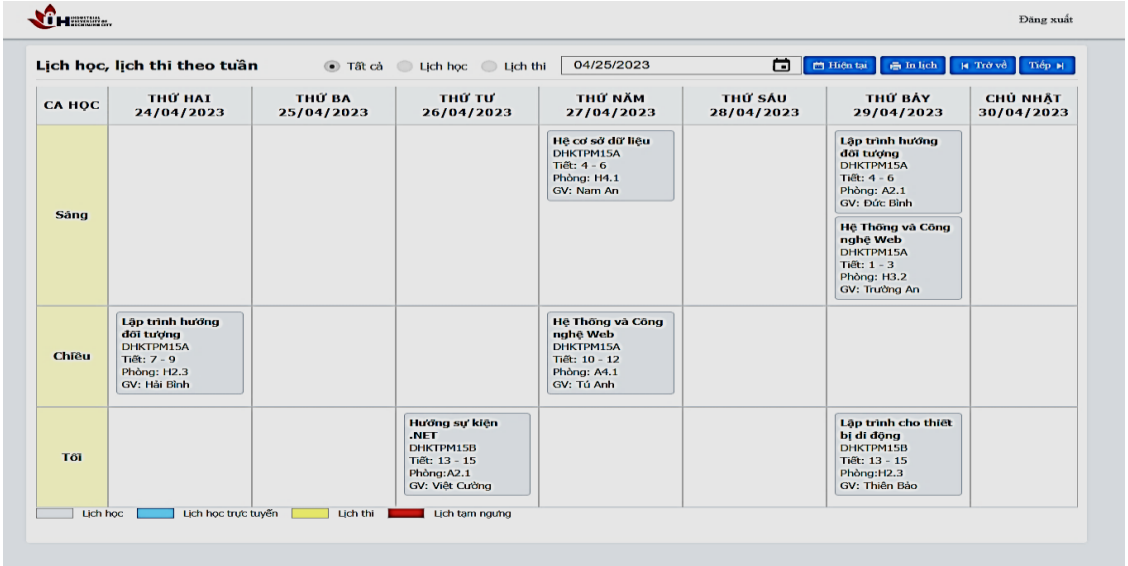

Hình 6: Môn học/học phần chờ đăng ký và đã đăng ký theo kỳ đã chọn




Hình 7: Chọn học phần và chọn lớp học phần

Sau khi đăng ký học phần thành công, sinh viên có thể chọn vào mục "lịch theo tuần" tại trang chủ để tiến hành kiểm tra lịch học của tất cả môn vừa đăng ký. Chức năng xem lịch học mặc định sẽ luôn hiển thị lịch trong 1 tuần và là tuần hiện tại, sinh viên muốn theo dõi tổng quan lịch học của khóa học có thể click chọn "tiếp theo" để xem tuần tiếp theo, "trở về" để lùi về 1 tuần hoặc "hiện tại" để hệ thống tải lịch học theo tuần hiện tại. Sinh viên cũng có thể chọn một ngày cụ thể để xem lịch học của ngày và tuần tương ứng (*Hình 8*).

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 



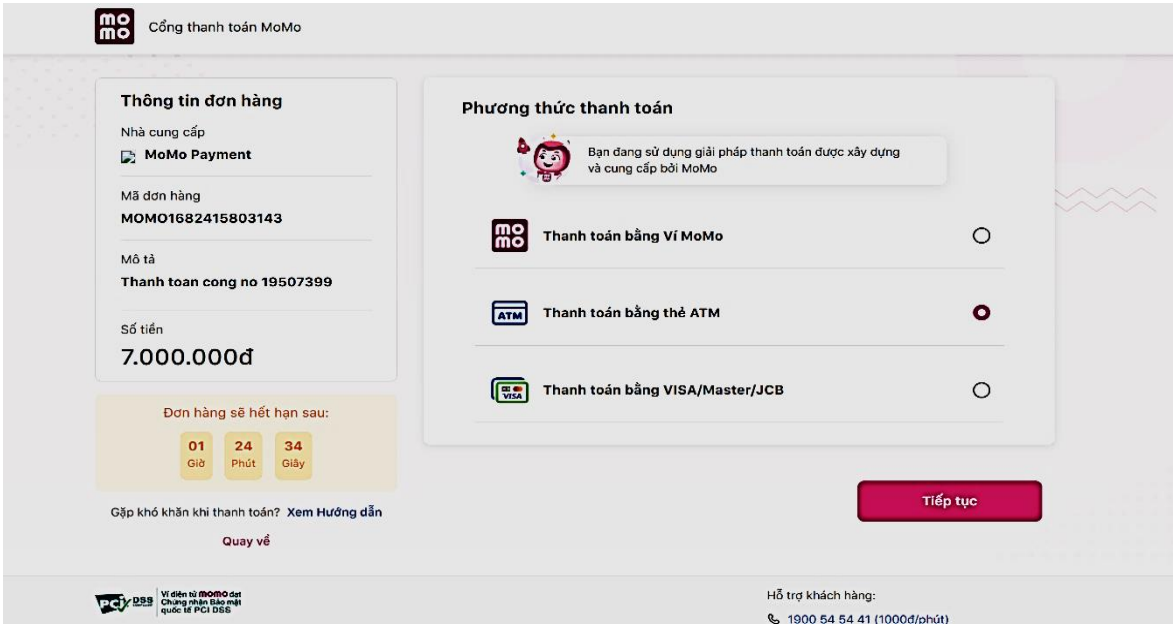

Hình 8: Xem lịch theo tuần

Sinh viên chọn vào mục xem điểm để tiến hành xem kết quả học tập theo học kỳ tương ứng.
Đối với sinh viên muốn thanh toán công nợ trực tuyến, hãy chọn vào "Thanh toán trực tuyến". Hệ thống sẽ tải các khoản cần thu cùng với số tiền cần phải thanh toán. Tại đây AVTEdu sẽ hỗ trợ một phương thức thanh toán đó là momo có liên kết với tài khoản ATM nội địa. Sinh viên sau khi click vào tùy chọn momo làm phương thức thanh toán, bắt đầu tiến hành thanh toán bằng cách click chọn vào nút thanh toán (*Hình 9*), hệ thống lập tức chuyển sang trang cổng thanh toán momo. Sinh viên cần chọn phương thức thanh toán bằng thẻ ATM và click chọn nút tiếp tục (*Hình 10*). Màn hình nhập liệu thông tin thẻ sẽ xuất hiện, sinh viên tiến hành nhập thông tin thẻ. Ở đây chúng tôi chỉ dừng lại ở mức độ test chức năng thanh toán trực tuyến và sẽ không có mục quản lý nguồn tiền, vì vậy thông tin thẻ sẽ là thông tin thẻ phục vụ cho mục đích kiểm thử và sẽ không sử dụng thông tin thẻ thật. Nếu thông tin thẻ hợp lệ và có đủ số tiền thanh toán, sinh viên có thể click vào nút thanh toán để tiến hành gạch nợ (*Hình 11*). Hệ thống hiển thị trang nhập mã xác thực OTP. Hệ thống tiến hành kiểm tra trạng thái thẻ, hạn mức thẻ và nguồn tiền.

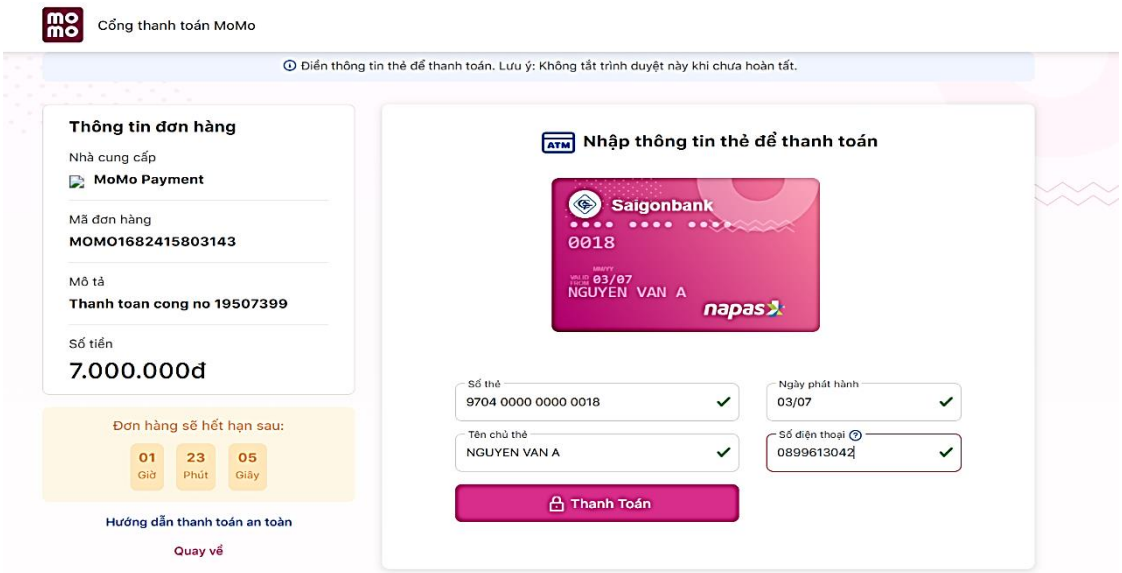



Hình 9: Giao diện thanh toán trực tuyến


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 10: Cổng thanh toán Momo chọn phương thức thanh toán



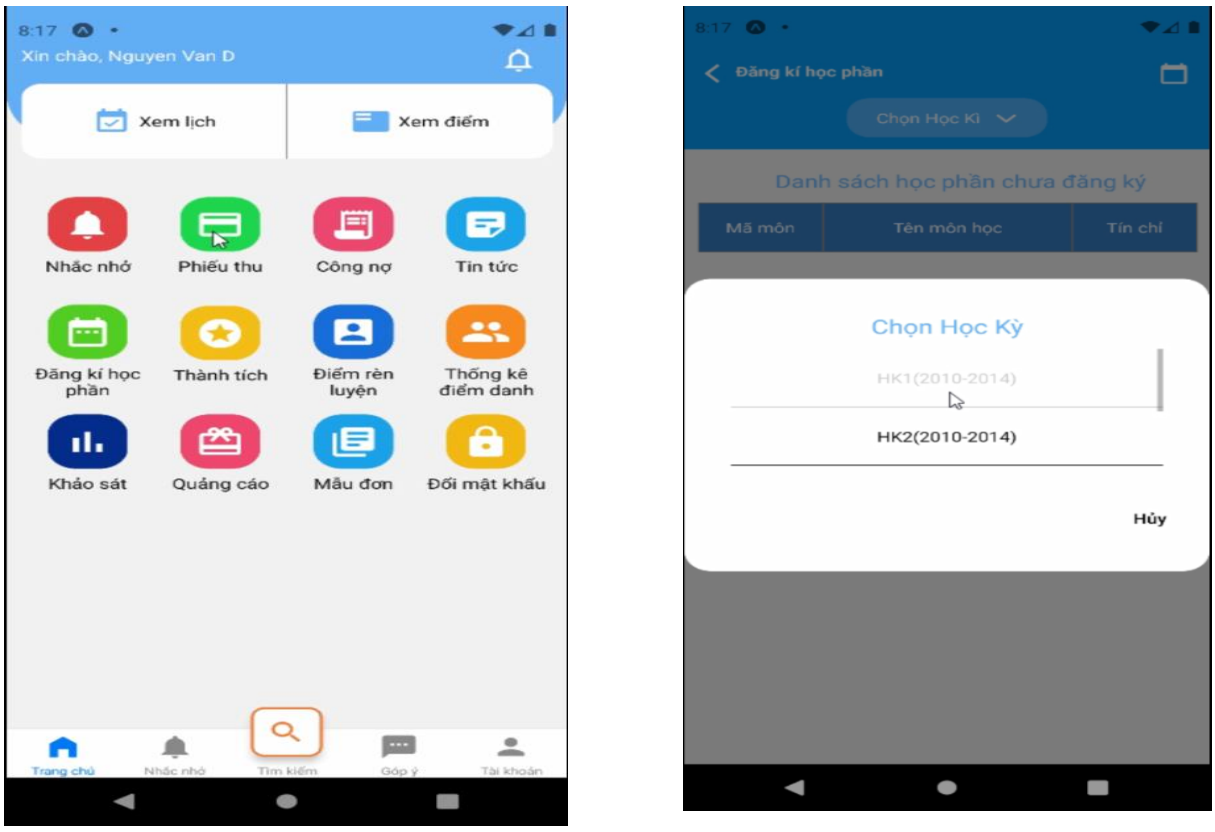

Hình 11: Trang nhập thông tin thẻ để thanh toán

### 3.4.2 Hiện thực hệ thống AVTEdu trên thiết bị di động

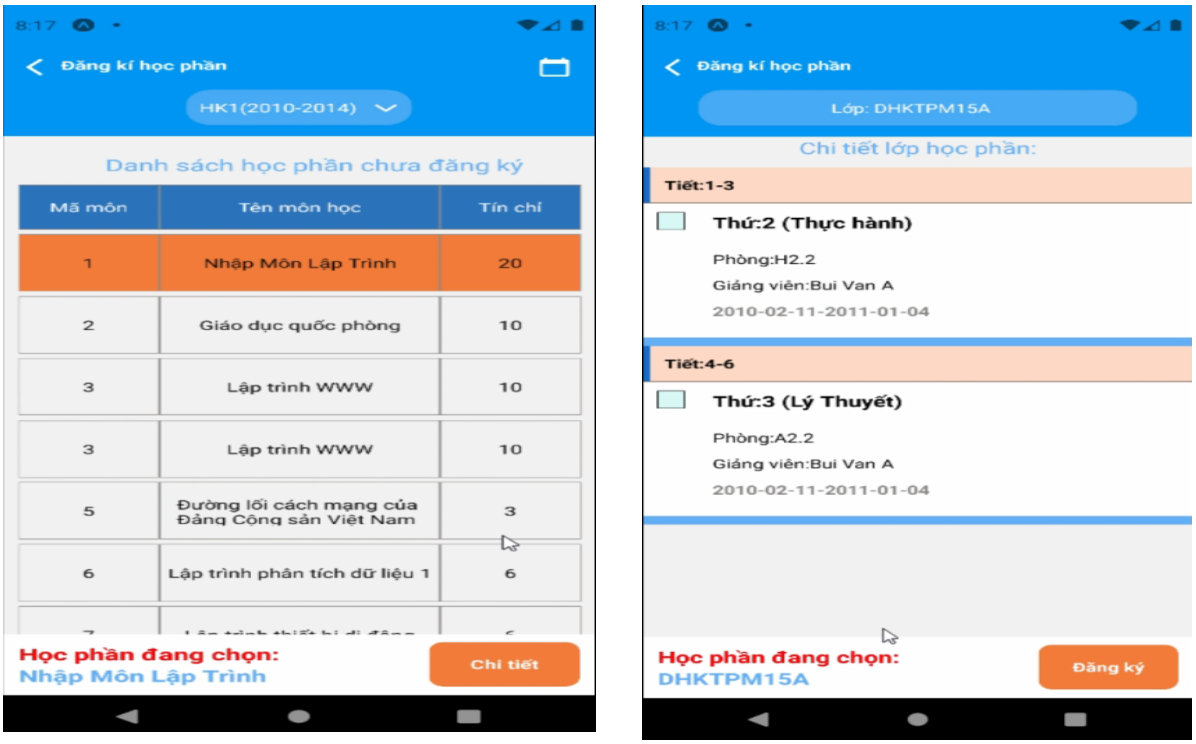
Sinh viên chọn vào chức năng đăng ký học phần và chọn học kỳ cần đăng ký (*Hình 12)*. Danh sách các lớp học phần tương ứng hiện ra, sinh viên click chọn chi tiết học phần và xem các thông tin về lớp thực hành, phòng, giảng viên và lịch học, cuối cùng đăng ký học phần (*Hình 13*).

Sau khi đăng ký học phần thành công, sinh viên có thể chọn vào mục xem lịch học. Chọn mốc thời gian cần xem để hiển thị lịch học.

Các chức năng trên hệ thống app của thiết bị di động sẽ đồng nhất với các chức năng trên Web. Sinh viên có thể dễ dàng đăng ký học phần, thanh toán công nợ cũng như thực hiện các chức năng tra cứu một cách dễ dàng và thuận tiện. Các giao diện trên app đảm bảo tính nhất quán, dễ dùng và các tính năng bảo mật cũng được đảm bảo an toàn cho người dùng.




Hình 12: Chọn học kỳ khi đăng ký học phần

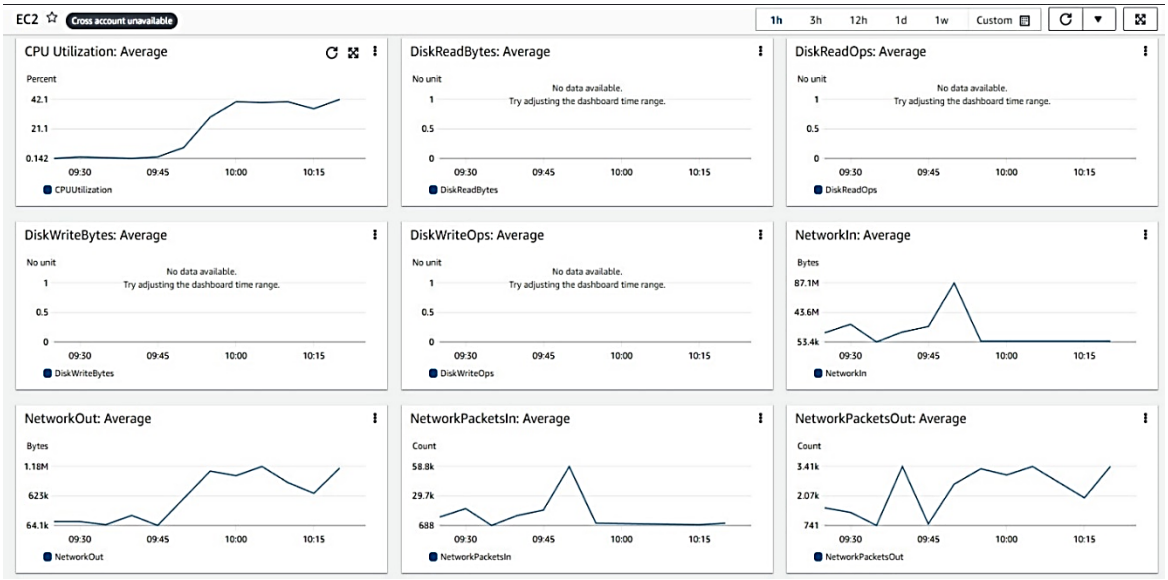



Hình 13: Chọn học phần và lớp học phần cần đăng ký


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 4 KIỂM NGHIỆM THỰC TẾ

Chúng tôi sử dụng dịch vụ EC2 của AWS để triển khai hệ thống. EC2 là một dịch vụ điện toán đám mây cho phép chạy các máy ảo (instances) linh hoạt trên nền tảng đám mây. Chúng tôi triển khai các máy chủ EC2 để xử lý và quản lý hệ thống AVTEDU. EC2 cung cấp khả năng scale cao và linh hoạt, cho phép chúng tôi tăng hoặc giảm số lượng máy chủ theo nhu cầu của hệ thống. Điều này giúp hệ thống AVTEDU hoạt động ổn định và đảm bảo hiệu suất cao trong quá trình xử lý. Đồng thời, trong quá trình thử nghiệm, chúng tôi đã ghi nhận và kiểm tra các thông số sử dụng tài nguyên trong EC2 trong các tình huống xử lý đồng thời nhiều giao dịch như đăng ký học phần. Việc này giúp chúng tôi xác định và điều chỉnh cấu hình ECS Cluster phù hợp với nhu cầu và tải của hệ thống AVTEDU (*Hình 14*).




Hình 14: Các thông số Mornitoring của hệ thống được triển khai

Bảng 1: Bảng ý nghĩa của các thông số Mornitoring

| | |
|---|---|
| **Disk reads (bytes)** | Các byte được đọc từ tất cả các khối lượng lưu trữ ứng dụng có sẵn cho ứng dụng. Số liệu này được sử dụng để xác định khối lượng dữ liệu mà ứng dụng đọc từ đĩa cứng. Điều này có thể được sử dụng để xác định tốc độ của ứng dụng. Nếu không có khối lượng lưu trữ nào, thì giá trị là 0 hoặc số liệu không được báo cáo.<br>Đơn vị: Byte |
| **Disk, read operations (operations)** | Các thao tác đọc đã hoàn thành từ tất cả các khối lượng lưu trữ ứng dụng có sẵn cho ứng dụng trong một khoảng thời gian cụ thể. Nếu không có khối lượng lưu trữ nào, thì giá trị là 0 hoặc số liệu không được báo cáo.<br>Đơn vị: count |
| **Disk writes (bytes)** | Các byte được ghi vào tất cả các khối lượng lưu trữ có sẵn cho ứng dụng đó. Số liệu này được sử dụng để xác định khối lượng dữ liệu mà ứng dụng ghi vào đĩa cứng. Điều này có thể được sử dụng để xác định tốc độ của ứng dụng. Nếu không có khối lượng lưu trữ nào, thì giá trị là 0 hoặc số liệu không được báo cáo.<br>Đơn vị: Byte |
| **Disk, write operations (operations)** | Các thao tác ghi đã hoàn thành vào tất cả khối lượng lưu trữ có sẵn cho ứng dụng trong một khoảng thời gian cụ thể. Nếu không có khối lượng lưu trữ nào, thì giá trị là 0 hoặc số liệu không được báo cáo. Đơn vị: count |
| **Network in (bytes)** | Số byte mà ứng dụng sẽ nhận được trên tất cả các giao diện mạng. Số liệu này xác định khối lượng lưu lượng mạng đến cho một ứng dụng duy nhất. Đơn vị: Byte |
| **Network out (bytes)** | Số byte được gửi bởi ứng dụng trên tất cả các giao diện mạng. Số liệu này xác định khối lượng lưu lượng mạng đi từ một ứng dụng duy nhất. |

| | |
|---|---|
| | Đơn vị: Byte |
| **Network packets in (count)** | Số lượng gói tin mà ứng dụng nhận được trên tất cả các giao diện mạng. Số liệu này xác định khối lượng lưu lượng đến về số lượng gói trên một ứng dụng. Đơn vị: count |
| **Network packets out (count)** | Số lượng gói tin mà ứng dụng gửi đi trên tất cả các giao diện mạng. Số liệu này xác định khối lượng lưu lượng đến về số lượng gói trên một ứng dụng. Đơn vị: count |

Ngoài ra, chúng tôi cũng triển khai phiên bản di động (mobile) của hệ thống trên Google Play và AppStore với tên gọi AVTEdu-App, nhằm mang lại sự tiện lợi và thuận tiện cho sinh viên trong việc sử dụng hệ thống AVTEDU.

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Bài báo này chúng tôi trình giải pháp cho bài toán tắc nghẽn cho các giao dịch trực tuyến mà sinh viên gặp phải trong thời gian qua tại trường Đại học Công nghiệp Tp.HCM. Giải pháp mà chúng tôi đề xuất là xây dựng một hệ thống gồm Web và ứng dụng trên thiết bị di động. Bên cạnh đó, chúng tôi cũng bổ sung cho các chức năng quản lý công nợ của sinh viên nhằm tăng cường tiện ích cho sinh viên. Chúng tôi đã xây dựng công cụ AVTEdu với các công nghệ chọn lựa phù hợp nhằm giải quuyết tối ưu của bài toán. Trong đó chúng tôi phân chia hệ thống thành hai phân hệ gồm Web và App trên di động. Chúng tôi triển khai hệ thống trên môi trường Cloud với các dịch vụ hỗ trợ như Load Balancer của AWS nhằm tự động hóa quá trình Scale (co giãn hệ thống phần cứng), giúp cho hệ thống trở nên ổn định và an toàn hơn. Công cụ AVTEdu cũng đã được triển khai cho sinh viên chuyên ngành Kỹ thuật phần mềm Khoa Công nghệ thông tin thử nghiệm. Trong đó, chúng tôi đã lồng ghép chức năng quản lý ví điện tử nhằm thực hiện các giao dịch thanh toán học phí online qua các kênh tài chính như Momo. Kết quả cũng được chúng tôi thu nhận, phân tích, đánh giá và hoàn thiện trong thời gian tới. Hiện công cụ AVTEdu cũng còn một số việc cần hoàn thiện như: Ứng dụng di động AVTEdu-App vẫn còn một số hạn chế về mặt tính năng, chưa có tính năng nhắc công nợ cũng như tích hợp vào hệ thống PMT sẵn có của trường. Vì thế trong tương lai, chúng tôi sẽ hoàn thiện các tính năng này để đáp ứng đầy đủ nhu cầu cho sinh viên trường Đại học Công nghiệp Tp.HCM. Chúng tôi sẽ triển khai chức năng thông báo theo cá nhân hóa trên ứng dụng di động nhằm giúp người dùng trải nghiệm với hệ thống một cách thân thiện và an toàn hơn.

## TÀI LIỆU THAM KHẢO

[1] Narula, S., & Jain, A. Cloud computing security: Amazon web service. Fifth International Conference on Advanced Computing & Communication Technologies (pp. 501-505). IEEE.(2015)

[2] L. M. Mario Casciaro, Node.js Design Patterns: Design and implement production-grade Node.js applications using proven patterns and techniques, 3rd Edition.

[3][Online]. Available: https://www.fusionhit.com/technologies/nodejs/. [Accessed 06 May 2023].

[4] Bangare, S., Gupta, S., Dalal, M., & Inamdar. Using Node. Js to build high speed and scalable backend database server. International Journal of Research in Advent Technology, 4, 19. (2016).

[5] B. Eisenman, Learning React Native: Building Native Mobile Apps with JavaScript, 2nd Edition, (2017).

[6] R. Wieruch, The Road to React: The React.js with Hooks in JavaScript Book, (2023 Edition) Kindle Edition.

[7] J. Carlson, Redis in Action, Foreword by Salvatore Sanfilippo.

[8] Macedo, T., & Oliveira, F. . Redis cookbook: Practical techniques for fast data manipulation. O'Reilly Media, Inc. .(2011)

[9][Online]. Available: https://topdev.vn/blog/redis-la-gi/.[Accessed 06 May 2023].

[10] Ferreira, F., Borges, H. S., & Valente, M. T. . On the adoption of JavaScript front-end frameworks. Software: Practice and Experience, 52(4), 947-966.(2022)

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F016**

# AMANDA- CÔNG CỤ HỖ TRỢ MUA SẮM TRỰC TUYẾN TRONG LĨNH VỰC THƯƠNG MẠI ĐIỆN TỬ DỰA TRÊN CÔNG NGHỆ BLOCKCHAIN

TÔN LONG PHƯỚC[1*], HOÀNG VĂN CHINH[1], NGUYỄN THỊ HẢO[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
** tonlongphuoc@iuh.edu.vn*

**Tóm tắt.** Trong kỷ nguyên số hiện nay, mua sắm trên các trang thương mại điện tử đang được xem là xu hướng. Hiện có rất nhiều ứng dụng thương mại điện tử khác nhau nhưng hầu hết đều sử dụng các hình thức thanh toán truyền thống như: qua ngân hàng, ví điện tử hoặc bằng tiền mặt. Trong quá trình thanh toán truyền thống đó, người dùng thường phải mất nhiều thời gian và công sức để thực hiện các bước liên quan đến thanh toán, bao gồm liên kết tài khoản, trả tiền mặt hoặc sử dụng thẻ tín dụng cũng như ký xác nhận thanh toán. Điều này gây bất tiện và lãng phí thời gian cho người dùng. Ngoài ra, các phương thức thanh toán truyền thống cũng phải đối mặt với các vấn đề về an toàn và bảo mật. Ngoài ra, thanh toán truyền thống còn phải đối mặt với vấn đề minh bạch và thống nhất cam kết giữa các bên. Người dùng có thể khó theo dõi và kiểm tra các giao dịch của mình một cách đáng tin cậy, đặc biệt là khi các giao dịch được thực hiện trên nhiều nền tảng khác nhau. Công cụ Amanda mà chúng tôi giới thiệu sẽ giải quyết các vấn đề trên. Amanda hỗ trợ các giao dịch sử dụng tiền số để thanh toán dựa vào Blockchain. Ngoài ra, công cụ sẽ lưu lại tất cả các nhật ký giao dịch bằng Blockchain, giúp đảm bảo tính minh bạch của hệ thống. Công cụ triển khai dựa trên một số công nghệ cốt lõi như NodeJS, ReactNative, ReactJS và một số dịch vụ tiền số trong việc quản lý ví điện tử. Hiện tại, Amanda được triển khai trên nền tảng internet với các giao dịch thử nghiệm thực hiện trên các đồng tiền số hiện có trên thị trường. Chúng tôi sẽ mở rộng các chức năng cũng như quảng bá tới người dùng công cụ này trong thời gian tới.

**Từ khóa.** Thương mại điện tử, Bảo mật, Blockchain, Tiền số, NodeJS, ReactJS.

## AMANDA- TOOL TO SUPPORT ONLINE SHOPPING IN E-COMMERCE ON BLOCKCHAIN TECHNOLOGY

**Abstract.** Shopping on e-commerce sites is now a trend in the era of digital transformation. Currently, there are many different e-commerce applications, but most of them use traditional forms of payment such as: via bank, e-wallet, or in cash. During that traditional payment process, users often have to spend a lot of time and effort to go through the steps related to payment, including linking accounts, paying cash or using credit cards, and Sign the payment receipt. This causes inconvenience and waste of time for users. In addition, traditional payment methods also face safety and security issues. In addition, the traditional payment also faces the problem of transparency and uniformity. It can be difficult for users to reliably track and check their transactions, especially when transactions are made on many different platforms. The Amanda tool that we recommend will solve the above problems. Amanda supports transactions using cryptocurrencies for blockchain payments. The implementation tool relies on some core technologies such as NodeJS, React-Native, ReactJS and some crypto services in managing the e-wallets. Currently, Amanda is deployed on the internet with experimental transactions. We will expand the functions as well as promote to users of this tool in the future.

**Keywords.** E-commerce, Security, Blockchain, Cryptocurrency, NodeJS, ReactJS.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

# 1 GIỚI THIỆU TỔNG QUAN

Hiện nay, thương mại điện tử đã trở thành một lĩnh vực phát triển nhanh chóng và thu hút sự quan tâm lớn từ cộng đồng người dùng trên toàn thế giới. Việc mua sắm trực tuyến đã trở thành một xu hướng thịnh hành, giúp người dùng tiết kiệm thời gian và tiện lợi hơn trong việc tìm kiếm và mua các sản phẩm và dịch vụ. Tuy nhiên, trong quá trình thanh toán trực tuyến, người dùng vẫn gặp phải nhiều thách thức và khó khăn. Các phương thức thanh toán truyền thống, bao gồm việc liên kết tài khoản ngân hàng, sử dụng ví điện tử hoặc thanh toán bằng tiền mặt, thường gây bất tiện và tốn nhiều thời gian. Hơn nữa, các vấn đề về an toàn, bảo mật và tính minh bạch trong quá trình thanh toán vẫn là mối quan ngại hàng đầu của người dùng. Nhằm giải quyết những thách thức này, một công cụ thanh toán tiền số tiên tiến đã xuất hiện - Amanda. Amanda được phát triển dựa trên công nghệ Blockchain, cung cấp một phương thức thanh toán mới mẻ và đột phá cho thương mại điện tử. Với Amanda, người dùng có thể tiến hành các giao dịch thanh toán sử dụng tiền số một cách dễ dàng và nhanh chóng. Công cụ này không chỉ giúp tiết kiệm thời gian và công sức cho người dùng, mà còn đảm bảo tính an toàn và bảo mật thông tin cá nhân và tài khoản ngân hàng. Quan trọng hơn, Amanda lưu trữ tất cả các giao dịch trong Blockchain, đảm bảo tính minh bạch và đáng tin cậy của hệ thống. Được triển khai trên nền tảng internet, Amanda hứa hẹn sẽ là công cụ được ưa chuộng đối với cộng đồng người dùng thương mại điện tử. Công cụ này đang tiếp tục mở rộng chức năng và nghiên cứu để đáp ứng nhu cầu ngày càng tăng của người dùng. Sự kết hợp giữa công nghệ Blockchain (block), NodeJS, ReactNative và ReactJS đã mang lại một công cụ thanh toán tiền số tiên tiến và đáng tin cậy cho thương mại điện tử hiện đại. Trong bối cảnh thương mại điện tử ngày càng phát triển và đòi hỏi sự tiện lợi và an toàn trong quá trình thanh toán, Amanda đóng vai trò quan trọng trong việc cung cấp một phương thức thanh toán mới dựa trên tiền số và Blockchain. Công cụ này hứa hẹn mang lại sự tiện ích và tin cậy cho người dùng, đồng thời đóng góp vào sự phát triển và thúc đẩy sự lan rộng của thương mại điện tử trong tương lai.

Bài báo của chúng tôi chia làm 5 phần: Phần 1 Giới thiệu tổng quan về hệ thống; Phần 2 Các công nghệ liên quan; Phần 3 Công cụ Amanda; Phần 4 Kiểm nghiệm thực tế và Phần 5 Kết luận và hướng phát triển. Trong đó Phần 3 là phần đóng góp chính của bài báo.

# 2 CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 Những framework hỗ trợ phát triển ứng dụng

**a. NestJS Framework**

Nest (NestJS) là một framework để xây dựng các ứng dụng phía máy chủ NodeJS hiệu quả, có thể mở rộng. NestJS sử dụng ngôn ngữ lập trình JavaScript, nhưng lại xây dựng và hỗ trợ đầy đủ cho ngôn ngữ TypeScript (Lập trình viên vẫn có thể viết bằng JavaScript) [1]. NestJS được tích hợp thêm lập trình hướng đối tượng (OOP) giúp cho việc xây dựng và tổ chức mã nguồn trở nên dễ dàng và rõ ràng hơn rất nhiều so với Express framework. Nest sử dụng các framework HTTP Server mạnh mẽ như Express (mặc định) và tùy chọn cũng có thể được định cấu hình để sử dụng Fastify. Nest cung cấp một mức trừu tượng hơn các framework Node.js phổ biến như Express/Fastify, nhưng cũng cho phép trực tiếp truy cập vào các API của chúng [1]. Điều này cho phép nhà phát triển tự do sử dụng nhiều module của bên thứ ba có sẵn cho nền tảng cơ bản.

**b. ReactJS**

ReactJS là một thư viện JavaScript mã nguồn mở được phát triển bởi Facebook [2]. Nó được sử dụng để xây dựng giao diện người dùng động và tương tác trong ứng dụng web. ReactJS giúp nhà phát triển tạo ra các thành phần UI đơn giản và dễ quản lý, giúp tăng hiệu suất và tái sử dụng mã. Chức năng chính của ReactJS là tạo ra một cách tiếp cận linh hoạt và hiệu quả trong việc xây dựng giao diện người dùng phản ứng (reactive) [2]. Nó cho phép nhà phát triển phân chia giao diện thành các thành phần độc lập và tái sử dụng được, giúp quản lý trạng thái và hiệu suất ứng dụng một cách hiệu quả.

**c. React-Native**

React Native là một framework mã nguồn mở dựa trên ReactJS, được phát triển bởi Facebook. Nó cho phép nhà phát triển xây dựng ứng dụng di động đa nền tảng bằng việc sử dụng JavaScript. Thay vì viết mã cho từng nền tảng riêng biệt như Android và iOS, React Native cho phép tái sử dụng mã nguồn và logic của ứng dụng trên cả hai nền tảng này [3].

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 2.2 Cloud computing

**a. EC2**

Amazon EC2 (Elastic Compute Cloud) là một dịch vụ điện toán đám mây do Amazon Web Services (AWS) cung cấp. EC2 cho phép người dùng thuê và triển khai các máy ảo (instances) trên cơ sở hạ tầng điện toán đám mây của Amazon [4].

EC2 cung cấp một môi trường linh hoạt cho việc chạy ứng dụng và dịch vụ trực tuyến. Người dùng có thể chọn loại và cấu hình của máy ảo, và sau đó sử dụng máy ảo này để chạy các ứng dụng, xử lý dữ liệu, lưu trữ và quản lý các tài nguyên điện toán của họ.

**b. Cloudinary**

Cloudinary là một dịch vụ quản lý hình ảnh và video trực tuyến. Nó cung cấp một nền tảng đám mây mạnh mẽ cho việc lưu trữ, xử lý và quản lý các tệp đa phương tiện. Với Cloudinary, người dùng có thể tải lên, xử lý và hiển thị hình ảnh và video trên ứng dụng web hoặc di động một cách dễ dàng [5].

Cloudinary cung cấp nhiều tính năng và chức năng hữu ích để làm việc với các tệp đa phương tiện, bao gồm: lưu trữ đám mây, xử lý hình ảnh và video, quản lý tài nguyên đa phương tiện, tích hợp dễ dàng.

**c. Redis Cloud**

Redis Cloud là một dịch vụ đám mây được cung cấp bởi Redis Labs, cho phép người dùng triển khai và quản lý cơ sở dữ liệu Redis trên môi trường đám mây. Redis là một hệ thống cơ sở dữ liệu khóa-giá trị (key-value) in-memory, nhanh chóng và linh hoạt, được sử dụng rộng rãi trong các ứng dụng yêu cầu xử lý dữ liệu nhanh và lưu trữ tạm thời [6].

## 2.3 BlockChain

Blockchain ban đầu là một cơ sở dữ liệu phân cấp lưu trữ thông tin trong các khối thông tin được liên kết với nhau bằng mã hóa và mở rộng theo thời gian. Mỗi khối thông tin đều chứa thông tin về thời gian khởi tạo và được liên kết tới khối trước đó, kèm một mã thời gian và dữ liệu giao dịch. Blockchain được thiết kế để chống lại sự thay đổi của dữ liệu: Một khi dữ liệu đã được mạng lưới chấp nhận thì sẽ không có cách nào thay đổi được nó [7]. Blockchain được đảm bảo nhờ cách thiết kế sử dụng hệ thống tính toán phân cấp với khả năng chịu lỗi cao. Nhờ thế nên Blockchain có thể đạt được sự đồng thuận phân cấp. Vì vậy Blockchain phù hợp để ghi lại những sự kiện, hồ sơ y tế, xử lý giao dịch, công chứng, danh tính và chứng minh nguồn gốc. Việc này có tiềm năng giúp xóa bỏ các hậu quả lớn khi dữ liệu bị thay đổi trong bối cảnh thương mại toàn cầu [8]. Blockchain lần đầu tiên được phát minh và thiết kế bởi Satoshi Nakamoto vào năm 2008 và được hiện thực hóa vào năm sau đó như là một phần cốt lõi của Bitcoin, khi công nghệ blockchain đóng vai trò như là một cuốn sổ cái cho tất cả các giao dịch. Qua việc sử dụng mạng lưới ngang hàng và một hệ thống dữ liệu phân cấp, Bitcoin blockchain được quản lý tự động. Việc phát minh ra blockchain cho Bitcoin đã làm cho nó trở thành loại tiền tệ kỹ thuật số đầu tiên giải quyết được vấn đề double spending (chi tiêu gian lận khi một lượng tiền được dùng 2 lần). Công nghệ này của Bitcoin đã trở thành nguồn cảm hứng cho một loạt các ứng dụng khác. Trong hệ thống *Amanda*, chúng tôi sử dụng Blockchain để lưu lại các lịch sử giao dịch, các hợp đồng. Dựa vào lịch sử giao dịch đó, hệ thống có thể truy ra được các vết của các khối block trước đó. Điều này có thể đảm bảo được tính toàn vẹn của dữ liệu. Khi xuất ra các lịch sử giao dịch, hệ thống sẽ xem xét các khối block có hợp lệ hay không? Có nghĩa là hệ thống sẽ kiểm tra mã hash của toàn bộ khối. Nếu một Blockchain hợp lệ - tất cả mã hash trước của khối là hợp lệ thì có thể xác định, các giao dịch đó là hợp lệ. Đồng thời, hợp đồng giao dịch đó cũng là hợp lệ và việc khớp lệnh cho các giao dịch này được thực thi trên các ví điện tử [9].

## 2.4 Hardhat

Hardhat là một công cụ phát triển hợp đồng thông minh (smart contract) dựa trên Ethereum. Nó cung cấp môi trường phát triển và bộ công cụ để xây dựng, triển khai và kiểm thử các ứng dụng và hợp đồng thông minh trên blockchain Ethereum [10]. Hardhat hỗ trợ việc viết mã thông minh bằng ngôn ngữ Solidity và TypeScript, và cung cấp các chức năng như biên dịch, triển khai, kiểm thử và mô phỏng trong môi trường phát triển cục bộ. Nó giúp các nhà phát triển tạo ra các ứng dụng phi tập trung (dApps) và hợp đồng thông minh một cách dễ dàng và hiệu quả. Hardhat cung cấp tích hợp sẵn với các thư viện phổ biến như Truffle và Ganache, và hỗ trợ quy trình phát triển có cấu trúc, giúp tăng cường sự tin cậy và sự linh hoạt trong quá trình phát triển ứng dụng blockchain [10].

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

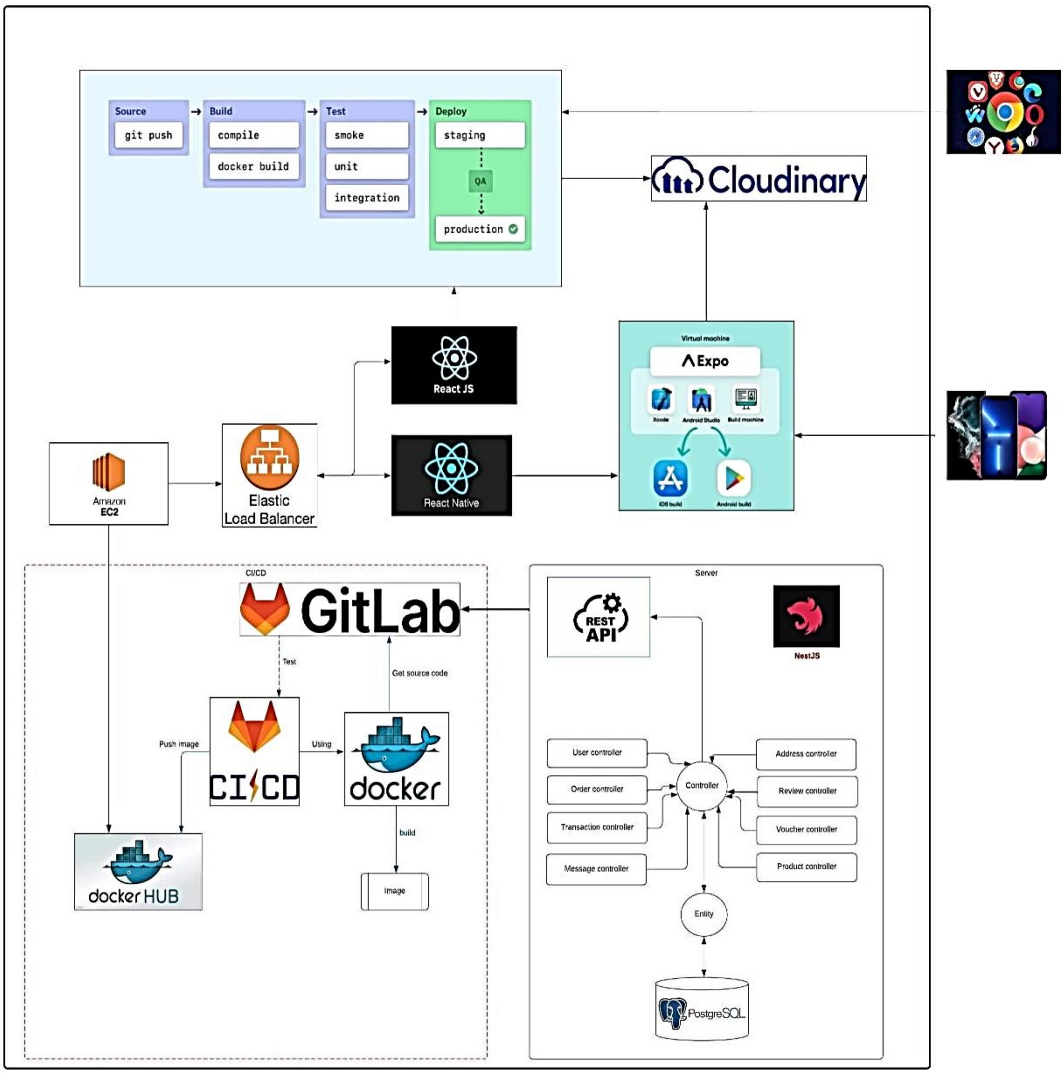
# 3 AMANDA: CÔNG CỤ HỖ TRỢ MUA SẮM TRỰC TUYẾN TRONG LĨNH VỰC THƯƠNG MẠI ĐIỆN TỬ DỰA TRÊN CÔNG NGHỆ BLOCKCHAIN

## 3.1 Giới thiệu

Để triển khai các giải pháp mà chúng tôi nêu ra trong phần giới thiệu, chúng tôi đề xuất xây dựng ứng dụng *Amanda*. Ứng dụng được triển khai dựa vào các công nghệ như NestJS, ReactJS/React-Navtive và Hardhat. Đồng thời, để ứng dụng *Amanda* người dùng (người thuê và cho thuê), chúng tôi cũng dùng các dịch vụ như AWS và Gitlab CI/CD. Hệ thống *Amanda* gồm ứng dụng trên điện thoại (mobile app) và hệ thống website. Trong đó, React Native tạo ứng dụng trên điện thoại hệ điều hành Android/iOS, phục vụ cho người dùng. ReactJS tạo website với các chức năng và dữ liệu đồng bộ với mobile app.

Điểm nhấn ở *Amanda* đó là sử dụng tiền điện tử cho các giao dịch trực tuyến. Khác với đa số những ứng dụng thương mại điện tử hiện tại ở Việt Nam. Bước đầu *Amanda* sẽ tập trung xây dựng lên một hệ thống với độ bảo mật, an toàn và đáng tin cậy cũng như độ linh hoạt giúp khách hàng giải quyết các vấn đề phát sinh một cách nhanh chóng nhất có thể. Mọi thứ được lưu ở blockchain dường như đã quá rõ ràng và minh bạch nên *Amanda* sẽ tập trung vào tự động hóa các vấn đề thay vì cần đến sự can thiệp của con người quá nhiều sẽ làm mất thời gian cũng như công sức của khách hàng để chờ đợi khi phát sinh các vấn đề liên quan đến các giao dịch trực tuyến trên hệ thống.

## 3.2 Kiến trúc phần mềm




Hình 1: Kiến trúc của Amanda


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Ứng dụng thương mại điện tử Amanda sử dụng kiến trúc Client – Server (*Hình 1*). Client – Server là một cấu trúc ứng dụng phân tán mà ở đó công việc được phân chia giữa bên cung cấp tài nguyên hoặc dịch vụ, được gọi là máy chủ, và bên yêu cầu dịch vụ, được gọi là máy khách. Thường thì các máy khách và máy chủ giao tiếp với nhau thông qua mạng máy tính trên các phần cứng riêng biệt, nhưng cả máy khách và máy chủ cũng có thể nằm trên cùng một hệ thống. Một máy chủ nối chủ chạy một hoặc nhiều chương trình máy chủ, đồng thời chia sẻ tài nguyên với các máy khách. Một máy khách thường không chia sẻ bất kì tài nguyên nào của nó cả, mà chỉ yêu cầu nội dung hoặc dịch vụ từ một máy chủ. Vì vậy, các máy khách sẽ thiết lập các phiên giao tiếp với máy chủ, và máy chủ sẽ chờ các yêu cầu đến.

Client: được coi như là người sử dụng các dịch vụ trên mạng do một hoặc nhiều máy chủ cung cấp và máy chủ được coi như là người cung cấp dịch vụ để trả lời các yêu cầu của các máy khác. Điều quan trọng là phải hiểu được vai trò hoạt động của nó trong một mô hình cụ thể, một máy khách trong mô hình này lại có thể là máy chủ trong một mô hình khác. Ví dụ cụ thể như một máy trạm làm việc như một máy khách bình thường trong mạng LAN nhưng đồng thời nó có thể đóng vai trò như một máy in chủ (printer server) cung cấp dịch vụ in ấn từ xa cho nhiều người khác (clients) sử dụng.

Server: là một máy tính nhiều người sử dụng (multiuser computer). Vì một máy chủ phải quản lý nhiều yêu cầu từ các máy khách trên mạng cho nên nó hoạt động sẽ tốt hơn nếu hệ điều hành của nó là đa nhiệm với các tính năng hoạt động độc lập song song với nhau như hệ điều hành UNIX, Windows. Server cung cấp và điều khiển các tiến trình truy cập vào tài nguyên của hệ thống. Các ứng dụng chạy trên máy chủ phải được tách rời nhau để một lỗi của ứng dụng này không làm hỏng ứng dụng khác. Tính đa nhiệm đảm bảo một tiến trình không sử dụng toàn bộ tài nguyên hệ thống.

### 3.3 Phân tích hệ thống Amanda

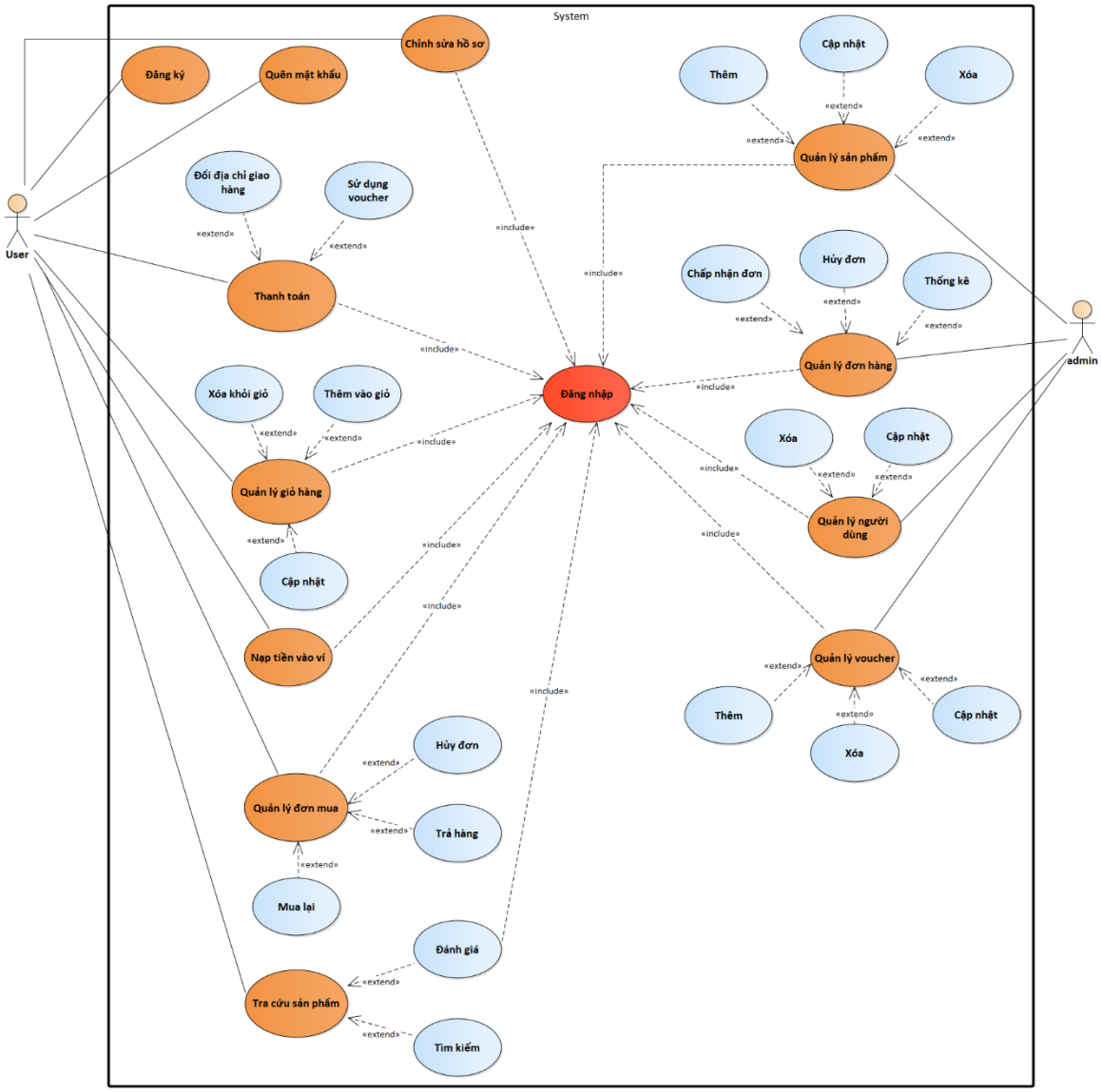
Đối tượng người dùng cho hệ thống Amanda gồm có chủ cửa hàng và người mua hàng. Trong đó chúng tôi xây dựng hệ thống tập trung vào các chức năng chính giành cho người mua hàng. Những chức năng này chúng tôi trình bày trong *Bảng 1*.

Bảng 1: Danh sách các chức năng trong Amanda

| Đối tượng | Tên chức năng | Mô tả |
|---|---|---|
| **Khách hàng** | Mua hàng | Người dùng thực hiện chọn sản phẩm, sau đó thêm vào giỏ hàng rồi thực hiện các bước xác thực để thanh toán qua ví Metamask. |
| | Nạp tiền vào ví | Amanda xây dựng một loại tiền riêng biệt sử dụng nội bộ đó là XU, khách hàng sẽ phải dùng ETH để nạp XU vào ví của mình.. |
| | Quản lý hồ sơ | Khách hàng có thể thực hiện các chức năng để cập nhật thông tin cá nhân của mình như số điện thoại, tên, địa chỉ giao hàng, email cá nhân,… |
| | Nhắn tin với chủ cửa hàng | Trao đổi thông tin theo thời gian thực với chủ cửa hàng. |
| | Đăng ký tài khoản | Đăng ký tài khoản để mua sắm. |
| | Đăng nhập | Đăng nhập để thực hiện được toàn bộ chức năng dành cho khách hàng. |
| **Chủ cửa hàng** | Quản lý sản phẩm | Thực hiện các chức năng thêm, xóa, sửa các sản phẩm được bán trong gian hàng. |
| | Quản lý hóa đơn | Chuyển đổi trạng thái của các đơn hàng được người dùng đặt mua. |
| | Quản lý khách hàng | Quản lý. |
| | Nhắn tin với khách hàng | Trao đổi thông tin với khách hàng theo thời gian thực. |

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Từ những chức năng trên, chúng tôi tập trung hiện thực các chức năng cho hệ thống ứng dụng Amanda từng bước, và bước đầu sẽ tập trung vào hai chức năng chính cho khách hàng sử dụng đó là: mua hàng và nạp tiền vào ví. Về phần chủ cửa hàng chúng tôi tập trung xây dựng các chức năng chính trước như là quản lý hóa đơn, quản lý sản phẩm, quản lý khách hàng, chạt (*Hình 2*).
Hệ thống kết hợp với một ứng dụng hỗ trợ quản lý tiền điện tử cũng như xác thực các giao dịch đó là ví điện tử Metamask. Vào thời điểm hiện tại, Metamask là một trong những ví tiền điện tử phổ biến nhất dùng để quản lý và giao dịch các tài sản kỹ thuật số trên các ứng dụng phi tập trung (Decentralized Applications - DApps) chạy trên blockchain Ethereum.




Hình 2: Mô hình Use-case của Amanda

Để có thể đăng nhập được Metamask chúng ta cần những đoạn mã bí mật theo thứ tự với độ phức tạp cao mà chỉ có người đăng ký tài khoản mới có thể có được những đoạn mã bí mật này. Nên việc mạo danh hay ăn cắp tài khoản gần như là không thể, nên người dùng có thể yên tâm khi sử dụng Metamask để quản lý cũng như để xác thực các giao dịch bằng tiền điện tử một cách an toàn và bảo mật.
Sau khi xác thực giao dịch bằng ví Metamask hệ thống còn sử dụng thêm các Smart Contract để tự động hóa các giao dịch và thỏa thuận giữa các bên. Smart contract là một chương trình tự thực thi dựa trên blockchain, giúp đảm bảo rằng các điều khoản trong hợp đồng được tuân thủ một cách chính xác và không thể thay đổi sau khi đã được thiết lập. Với điều này thì các giao dịch của khách hàng sẽ luôn luôn chính xác

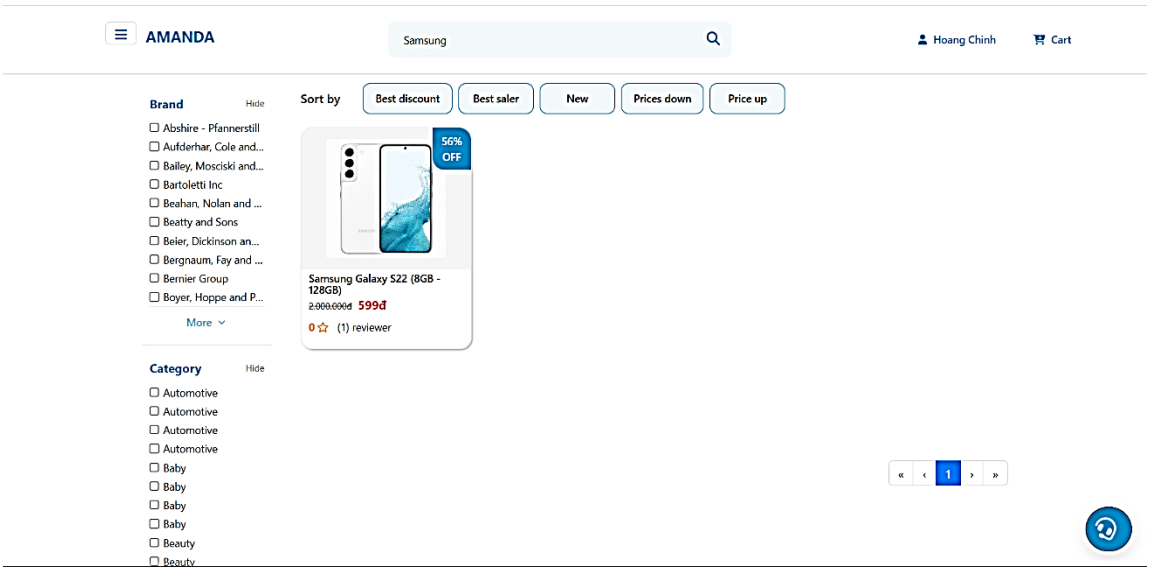
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

và toàn vẹn. Nếu xảy ra những sự tranh chấp chúng ta hoàn toàn có thể truy vết các giao dịch này ở mạng lưới blockchain Ethereum để có thể giải quyết tranh chấp một cách chính xác và công bằng.

## 3.4 Giao diện

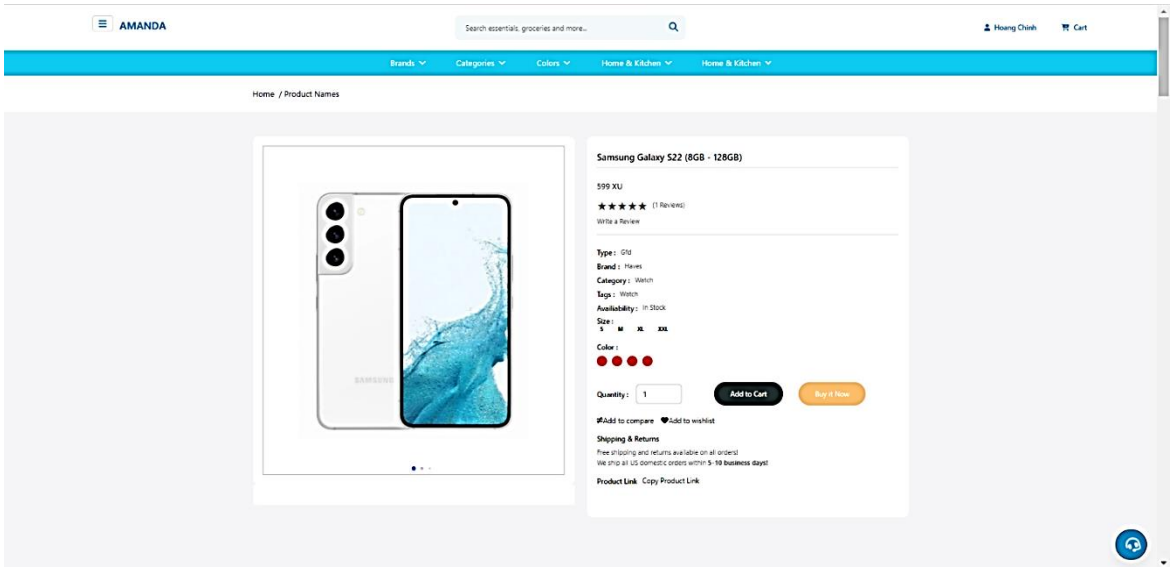
### 3.4.1 Hệ thống Website của Amanda

Để thuận tiện cho việc tìm kiếm sản phẩm, chúng tôi đã xây dựng lên giao diện rất đơn giản, dễ tiếp cận nhưng cũng có đầy đủ các chức năng cần thiết. Các sản phẩm được hiển thị ra một cách rõ ràng, hệ thống còn hỗ trợ thêm các bộ lọc giúp cho người dùng có thể lọc các sản phẩm cần thiết theo nhu cầu của mình. Đồng thời hệ thống cũng hỗ trợ người dùng phân trang trong trường hợp kết quả tìm kiếm ra quá nhiều sản phẩm (*Hình 3*).



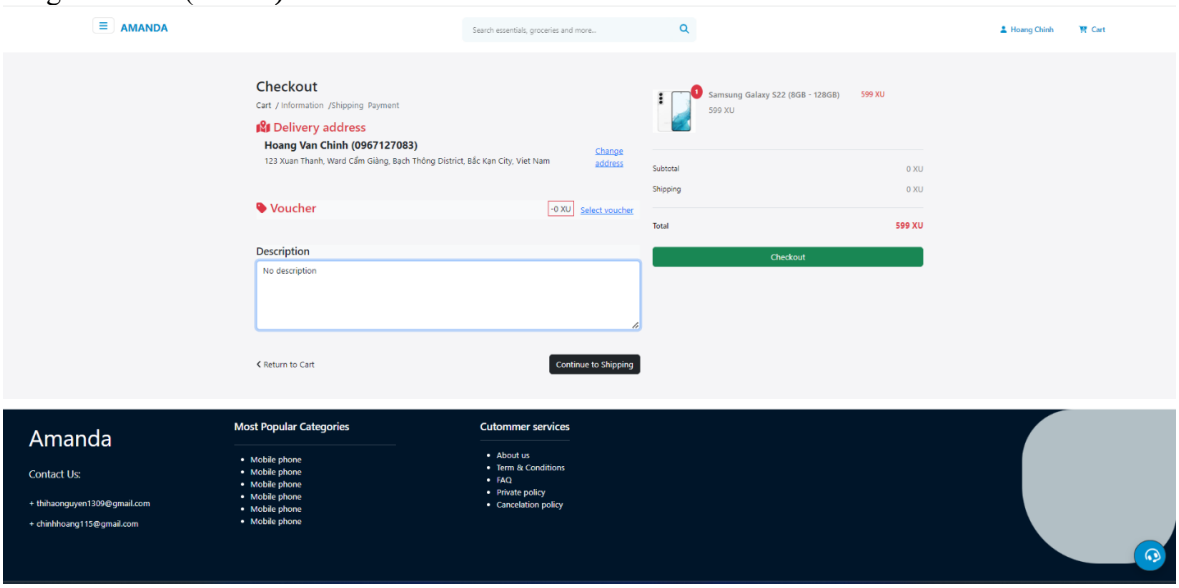

Hình 3: Giao diện trang tìm kiếm *Amanda*

Khi người dùng chọn sản phẩm để xem chi tiết về sản phẩm đó, hệ thống sẽ có nhiều sự lựa chọn cho khách hàng như số lượng, màu sắc, kích cỡ. Đồng thời sau khi xem thông tin sản phẩm nếu người dùng hài lòng có thể nhấn mua ngay hoặc thêm vào giỏ hàng (*Hình 3*).



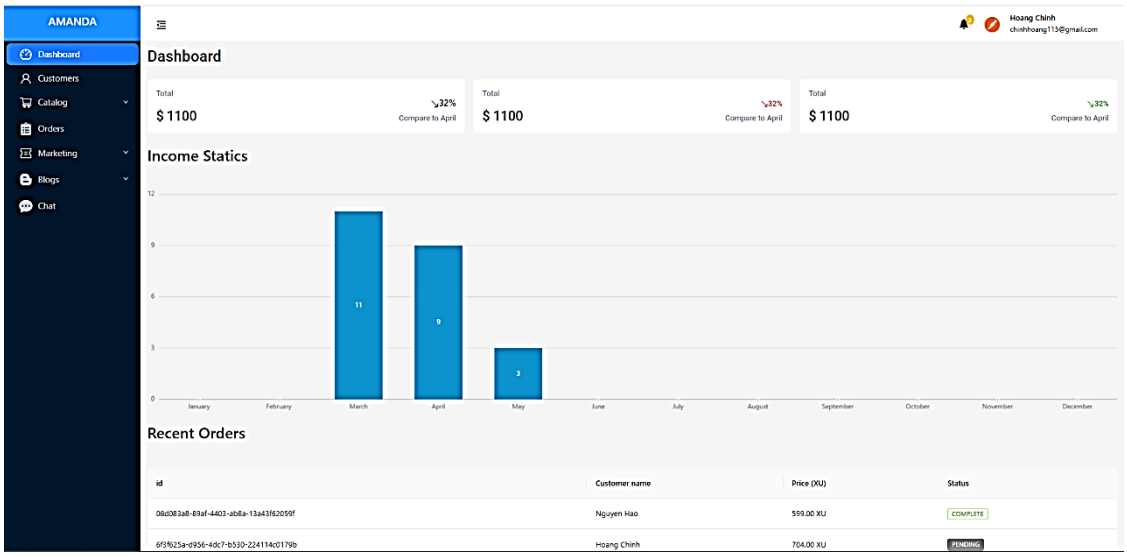

Hình 4: Giao diện thông tin sản phẩm được tìm thấy trên hệ thống *Amanda*

Khi người dùng chọn mua ngay thì hệ thống sẽ chuyển đến trang thanh toán, tại đây người dùng có thể có nhiều tùy chọn như thay đổi địa chỉ giao hàng, chọn mã giảm giá nếu có và có thể thêm ghi chú cho đơn hàng của mình (Hình 4).




Hình 5: Trang thông tin thanh toán và giao hàng trên hệ thống *Amanda*

Thống kê là một phần quan trọng giúp cho chủ Shop có thể đưa ra những chiến lược kinh doanh phù hợp với sự phát triển của hệ thống và sự biến đổi của thị trường (*Hình 5*, *Hình 6*).




Hình 6: Trang thống kê tổng quát trên hệ thống *Amanda*

## 4 KIỂM NGHIỆM THỰC TẾ

Chúng tôi đã tiến hành triển khai hệ thống Amanda lên máy ảo EC2 của AWS với cấu hình như sau: loại instance t2.micro, số CPU là 1, RAM 1GB, ROM 8GB, với hệ điều hành Linux. Trên máy ảo này của AWS chúng tôi cài đặt Docker phiên bản 20.10.21 để cài đặt các container của ứng dụng lên máy ảo EC2. Với loại tiền điện tử dùng riêng cho hệ thống Amanda chúng tôi đã triển khai và xác thực trên mạng testnet của BNB (Binance Smart Chain). Hệ thống hiện tại đang trong quá trình thử nghiệm nên máy chủ clould sẽ sử dụng một cấu hình không quá cao để thử nghiệm. Trong tương lai khi bước vào chạy ứng dụng trong môi trường thực tế, cấu hình sẽ được nâng cấp để có thể đáp ứng được nhu cầu của người dùng.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Bài báo đã giới thiệu công cụ Amanda nhằm giải quyết các vấn đề liên quan đến thanh toán trong mua sắm trực tuyến. Các phương thức thanh toán truyền thống thường gây bất tiện và lãng phí thời gian cho người dùng, đồng thời đối mặt với các vấn đề về an toàn và bảo mật. Amanda sử dụng công nghệ Blockchain để cung cấp một phương thức thanh toán tiện lợi, an toàn và minh bạch. Công cụ Amanda đã được triển khai trên nền tảng Internet và đã thành công trong việc thực hiện các giao dịch thử nghiệm sử dụng các đồng tiền số hiện có trên thị trường. Sự kết hợp giữa các công nghệ cốt lõi như NodeJS, Hardhat, ReactNative và ReactJS cùng với việc sử dụng các dịch vụ tiền số để quản lý ví điện tử đã tạo nên một công cụ mạnh mẽ để đáp ứng nhu cầu mua sắm trực tuyến hiện đại. Amanda là một bước tiến quan trọng trong việc cải thiện quá trình thanh toán trong mua sắm trực tuyến cả trên nền tảng Web và App cho di động. Với khả năng sử dụng Blockchain và các công nghệ tiên tiến, Amanda đáp ứng nhu cầu của người dùng trong việc thực hiện các giao dịch mua sắm một cách tiện lợi, an toàn và minh bạch. Hiện công cụ đã triển khai trong môi trường thật với một số giao dịch được triển khai thanh toán bằng các đồng tiền số thông dụng (stablecoin). Trong tương lai, Amanda sẽ tiếp tục phát triển và mở rộng các chức năng để cung cấp một trải nghiệm mua sắm trực tuyến tốt hơn cho người dùng. Đồng thời, công cụ này sẽ được quảng bá để thu hút người dùng và tạo ra sự tin tưởng trong việc sử dụng các giao dịch thanh toán tiền số.

## TÀI LIỆU THAM KHẢO

[1] NestJS (2023), [online], https://docs.nestjs.com, truy cập ngày 19/01/2023

[2] ReactJS (2023), [online], https://legacy.reactjs.org, truy cập ngày 19/01/2023

[3] React-native (2023), [online], https://reactnative.dev, truy cập ngày 19/01/2023

[4] Amazon web service (2023), [online], https://docs.aws.amazon.com , truy cập ngày 19/01/2023

[5] Cloudinary SDK for NodeJS (2023), [online], https://cloudinary.com/documentation/node_integration , truy cập ngày 19/01/2023

[6] Redis (2023), [online], https://redis.io/docs , truy cập ngày 19/01/2023

[7] Treiblmaier, Horst, and Christian Sillaber. *The impact of blockchain on e-commerce: a framework for salient research topics*. Electronic Commerce Research and Applications 48 (2021): 101054.

[8] Zhu, Xingxiong, and Dong Wang. *Research on blockchain application for E-commerce, finance and energy*. IOP Conference Series: Earth and Environmental Science. Vol. 252. No. 4. IOP Publishing, 2019.

[9] Lahkani, Marjaneh Jahangiri, et al. *Sustainable B2B E-commerce and blockchain-based supply chain finance*. Sustainability 12.10 (2020): 3968.

[10] Jain, Shashank Mohan. *Hardhat: A Brief Introduction to Web3*, Decentralized Web Fundamentals for App Development. Berkeley, CA: Apress, 2022. 167-179.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F017**

# MEDIHEARTH: GIẢI PHÁP CHĂM SÓC SỨC KHỎE THÔNG MINH CHO BỆNH NHÂN TIM MẠCH DỰA TRÊN CÔNG NGHỆ INTERNET VẠN VẬT

TÔN LONG PHƯỚC[1*], ĐINH QUANG HUY[1], NGUYỄN THÀNH LUÂN[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*[*]tonlongphuoc@iuh.edu.vn, huydinhse@gmail.com, nguyenthanhluantqd@gmail.com*

**Tóm tắt.** Hiện nay, việc chăm sóc sức khỏe bệnh nhân tim mạch được triển khai trên nền tảng thông minh đang trở thành một xu hướng. Tuy nhiên, việc ứng dụng công nghệ IoT trong lĩnh vực này vẫn còn nhiều hạn chế nguyên nhân chính là do việc triển khai và sử dụng các ứng dụng thông minh trong y tế vẫn chưa được quan tâm đúng mức. Đồng thời, việc kết hợp công nghệ IoT trong lĩnh vực chăm sóc sức khỏe vẫn còn nhiều thử thách. Một trong những hạn chế của việc áp dụng IoT trong chăm sóc sức khỏe là độ tin cậy của dữ liệu được thu thập từ các thiết bị. Việc xử lý và phân tích dữ liệu cũng đòi hỏi trình độ chuyên môn cao. Ngoài ra, việc đảm bảo bí mật thông tin bệnh nhân cũng là một trong những thách thức cần được quan tâm. Để giải quyết vấn đề này, chúng tôi đề xuất một công cụ có tên là MediHealth. MediHealth là một hệ thống theo dõi nhịp tim thông minh với thiết bị đeo tay và đưa ra các cảnh báo trên web và ứng dụng di động khi nhịp tim bất thường. Đồng thời, công cụ sử dụng thuật toán trí tuệ nhân tạo để đưa ra các cảnh báo, cùng các thông số khác do bác sĩ cung cấp, giúp phát hiện thêm các bệnh lý tim mạch tiềm ẩn. Hệ thống được xây dựng bằng ReactJS cho web, Flutter cho mobile, Nodejs cho phần web server và công nghệ web-socket để đưa ra các thông báo theo thời gian thực cho các bác sỹ và bệnh nhân. Hiện công cụ đã được thử nghiệm cho một số bệnh nhân tại TP.HCM. Trong tương lai, chúng tôi sẽ mở rộng phạm vi bệnh mà hệ thống có thể hỗ trợ, đồng thời, chúng tôi sẽ thiết kế lại thiết bị để thu thập nhiều dữ liệu chính xác.

**Từ khóa.** Chăm sóc sức khỏe, IoT, Trí tuệ nhân tạo, ReactJS, Flutter, Nodejs, Web-socket, AWS.

## MEDIHEALTH: SMART CARDIOVASCULAR CARE SOLUTION BASED on IoT TECHNOLOGY

**Abstract.** Currently, the health care of cardiovascular patients deployed on smart platforms is a trend. However, the application of IoT technology in this field still has many limitations as the deployment and use of smart applications in healthcare still face many challenges. One of the limitations of IoT adoption in healthcare is the reliability of data collected from devices. The processing and analysis of data also requires a high level of expertise. In addition, ensuring the confidentiality of patient information is also one of the challenges that need to be considered. To solve this problem, we recommend a tool called MediHealth. MediHealth is a smart heart rate monitoring system and gives web and mobile alerts when the heart rate is abnormal. At the same time, the tool uses an artificial intelligence algorithm to give warnings, and with other parameters provided by the doctor, additionally detects potential cardiovascular diseases. The system is built using ReactJS for the web, Flutter for mobile, Nodejs for the web server part and web-socket technology to provide real-time notifications to the doctors. Currently, the tool has been tested for some patients in Ho Chi Minh City. In the future, we will expand the range of diseases the system can support, and at the same time, we will redesign the device to be more compact and convenient to receive more accurate data.

**Keywords.** Healthcare, IoT, Artificial Intelligence, ReactJS, Flutter, Nodejs, Web-socket, AWS.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

# 1 GIỚI THIỆU TỔNG QUAN

Với sự phát triển chóng mặt của công nghệ số hiện nay, Internet of Things (IoT) đang trở thành một xu hướng đáng chú ý và nhận được sự quan tâm và đầu tư đáng kể, đặc biệt trong lĩnh vực chăm sóc sức khỏe. IoT là một hệ thống được thiết kế để kết nối và quản lý các thiết bị thông minh, cho phép chúng ta điều khiển và giám sát từ xa thông qua internet. Trong lĩnh vực chăm sóc các bệnh lý về tim mạch, tiến bộ của IoT đã mở ra nhiều tiềm năng ứng dụng đáng chú ý. Theo số liệu của Tổ chức Y tế Thế giới năm 2019, tử vong do bệnh tim mạch chiếm tới 39,5%, trong đó: bệnh mạch máu não (55,4%), bệnh tim thiếu máu cục bộ (32%), bệnh tim do tăng huyết áp (6,9%) và bệnh tim mạch khác (5,7%) [1]. Đó cũng là vấn đề nan giải của các tổ chức y tế về các tình trạng bệnh lý liên quan đến tim mạch diễn ra hiện nay.

Hiện nay, gánh nặng bệnh tật vẫn tiếp tục gia tăng, đặc biệt là gánh nặng về bệnh lý tim mạch và xu hướng trẻ hóa ở những người trong độ tuổi lao động. Do đó, chúng ta cần theo dõi, kiểm tra và phát hiện các triệu chứng, bệnh lý sớm để có thể điều trị kịp thời và tránh các biến chứng khác. Để giảm thiểu tác động của bệnh này, chúng tôi đã sử dụng công nghệ IoT trong việc chăm sóc sức khỏe của bệnh nhân được tốt hơn. MediHealth là hệ thống được phát triển với mục tiêu giảm bớt áp lực cho nhân viên y tế và tự động hóa quy trình theo dõi sức khỏe của bệnh nhân. Được trang bị các thiết bị đeo y tế, hệ thống này có khả năng thu thập các chỉ số sức khỏe của bệnh nhân và truyền dữ liệu đến bác sĩ theo dõi ngay trong thời gian thực. Đặc biệt, hệ thống sử dụng dữ liệu và tập luật y tế kết hợp với trí tuệ nhân tạo (AI) [2] để theo dõi và phân tích tự động. Điều này giúp bác sĩ có thể từ xa theo dõi tình trạng sức khỏe của bệnh nhân và đưa ra quyết định cần thiết một cách kịp thời. Ngoài ra, hệ thống cũng cung cấp thông báo để xử lý các tình huống y tế khẩn cấp.

Hệ thống MediHealth mang lại nhiều lợi ích đáng kể cho bệnh nhân và chuyên gia y tế. Dữ liệu sức khỏe được thu thập có thể được chia sẻ và phân tích bởi các chuyên gia y tế, từ đó đưa ra chẩn đoán chính xác và phương pháp điều trị hiệu quả. Bác sĩ và các chuyên gia y tế có khả năng theo dõi tình trạng sức khỏe tim mạch của bệnh nhân từ xa thông qua các ứng dụng và giao diện trực tuyến. Họ nhận được cảnh báo ngay khi có sự thay đổi bất thường trong dữ liệu sức khỏe, giúp họ can thiệp kịp thời và đưa ra quyết định chăm sóc phù hợp.

Kết hợp IoT và chăm sóc sức khỏe tim mạch mang lại nhiều lợi ích đáng kể, bao gồm cải thiện chất lượng chăm sóc sức khỏe, giảm thời gian và công sức của bác sĩ và nhân viên y tế trong việc giám sát bệnh nhân, cũng như tăng khả năng tự quản lý sức khỏe của bệnh nhân [3]. Điều này đóng góp vào việc cải thiện chất lượng cuộc sống của những người mắc bệnh tim mạch và giảm tỷ lệ tử vong toàn cầu do bệnh này.

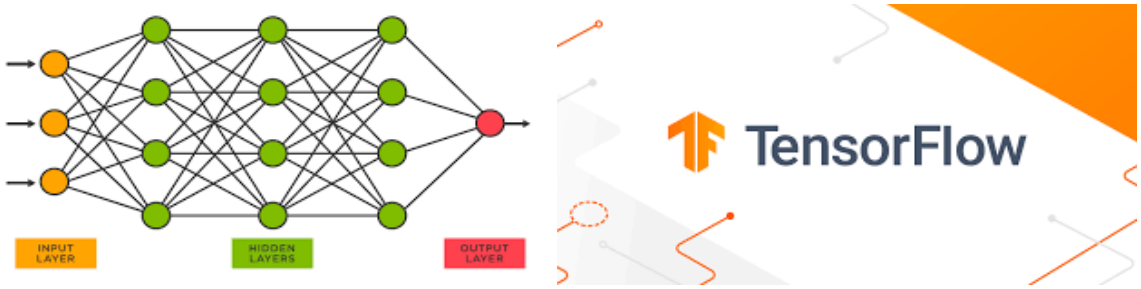
Bài báo của chúng tôi chia làm 5 phần: Phần 1 Giới thiệu tổng quan về hệ thống; Phần 2 Các công nghệ liên quan; Phần 3 Công cụ MEDIHEALTH; Phần 4 Kiểm nghiệm thực tế và Phần 5 Kết luận và hướng phát triển. Trong đó Phần 3 là phần đóng góp chính của bài báo.

# 2 CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 TensorFlow – Neural Network

TensorFlow là một thư viện mã nguồn mở được phát triển bởi Google, cung cấp các công cụ để xây dựng và huấn luyện các mô hình Machine Learning [4]. Neural Network là một mô hình Machine Learning được lấy cảm hứng từ cấu trúc của hệ thống thần kinh sinh học, với khả năng học hỏi và tự điều chỉnh dựa trên dữ liệu huấn luyện. TensorFlow có một số ưu điểm: TensorFlow cung cấp nhiều công cụ và thư viện hỗ trợ cho việc xây dựng và huấn luyện mô hình Neural Network, giúp cho quá trình này trở nên đơn giản hơn; Neural Network có khả năng học hỏi và tự điều chỉnh dựa trên dữ liệu huấn luyện, cho phép mô hình tự động tìm ra các đặc trưng quan trọng và cải thiện độ chính xác dự đoán (Hình 2.1); Kết quả dự đoán của mô hình Neural Network có thể đạt được độ chính xác cao, đặc biệt là trong các bài toán phức tạp như dự đoán bệnh tim mạch. Ngoài ra, cũng có một số nhược điểm: Việc xây dựng và huấn luyện mô hình Neural Network yêu cầu nhiều kinh nghiệm và kiến thức chuyên sâu về Machine Learning, đặc biệt là trong việc lựa chọn kiến trúc mô hình và thiết lập các tham số; Việc huấn luyện mô hình Neural Network yêu cầu một lượng lớn dữ liệu huấn luyện, đồng thời cần phải đảm bảo tính đại diện của dữ liệu để tránh hiện tượng overfitting [5].

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 1: TensoFlow – Mô hình Neural Network [6]

## 2.2 ReactJS – Flutter

ReactJS và Flutter đều là công nghệ phát triển ứng dụng, nhưng có ngôn ngữ lập trình và mục tiêu sử dụng khác nhau. ReactJS là một thư viện JavaScript được sử dụng để xây dựng các ứng dụng web động. Nó tập trung vào việc tạo ra giao diện người dùng (UI) đáp ứng và tương tác mượt mà với các thành phần khác nhau của ứng dụng web. Còn Flutter, trên một khía cạnh khác, là một framework phát triển ứng dụng di động được phát triển bởi Google. Flutter cho phép xây dựng các ứng dụng di động đa nền tảng (cross-platform) bằng việc sử dụng một ngôn ngữ lập trình duy nhất, là Dart. Với Flutter, ta có thể tạo ra giao diện người dùng đẹp mắt và đáp ứng trên nhiều nền tảng như Android và iOS [7, 8].

## 2.3 Firebase

Firebase là một nền tảng dịch vụ đám mây được cung cấp bởi Google, giúp phát triển ứng dụng web và di động dễ dàng hơn. Với Firebase, nhà phát triển có thể sử dụng các dịch vụ như xác thực người dùng, lưu trữ dữ liệu, thông báo đẩy, và nhiều hơn nữa để xây dựng ứng dụng chất lượng cao. Trong Firebase, một dịch vụ có tên là FCM (Firebase Cloud Messaging), là một dịch vụ thông báo đẩy. Nó cho phép nhà phát triển gửi thông báo đẩy từ máy chủ đến các thiết bị di động như điện thoại thông minh. Với việc tích hợp FCM vào ứng dụng, nhà phát triển có thể tận dụng sự linh hoạt và khả năng tùy chỉnh để gửi thông báo đẩy đến người dùng, giữ liên lạc và tăng cường tương tác với ứng dụng của mình trên các thiết bị di động [9].

## 2.4 Thiết bị IoT

Như Hình 2, chúng tôi đã thiết kế và lắp đặt để có thể tạo nên một thiết bị đang hoạt động. Chúng tôi đã tham khảo và dựa trên các thông số cũng như chức năng của các thiết bị IoT để có thể đưa ra quyết định sử dụng nó trong dự án này. Cụ thể: ESP8266 NodeMCU [10] là một vi điều khiển dựa trên module wifi ESP8266. Nó là một phiên bản được phát triển dựa trên Arduino và có thể lập trình và quản lý thông qua một cổng USB. ESP8266 NodeMCU hỗ trợ nhiều chức năng và tính năng bao gồm cả giao tiếp WiFi, GPIO, PWM, I2C và UART, cho ESP8266 NodeMCU có thể sử dụng để phát triển các ứng dụng IoT, như điều khiển các thiết bị thông qua mạng WiFi, thu thập dữ liệu từ các cảm biến và truyền dữ liệu đến các máy chủ hoặc ứng dụng điện toán đám mây khác.

Cảm biến nhịp tim và oxy trong máu MAX30102 [11] là một mô-đun đo nhịp tim và oxy trong máy tích hợp. Nó bao gồm đèn LED bên trong, bộ tách sóng quang, các bộ phận quang học và các thiết bị điện tử có tiếng ồn thấp với khả năng loại bỏ ánh sáng xung quanh.

Cảm biến nhịp tim và oxy trong máu Max30102 hoạt động trên một nguồn cung cấp điện 1.8V và một nguồn cấp điện 5.0V riêng biệt cho các đèn LED bên trong.

Giao tiếp thông qua giao diện tương thích I2C tiêu chuẩn và có thể được tắt thông qua phần mềm với chế độ chờ bằng không. Nó cho phép các thanh ray nguồn vẫn được cấp nguồn ở mọi thời điểm.

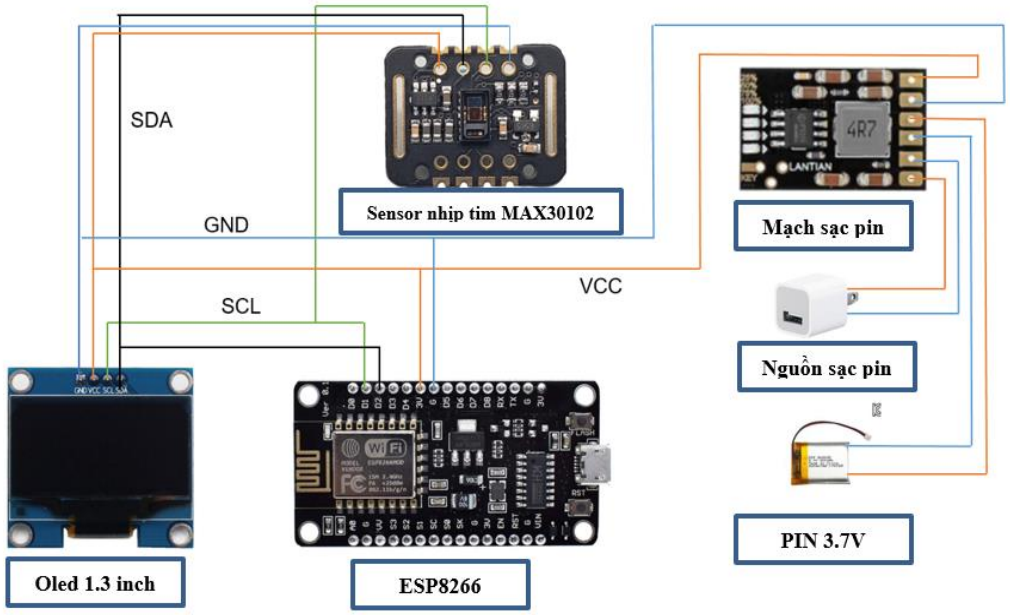
Màn hình Oled 1.3 inch 128×64 I2C [12] cho khả năng hiển thị đẹp, sang trọng, rõ nét vào ban ngày và khả năng tiết kiệm năng lượng tối đa với mức chi phí phù hợp, màn hình sử dụng giao tiếp I2C cho chất lượng đường truyền ổn định và rất dễ giao tiếp chỉ với 2 chân GPIO.

Mạch sạc pin lithim 3.7V [13, 14] có ngõ ra 5V 2A. Có chức năng sạc cho pin lithium và làm bộ cấp nguồn 5V dòng lên tới 2A. Thích hợp cho các hệ thống cần cấp nguồn liên tục, ngay cả khi mất điện sẽ có nguồn nuôi được tăng áp từ pin lithium.

Trên module có tích hợp LEDs báo dung lượng pin và báo sạc. Nút nhấn kích hoạt module hoạt động hoặc tắt. Cho phép vừa sạc vừa cung cấp điện ở ngõ ra. Nhấn nút một lần để kích hoạt module. Nhấn và giữ để

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

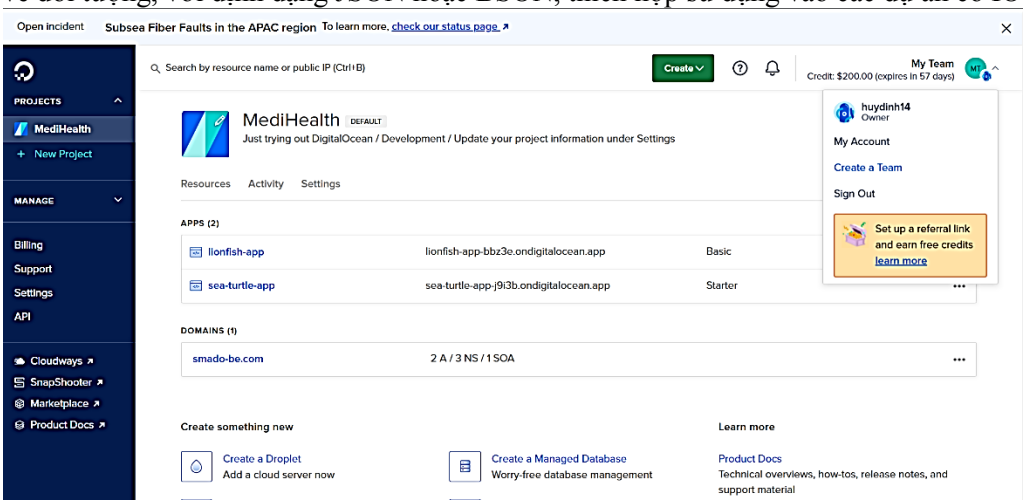
tắt module. Module sẽ tự động kích hoạt khi có thiết bị kết nối. Nếu dòng tiêu thụ nhỏ hơn 50mA, sau 30s module sẽ tự tắt.




Hình 2: Sơ đồ lắp ráp linh kiện các thiết bị IoT của thiết bị đeo tay trong Medihealth

## 2.5 NoSQL - MongoDB

MongoDB là hệ quản trị cơ sở dữ liệu NoSQL (non-relational), được phát triển bởi công ty MongoDB Inc. Các dữ liệu trong MongoDB được lưu trữ dưới dạng tài liệu (document), đó là một bản ghi độc lập chứa thông tin về đối tượng, với định dạng JSON hoặc BSON, thích hợp sử dụng vào các dự án có IOT [15].




Hình 3: Công cụ MediHealth đưa lên Cloud của Digital Ocean

## 2.6 Digital Ocean

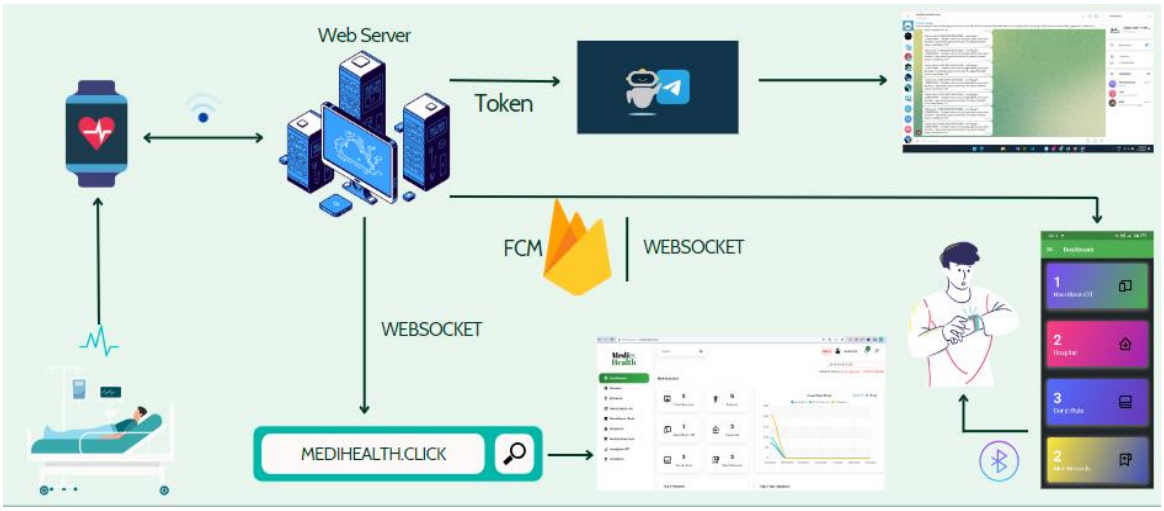
DigitalOcean là một nhà cung cấp dịch vụ đám mây với các giải pháp cho máy chủ ảo, máy chủ riêng và lưu trữ đám mây (*Hình 3*). Nó cung cấp một giao diện đơn giản và dễ sử dụng cho phép người dùng tạo và quản lý các máy chủ ảo, lưu trữ và cơ sở dữ liệu [16].

# 3 CÔNG CỤ MEDIHEALTH

## 3.1 Giới thiệu

Chúng tôi đã phát triển công cụ Medihealth với sự kết hợp của nhiều công nghệ tiên tiến, tuân thủ mục tiêu đã đề ra. Phần Backend được xây dựng trên nền tảng NodeJS [17,18], đảm nhận nhiệm vụ tạo Restful-API và xử lý yêu cầu từ phía clients, đồng thời trả về phản hồi phục vụ cho clients. Dữ liệu được lưu trữ trong MongoDB để tăng tốc độ truy xuất dữ liệu, phù hợp cho các dự án thời gian thực. Phần Frontend được xây dựng trên nền tảng ReactJS, đảm nhận việc xây dựng giao diện Website và gửi yêu cầu xuống Server, hiển thị biểu đồ các chỉ số sức khỏe của bệnh nhân cho bác sĩ hoặc chuyên gia theo dõi thời gian thực. ReactJS cho phép thực hiện các chức năng quản lý hệ thống như quản lý bệnh nhân, bác sĩ, tập luật và nhiều chức năng khác. Phần ứng dụng di động cho Android/IOS được xây dựng trên nền tảng Flutter, phục vụ chức năng thông báo tình trạng bất thường *(Hình 3.1*) của bệnh nhân tới bác sĩ chịu trách nhiệm và các chức năng liên quan như quản lý hồ sơ bệnh án, xem bệnh nhân đang được điều trị, ... Để đảm bảo áp dụng ứng dụng vào thực tế, chúng tôi đã sử dụng dịch vụ của Digital Ocean để tạo máy ảo và cài đặt Server. Để đảm bảo bảo mật và tin cậy cho dữ liệu trong hệ thống, chúng tôi đã sử dụng các giải pháp mã hóa dữ liệu trong quá trình truyền thông và lưu trữ.

Việc chúng tôi sử dụng công nghệ như WebSocket sẽ giúp cho các thông báo sẽ được gửi đi trong thời gian thực một cách nhanh chóng và công nghệ FCM từ Firebase sẽ gửi các thông báo đến thiết bị di động, cho dù thiết bị đó đang không sử dụng ứng dụng MediHealth. Cùng với đó, WebServer là trung tâm xử lý các yêu cầu, cũng như nhận và gửi các thông báo từ các thiết bị IOT, xử lý dữ liệu và đồng bộ dữ liệu giữa thiết bị và người dùng một cách ổn định nhất. Ngoài ra, Bot Telegram cũng sẽ hỗ trợ trong việc thông báo và lưu lại các thông số cảnh báo cần thiết trong quá trình sử dụng.

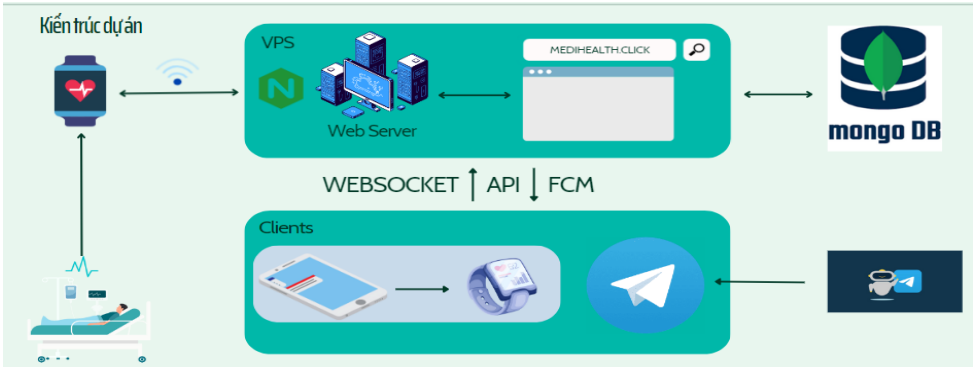



Hình 4: Sơ đồ hoạt động cảnh báo của hệ thống Medihealth

## 3.2 Kiến trúc của hệ thống

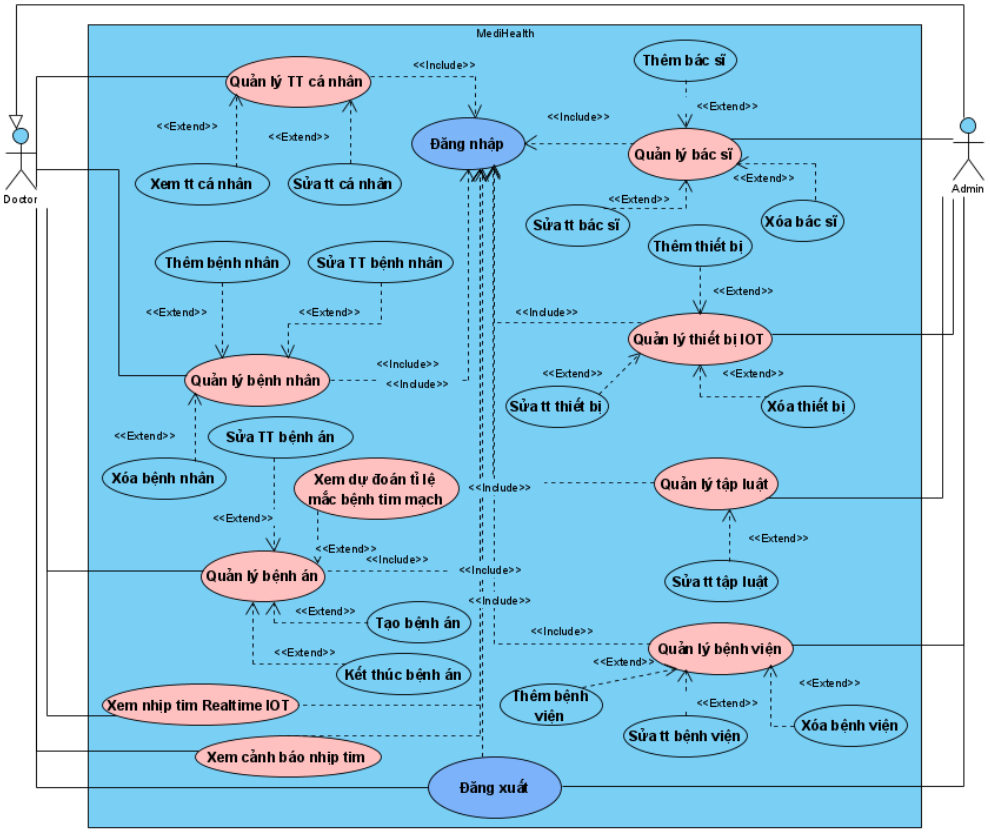
Hệ thống Medihealth áp dụng kiến trúc MVC (Model-View-Controller). MVC là một mẫu kiến trúc phần mềm để tạo lập giao diện người dùng trên máy tính và được áp dụng rộng rãi trên các kiến trúc của một hệ thống Website [19, 20]. Kiến trúc MVC sẽ được chia làm 3 phần đúng theo tên viết tắt của nó là: View – Model – Controller. View: sử dụng công nghệ ReactJS cho phía web, Flutter cho phía app để thực thi giao diện người dùng. Sử dụng các phương thức HTTP/HTTPS để kết nối với Controller của server. Controller: Lấy dữ liệu từ Model để xây dựng các RESTful API để View có thể truy cập được và lấy dữ liệu hiển thị ra giao diện. Model: Chịu trách nhiệm lưu trữ các dữ liệu từ phía View vào cơ sở dữ liệu. Hơn nữa, Model tạo các entity để tham chiếu tới các đối trượng trong cơ sở dữ liệu. Chúng tôi đã sử dụng VPS (Virtual Private Server) trên Digital Ocean để có thể đưa dự án lên và liên kết làm việc với các clients khác (*Hình 5*).


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 5: Kiến trúc trong hệ thống Medihealth

### 3.3 Phân tích hệ thống Medihealth




Hình 6: Sơ đồ Use Case của hệ thống MediHealth

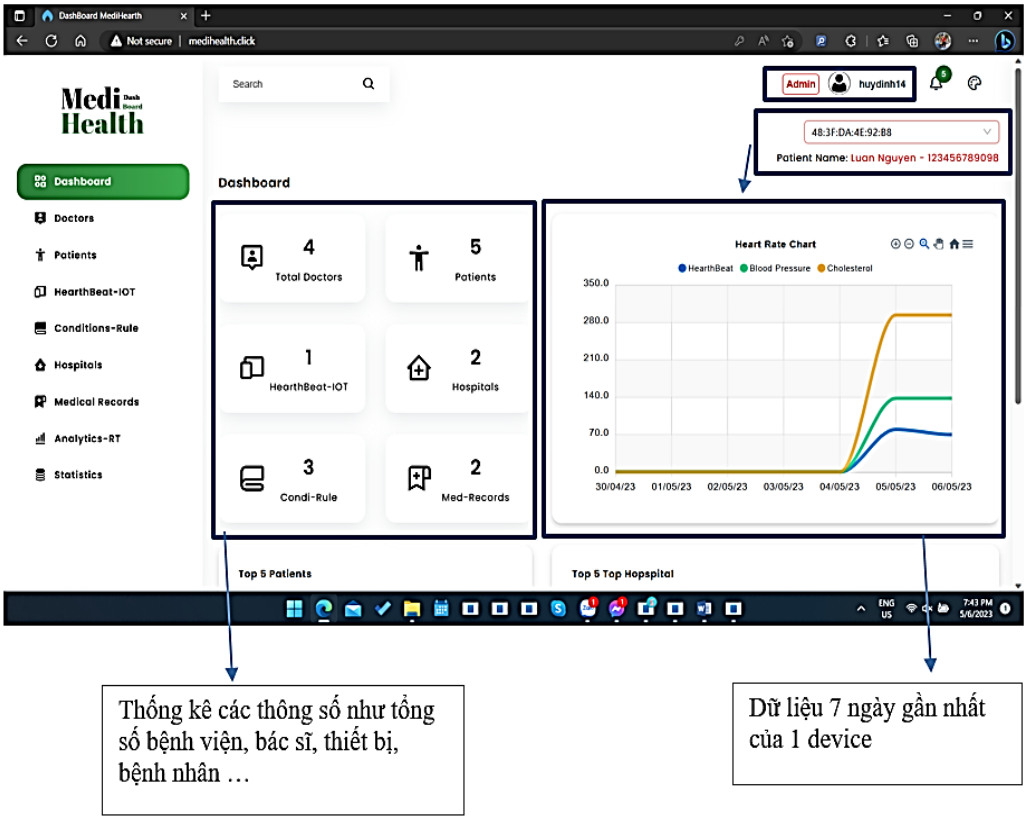
Đối tượng người dùng cho hệ thống Medihealth gồm ba tác nhân chính: *Chuyên gia*, *Bác sĩ* và *Thiết bị IoT*. Trong đó, chúng tôi tập trung vào chức năng chính của hệ thống là phát hiện bất thường, xây dựng tập luật hỗ trợ việc cảnh báo, theo dõi tình trạng bệnh nhân cũng như phát hiện bất thường của bệnh nhân trong thời gian thực. Ngoài ra, những chức năng phụ cũng được đưa vào hệ thống để hỗ trợ việc quản lí hệ thống và làm tăng trải nghiệm của người dùng. Từ những chức năng đó, chúng tôi tập trung vào các chức năng chính như: Xem tình trạng bệnh nhân, gửi dữ liệu bệnh nhân lên hệ thống, đồng bộ trạng thái thiết bị, gửi thông báo đến bác sĩ. Sau đó chúng tôi mở rộng các chức năng tương ứng.
Trong mô hình *use-case* chúng tôi trình bày, gồm có hai thành phần *actor* chính mô tả cho hai đối tượng người dùng mà chúng tôi hướng đến (*Admin, Doctor*). Ngoài ra, chúng tôi cũng mở rộng các *use-case* cho các chức năng của hệ thống nhằm tạo điều kiện thuận lợi cho người dùng là chuyên gia và bác sĩ.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Từ các usecase đã phân tích ở trên thì chúng tôi đã phác thảo ra được lược đồ quan hệ thực thể của Medihearth. Trong đó, chúng tôi chú trọng đến bảng *Medical Record* và bảng *Notification,* hai bảng này giúp khá nhiều dữ liệu cho chức năng chính là xem tình trạng bệnh nhân và gửi thông báo cho bác sĩ khi gặp bất thường. Ngoài ra thì còn các bảng dữ liệu khác bổ trợ cho chức năng chính đồng thời làm rõ các mối quan hệ trong lúc xử lý dữ liệu (*Hình 6*).

### 3.4 Giao diện đồ họa

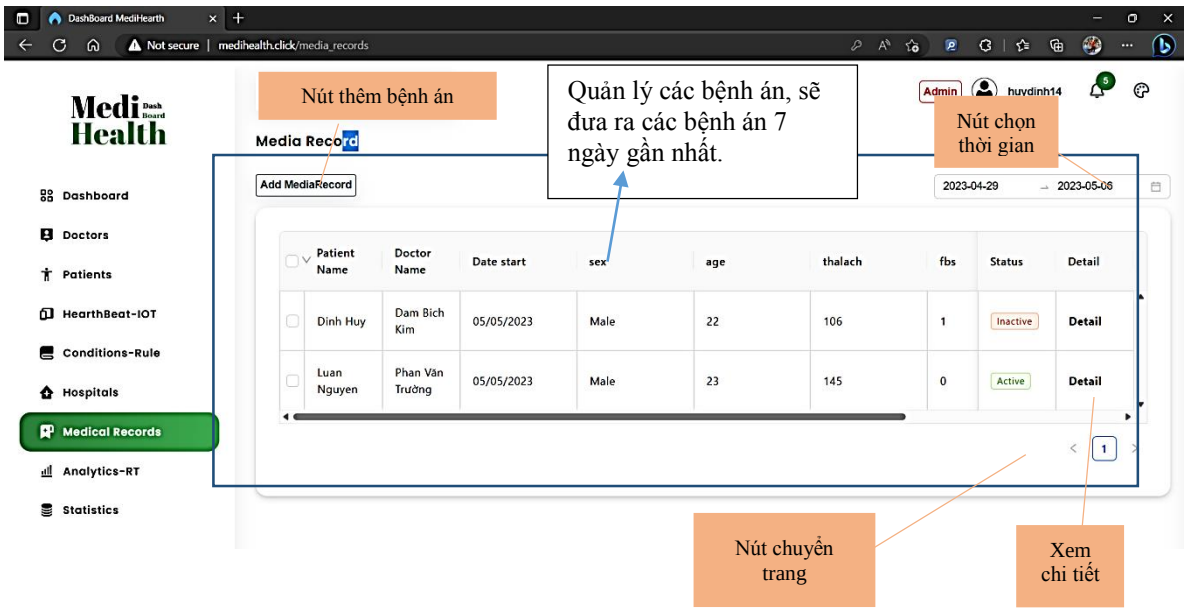
Hệ thống Medihealth triển khai nền tảng Web app và Mobile app. Web app được triển khai bằng ReactJS, Mobile app được xây dựng bằng Flutter để chạy trên cả hai nền tảng Android/iOS. Công cụ này chúng tôi quan tâm đến trải nghiệm của người dùng, giao diện thân thiện và đảm bảo tính ổn định cho cả Website và app. Sau đây chúng tôi xin giới thiệu một vài giao diện và chức năng của Medihealth trên Web và App. Sau khi chuyên gia hoặc bác sĩ đăng nhập sẽ hiện ra giao diện Dashboard hiển thị biểu đồ đường của một bệnh nhân. Biểu đồ này thống kê bảy ngày gần nhất tính đến ngày hiện tại, giúp bác sĩ theo dõi tình trạng của bệnh nhân dễ dàng hơn (*Hình 7*).




Hình 7: Giao diện dashboard hệ thống Medihealth trên nền Web

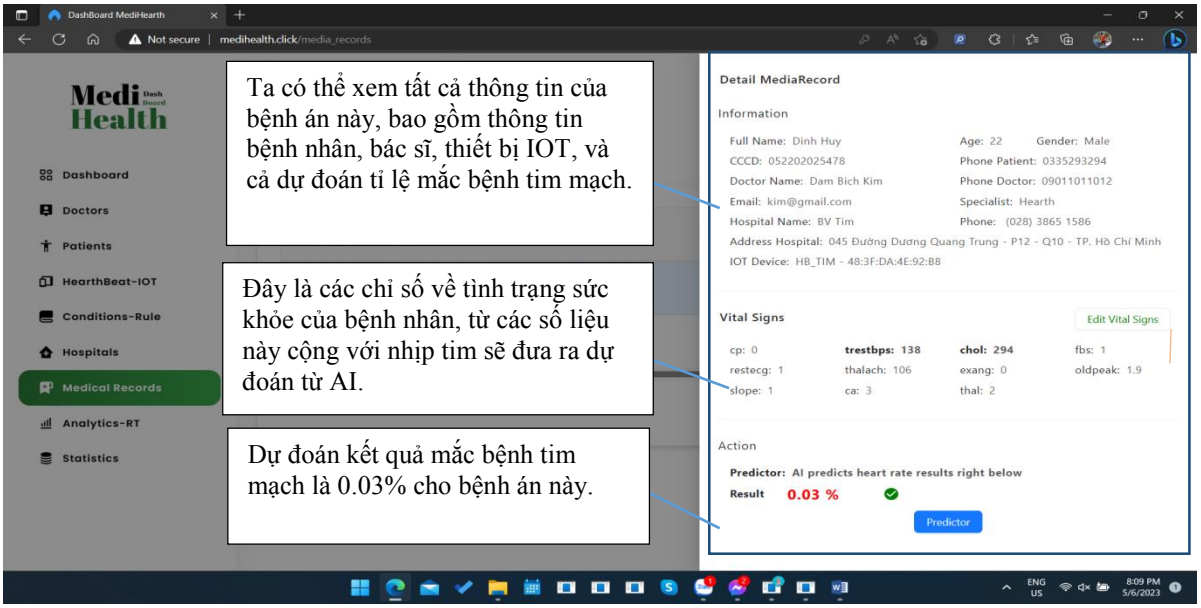
*Dashboard là giao diện trang chủ của Website. Theo dõi bệnh nhân tùy theo lựa chọn của bác sĩ hoặc chuyên gia. Hiển thị biểu đồ đường với các chỉ số trung bình như nhịp tim, huyết áp, chỉ số Cholesterol trong bảy ngày gần nhất. Người dùng chọn bệnh nhân để xem tình trạng của bệnh nhân vừa chọn. Biểu đồ đường màu xanh lam thể hiện nhịp tim của bệnh nhân, biểu đồ đường màu xanh lá thể hiện huyết áp và đường màu cam thể hiện chỉ số Cholesterol trong máu của bệnh nhân (Hình 8).*


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 8: Giao diện quản lý bệnh án của hệ thống MediHealth

Quản lý bệnh án: giao diện hỗ trợ xem danh sách thiết bị của hệ thống, cũng như tạo và xem thông tin bệnh án. Nút "*Add MediaRecord*" dùng để thêm thông tin một bệnh án mới. Nút "*Detail*" dùng để xem thông tin một bệnh án *(Hình 9)*.

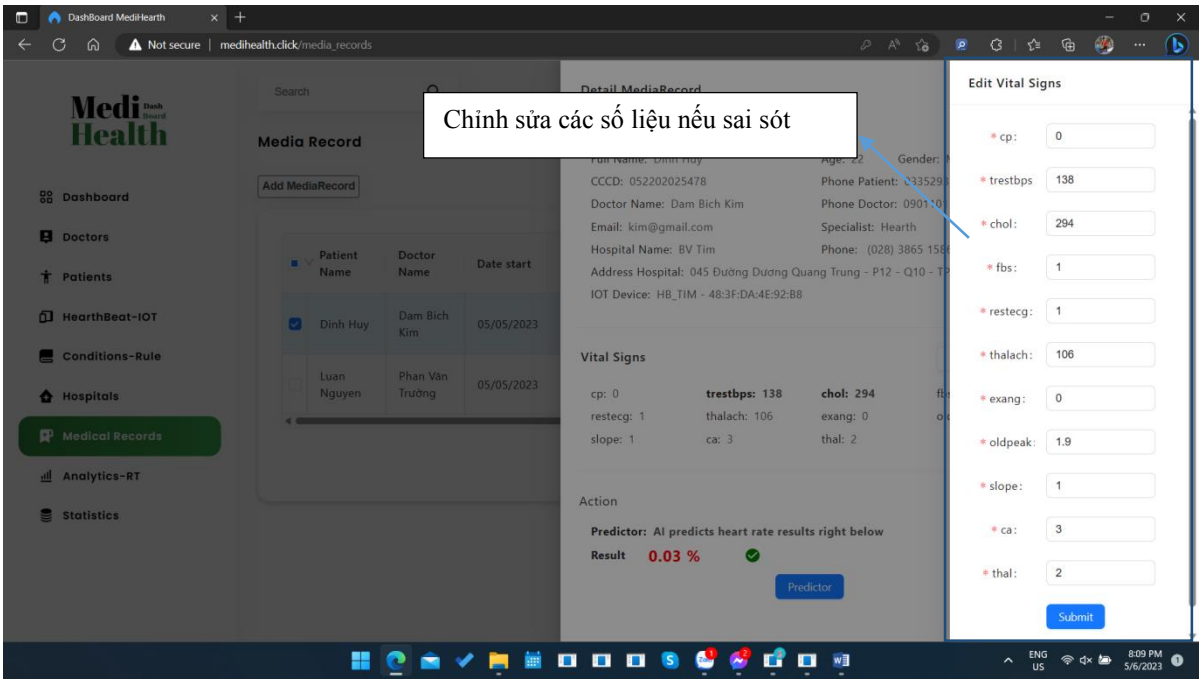



Hình 9: Giao diện chi tiết bệnh án

Ở giao diện này, nút "*Predictor"*: Nút thực hiện đánh giá tình trạng bệnh từ dữ liệu sức khoẻ, được đánh giá bởi Trí tuệ nhân tạo. Kết quả sẽ dự đoán bệnh án này sẽ có tỉ lệ mắc bệnh tim mạch là bao nhiêu phần trăm. Từ đó bác sĩ có thể dựa vào và đưa ra chẩn đoán khách quan hơn (*Hình 9*).
Khi ta muốn chỉnh sửa tập các chỉ số về tình trạng sức khoẻ, nút "*Edit Vital Sign"* sẽ giúp ta sửa các chỉ số về tình trạng sức khoẻ của bệnh án này (*Hình 10*).
Ở đây ta có thể thấy các các hiệu về bệnh tim mạch, chúng tôi sẽ giải thích cụ thể các ký hiệu này như sau: age: Tuổi của bệnh nhân; sex: Giới tính của bệnh nhân (1 = Nam, 0 = Nữ); cp: Loại đau ngực mà bệnh nhân trải qua. Có 4 loại khác nhau, từ 0 đến 3; trestbps: Huyết áp tâm thu của bệnh nhân trong tình trạng nghỉ

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

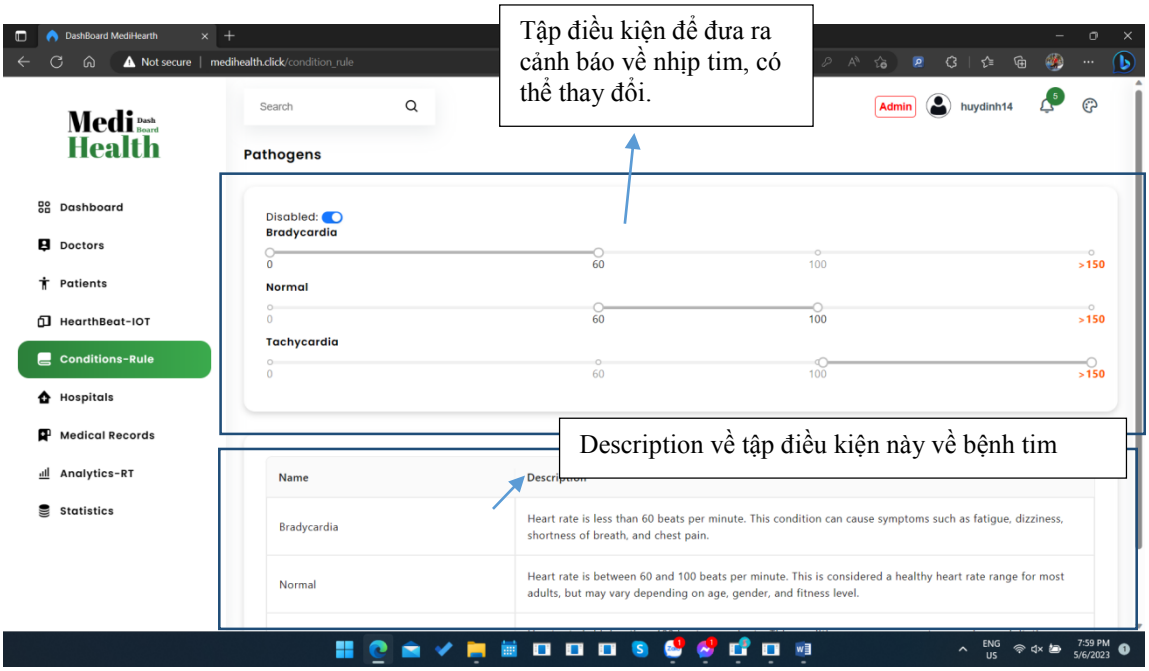
ngơi; chol: Mức độ Cholesterol trong máu của bệnh nhân; fbs: Mức độ đường huyết nhiễm trùng (0 = không, 1 = có); restecg: Kết quả điện tâm đồ (0, 1 hoặc 2); thalach: Nhịp tim tối đa đạt được của bệnh nhân; exang: Đau ngực do tập thể dục; oldpeak: Giảm đáp lực ST được ghi nhận sau thử nghiệm tập thể dục so với trước khi thử nghiệm; slope: Góc nghiêng của đoạn giảm đáp lực ST (0, 1 hoặc 2); ca: Số lượng động mạch vành bị hẹp chủ yếu (0-3); thal: Loại bệnh tim của bệnh nhân (1: thalassemia trung bình 2: thalassemia nặng 3: không mắc bệnh thalassemia).




Hình 10: Giao diện chỉnh sửa tập chỉ số về tình trạng sức khỏe của bệnh nhân

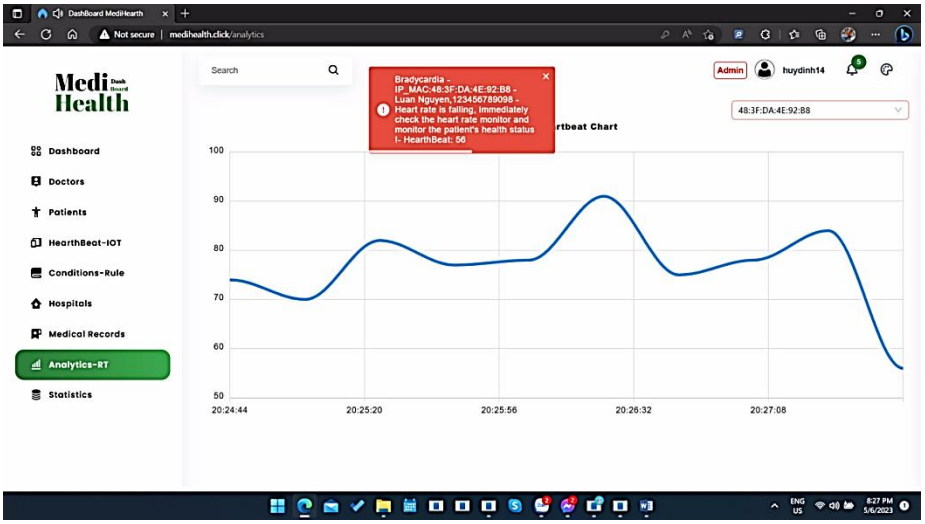
Từ các tập chỉ số này, cùng với các giá trị khác từ thiết bị đo được và thông tin của bệnh nhân, là các giá trị đầu vào của mô hình AI mà chúng tôi đã đề cập ở trên. Các tập giá trị này sẽ là dữ liệu đầu vào và đầu ra là một con số chỉ ra bệnh án đó chiếm bao nhiêu phần trăm tỉ lệ mắc bệnh tim mạch (*Hình 10*).
*Quản lý mầm bệnh: giao diện hỗ trợ xem thông tin về khoảng nhịp tim, cũng như cập nhật dữ liệu. "Disable": Nó thể hiện các khoảng mà nhịp tim rơi vào, Bradycardia là thấp [0-60], Normal [60-100] là nhịp tim bình thường, còn Tachycardia (> 100) thuộc nhịp tim cao, dựa vào các tập điều kiện này mà hệ thống sẽ đưa ra các cảnh báo tới bác sĩ (Hình 11).*

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

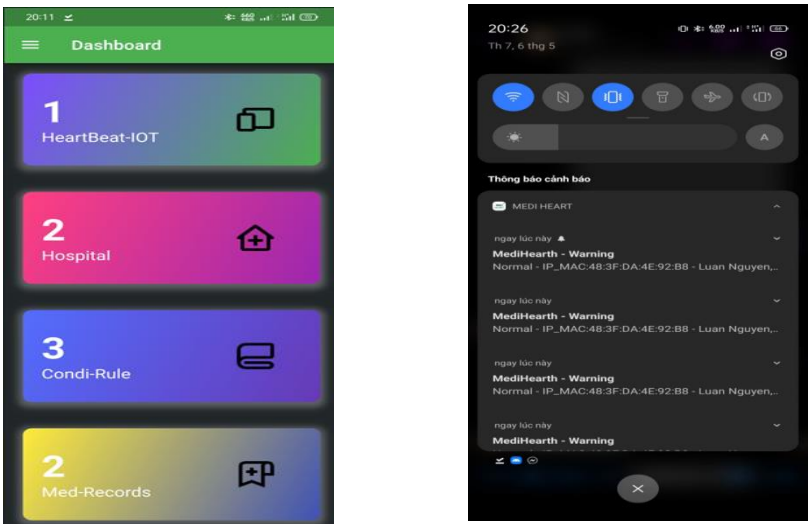



Hình 11: Giao diện quản lý tập luật từ chuyên gia

Để xem nhịp tim trực tiếp, ta cần phải chọn thiết bị cần xem. Ở đây ta có thể thấy, cảnh báo nhịp tim có màu đỏ khi nhịp tim xuống thấp sẽ được gửi cảnh báo với các thông tin cụ thể về thông tin bệnh nhân, nhịp tim bao nhiêu và thông tin thiết bị, giúp bác sĩ có thể nắm được thông tin nhanh và chính xác, từ đó đưa ra quyết định chính xác hơn (*Hình 12*).




Hình 12: Giao diện xem nhịp tim trực tiếp từ thiết bị bệnh nhân đang đeo



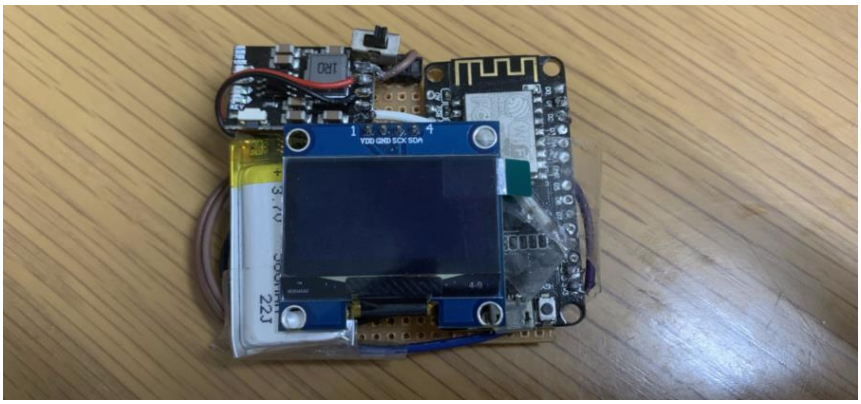

Hình 13: Màn hình chính và thông báo bất thường trên mobile

Về app mobile, đây là giao diện Mobile để thị thông tin tổng quan của hệ thống và thông báo của hệ thống khi có giá trị nhịp tim bất thường (*Hình 13)*.




Hình 14: Giao diện cảnh báo trên SmartWatch

Ngoài ra còn có thông báo trên smartwatch *SamSung Watch 4* khi kết nối Bluetooth với điện thoại di động *(Hình 14).*



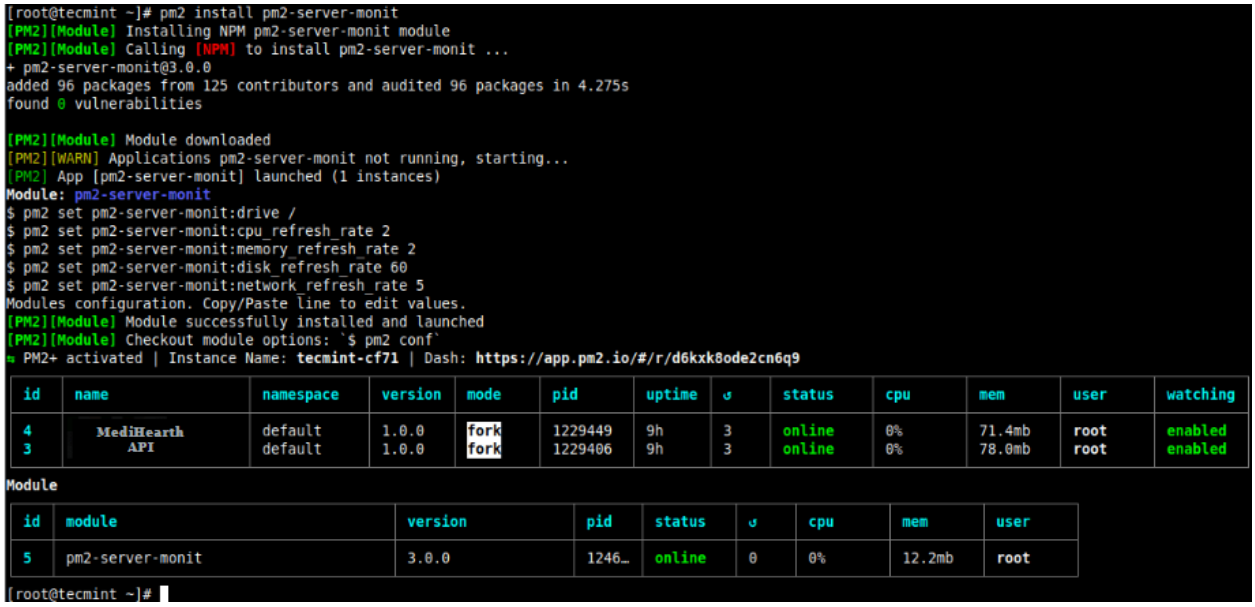
Hình 15: Thiết bị IoT sau khi hoàn thành lắp đặt

*Cuối cùng, đây là hình ảnh thiết bị chúng tôi đã xây dựng nó cùng với các linh kiện như đã nêu trên. Thiết kế khá nhỏ gọn nhưng còn cần phát triển thêm nữa để người dùng có thể đeo thoải mái nhất (Hình 15).*

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 4 KIỂM NGHIỆM THỰC TẾ

Chúng tôi sử dụng dịch vụ trên Digital Ocean để triển khai hệ thống, sử dụng PM2 [20] quản lý các tiến trình trong ứng dụng, PM2 cung cấp các tính năng như giám sát quy trình, tự động khởi động lại ứng dụng và cân bằng tải. Nó có thể chạy nhiều phiên bản ứng dụng của bạn, phân phối các yêu cầu đến trên chúng và tự động khởi động lại các quy trình bị lỗi (*Hình 16*).




Hình 16: Màn hình quản lý hệ thống Mediheath khi triển khai trên Digital Ocean

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này chúng tôi đã trình bày giải pháp kết hợp giữa IoT và máy học (tập luật) để giải quyết bài toán xây dựng hệ thống theo dõi, giám sát bệnh nhân theo thời gian thực. Chúng tôi đã xây dựng công cụ có tên là Medihealth dựa vào hướng tiếp cận này. Medihealth gồm hai phân hệ Web và App trên di động, đồng thời thiết kế một thiết bị đeo tay dành cho bệnh nhân có sự hỗ trợ, kết nối Internet. Công cụ được triển khai trên các nền tảng công nghệ như ReactJS cho Web, Flutter cho di động, Nodejs cho server và IoT cho thiết bị đeo tay. Đồng thời công cụ cũng đã triển khai trên các dịch vụ điện toán đám mây (Digital Ocean) nhằm tối ưu thời gian thực cho các thông số theo dõi bệnh nhân. Công cụ cung cấp các chức năng chính cho các người dùng bao gồm bác sỹ, bệnh nhân, nhân viên y tế. Trong đó bác sỹ có thể quản lý hồ sơ bệnh nhân cũng như theo dõi bệnh nhân thông qua thiết bị IoT và hệ thống Web, App. Bác sỹ có thể nhận được những thông tin cảnh báo theo thời gian thực từ bệnh nhân đang theo dõi qua hệ thống. Ngoài ra bệnh nhân cũng có thể xem nhịp tim theo thời gian thực, xem thống kê các thông số bệnh tim mạch trên biểu đồ từ đó có thể nhận được những cảnh báo kịp thời từ hệ thống. Bên cạnh đó, hệ thống có thể dễ dàng mở rộng cho các nhóm bệnh (thông qua các tập luật), từ đó các thiết bị đeo tay của bệnh nhân có thể được dùng trong nhiều nhóm bệnh. Một số ưu điểm của hệ thống bao gồm giao diện trực quan, dễ sử dụng, thiết bị có triển khai đại trà từ đó giảm quá tải nguồn lực y tế trong tình hình dịch bệnh hiện nay. Hiện công cụ Medihealth đã được chúng tôi thử nghiệm cho một số bệnh nhân tại TP.HCM. Tuy nhiên, công cụ vẫn tồn tại một số nhược điểm như độ trễ của các cảm biến trên thiết bị IoT đặc biệt trong lĩnh vực chăm sóc sức khỏe các độ trễ này là rủi ro cao. Trong tương lai, chúng tôi sẽ phát triển công cụ theo các hướng là bổ sung tính năng cảnh báo SOS và thông tin GPS, cải thiện quản lý và thống kê, áp dụng kiến trúc microservice và xây dựng hệ thống log để theo dõi hệ thống hoạt động hiệu quả và an toàn hơn.

## TÀI LIỆU THAM KHẢO

[1]. "Số liệu của Tổ chức Y tế Thế giới năm 2019", [Online], Xu hướng mắc bệnh tim mạch, Accessed: 1/5/2023.

[2]. AI, [Online], IoT - Nền tảng công nghệ cho y tế thông minh (mic.gov.vn), Accessed: 1/5/2023.

[3]. L.-P. Tôn, L.-S. Lê, and M.-S. Nguyen, "Micraspis: A Computer-Aided Proposal Toward Programming and Archi- tecting Smart IoT Wearables," IEEE Access, vol. 9, pp. 105 393–105 408, 2021.

[4]. Tensoflow – Neural Network, [Online], TensorFlow Neural Network Tutorial (stackabuse.com), Accessed: 1/5/2023.

[5]. Machine Learning, [Online], ML, Accessed: 1/5/2023.

[6]. TensorFlow, [Online], https://blog.tensorflow.org/2022/10/building-the-future-of-tensorflow.html, Accessed: 1/5/2023.

[7]. ReactJS, [Online], https://reactjs.org , Accessed: 1/5/2023.

[8]. Flutter, [Online], https://flutter.xn--dev-nga/, Accessed: 1/5/2023.

[9]. Firebase, [Online], https://firebase.google.com, Accessed: 1/5/2023.

[10]. ESP8266 NODEMCU, [Online], ESP8266 , Accessed: 1/5/2023.

[11]. Max30102, [Online], MAX30102 High-Sensitivity Pulse Oximeter and Heart-Rate Sensor for Wearable Health | Maxim Integrated (stg-maximintegrated.com), Accessed: 1/5/2023.

[12]. Oled 1.3inch, [Online], Interfacing 1.3 Inch I2C OLED Display Module with Arduino - Electropeak, Accessed: 1/5/2023.

[13]. Mạch sạc pin lithium 3.7V, [Online], Mạch sạc pin Lithium 3.7v/4.7v có ngõ ra 5V 2A Linh Kiện 888 (linhkien888.vn), Accessed: 1/5/2023.

[14]. Pin Lithium, [Online], PinLithium,

[15]. MongoDB, [Online], https://www.mongodb.com/Atlas, Accessed: 1/5/2023.

[16]. Digital Ocean, [Online], https://cloud.digitalocean.com, Accessed: 1/5/2023.

[17]. NodeJs, [Online], https://nodejs.org, Accessed: 1/5/2023

[18]. K. Faaborg and S. Pasquali. (2017). Mastering Node.js Second Edition. Packt Publishing

[19]. C. Doxsey. (2012). An Introduction to Programming in Go. CreateSpace Independent Publishing Platform.

[20]. PM2, [Online], PM2-Link, Accessed: 1/5/2023.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F018**

# CADIOCARE: XÂY DỰNG HỆ THỐNG HỖ TRỢ CHĂM SÓC SỨC KHỎE THÔNG MINH CHO BỆNH NHÂN TIM MẠCH VỚI SỰ HỖ TRỢ ChatGPT

TÔN LONG PHƯỚC[1*], VÕ THÀNH NHỚ[1], BÙI SỸ SƠN[1]

[1]*Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
**tonlongphuoc@iuh.edu.vn*

**Tóm tắt.** Hiện nay, bệnh tim mạch là một trong những căn bệnh phổ biến và nguy hiểm trên thế giới. Các loại bệnh về tim mạch bao gồm đau thắt ngực, đột quỵ, bệnh động mạch vành và suy tim. Tại Việt Nam, xu hướng tử vong do bệnh tim mạch ngày càng gia tăng, với tỷ lệ tử vong tăng từ 127,3/100.000 dân (năm 2000) lên 164,9/100.000 dân hiện nay. Từ đó cần có một hệ thống chăm sóc sức khỏe thông minh cho bệnh nhân tim mạch. Để giải quyết vấn đề này, chúng tôi đã đề xuất công cụ có tên là CardioCare - một nền tảng chăm sóc sức khỏe thông minh dành cho bệnh nhân mắc bệnh tim mạch. Điểm đặc biệt của dự án là cả bệnh nhân và bác sĩ đều có thể theo dõi tình trạng sức khỏe bệnh nhân hàng ngày, bệnh nhân có thể liên hệ trực tiếp với bác sĩ chuyên khoa tim mạch qua tin nhắn và cuộc gọi video. Công cụ cũng hỗ trợ chatbot để giải đáp thắc mắc của người dùng và nhiều chức năng hữu ích khác. Chúng tôi đã sử dụng các công nghệ tiên tiến như NodeJS, NestJs, TypeScript, ReactJS, Flutter và các công nghệ hỗ trợ như ChatGPT, Google Cloud, ZegoCloud, Serper, AWS, Firebase, socket IO, Github để xây dựng Front-end, Back-end và ứng dụng chạy trên thiết bị di động. Hiện tại, công cụ đã hoàn thành và đã đưa vào thử nghiệm cho các bệnh nhân mắc bệnh tim mạch tại TP.HCM. Trong thời gian tới, chúng tôi sẽ cố gắng hoàn thiện và nhân rộng công cụ ra toàn quốc. Chúng tôi sẽ phát triển các chức năng để quản lý, theo dõi bệnh án của người dùng đối với tất cả các nhóm bệnh phổ biến.

**Từ khóa.** Ứng dụng Web, Ứng dụng di động, AWS services, ChatGPT, Chăm sóc sức khỏe, AWS.

## CADIOCARE: BUILDING AN INTELLIGENT HEALTHCARE SUPPORT SYSTEM FOR CARDIOVASCULAR PATIENTS WITH ChatGPT SUPPORT

**Abstract.** Currently, cardiovascular disease is one of the most common and dangerous diseases in the world. Types of cardiovascular disease include angina, stroke, coronary artery disease, and heart failure. In Vietnam, the trend of deaths due to cardiovascular disease is increasing, with the mortality rate increasing from 127.3/100,000 people (in 2000) to 164.9/100,000 people at present. To address this issue, we have created CardioCare - a smart healthcare platform for patients with cardiovascular disease. The project's special feature is that both patients and doctors can monitor their daily health status, and patients can contact a specialist in cardiovascular disease directly via messaging and video calls. The project also supports a chatbot to answer users' questions and many other useful functions. We have used advanced technologies such as NodeJS, NestJs, TypeScript, ReactJS, and Flutter and supporting technologies such as ChatGPT, Google Cloud, ZegoCloud, Serper, AWS, Firebase, socket IO, Github to build Front-end, Back-end, and Mobile. Currently, the project has been completed and is ready for testing for patients with cardiovascular disease in Ho Chi Minh City. In the future, we will strive to improve and expand the project nationwide and develop it to manage and monitor users' diseases for all types of diseases.

**Keywords.** Web application, Mobile application, AWS services, ChatGPT, Healthcare, AWS

## 1 GIỚI THIỆU TỔNG QUAN

Chúng tôi xin giới thiệu công cụ "Cadiocare: hệ thống hỗ trợ chăm sóc sức khỏe thông minh cho bệnh nhân tim mạch với sự hỗ trợ ChatGPT", một giải pháp đáng chú ý trong việc cải thiện chất lượng chăm sóc sức

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

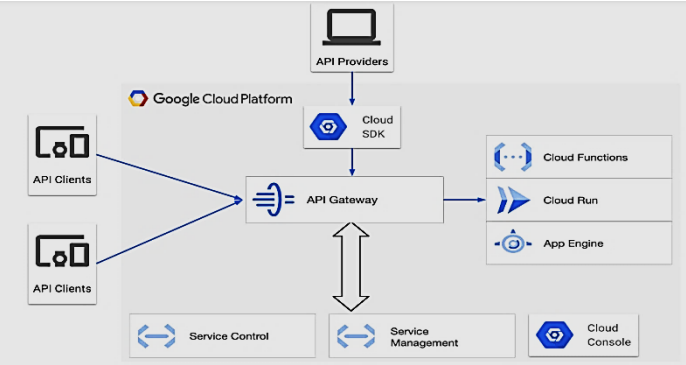
khỏe cho bệnh nhân tim mạch. Bệnh tim mạch là một vấn đề sức khỏe quan trọng và phức tạp, đòi hỏi sự quan tâm và chăm sóc đều đặn. Tuy nhiên, việc theo dõi và tự quản lý bệnh thường gặp khó khăn như là giám sát và ghi nhận các chỉ số sức khỏe: Bệnh nhân tim mạch thường phải đo và ghi lại các chỉ số như nhịp tim, huyết áp, mức đường huyết hoặc cân nặng. Điều này đòi hỏi sự quan tâm và công sức để thực hiện và ghi lại thông tin này một cách chính xác và đầy đủ, tìm kiếm thông tin và tư vấn: Bệnh nhân thường cần thông tin và tư vấn về bệnh tim mạch, quy trình điều trị, lối sống lành mạnh và các biện pháp phòng ngừa. Tuy nhiên, việc tìm kiếm thông tin đáng tin cậy và giải đáp các câu hỏi có thể mất thời gian và không luôn đảm bảo được tính chính xác và hiệu quả, khả năng định vị trong tình huống khẩn cấp: Trong trường hợp bệnh nhân gặp tình huống khẩn cấp như cơn đau tim, việc định vị và kết nối nhanh chóng với nhân viên y tế là rất quan trọng. Tuy nhiên, việc xác định đúng vị trí và thông báo tình huống khẩn cấp có thể gặp khó khăn trong một tình huống căng thẳng và khẩn cấp. Vì vậy, chúng tôi đã đề xuất xây dựng một hệ thống thông minh, sử dụng ChatGPT, nhằm mang đến sự hỗ trợ toàn diện và tiện lợi cho bệnh nhân tim mạch.

Hệ thống của chúng tôi tiếp nhận các chỉ số sau đó đưa ra chẩn đoán và gửi thông báo tình trạng sức khỏe hàng ngày đến cả bệnh nhân và bác sĩ điều trị. Điều này giúp bệnh nhân có thể tự theo dõi tình trạng sức khỏe và bác sĩ cũng có thể biết được diễn biến sức khỏe của bệnh nhân. Hệ thống còn có thể giúp bệnh nhân kết nối trực tiếp với bác sĩ chuyên khoa về tim mạch có thể liên lạc trực tiếp thông qua tin nhắn, gọi điện và đặt lịch hẹn với bác sĩ. Bên cạnh đó, hệ thống còn được trang bị tính năng chatbot để cung cấp tư vấn tự động, giải đáp các câu hỏi phổ biến về bệnh tim mạch. Điều này giúp giảm thiểu thời gian chờ đợi và tăng cường sự tiện lợi trong việc tìm kiếm thông tin y tế. Đặc biệt, hệ thống của chúng tôi cũng tích hợp sử dụng Google Map để hỗ trợ định vị bệnh nhân trong trường hợp khẩn cấp. Khi bệnh nhân gặp tình huống cần sự giúp đỡ ngay lập tức, họ có thể kích hoạt tính năng SOS và hệ thống sẽ tự động gửi thông tin vị trí đến bác sĩ điều trị để được giúp đỡ nhanh chóng và chính xác. Kế hoạch triển khai của chúng tôi bao gồm các giai đoạn nghiên cứu yêu cầu, thiết kế chi tiết, phát triển hệ thống, kiểm thử và đánh giá hiệu suất, và cuối cùng là triển khai thực tế. Chúng tôi cam kết đảm bảo chất lượng và hiệu quả của hệ thống, đồng thời đáp ứng đầy đủ các yêu cầu và mong muốn của bệnh nhân và nhà điều trị. Nghiên cứu này hứa hẹn đóng góp vào việc cải thiện chất lượng chăm sóc sức khỏe cho bệnh nhân tim mạch và đồng thời định hướng phát triển lĩnh vực chăm sóc sức khỏe thông minh. Chúng tôi hy vọng rằng giải pháp của chúng tôi sẽ mang lại lợi ích thiết thực và tạo ra sự thay đổi tích cực trong cuộc sống của những người bị bệnh tim mạch.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 Google Cloud Platform

Google Cloud Platform (GCP) là tập hợp các dịch vụ điện toán đám mây, được Google giới thiệu vào ngày 07 tháng 4 năm 2008. GCP chạy trên cơ sở hạ tầng giống như các sản phẩm của Google, chẳng hạn như công cụ tìm kiếm Google, YouTube, v.v. GCP cung cấp nhiều công cụ và dịch vụ khác nhau giúp quản lý cơ sở hạ tầng, cung cấp máy chủ và định cấu hình mạng [1]. Hoạt động của kiến trúc gọi api trên google cloud platform như sau: Api provider sẽ cung cấp api cho người dùng sau đó lưu vào cloud SDK. Để sử dụng API từ nhà cung cấp thì người dùng phải thông qua API Gateway [2].




Hình 1: Kiến trúc gọi api trên Google Cloud Platform [2]

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 2.2 ChatGPT

ChatGPT, viết tắt của Chat Generative Pre-training Transformer, là một chatbot sử dụng công nghệ AI phổ biến hiện nay. Chúng tôi đã tích hợp ChatGPT vào hệ thống của mình để cải thiện khả năng hiểu và tạo ngôn ngữ tự nhiên của hệ thống. ChatGPT là một chatbot do OpenAI phát triển, dựa trên GPT-3.5. GPT-3.5 là một dòng mô hình ngôn ngữ lớn được đào tạo trên một bộ dữ liệu khổng lồ gồm văn bản và mã. ChatGPT có thể được sử dụng để tạo văn bản, dịch ngôn ngữ, viết các loại nội dung sáng tạo khác nhau và trả lời câu hỏi của bạn một cách đầy đủ thông tin. Chúng tôi đã sử dụng ChatGPT để cải thiện khả năng hiểu các câu hỏi của người dùng và trả lời các câu hỏi đó một cách đầy đủ thông tin. Chúng tôi cũng đã sử dụng ChatGPT để hỗ trợ người dùng được tư vấn các thông tin sức khỏe một cách nhanh chóng, cũng như hỗ trợ bác sĩ trong công việc chăm sóc sức khỏe khách hàng. Chúng tôi tin rằng việc tích hợp ChatGPT vào hệ thống của mình sẽ giúp chúng tôi cung cấp cho người dùng trải nghiệm tốt hơn [3, 4].

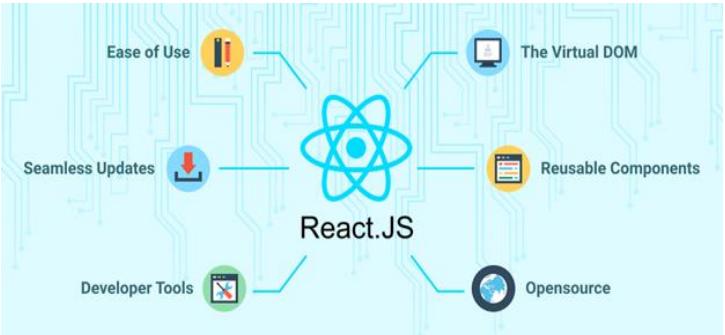
## 2.3 Zego Cloud

Zego cloud là một nền tảng điện toán đám mây tiên tiến được thiết kế để cung cấp các dịch vụ và giải pháp đám mây linh hoạt và tin cậy cho các doanh nghiệp và tổ chức. Với sự kết hợp giữa công nghệ hàng đầu và khả năng mở rộng linh hoạt, Zego cloud mang đến một môi trường đám mây mạnh mẽ cho việc lưu trữ, quản lý và triển khai ứng dụng trực tuyến. Zego cloud cung cấp một API gọi video mạnh mẽ và linh hoạt cho các ứng dụng và dịch vụ trực tuyến. API gọi video của Zego cloud cho phép các nhà phát triển tích hợp chức năng gọi video trực tiếp vào ứng dụng của họ, mang lại trải nghiệm tương tác thời gian thực cho người dùng. Với API gọi video của Zego cloud, bạn có thể xây dựng các tính năng gọi video như cuộc gọi nhóm, cuộc gọi 1-1 và phân chia màn hình trong ứng dụng của mình. API hỗ trợ cả âm thanh và video chất lượng cao, đảm bảo rằng người dùng có thể tương tác một cách rõ ràng và mượt mà. Bạn có thể tận dụng API gọi video để xây dựng ứng dụng trực tuyến đa nền tảng, từ ứng dụng di động đến ứng dụng web và desktop [5].



Hình 2: Cuộc gọi trực tuyến trên ZegoCloud [5]

## 2.4 ReactJS

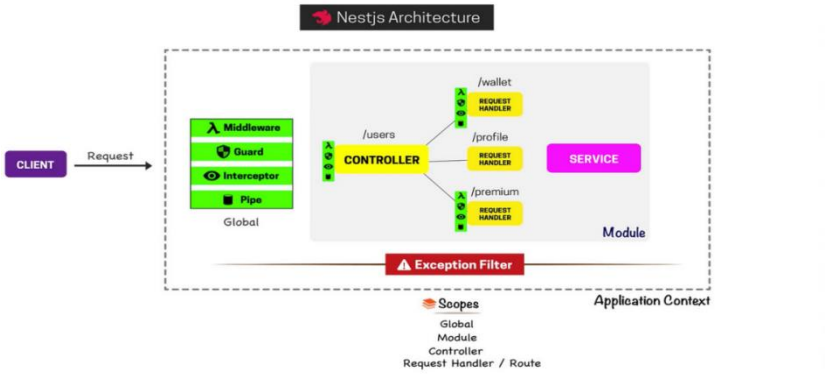



Hình 3: Các đặc điểm chính của Reactjs [6]

ReactJS là một thư viện JavaScript mã nguồn mở của Facebook, giúp xây dựng giao diện người dùng linh hoạt và hiệu quả trong ứng dụng web và di động. Nó sử dụng mô hình thành phần JSX và React Hooks để tạo giao diện dễ đọc, quản lý trạng thái ứng dụng và tăng tính mô-đun. ReactJS là một công cụ phổ biến trong việc xây dựng giao diện người dùng tương tác.

## 2.5 NestJs

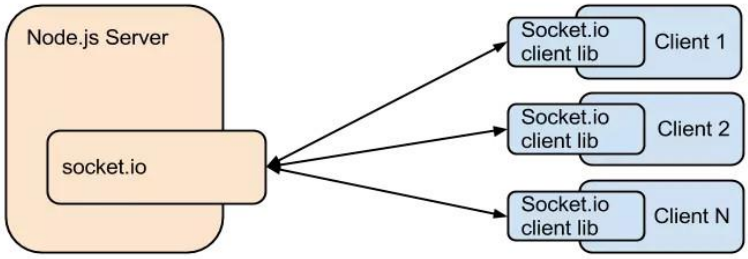
NestJS là một framework phát triển ứng dụng back-end với TypeScript và JavaScript, xây dựng trên Node.js. Với mô hình controller-service, Dependency Injection và middleware, NestJS giúp xây dựng ứng dụng back-end chất lượng cao, dễ quản lý và có tính mô-đun cao. Với cộng đồng phát triển mạnh mẽ và tích hợp linh hoạt, NestJS trở thành một lựa chọn phổ biến cho việc phát triển ứng dụng back-end đáng tin cậy và mạnh mẽ trên nền tảng Node.js [8].




Hình 4: Kiến trúc của NestJs [7]

## 2.6 SocketIO

Socket.IO là một thư viện cho phép truyền thông hai chiều, dựa trên sự kiện và có độ trễ thấp giữa máy khách và máy chủ. Nó được xây dựng trên giao thức WebSocket và cung cấp các đảm bảo bổ sung như sự rớt kết nối tự động hoặc sử dụng HTTP long-polling nếu không thể sử dụng WebSocket [9].




Hình 5: Cơ chế hoạt động của Socket IO [9]

# 3 CÔNG CỤ CARDIOCARE

## 3.1 Giới thiệu

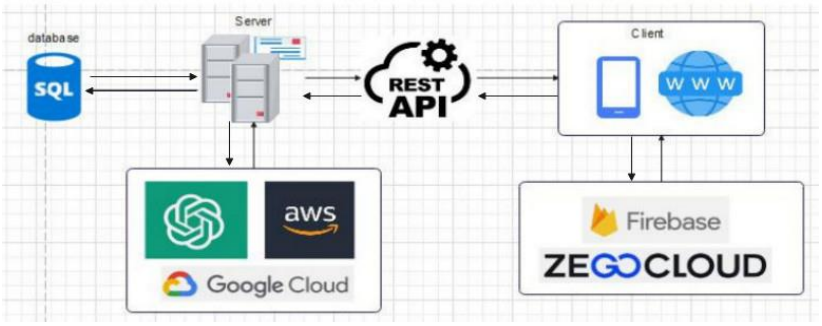
Hệ thống của chúng tôi được xây dựng bằng một số công nghệ chính như Node.js, Nest.js, JavaScript, Reactjs, Flutter, Socket.IO, Recharts, JSON Web Token (JWT) và cơ sở dữ liệu MySQL. Ngoài ra, chúng tôi đã tích hợp một số công nghệ bên ngoài như ChatGPT, AWS, Firebase Authentication [10], Zego Cloud, Serper và Google Cloud. Hệ thống được triển khai trên hai nền tảng là website và ứng dụng di động. Nest.js được sử dụng để xây dựng máy chủ cho hệ thống, phối hợp với MySQL để lưu trữ dữ liệu. React.js được sử dụng để xây dựng website, trong khi Flutter được sử dụng để phát triển ứng dụng di động cho cả hai nền tảng Android và iOS. Cả hai đều có các chức năng tương tự và được thiết kế theo yêu cầu của bài toán. Socket.IO được sử dụng để cung cấp tính năng giao tiếp thời gian thực (real-time) cho các chức năng như nhắn tin, thông báo, và nhiều hơn nữa. JWT được sử dụng để mã hóa dữ liệu dựa trên JSON, giúp xác thực và truyền dữ liệu an toàn giữa các bên. Recharts được sử dụng để xây dựng biểu đồ, giúp người dùng có cái nhìn trực quan hơn về dữ liệu. Firebase Authentication là một dịch vụ điện toán đám mây nổi tiếng, hỗ trợ đăng ký và xác thực thông qua số điện thoại và email. Chúng tôi sử dụng AWS (Amazon Web Services) và dịch vụ S3 của nó để lưu trữ hình ảnh, video và các tài nguyên khác [11]. Serper là một công cụ tìm

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

kiếm nhanh chóng giúp người dùng tìm kiếm thông tin. Zego Cloud cung cấp các dịch vụ hỗ trợ giao tiếp trực tuyến. Google Cloud là một nền tảng điện toán đám mây mạnh mẽ, trong hệ thống của chúng tôi, chúng tôi sử dụng Google Maps để người dùng có thể tìm kiếm, định vị và nhận gợi ý về các địa điểm mong muốn. Cuối cùng, chúng tôi sử dụng OpenAI làm nền tảng điện toán đám mây để triển khai ChatGPT, một hệ thống tự động trả lời cho người dùng trong lĩnh vực chăm sóc sức khỏe [12, 13].

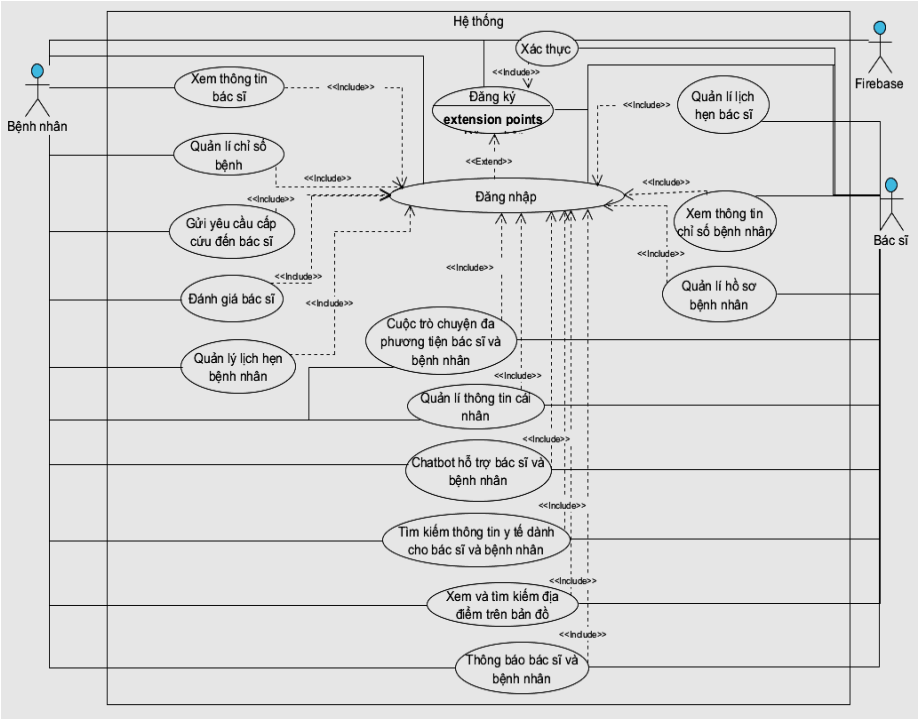
## 3.2 Kiến trúc hệ thống

Ứng dụng của chúng tôi áp dụng kiến trúc RESTful API để xây dựng và thiết kế. Người dùng có thể tương tác với ứng dụng thông qua giao diện web hoặc di động. Khi người dùng thực hiện các hành động như đăng nhập, đăng ký hoặc thực hiện các tác vụ khác trên giao diện, yêu cầu của người dùng sẽ được gửi đến máy chủ. Máy chủ sẽ xử lý yêu cầu của người dùng, truy xuất dữ liệu từ cơ sở dữ liệu và trả về kết quả tương ứng cho người dùng. Nếu người dùng muốn sử dụng chatbot AI, máy chủ sẽ gọi ChatGPT để lấy kết quả từ câu hỏi của người dùng và trả về cho người dùng. Máy chủ sử dụng dịch vụ Google Map của Google Cloud để cung cấp định vị của người dùng và thông tin về các cơ sở y tế gần nhất. Cơ sở dữ liệu chứa thông tin về người dùng, thông tin liên quan khác và cung cấp cho máy chủ để quản lý và truy xuất dữ liệu. Kiến trúc này cho phép các thành phần của ứng dụng hoạt động độc lập và tương tác với nhau để cung cấp cho người dùng các tính năng và trải nghiệm tốt nhất.



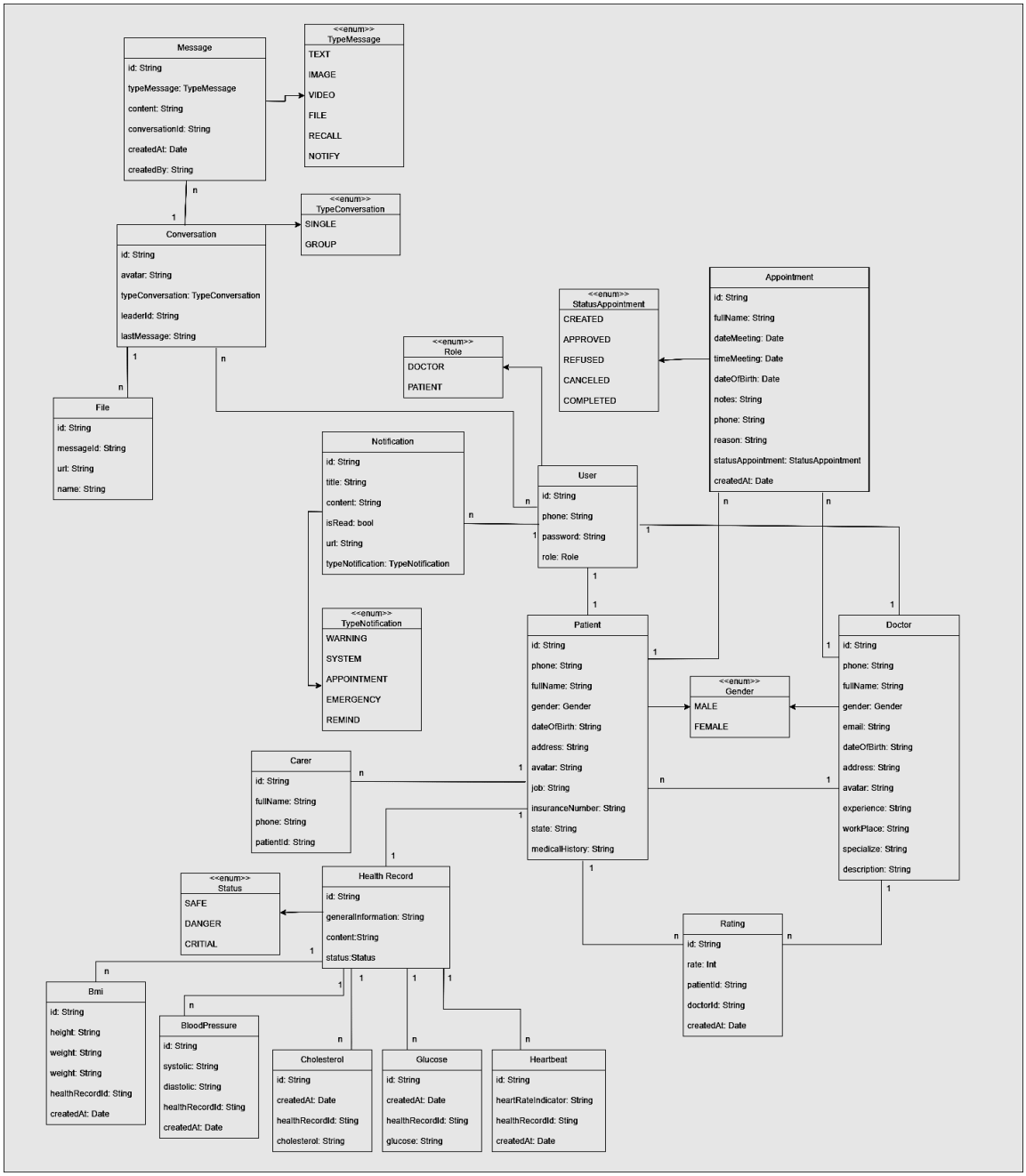

Hình 6: Mô hình Restful API

## 3.3 Phân tích hệ thống CARDIOCARE




Hình 7: Sơ đồ Use-case của Cadiocare

Đối tượng của CardioCare là các bệnh nhân bị các vấn đề tim mạch và các bác sĩ chuyên ngành tim mạch. Hệ thống mục đích giúp bác sĩ có thể theo dõi chỉ số cho các bệnh nhân tim mạch từ xa, và bệnh nhân có thể tìm hiểu, tư vấn thông qua AI, giải quyết việc bệnh nhân phải đến bệnh viện mới nhận được sự tư vấn của bác sĩ. Sơ đồ usecase có hai actors chính là bệnh nhân, bác sĩ và một actors phụ là Firebase (*Hình 7*).




Hình 8: Sơ đồ Class Diagram của Cadiocare

Lớp User dùng để lưu thông tin đăng nhập và phân quyền người dùng là bác sĩ hay bệnh nhân để vào màn hình tương ứng. Lớp Doctor và lớp Patient lưu thông tin bác sĩ và bệnh nhân và bệnh nhân sẽ có một bác sĩ. Tương ứng với một bệnh nhân sẽ có lớp HealthRecord để lưu thông tin sức khỏe hàng ngày và lưu qua từng lớp Bmi, Cholesterol, Glucose, Heartbeat, BloodPressure. Lớp Notification lưu các thông báo cho từng bệnh nhân và bác sĩ. Lớp Conversation thực hiện các cuộc trò chuyện của bệnh nhân và bác sĩ và những tin nhắn được lưu vào từng lớp Message tương ứng (*Hình 8*). Lớp Appointment dùng để thực hiện lưu các thông tin lịch hẹn của bác sĩ và bệnh nhân.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.4 Giao diện của hệ thống

Trong giao diện website sử dụng các công nghệ ChatGPT hỗ trợ chat AI, công nghệ Zego Cloud sử dụng để thực hiện các cuộc gọi video, khám trực tuyến, công nghệ Serper hỗ trợ người dùng có thể tìm kiếm thông tin mong muốn.

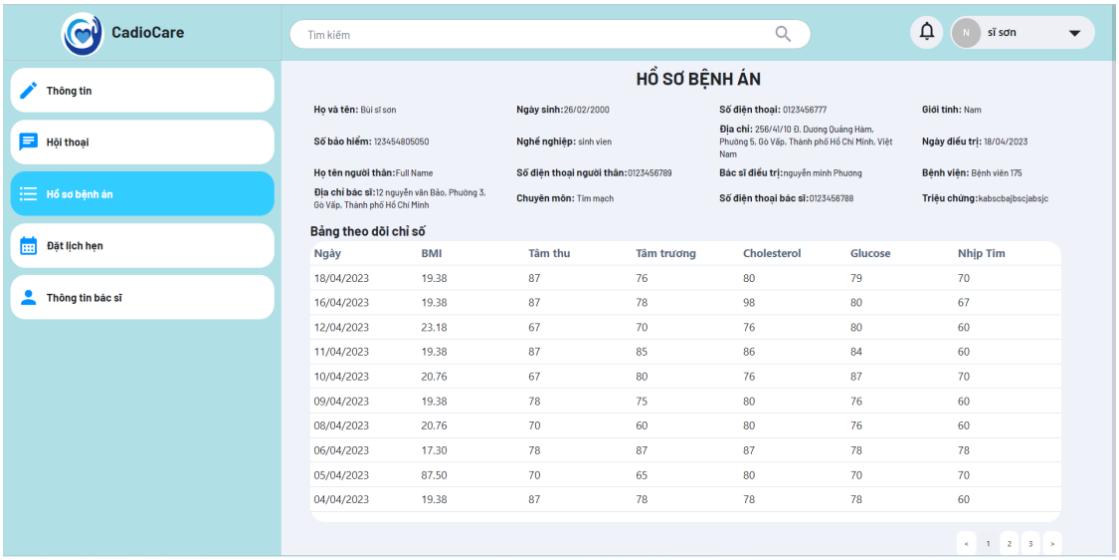
#### 3.4.1 Một số giao diện website

Trang chủ dành cho bệnh nhân của chúng tôi được thiết kế với nhiều chức năng hữu ích. Khi bạn truy cập vào trang chủ, bạn sẽ thấy một menu nằm ở bên trái cho phép bạn chuyển đến các trang khác trong hệ thống (*Hình 9*).




Hình 9: Giao diện quản lý của bệnh nhân

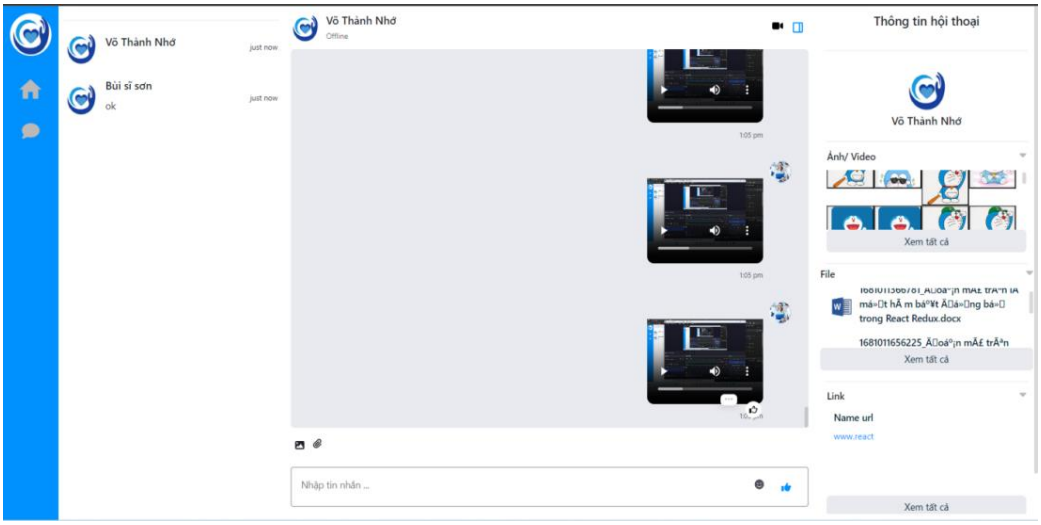
Bạn có thể nhấp vào biểu tượng "Thêm thông tin" để thêm các chỉ số bệnh hằng ngày của bạn. Thông tin về chỉ số và trạng thái bệnh sẽ được hiển thị một cách chi tiết và chính xác. Ngoài ra, chúng tôi cung cấp biểu đồ tình trạng bệnh trong vòng 10 ngày gần đây để bạn có cái nhìn tổng quan về sự phát triển của bệnh. Nếu bạn cần cấp cứu khẩn cấp, hãy nhấp vào biểu tượng cấp cứu và chúng tôi sẽ đáp ứng ngay lập tức.




Hình 10: Giao diện xem thông tin chỉ số bệnh của bệnh nhân

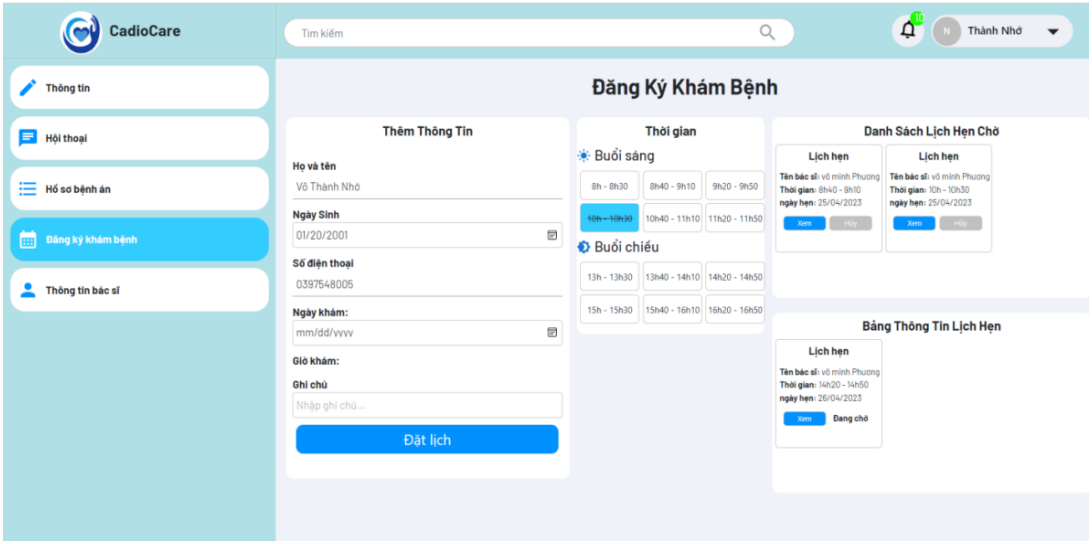
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Màn hình (*Hình 10*) hồ sơ bệnh án người bệnh có thể xem lại hồ sơ của mình và bản chỉ số nhập hằng ngày của mình




Hình 11: Giao diện trò chuyện giữa bệnh nhân và bác sĩ

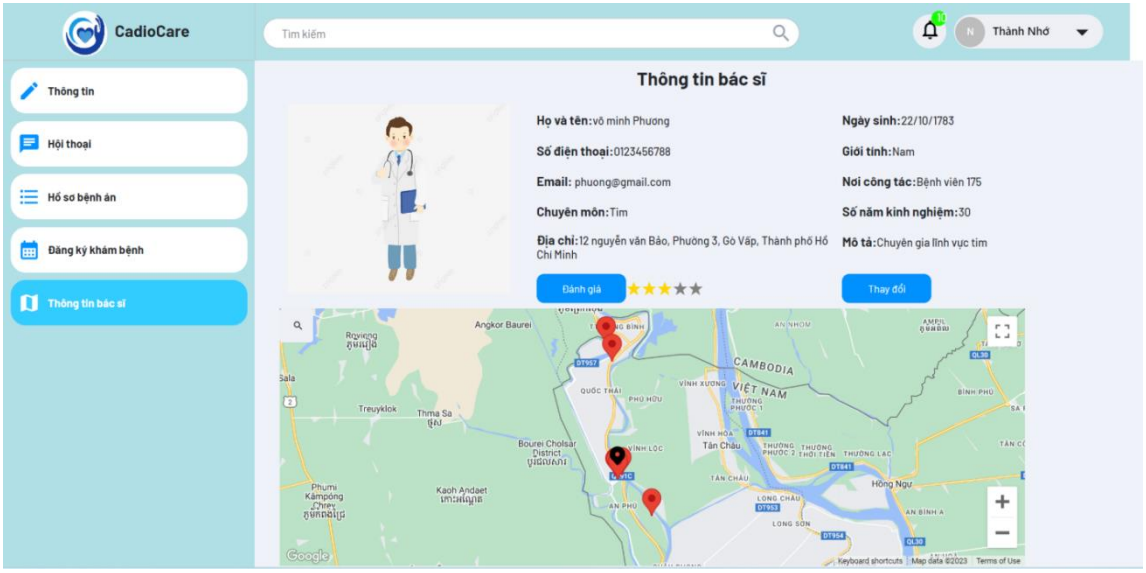
Ở màn hình này (*Hình 11*), người dùng có nhiều chức năng để quản lý cuộc trò chuyện của mình một cách dễ dàng. Bạn có thể xem danh sách các cuộc trò chuyện hiện có của mình, và khi chọn một cuộc trò chuyện cụ thể, hệ thống sẽ hiển thị thông tin chi tiết về cuộc trò chuyện đó, bao gồm các tin nhắn, video, ảnh và file đã gửi.




Hình 12: Giao diện đăng ký khám bệnh

Màn hình này (Hình 12) cung cấp cho bệnh nhân các chức năng quan trọng để quản lý lịch khám bệnh và tương tác với bác sĩ điều trị. Bệnh nhân có thể dễ dàng thực hiện các hành động sau đây. Trước tiên, bệnh nhân có thể đăng ký lịch khám bệnh với bác sĩ điều trị thông qua giao diện này. Đồng thời, họ có thể cung cấp các thông tin cần thiết trước khi đặt lịch để giúp bác sĩ có cái nhìn toàn diện hơn về tình trạng sức khỏe.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh



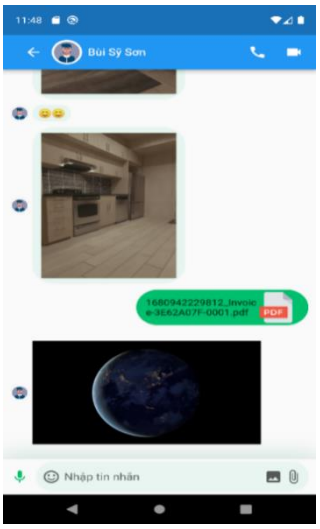

Hình 13: Giao diện xem thông tin bác sĩ và bản đồ

*Hình 13*, cung cấp cho bệnh nhân một loạt chức năng hữu ích để tương tác với thông tin và đánh giá các bác sĩ. Dưới đây là các chức năng mà bệnh nhân có thể sử dụng.

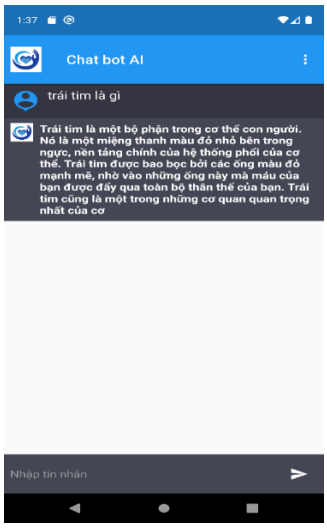
### 3.4.2 Một số giao diện ứng dụng di động




Hình 14: Màn hình trang chủ



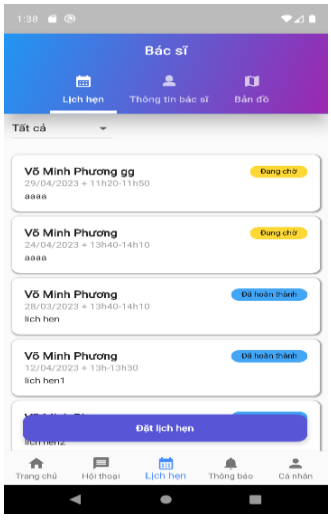

Hình 15: Màn hình hội thoại



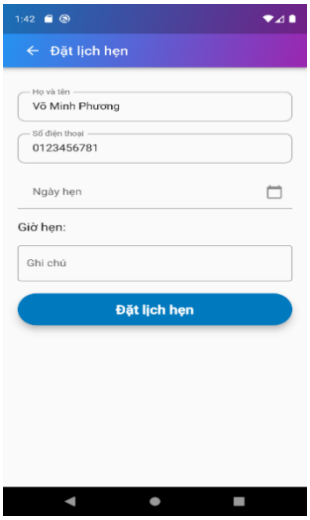

Hình 16: Chatbot

Một số màn hình trên ứng dụng di động của hệ thống Cadiocare. Người dùng có thể xem thông tin hồ sơ bệnh án của mình và thực hiện các chức năng trao đổi thông tin trên hệ thống với bác sĩ hoặc với hệ thống Chatbot thông minh ChatGPT (*Hình 14*, *Hình 15*, *Hình 16*). Một số chức năng hỗ trợ quản lý các lịch hẹn giữa bệnh nhân và bác sĩ (*Hình 17*, *Hình 18*, *Hình 19*).

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 17: Màn hình xem lịch hẹn




Hình 18: Màn hình đặt lịch hẹn




Hình 19: Màn hình thông báo

## 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này chúng tôi trình bày giải pháp chăm sóc sức khỏe thông minh có sự kết hợp ChatGPT. Để triển khai giải pháp trên, chúng tôi đã xây dựng thành công một hệ thống hỗ trợ chăm sóc sức khỏe thông minh có tên Cadiocare. Hệ thống được thiết kế để cung cấp cho người dùng thông tin hữu ích về sức khỏe và hỗ trợ cho các vấn đề liên quan đến bệnh tim mạch. Hệ thống đã giải quyết được cơ bản bài toán đặt ra là hỗ trợ theo dõi, chăm sóc sức khoẻ và điều trị tại nhà cho bệnh nhân tim mạch, giúp kết nối bệnh nhân với bác sĩ một cách nhanh chóng. Nhờ các tính năng thông minh (tích hợp ChatGPT để hỗ trợ tư vấn bệnh nhân) và tự động trong việc cảnh báo hay trao đổi thông tin dựa vào các tập luật của chuyên gia. Hệ thống còn có các chuyên gia theo dõi hỗ trợ mà bệnh nhân có thể điều trị tại nhà hiệu quả an toàn chính xác hơn, có thể hỗ trợ bệnh nhân ngay cả trong tình huống khẩn cấp nhờ tính năng định vị SOS. Tính thực tiễn của hệ thống cao trong bối cảnh tình trạng bệnh nhân mắc các bệnh về tim và mong muốn điều trị tại nhà ngày càng nhiều. Hiện hệ thống đã triển khai thử nghiệm cho một số bệnh nhân tại Tp.HCM. Tuy nhiên, hệ thống cũng còn một số hạn chế cần phải hoàn thiện. Trong tương lai, chúng tôi sẽ tiếp tục phát triển hệ thống bao gồm việc tích hợp các công nghệ nhận dạng giọng nói và xử lý ngôn ngữ tự nhiên để cải thiện khả năng tương tác với người dùng, cũng như mở rộng phạm vi ứng dụng của hệ thống để hỗ trợ các bệnh nhân có các vấn đề sức khỏe khác.

## REFERENCES

[1] Google Cloud (2023), [online], https://cloud.google.com/api-gateway/docs/architecture-overview, Accessed 25/05/2023

[2] Google Cloud (2023), [online], www.educba.com/google-cloud-platform Accessed 15/04/2023

[3] Chat GPT (2023), [online], https://akenda.vn/10-meo-su-dung-chatgpt-hieu-qua-tu-lan-dau-tien/ – Accessed 25/04/2023

[4] Vi.wikipedia.org (2023), [online], https://vi.wikipedia.org/wiki/ChatGPT Accessed 15/04/2023

[5] ZegoCloud (2023), [online], https://www.zegocloud.com/ Accessed 15/04/2023

[6] Reactjs [online] https://www.faqs.com.pk/what-is-react-js/– [Accessed 25/05/2023]

[7] NestJs [online] https://peakd.com/hive-181335/@sagarkothari88/learning-nestjs-and-contributing-to – [Accessed 15/04/2023]

[8] NodeJs vs SocketIO[online] https://viblo.asia/p/buoc-dau-lam-quen-voi-nodejs-va-socketio-MJyGjQrWvPB – [Accessed 15/04/2023]

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

[9] Socket IO [online] https://socket.io/docs/v4/ – [Accessed 15/04/2023]

[10] Firebase (2023), [online], https://firebase.google.com/docs/auth Accessed 15/04/2023

[11] GUPTA, Bulbul; MITTAL, Pooja; MUFTI, Tabish. A review on amazon web service (AWS), microsoft azure & google cloud platform (gcp) services. In: Proceedings of the 2nd International Conference on ICT for Digital, Smart, and Sustainable Development, ICIDSSD 2020, 27-28 February 2020, Jamia Hamdard, New Delhi, India. 2021.

[12] BISWAS, Som S. Role of chat gpt in public health. Annals of biomedical engineering, 2023, vol 51(5): 868-869.

[13] CASCELLA, Marco, et al. Evaluating the feasibility of ChatGPT in healthcare: an analysis of multiple clinical and research scenarios. Journal of Medical Systems, 2023, vol 47(1): 33-56

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F019**

# BUGHOUSE - ỨNG DỤNG QUẢN LÝ HỢP ĐỒNG THÔNG MINH CHO CÁC DỊCH VỤ CHO THUÊ PHÒNG DỰA TRÊN KỸ THUẬT BLOCKCHAIN

TÔN LONG PHƯỚC[1*], VÕ THỊ TRÀ GIANG[1], ĐOÀN NGỌC QUỐC BẢO[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*tonlongphuoc@iuh.edu.vn*

**Tóm tắt.** Với sự gia tăng số lượng sinh viên, người lao động từ các tỉnh tập trung tại các đô thị lớn để học tập và làm việc. Nhu cầu tìm trọ gần trường, gần chỗ làm việc đúng yêu cầu, an ninh đảm bảo vẫn là bài toán khó. Ngoài ra, các hợp đồng thuê trọ hiện nay cũng chưa được quản lý nhằm bảo vệ quyền lợi người thuê cũng như người cho thuê. Để giải quyết bài toán này, chúng tôi đề xuất công cụ có tên là BUGHOUSE. Công cụ BUGHOUSE là một ứng dụng cho phép người dùng tìm kiếm, quản lý phòng trọ trên nền tảng Blockchain sử dụng cơ chế của hợp đồng thông minh (smart contract) để minh bạch tất cả giao dịch. Qua đó các giao dịch dựa trên các thỏa thuận hợp đồng từ trước, sẽ được tự động hóa quá trình chuyển nhận tài chính và lưu trữ thông tin giao dịch. Chúng tôi chọn công nghệ NodeJs cho Back-end, ReactJs cho Front-end nền tảng Web. Với nền tảng di động, chúng tôi chọn React-Native. Ngoài ra hệ thống của chúng tôi được triển khai trên các nền tảng điện toán đám mây, tích hợp các dịch vụ Render của AWS. Hiện, công cụ được triển khai thử nghiệm cho thị trường tại TP.HCM. Trong tương lai, chúng tôi sẽ mở rộng phạm vi ứng dụng đồng thời quảng bá công cụ đến người dùng.

**Từ khóa.** Ứng dụng thuê trọ, NodeJS, Hợp đồng thông minh, Chuỗi khối, Điện toán đám mây, AWS.

## BUGHOUSE- A BLOCKCHAIN-BASE SMART CONTRACT APPLICATION FOR TRANSPARENT ROOM RENTAL MANAGER

**Abstract.** The large number of students and workers from the provinces who gather in big cities to study and work is increasing. Therefore, the need to find accommodation near the school, near the required workplace, and ensure security is still a difficult problem. In addition, the current rental contracts have not been managed to protect the interests of tenants as well as lessors. To solve this problem, we propose a tool called BUGHOUSE. BUGHOUSE is an application that allows users to search and manage accommodations on the Blockchain platform using the mechanism of smart contracts to make all transactions transparent. Thereby, transactions based on pre-existing contractual agreements will automate the process of financial transfer and storage of transaction information. We choose NodeJs technology for Back-end, and ReactJs for Front-end for Web. For the mobile platform, we choose React-Native. In addition, our system is deployed on cloud computing platforms, integrating AWS Render services. It is also piloted for the market in Ho Chi Minh City. In the future, we will expand the scope of the application while promoting the tool to users.

**Keywords.** Rental House Application, NodeJS, Smart Contract, Blockchain, Cloud Computing, AWS.

## 1 GIỚI THIỆU TỔNG QUAN

Trong bối cảnh TP.HCM đang trở thành một thành phố lớn với nhu cầu tìm kiếm chỗ ở ngày càng tăng cao, các vấn đề liên quan đến việc thuê nhà trở thành một thách thức đối với sinh viên và người đi làm mới đến thành phố. Các yêu cầu cơ bản như chất lượng sống, giá cả phù hợp và an ninh đòi hỏi người tìm nhà phải đối mặt với nhiều khó khăn.

Ngoài ra, việc sử dụng các hợp đồng thuê truyền thống cũng mang theo nhiều rủi ro cho cả người thuê và người cho thuê. Các hợp đồng này có thể không rõ ràng về điều khoản, gặp sai sót trong quá trình tạo hợp

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

đồng và thường có thời hạn dài, gây khó khăn khi muốn thay đổi điều khoản hoặc chấm dứt hợp đồng. Để đảm bảo tính công minh, việc có sự giám sát từ bên thứ ba là cần thiết.
Người cho thuê cũng đối mặt với nhiều khó khăn trong việc quản lý nhà trọ. Việc tìm kiếm và thu hút khách hàng tiềm năng là một nhiệm vụ khó khăn do sự cạnh tranh cao trong ngành cho thuê trọ tại các thành phố lớn. Quản lý nhiều phòng và các dịch vụ đi kèm cũng tạo ra rủi ro sai sót. Hơn nữa, việc quản lý hoá đơn hàng tháng cũng gặp nhiều hạn chế và cần được chú trọng. Sự minh bạch trong các giao dịch giữa người thuê và người cho thuê là cần thiết để đảm bảo tính công bằng cho cả hai bên.
Để giải quyết các vấn đề này, chúng tôi đề xuất ứng dụng Bughouse - một phần mềm quản lý dịch vụ thuê phòng tiện ích được xây dựng dựa trên công nghệ Blockchain. Hệ thống sẽ được triển khai trên cả hai nền tảng website và ứng dụng di động nhằm có thể đáp ứng các nhu cầu của người dùng một cách tốt nhất. Với giao diện thân thiện dễ dàng sử dụng, chúng tôi hiển thị danh sách phòng cùng với các thông tin như tên phòng, mô tả, giá tiền, địa chỉ, … để thuận tiện cho người dùng tìm kiếm được phòng phù hợp với nhu cầu.
Công nghệ Blockchain mang lại nhiều lợi ích quan trọng trong quản lý giao dịch. Blockchain là một hệ thống phân tán và không thể thay đổi, trong đó mọi giao dịch và thông tin được lưu trữ trong các khối liên kết an toàn và minh bạch. Hiện nay, Blockchain đã trở thành một công nghệ phổ biến và được áp dụng trong nhiều lĩnh vực quản lý giao dịch với những ưu điểm đáng kể [1].
Tính minh bạch là một ưu điểm chính của Blockchain. Các giao dịch trên Blockchain được lưu trữ trong các khối liên kết và không thể thay đổi, tạo ra một bản sao chính xác và công khai của toàn bộ lịch sử giao dịch. Điều này đảm bảo tính minh bạch và đáng tin cậy, cho phép người dùng kiểm tra và xác minh thông tin giao dịch một cách dễ dàng.
Tính an toàn và bảo mật là một lợi thế quan trọng của Blockchain. Công nghệ này sử dụng các thuật toán mật mã mạnh mẽ để bảo vệ thông tin giao dịch. Mỗi giao dịch trên Blockchain được xác nhận bởi một mạng lưới ngang hàng của các người tham gia, làm cho việc tấn công và thay đổi dữ liệu trở nên khó khăn.
Blockchain cũng mang lại tốc độ và hiệu suất cao. Nó có khả năng xử lý giao dịch nhanh chóng và hiệu quả, vượt qua các hạn chế về thời gian và quy trình trong việc xác nhận giao dịch truyền thống. Điều này đặc biệt hữu ích trong các hệ thống giao dịch đòi hỏi tốc độ và khả năng mở rộng, như thanh toán và giao dịch tài chính. Một ưu điểm khác của Blockchain là loại bỏ hoặc giảm thiểu sự phụ thuộc vào các bên trung gian. Blockchain giúp giảm chi phí và thời gian giao dịch, đồng thời tạo ra môi trường trực tiếp giữa các bên tham gia, tăng tính công bằng và sự tự chủ. Hơn nữa, công nghệ Blockchain còn cung cấp tính năng tự động thực thi thông qua hợp đồng thông minh (smart contract). Hợp đồng thông minh là các chương trình tự thực thi các điều khoản và điều kiện của hợp đồng một cách tự động. Điều này loại bỏ sự phụ thuộc vào bên thứ ba trung gian và đảm bảo thực hiện chính xác các điều khoản hợp đồng, giảm thiểu rủi ro và chi phí liên quan [2]. Đây cũng là tính năng chúng tôi đề xuất sẽ sử dụng cho hệ thống quản lý thuê phòng Bughouse. Hợp đồng sẽ được thiết lập các điều khoản rõ ràng dựa trên yêu cầu của cả hai bên, giúp tránh được các tranh chấp tìm ẩn và đảm bảo quyền lợi của các bên. Hợp đồng sẽ được tự động thực thi các chức năng khi các điều khoản được hai bên chấp thuận và ký kết. Chúng tôi còn đề xuất ví điện tử riêng để thuận tiện cho cho các giao dịch trong hệ thống.
Bài báo của chúng tôi chia làm 5 phần: Phần 1 Giới thiệu tổng quan về hệ thống; Phần 2 Các công nghệ liên quan; Phần 3 Công cụ Bughouse; Phần 4 Kiểm nghiệm thực tế và Phần 5 Kết luận và hướng phát triển. Trong đó Phần 3 là phần đóng góp chính của bài báo.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 NodeJS

Node.js là một nền tảng dựa trên JavaScript được sử dụng chủ yếu để tạo các ứng dụng web chuyên sâu về I/O chẳng hạn như ứng dụng trò chuyện, trang web phát trực tuyến đa phương tiện, v.v. Nó được xây dựng trên công cụ JavaScript V8 của Google Chrome. Ứng dụng web là phần mềm chạy trên máy chủ và được hiển thị bởi trình duyệt máy khách truy cập tất cả các tài nguyên của ứng dụng thông qua Internet [3]. Node.js cho phép phát triển các ứng dụng mạng chạy phía máy chủ, như các ứng dụng web, API, ứng dụng di động, và nhiều hơn nữa. Nó cung cấp khả năng xử lý yêu cầu đồng thời (concurrency) cao, khả năng xây dựng các ứng dụng thời gian thực và khả năng mở rộng tốt.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

## 2.2 ReactJS - React-Native

ReactJS là một thư viện giao diện người dùng dựa trên thành phần, mã nguồn mở, chỉ chịu trách nhiệm cho lớp xem của ứng dụng. ReactJS không phải là một framework, nó chỉ là một thư viện do Facebook phát triển để giải quyết một số vấn đề mà chúng tôi gặp phải trước đó. Thư viện này hỗ trợ cho việc tạo giao diện người dùng cùng các xử lý tương tác của người dùng với giao diện [4]. ReactJS sử dụng mô hình Component-Based để xây dựng giao diện người dùng. Một component trong ReactJS đại diện cho một phần của giao diện, nơi bạn có thể xây dựng các thành phần nhỏ hơn và kết hợp chúng lại để tạo thành giao diện hoàn chỉnh. Điều này giúp quản lý và tái sử dụng mã dễ dàng hơn, giúp tăng tính module và hiệu suất của ứng dụng.

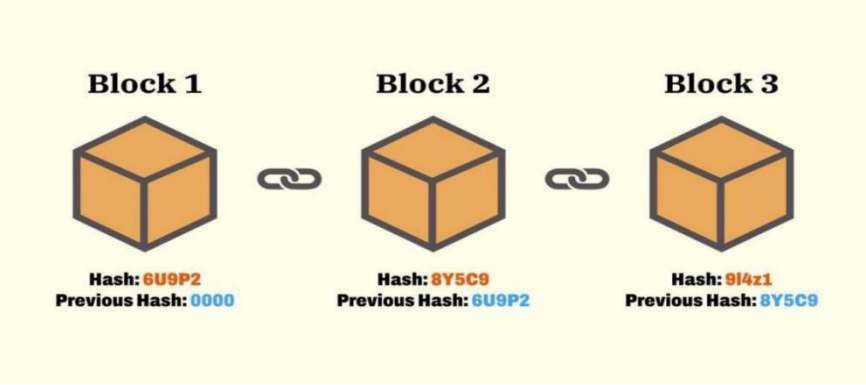
React Native là một framework được sử dụng để xây dựng ứng dụng di động đa nền tảng, dựa trên JavaScript và React - một thư viện phát triển giao diện người dùng được tạo bởi Facebook [5]. React Native cho phép xây dựng các ứng dụng di động bằng cách sử dụng các thành phần và API của JavaScript và React, và sau đó biên dịch thành mã nguồn tương ứng cho các nền tảng di động như Android và iOS. Điều này cho phép bạn viết code một lần và chạy ứng dụng trên nhiều nền tảng, giúp giảm thời gian và công sức cần thiết để phát triển và duy trì ứng dụng di động.

## 2.3 Solidity

Solidity là ngôn ngữ cấp cao (high-level), curly-bracket, hướng đối tượng thường dùng để thực hiện các hợp đồng thông minh (Smart Contract). Solidity chịu ảnh hưởng nhiều bởi C++, Python, Javascript [6]. Solidity là một ngôn ngữ hướng đối tượng, nó có hết các đặt tính kế thừa, trừu tượng, đóng gói, đa hình. Điều này giúp solidity phát triển các hợp đồng thông minh dễ dàng hơn, tải sử dụng mã nguồn linh hoạt. Solidity sở hữu cú pháp tương tự JavaScript, cú pháp đơn giản, dễ tiếp cận với người dùng hơn. Ngoài ra, nó còn hỗ trợ kiểm tra dữ liệu tĩnh, giúp ngăn lỗi trong quá trình thực thi hợp đồng thông minh (Smart Contract). Solidity cũng hỗ trợ tính năng bảo mật như kiểm tra tính hợp lệ của giao dịch, hạn chế truy cập vào dữ liệu của hợp đồng thông minh từ bên ngoài, và các tính năng khác để đảm bảo tính an toàn và đáng tin cậy của hợp đồng thông minh.

## 2.4 Blockchain

Blockchain được thiết lập như một công nghệ trong đó cơ sở dữ liệu phân tán ghi lại tất cả các giao dịch đã xảy ra trong mạng ngang hàng. Nó được coi là một mô hình điện toán phân tán đã khắc phục thành công vấn đề liên quan đến niềm tin của một bên thứ ba. Mỗi giao dịch khi sinh ra được chèn vào chuỗi các khối và mỗi khối được tạo bởi một hàm băm duy nhất (chuỗi chữ và số do mã hóa dữ liệu bằng khóa riêng và khóa chung mật mã), một nonce (một số duy nhất cho khối) và bởi một hàm băm từ khối trước đó (*Hình 1*). Điều này khiến cho Blockchain không thể thay đổi đảm bảo tính an toàn và bảo mật [1, 2].



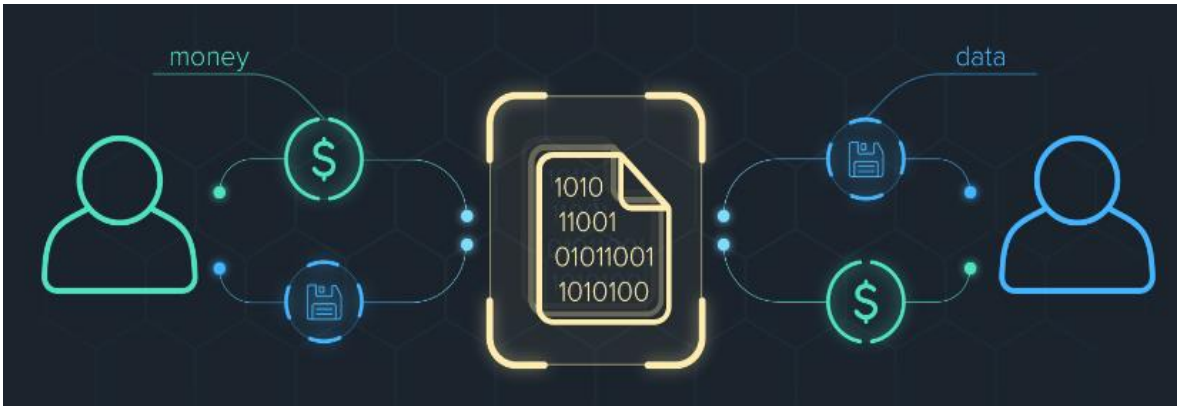

Hình 1: Liên Kết Giữa Các Khối trong Blockchain

## 2.5 Smart Contract

Hợp đồng thông minh (Smart Contract) là một chương trình hoặc giao thức đơn giản được lưu trữ trên Blockchain, có khả năng tự động thực thi các thỏa thuận được đáp ứng trước đó mà không cần thông qua

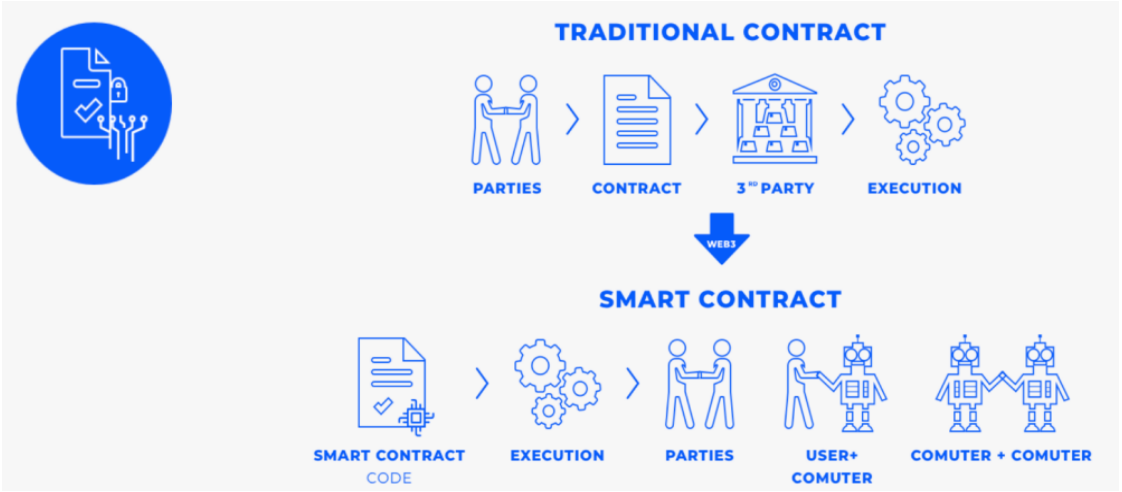
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

bất kỳ bên trung gian nào. Chúng được viết bằng các ngôn ngữ cấp cao như Python hoặc Solidity cho các ứng dụng trên nền tảng Ethereum và được biên dịch bởi Solidity hoặc Serpent. Khi hợp đồng thông minh được thực thi thành công, nó sẽ được triển khai lên nền tảng Blockchain và được gán một địa chỉ duy nhất [1, 2]. Một cách đơn giản, hợp đồng thông minh có thể coi như một tài khoản trên nền tảng Blockchain, bao gồm số dư và là mục tiêu của các giao dịch (*Hình 2*). Tuy nhiên, điểm khác biệt chính là hợp đồng thông minh không được điều khiển bởi người dùng mà được tự động triển khai. Người dùng có thể tương tác với hợp đồng thông qua các giao dịch. Một đặc điểm quan trọng của hợp đồng thông minh là chúng không thể bị xóa và các tương tác với chúng không thể đảo ngược theo mặc định. Điều này đảm bảo tính toàn vẹn và không thể thay đổi của các điều khoản và điều kiện trong hợp đồng. Hợp đồng thông minh mang lại sự tự động, minh bạch và không thể thay đổi trong việc thực hiện các thỏa thuận. Chúng giúp loại bỏ sự phụ thuộc vào các bên trung gian, giảm thiểu rủi ro và tạo ra một môi trường công bằng và tin cậy cho các giao dịch trên Blockchain [7]. Trong hệ thống Bughouse, chúng tôi sử dụng Smart Contract để tự động quản lý các hoạt động của phòng thuê. Từ ký kết hợp đồng, gia hạn, thanh toán đến quản lý trạng thái, Smart Contract đảm bảo tính chính xác và minh bạch. Điều này giúp tiết kiệm thời gian, giảm sai sót và tạo một môi trường tin cậy cho việc quản lý phòng thuê.




Hình 2: Sơ Đồ Mô Tả SmartContract [10]

Trong hệ thống Smart House, chúng tôi sử dụng Smart Contract (Hình 3) để tự động quản lý các hoạt động của phòng thuê. Từ ký kết hợp đồng, gia hạn, thanh toán đến quản lý trạng thái, Smart Contract đảm bảo tính chính xác và minh bạch. Điều này giúp tiết kiệm thời gian, giảm sai sót và tạo một môi trường tin cậy cho việc quản lý phòng thuê.




Hình 3: Hợp đồng truyền thống và Smart Contract [10]

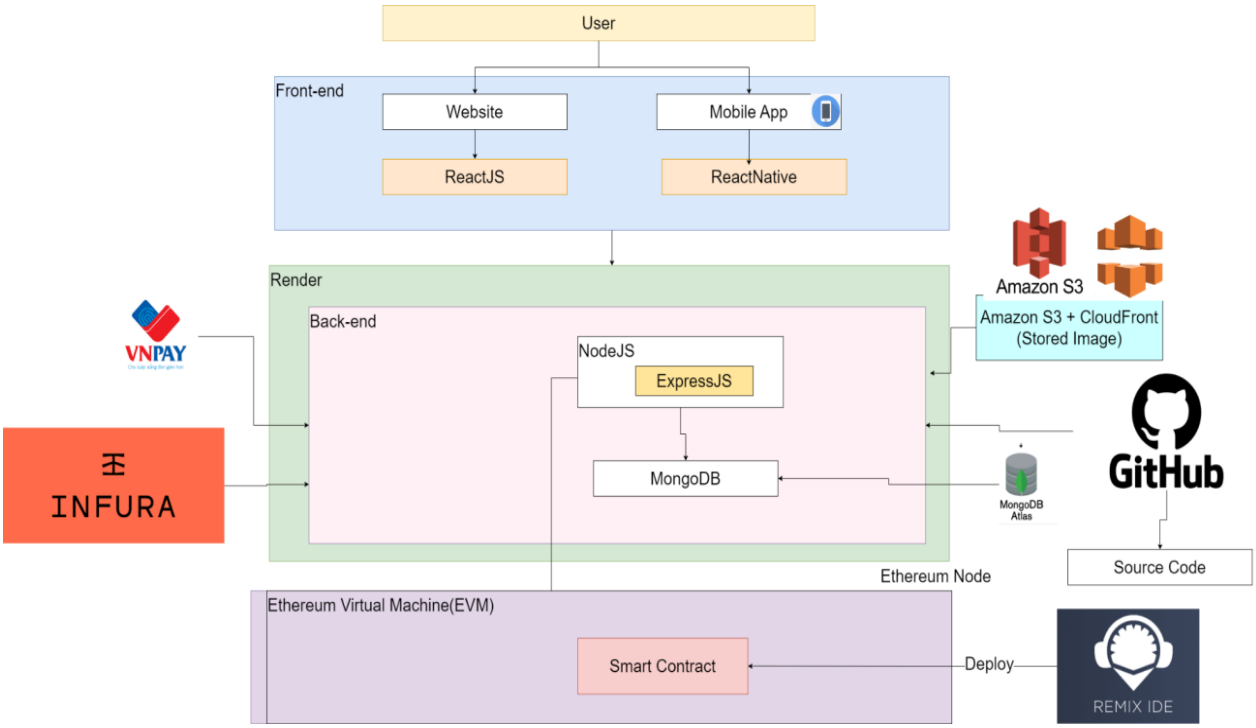
# 3 CÔNG CỤ BUGHOUSE

## 3.1 Giới thiệu

Để triển khai các giải pháp mà chúng tôi nêu ra trong phần giới thiệu, chúng tôi đề xuất xây dựng ứng dụng *Bughouse*. Ứng dụng triển khai dựa trên các công nghệ như NodeJS, ReactJS/React-Native. Và để Bughouse có tính ứng dụng hơn, chúng tôi sử dụng các dịch vụ của Render và Vercel để triển khai. Ứng dụng Bughouse được triển khai tên hai nền tảng website và ứng dụng điện thoại. Trong đó, về phần giao diện cho ứng dụng chúng tôi sử dụng ReactJS để xây dựng giao diện website, React Native để xây dựng giao diện di động. NodeJS sẽ đảm nhận việc tạo API để xử lý toàn bộ request của hệ thống. Solidity để triển khai Smart Contract lên Blockchain. MongoDB để lưu trữ dữ liệu của toàn bộ hệ thống [8]. Ngoài ra hệ thống còn được tích hợp các dịch vụ như: AWS S3 và CloudFront. Trong đó thì S3 dùng để lưu trữ dữ liệu dưới dạng file/folder và CloudFront để giúp tăng tốc độ truy cập file trong S3 [9].

## 3.2 Kiến trúc phần mềm

Phần mềm sử dụng mô hình NodeJS và ExpressJS, mô hình MVC (Model-View-Controller) Models đại diện cho dữ liệu và logic liên quan đến dữ liệu trong ứng dụng. Các models trong Bughouse có thể đại diện cho các khái niệm như User (người dùng), Room (phòng), Contract (Hợp đồng), … Models sẽ định nghĩa cấu trúc dữ liệu, quan hệ giữa các đối tượng, và các phương thức để tương tác với dữ liệu, chẳng hạn như tạo mới, lấy, cập nhật hoặc xóa. Views là thành phần chịu trách nhiệm hiển thị giao diện người dùng và tương tác với người dùng. Trong ứng dụng Bughouse, views là template engine Handlebars để tạo ra giao diện người dùng như giao diện mail, otp. Views sẽ nhận dữ liệu từ Controllers và sắp xếp, định dạng nó để hiển thị cho người dùng. Controllers là thành phần điều khiển và quản lý luồng dữ liệu và logic trong ứng dụng. Controllers trong Bughouse sẽ xử lý các yêu cầu từ người dùng, gọi các phương thức từ models để tương tác với dữ liệu, và trả về các views tương ứng để hiển thị cho người dùng. Controllers cũng có thể xử lý các tác vụ như xác thực, quản lý phiên, kiểm tra quyền truy cập, và thực hiện các thao tác logic phức tạp khác. Routing: ExpressJS sẽ được sử dụng để định tuyến các yêu cầu HTTP đến các controllers tương ứng. Các tệp tin route sẽ xác định các đường dẫn và phương thức HTTP tương ứng và ánh xạ chúng đến các controllers để xử lý (Hình 4).



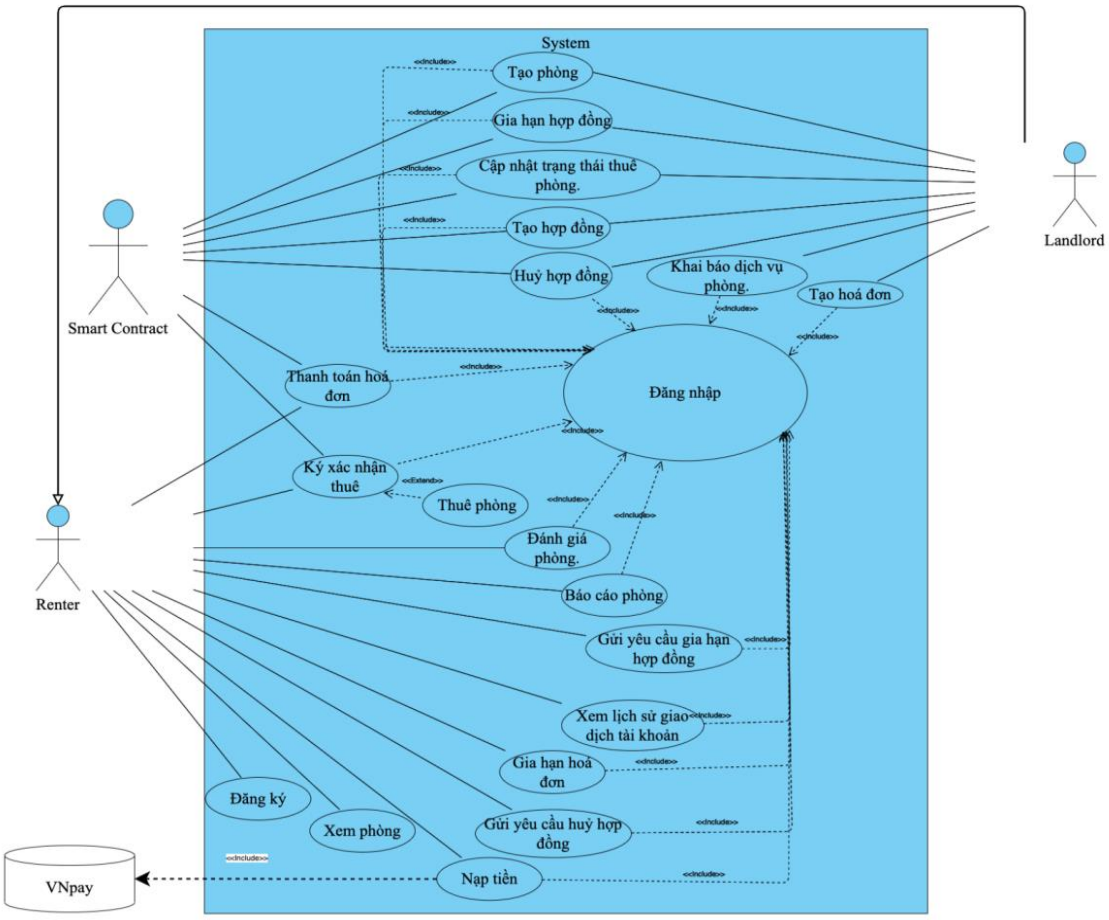

Hình 4: Sơ đồ kiến trúc phần mềm công cụ BUGHOUSE


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.3 Phân tích hệ thống Bughouse

Đối tượng người dùng cho hệ thống Bughouse gồm có người thuê và và người cho thuê. Trong đó, chúng tôi tập trung vào các chức năng chính của hệ thống là hỗ trợ người muốn cho thuê cũng như đi thuê phòng và tự động hóa tuần tự của các giao dịch trong hợp đồng đã được lập bởi người dùng. Ngoài ra, các chức năng phụ khác cũng được đưa vào để người dùng có thể dễ dàng sử dụng hơn như hỗ trợ các ví điện tử, thực hiện các giao dịch thanh toán thông qua ví điện tử VNPay.

Từ những chức năng trên, chúng tôi tập trung hiện thực các chức năng chính trong hệ thống App cũng như Web *Bughouse* như: Tạo phòng, thuê phòng, ký xác nhận và mốt chức năng tự đồng dưới hệ thống. Sau đó mới mở rộng các chức năng như *Hình 5*. Trong lược đồ use-case, chúng tôi sẽ có hai actor chính là Landlord (Người cho thuê) và Renter (Người thuê). Trong đó các chức năng chính của hệ thống chính là: *Use-case Tạo phòng, Use-case Ký xác nhận, Use-case Thanh toán hoá đơn.* Bên cạnh đó, chúng tôi cũng mở rộng các use-case cho các chức năng của hệ thống tích hợp thêm các dịch vụ của bên thứ ba như VNPay nhằm tạo điều kiện thuận lợi cho người dùng.




Hình 5: Mô hình Use-case của Bughouse

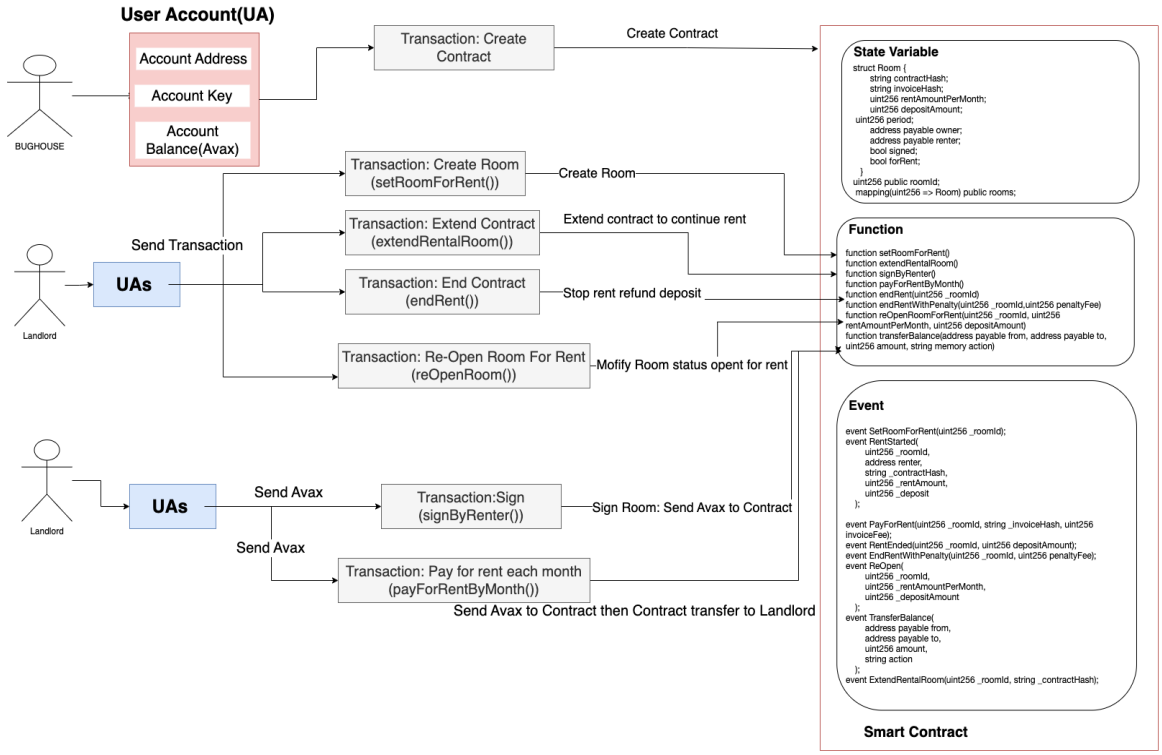
### 3.4 Phân tích Smart Contract của Bughouse

Hợp đồng thông minh (Smart Contract) đứng giữa người cho thuê và người đi thuê với mục đích là đảm bảo thỏa thuận cho thuê được ký, tiền thuê được chuyển cho người đi thuê đúng hạn, tiền cọc được giữ bởi hợp đồng thông minh, gia hạn và kết thúc hợp đồng được thực hiện chính xác (Hình 5).

Người cho thuê và người thuê được cấp tài khoản. Các tài khoản này được kiểm soát bởi các khoá riêng. Người dùng có thể dùng tài khoản này để tạo các giao dịch chuyển tiền hoặc gọi các chức năng trên hợp đồng. Hợp đồng sẽ được phân quyền theo từng đối tượng trong hệ thống [11].

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Đối với người cho thuê có các quyền như tạo phòng với các chỉ số tiền thuê, tiền cọc, thời hạn, trạng thái của phòng khi vừa được khởi tạo sẽ là "!forrent". Gia hạn thời gian thuê phòng người cho thuê sau khi nhận được yêu cầu gia hạn của người đi thuê thì sẽ gọi lên hợp đồng thực thi phương thức gia hạn hợp đồng với thông tin hợp đồng sau khi được cập nhật lại thời hạn, và thời hạn gia hạn để hợp đồng cập nhật lại thông tin phòng. Huỷ hợp đồng khi người cho thuê hoặc người đi thuê gửi yêu cầu huỷ hợp đồng. Hợp đồng kiểm tra lại thời hạn thuê phòng, nếu thời hạn hợp đồng kết thúc thì tiền cọc sẽ được hợp đồng chuyển về ví người thuê. Ngược lại, tiền cọc sẽ được chuyển về ví của chủ trọ như thoả thuận từ trước. Trạng thái phòng sẽ được chuyển sang trạng thái "not-ready" và thông tin người đi thuê sẽ bị xóa. Mở lại trạng thái cho thuê phòng người cho thuê gọi lên hợp đồng thực thi phương thức mở lại trạng thái phòng cùng với thông tin của phòng cần mở lại. Hợp đồng chuyển trạng thái phòng "for rent".
Đối với người đi thuê thì sẽ có những quyền như ký xác nhận thuê phòng là người thuê xác nhận thuê phòng qua hợp đồng đồng thời chuyển tiền thuê và tiền cọc cho hợp đồng, hợp đồng giữ lại tiền cọc và chuyển tiền thuê cho người cho thuê. Trạng thái phòng cho thuê sẽ được chuyển sang trạng thái "rented", và việc ký này khi thành công thì không thể thực hiện xác nhận một lần nữa, tránh việc ghi đè thông tin thuê hiện tại. Thanh toán tiền phòng hàng tháng khi người thuê thanh toán hoá đơn tiền phòng hợp đồng kiểm tra số dư của người dùng. Hợp đồng nhận tiền từ người thuê và chuyển tiền cho người cho thuê.

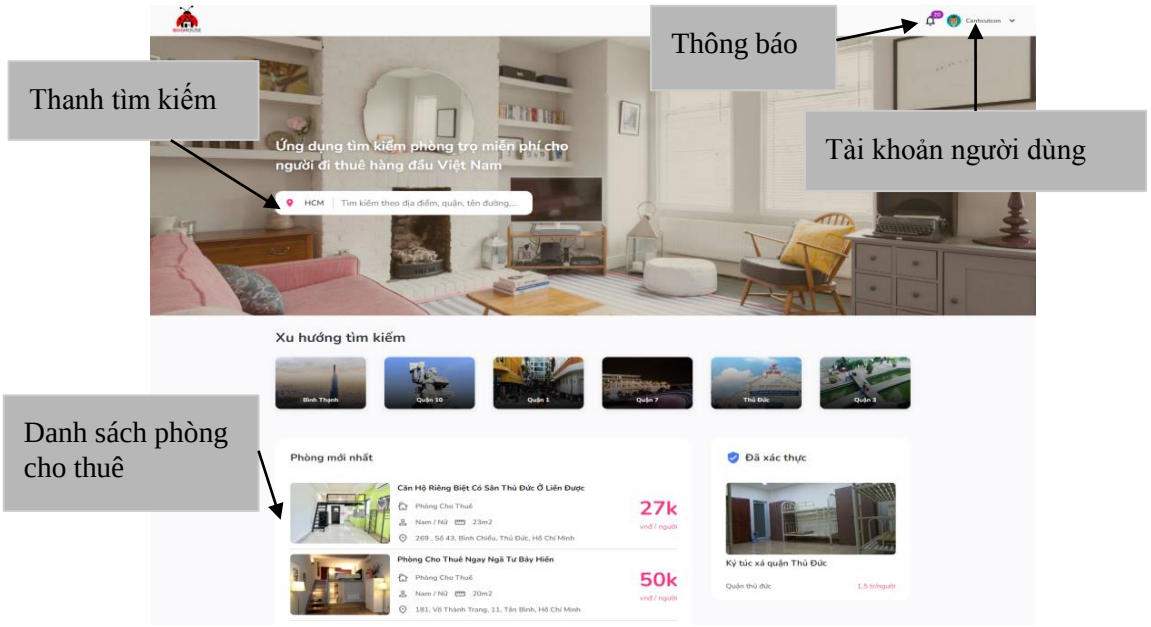



Hình 6: Sơ đồ cấu tạo của Smart Contract Bughouse

## 3.5 Giao diện đồ họa

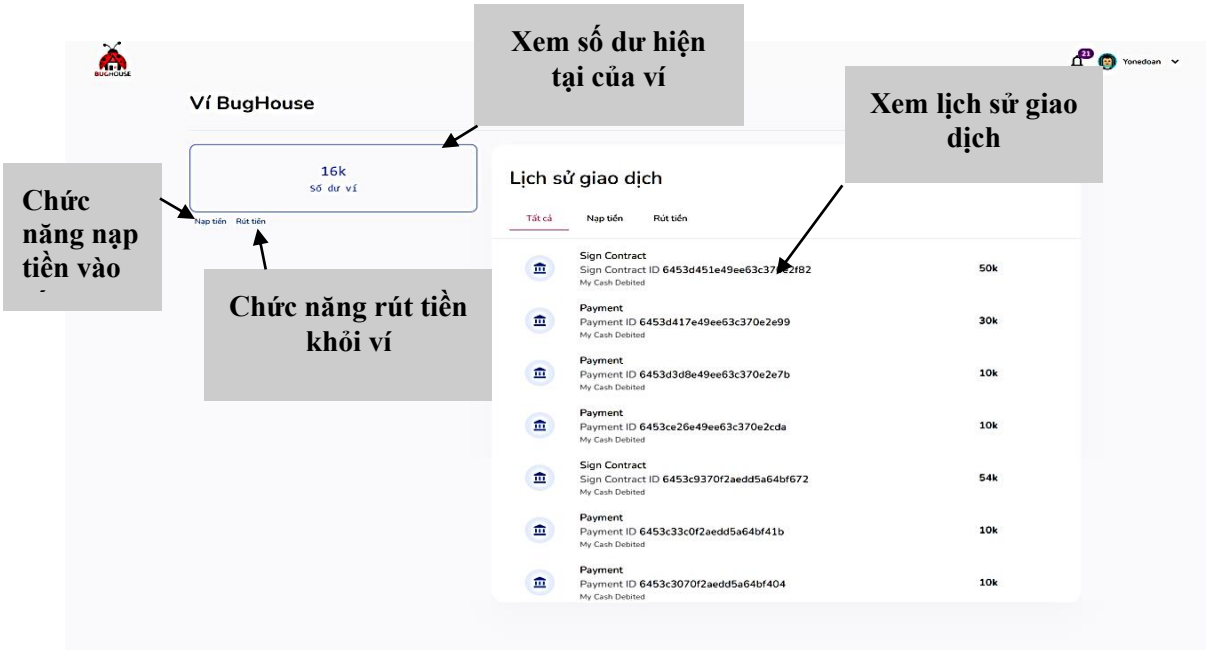
Công cụ Bughouse triển khai trên hai nền tảng là web app và mobile app. Công cụ này sẽ có giao diện người dùng thân thiện và đảm bảo tính ổn định trên các dòng điện thoại khác nhau. Chúng tôi cũng xin giới thiệu vài giao diện cho các chức năng trên ứng dụng trên.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh



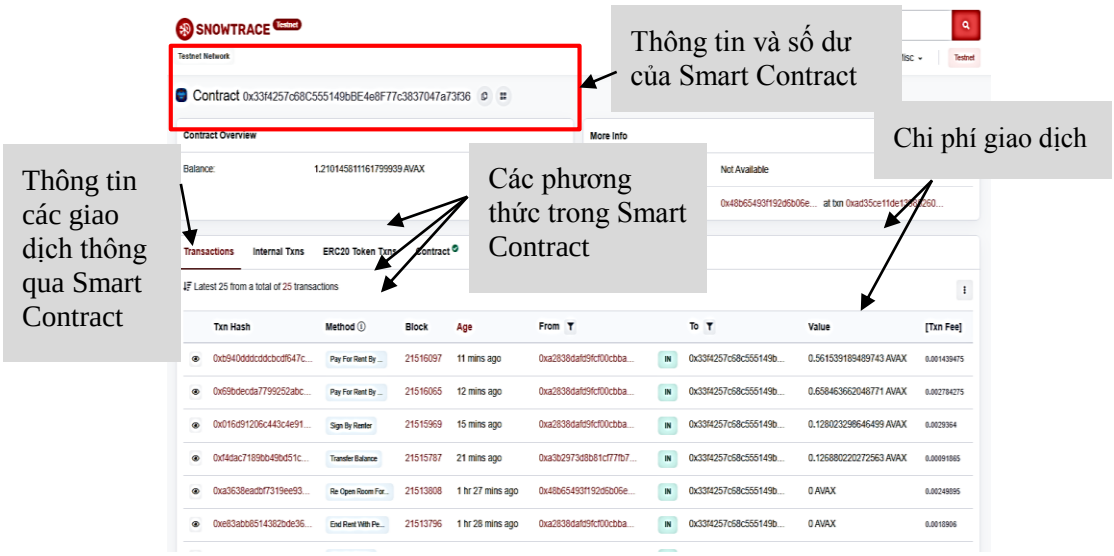

Hình 7: Giao diện trang chủ trên website BUGHOUSE

Giao diện trang chủ của hệ thống có các thành phần như: thanh tìm kiếm, danh sách phòng cho thuê, thông báo, tài khoản người dùng. Với chức năng tìm kiếm người dùng sẽ nhập địa chỉ muốn thuê trọ vào ô tìm kiếm, bấm nút Lọc để thêm các tiêu chí tìm kiếm như giá phòng, diện tích phòng, khoảng cách đến căn hộ. Sau đó người dùng chọn chức năng tìm kiếm. Kết quả tìm kiếm phù hợp sẽ được hiển thị trực tiếp lên trên bản đồ hoặc hiển thị dưới dạng danh sách. Chức năng thông báo khi nhấn vào sẽ mở ra cửa sổ hiển thị toàn bộ thông báo của hệ thống đến với người dùng (*Hình 6*). Người dùng cũng có thể kiểm tra tài khoản giao dịch của mình thông qua ví điện tử mà hệ thống cung cấp (*Hình 7*).



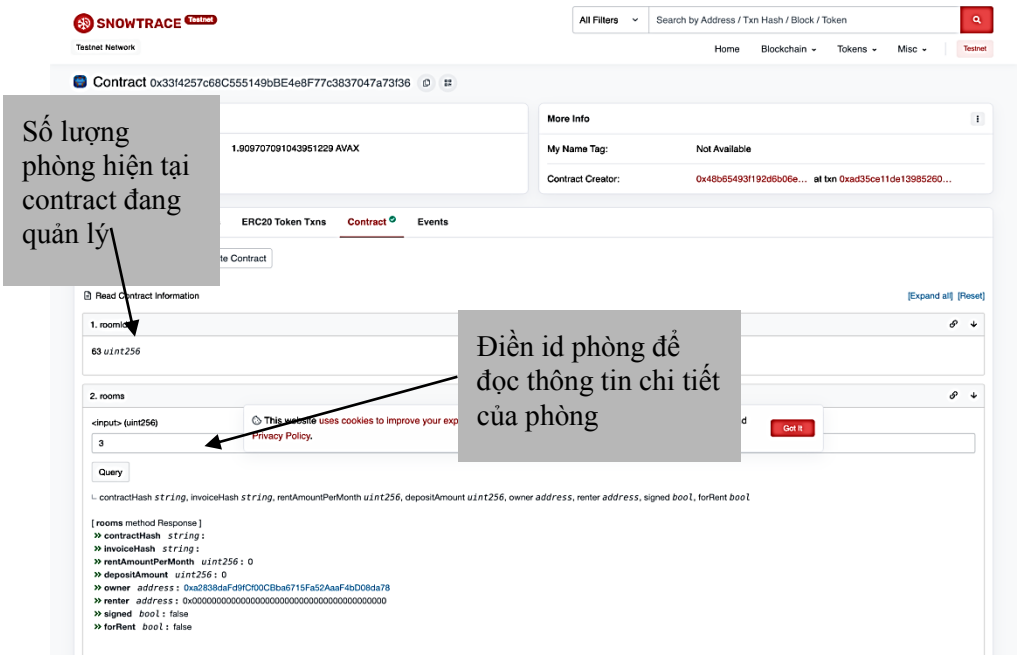

Hình 8: Giao diện thông tin ví điện tử trong BugHouse

Giao diện này cho phép xem số lượng phòng hiện tại Smart Contract đang quản lý và cho phép nhập id phòng để xem thông tin trạng thái phòng hiện tại (*Hình 8*).




Hình 9: Giao diện quản lý Smart Contract

Smart Contract sau khi được triển khai thành công với Fuji Testnet sẽ được quản lý trên Snow Trace Testnet. Snow Trace Testnet là một mạng thử nghiệm được xây dựng dựa trên nền tảng Blockchain Avalanche. Nó được tạo ra để giúp các nhà phát triển và nhà nghiên cứu có thể kiểm tra các ứng dụng Blockchain của mình mà không cần sử dụng tiền thật (*Hình 9*).

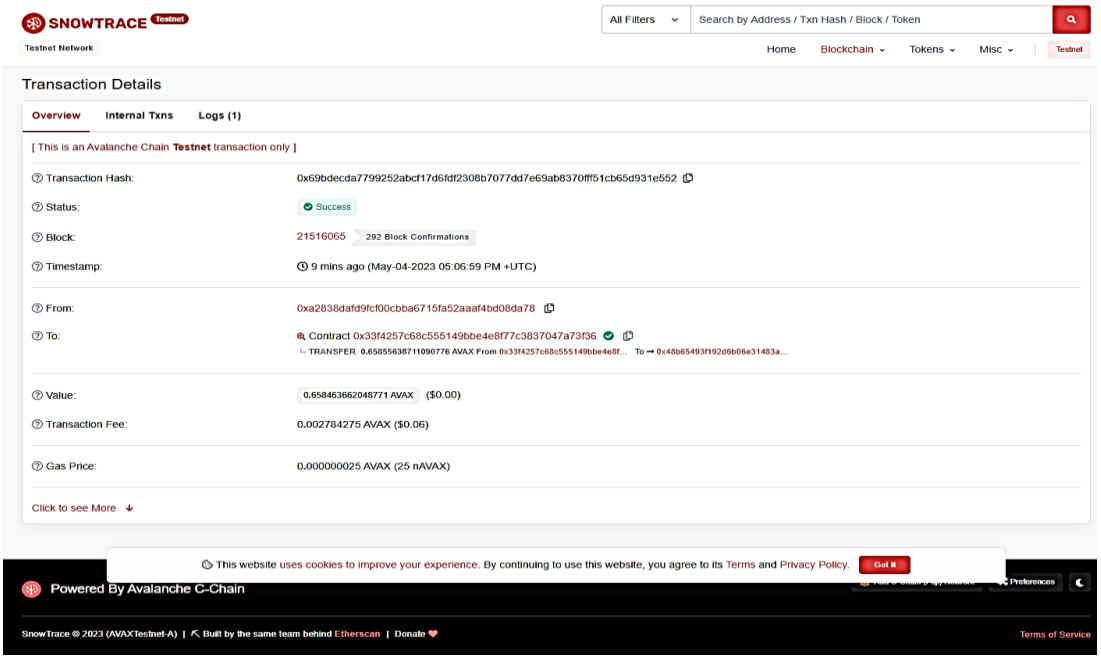



Hình 10: Giao diện tra cứu giao dịch trên Blockchain

Snow Trace Testnet cho phép các nhà phát triển triển khai các ứng dụng, hợp đồng thông minh (Smart Contract) và các dịch vụ Blockchain của mình trên một mạng lưới thử nghiệm có tính năng đầy đủ, tốc độ


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

cao và chi phí thấp. Hệ thống sử dụng Snow Trace Testnet để theo dõi quản lý, truy vết các giao dịch diễn ra trên Smart Contract (*Hình 10*).




Hình 11: Giao diện tổng quan giao dịch trên Blockchain

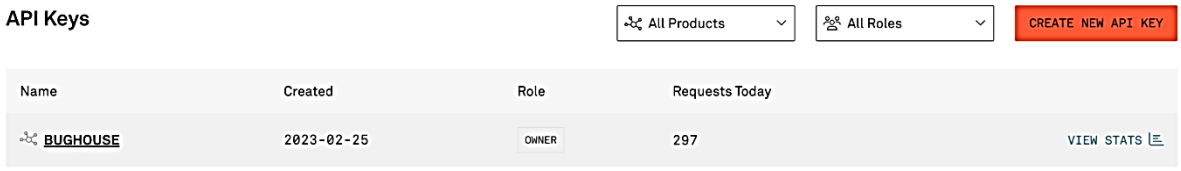
## 4 KIỂM NGHIỆM THỰC TẾ

Chúng tôi sử dụng dịch vụ Web Services của Render để triển khai hệ thống, dịch vụ này cho phép triển khai hệ thống một cách đơn giản và tự động hoá. Bên cạnh đó Render cũng hỗ trợ quản lý tài nguyên và mở rộng ứng dụng một cách linh hoạt như *Hình 12*. Trong hình này, chúng tôi đã thử nghiệm triển khai hệ thống lên Render, đây là màn hình log của server.

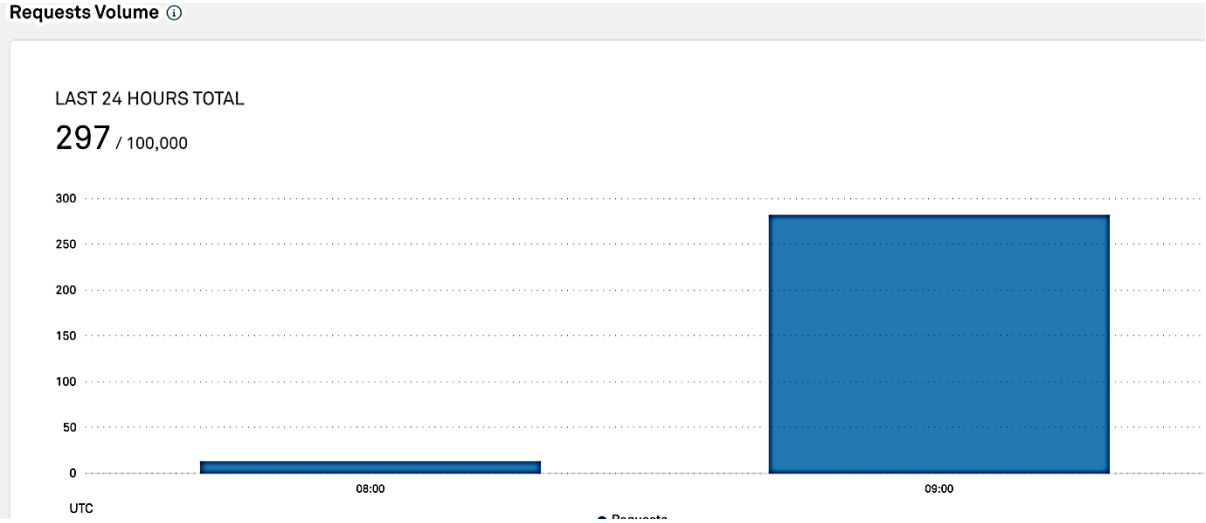



Hình 12: Màn hình log của hệ thống Bughouse trên Web Services của Render

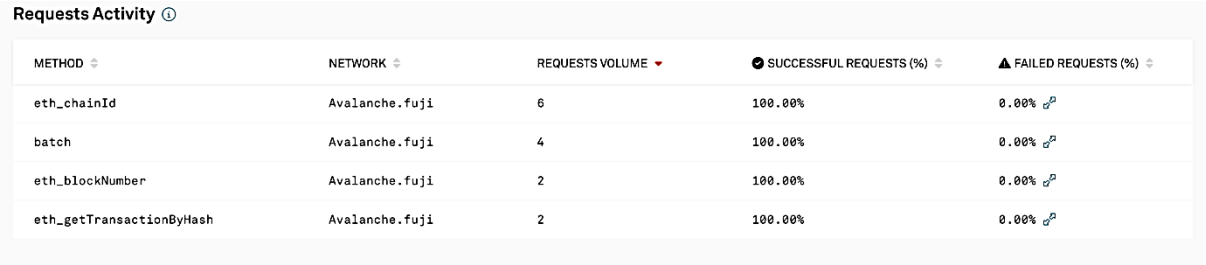
Bughouse sử dụng dịch vụ trợ Web3 của Infura làm trung gian để gửi các yêu cầu đến các mạng trên Blockchain. Infura hỗ trợ Bughouse dễ dàng tương tác với blockchain Ethereum và các blockchain khác một cách thuận tiện và đáng tin cậy (*Hình 13*, *Hình 14*, *Hình 15*).




Hình 13: Màn hình API key Infura của Bughouse




Hình 14: Màn hình hiển thị hiệu suất và dữ liệu sử dụng API




Hình 15: Màn hình hiển thị tổng khối lượng yêu cầu, yêu cầu thành công và thất bại, trên mỗi mạng và lựa chọn phương thức

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài báo này, chúng tôi đã trình bày hai giải pháp cho việc xây dựng ứng dụng Bughouse, đó là ứng dụng di động và ứng dụng web. Nhằm giải quyết bài toán xây dựng hệ thống tìm kiếm và quản lý thuê phòng trên nền tảng Blockchain. Cả website lẫn ứng dụng đều đã có có đầy đủ các chức năng cơ bản, trở thành một hệ thống đáng tin cậy và linh hoạt để người thuê tìm kiếm và thuê trọ một cách thuận tiện, đồng thời hỗ trợ cho người dùng quản lý phòng trọ và tự động tính toán tiền thuê. Bên cạnh đó hệ thống là bên thứ ba để giám sát các giao dịch bên thuê và bên cho thuê. Các giao dịch sẽ đảm bảo tính an toàn khi hệ thống cung cấp ví điện tử cho các giao dịch này. Hệ thống sử dụng các công nghệ mới như: AWS, Render, Remix IDE, NodeJS, ReactJS, React Native, Web3 … Bên cạnh đó hệ thống sử dụng công nghệ Blockchain cung cấp tính bảo mật và đáng tin cậy cho các giao dịch giữa người thuê và người cho thuê. Triển khai Smart Contract thành công nhằm tự động thực thi các chức năng liên quan đến việc cho thuê phòng như đặt phòng, mở rộng thời gian thuê, bắt đầu và kết thúc thuê, thanh toán tiền thuê và phạt phí khi không tuân thủ thỏa thuận. Hiện hệ thống đã triển khai trên mô hình thực tại Tp.HCM. Trong tương lai, chúng tôi sẽ mở rộng quy mô của vùng thuê sang các tỉnh thành hoặc đất nước khác. Mở rộng thêm các loại hình cho

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

thuê khác như: ở ghép, thuê nhà, …phát triển các chức năng tùy biến xử lý các giao dịch thỏa thuận, nhằm đem đến cho người dùng nhiều lựa chọn hơn. Cập nhập thêm tính năng gợi ý phòng thông minh theo nhu cầu của người dùng. Tối ưu các tính năng có hiệu năng thấp, để có thể tăng trải nghiệm của người dùng lên mức cao nhất.

## TÀI LIỆU THAM KHẢO

[1]. Alharby, M., Van Moorsel, A.: Blockchain-based smart contracts: A systematic mapping study. arXiv preprint arXiv:1710.06372 (2017)

[2]. Karamitsos, I., Papadaki, M., & Barghuthi, N. B. A. (2018, 7). Design of the Blockchain Smart Contract: A Use Case for Real Estate. *Scientific Research Publishing*. https://www.scirp.org/journal/paperinformation.aspx?paperid=85741

[3]. Neeraj26pathak. (2021, October 19). *Node.js Web Application Architecture*. GeeksforGeeks. Retrieved May 29, 2023, from https://www.geeksforgeeks.org/node-js-web-application-architecture/

[4]. Geeksforgeeks. (2023, March 10). *ReactJS Tutorials*. GeeksforGeeks. Retrieved May 29, 2023, from https://www.geeksforgeeks.org/reactjs-tutorials/

[5]. Paul, A., & Nalwaya, A. (2019). *React Native for Mobile Development: Harness the Power of React Native to Create Stunning IOS and Android Applications*. Apress.

[6]. The Solidity Authors. (2023, January 5). Solidity. Retrieved May 30, 2023, from https://docs.soliditylang.org/en/v0.8.19/.

[7]. Thạc Sĩ Nguyễn Nhi Quang (2021), "Ứng dụng Blockchain trong giao dịch L/C tại các ngân hàng thương mại Việt Nam", Tạp chí công nghệ và ngân hàng số, ngày 11/08/2021

[8]. Ongo, G., Kusuma, G.P. Hybrid database system of mysql and mongodb in web application development. In Proceeding of the 2018 International Conference on Information Management and Technology (ICIMTech), IEEE (2018) 256–260

[9]. Amazon Web Service – CloudFront (2023), [online], Content Delivery Network, https://aws.amazon.com/cloudfront/, Accessed 20/05/2023

[10]. Stepik. (2019, August 7). *Step 1 – Smart Assets – Stepik*. Stepik. Retrieved July 27, 2023, from https://stepik.org/lesson/237609/step/1

[11] Loannis Karamitsos, Maria Papadaki, Nedaa Baker Al Barghuthi. 2018. "Design of the Blockchain Smart Contract: A Use Case for Real Estate." Scientific Research Publishing. https://www.scirp.org/journal/paperinformation.aspx?paperid=85741.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F020**

# HEALTH MONITORING - HỆ THỐNG CẢNH BÁO CHĂM SÓC SỨC KHỎE DỰA TRÊN NỀN TẢNG IoT

TÔN LONG PHƯỚC[1*], MAI VĂN TRƯỜNG[1], NGUYỄN HOÀNG DUY[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*[*]tonlongphuoc@iuh.edu.vn, mvantruong.dev@gmail.com, nguyenhoangduy.dev@gmail.com*

**Tóm tắt.** Đại dịch Covid-19 đã có ảnh hưởng lớn đến ngành y tế trên toàn thế giới, đặc biệt là ở Việt Nam. Nhân viên y tế và bác sĩ phải chăm sóc một lượng bệnh nhân tăng đột biến, gây ra nhiều rủi ro khiến các bệnh nhân không được giám sát, chăm sóc và cảnh báo kịp thời. Nhận thấy vấn đề này, chúng tôi đã phát triển công cụ giám sát bệnh nhân, gọi là Health Monitoring, dựa trên Internet of Things (IoT) với hai phân hệ Web ứng dụng và ứng dụng trên di động, kết hợp trí tuệ nhân tạo để hỗ trợ bác sĩ theo dõi tình trạng bệnh nhân từ xa và cảnh báo khi phát hiện bất thường trên cơ thể bệnh nhân trong thời gian thực. Công cụ này bao gồm thiết bị đeo tay IoT sử dụng board mạch Arduino và ESP8266 để xử lý dữ liệu từ cảm biến và giám sát bệnh nhân trong thời gian thực. Hệ thống Web sử dụng ReactJS để xử lý phía frontend và Java (Spring Boot) để xử lý phía backend, trong khi ứng dụng di động được viết bằng React Native. Cả các phân hệ Web, app và thiết bị IoT đều được kết nối với nhau thông qua websocket để truyền dữ liệu giữa các phân hệ và cung cấp cảnh báo kịp thời. Hiện nay, chúng tôi đã thử nghiệm công cụ này với các sinh viên tại trường Đại học Công nghiệp TP.HCM và đã thu được kết quả tích cực, bao gồm việc thu thập và cảnh báo theo thời gian thực cho bác sĩ từ bệnh nhân trên các phân hệ của hệ thống. Trong tương lai, chúng tôi sẽ phát triển công cụ này cho nhiều nhóm bệnh khác nhau và nâng cao độ chính xác cho việc chẩn đoán và cảnh báo bằng cách thu thập nhiều thông số hơn từ thiết bị đeo tay.

**Từ khóa.** IoT, Giám sát bệnh nhân, Phát hiện thời gian thực, Cảm biến, giám sát từ xa, ReactJS, React Native, Spring Boot, Arduino, ESP8266, Websocket.

## HEALTH MONITORING - IoT-BASED HEALTH CARE SYSTEM

**Abstract.** The Covid-19 pandemic has significantly impacted the healthcare industry worldwide, especially in Vietnam. Healthcare workers and doctors have had to care for a sudden surge of patients, resulting in many risks and patients not being monitored, cared for, or alerted on time. To address these issues, we have developed a patient monitoring tool called Health Monitoring, based on the Internet of Things (IoT), with two subsystems: a web application and a mobile application, combined with artificial intelligence to assist doctors in remotely monitoring and detecting abnormalities in real-time. The tool includes an IoT wristband device using Arduino and ESP8266 to process data from sensors and monitor patients in real time. The web system uses ReactJS to process the frontend and Java (Spring Boot) to process the backend, while the mobile application is written in React Native. All web, app, and IoT subsystems are connected via WebSockets to transmit data between subsystems and provide timely alerts. We have tested this tool with students at the Industrial University of Ho Chi Minh City and obtained favorable results, including real-time data collection and alerts for doctors from patients across subsystems. In the future, we will develop this tool for various patient groups and improve diagnostic accuracy and alerting by collecting more data from the wristband device.

**Keywords.** IoT, patient monitoring, real-time detection, sensor, remote monitoring, ReactJS, React Native, Spring Boot, Arduino, ESP8266, websocket.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

# 1 GIỚI THIỆU TỔNG QUAN

Trong thời đại công nghệ số phát triển nhanh chóng như hiện nay, Internet of Things (IoT) đã trở thành một trong những xu hướng công nghệ đang được quan tâm và đầu tư rất nhiều. IoT là một hệ thống được thiết kế để kết nối và quản lý các thiết bị thông minh, cho phép chúng ta có thể điều khiển và giám sát các thiết bị từ xa thông qua internet. Sự phát triển của IoT đã mở ra nhiều tiềm năng ứng dụng trong nhiều lĩnh vực khác nhau, trong đó có lĩnh vực chăm sóc sức khỏe. Các thiết bị y tế thông minh giúp bệnh nhân có thể tự theo dõi tình trạng sức khỏe của mình và cung cấp dữ liệu cho bác sĩ để đưa ra chẩn đoán và điều trị tốt hơn.

Hiện nay, chăm sóc sức khỏe đang là một trong những vấn đề được quan tâm hàng đầu trong xã hội. Tuy nhiên, việc quản lý và giám sát sức khỏe của mỗi người vẫn còn gặp nhiều khó khăn. Trong năm 2019 thế giới phải đối mặt với đại dịch Covid – 19 do vi-rút có tên SARS-CoV-2 gây ra, căn bệnh này ảnh hưởng rất lớn đến nhiều lĩnh vực trong đời sống và xã hội, gây một sức ép khủng khiếp lên hệ thống y tế toàn cầu, trong thời điểm đó các bệnh viện gần như quá tải, bác sĩ và các nhân viên y tế gần như kiệt quệ, với số lượng bệnh nhân gia tăng theo cấp số nhân. Hằng ngày các y, bác sĩ không thể túc trực bên cạnh bệnh nhân theo dõi tình trạng của họ cũng như phát hiện bất thường từ cơ thể họ một cách nhanh chóng và kịp thời. Hơn nữa, những người cao tuổi hoặc những người sống một mình cũng cần có một hệ thống cảnh báo chăm sóc sức khỏe để đảm bảo an toàn cho sức khỏe của họ.

Với sự phát triển của IoT, việc xây dựng hệ thống cảnh báo chăm sóc sức khỏe dựa trên nền tảng IoT trở nên khả thi hơn bao giờ hết và nhận thấy những bất cập trên với mục đích giảm thiểu áp lực của các bác sĩ và nhân viên y tế cũng như tự động hóa quy trình theo dõi bệnh nhân, nghiên cứu này được sinh ra với mục tiêu qua việc theo dõi sức khỏe từ xa với thiết bị đeo tay y tế được cung cấp, hệ thống này sẽ thu thập các chỉ số sức khỏe của bệnh nhân chuyển những thông số đó đến với Website và Application của bác sĩ và chuyên gia trong thời gian thực. Nếu nhận thấy những bất thường của bệnh nhân từ thiết bị IoT hệ thống sẽ thực hiện cảnh báo đến Application và đèn của thiết bị IoT. Hệ thống dựa vào các tập luật y tế được ban hành, sử dụng dữ liệu để theo dõi và phân tích tự động, nhờ đó các bác sĩ có thể theo dõi tình trạng của bệnh nhân từ xa ngay lập tức để đưa ra quyết định cần thiết và kịp thời cũng như là nhận được những thông báo từ hệ thống để xử lý tình huống y tế khẩn cấp.

Mục tiêu nghiên cứu nhằm mục đích xây dựng hệ thống cảnh báo thông minh từ xa kết hợp với thiết bị IoT tập trung vào những bệnh nhân mắc bệnh Covid – 19 là chính. Hệ thống sẽ được xây dựng cả hai nền tảng Website và Application kết hợp với thiết bị IoT, trên nền tảng Website được xây dựng theo dạng CMS (Content Management System) hay còn được gọi là hệ thống quản trị nội dung giúp cho bác sĩ hoặc chuyên gia có thể cập nhật thay đổi dữ liệu trên Website cũng như lấy được dữ liệu từ thiết bị IoT trong thời gian thực và hiển thị lên Website. Đối với nền tảng Application được thiết kế dành cho bác sĩ, khi nhận được tình trạng bất thường của bệnh nhân từ thiết bị IoT, app có nhiệm vụ thông báo đến cho bác sĩ tình trạng bất thường của bệnh nhân, bác sĩ sẽ có thể kịp thời quan sát tình trạng của bệnh nhân. Việc kết hợp hai nền tảng Website và Application sẽ giúp người dùng có thể quản lý và theo dõi sức khỏe của mình một cách toàn diện trên nhiều nền tảng khác nhau. Cả hai nền tảng đều có thể được tích hợp với các thiết bị IoT để thu thập và quản lý dữ liệu sức khỏe của người dùng. Việc cảnh báo tình trạng sức khỏe nguy hiểm cũng được thực hiện trên cả hai nền tảng để đảm bảo người dùng có thể can thiệp kịp thời và giảm thiểu rủi ro cho sức khỏe của họ.

Bài báo của chúng tôi chia làm 5 phần: Phần 1 Giới thiệu tổng quan về hệ thống; Phần 2 Các công nghệ liên quan; Phần 3 Công cụ Health Monitoring; Phần 4 Kiểm nghiệm thực tế và Phần 5 Kết luận và hướng phát triển. Trong đó Phần 3 là phần đóng góp chính của bài báo.

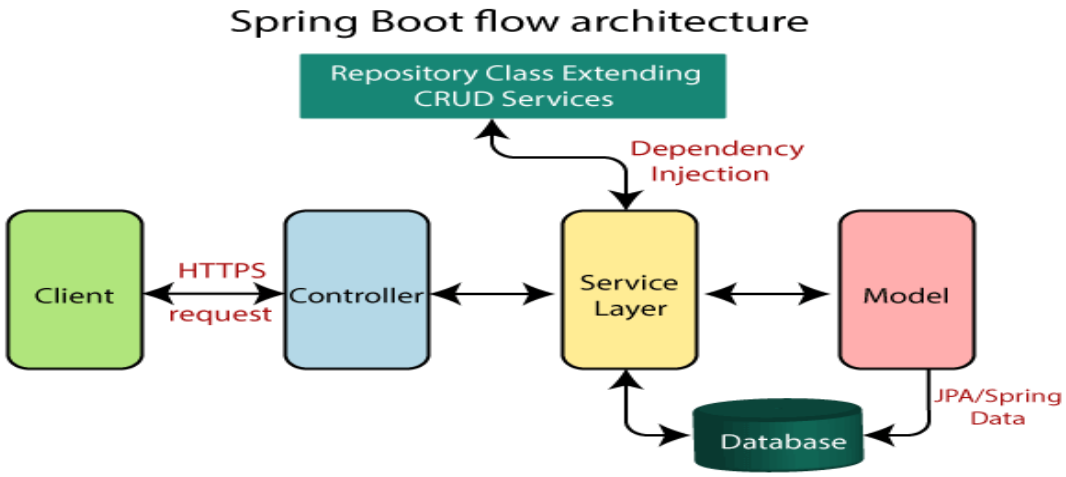
# 2 CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 Những framework hỗ trợ phát triển ứng dụng

### a) Spring Boot

Spring Boot là một extension của Spring Framework dùng để phát triển ứng dụng Java, được sử dụng để tạo ra các ứng dụng web, mobile và desktop. Spring Boot giúp cho việc phát triển ứng dụng trở nên nhanh chóng và dễ dàng hơn bằng cách cung cấp nhiều tính năng mặc định và giảm thiểu sự cấu hình. Kiến trúc Spring Boot bao gồm các thành phần cơ bản như: Repository Class Extending, Service Layer, Model, Controller (*Hình 1*). Repository Class Extending: hỗ trợ quá trình xây dựng hệ thống một cách nhanh chóng

hoặc cũng có thể là các lớp chứa các phương thức xử lý, truy xuất dữ liệu (CRUD) do lập trình viên viết ra. Service Layer là lớp dịch vụ chứa các phương thức (hàm) xử lý các yêu cầu như chỉnh sửa, lớp Service thường sử dụng các phương thức của các lớp trong Repository Class Extending (bằng cách gọi tới một hay kết hợp nhiều phương thức với nhau) để truy xuất, thêm, xóa dữ liệu trong cơ sở dữ liệu. Model là thành phần được đính kèm theo các truy vấn của Services Layer xuống cơ sở dữ liệu. Model đóng nhiều vai trò quan trọng như phát sinh hệ cơ sở dữ liệu theo liên kết các lớp của lập trình viên, chuyển đổi kết quả truy xuất từ cơ sở dữ liệu thành các thực thể lớp. Controller là thành phần giao tiếp trực tiếp với phía client, tồn tại dưới dạng các API. Model có trách nhiệm xử lý yêu cầu của client, xác minh các yêu cầu đó. Model truy cập đến phương thức phù hợp trong Service Layer để đạt được kết quả mong muốn, trả kết quả về cho phía client. [1, 2]. Springboot có một số ưu điểm sau: Giảm thiểu thời gian phát triển; tăng tính ổn định; tính dễ quản lý: Spring Boot giúp cho việc quản lý ứng dụng trở nên dễ dàng hơn bằng cách cung cấp các công cụ quản lý và khả năng giám sát. Ngoài ra Springboot cũng có những mặt hạn chế: Cần kiến thức về Spring Framework; tính phức tạp trong một số trường hợp: Trong một số trường hợp, Spring Boot có thể trở nên phức tạp và khó hiểu đối với người mới bắt đầu hoặc với các ứng dụng có tính độc lập cao.




Hình 1: Kiến trúc Spring Boot [2]

**b) ReactJS – React Native**

ReactJS và React Native đều là các công nghệ phát triển ứng dụng web được phát triển bởi Facebook. Tuy nhiên, chúng có mục đích và ứng dụng khác nhau. ReactJS là một thư viện JavaScript được sử dụng để xây dựng các ứng dụng web động. ReactJS cho phép các nhà phát triển tạo ra các giao diện người dùng (UI) động và tương tác với các thành phần khác nhau của ứng dụng web một cách dễ dàng. ReactJS sử dụng một phương pháp lập trình gọi là "reactive programming" để tạo ra các ứng dụng web động nhanh chóng và dễ bảo trì. React Native là một framework phát triển ứng dụng di động được phát triển dựa trên ReactJS [3]. React Native cho phép các nhà phát triển tạo ra các ứng dụng di động đa nền tảng (cross-platform) bằng cách sử dụng JavaScript. Với React Native, các nhà phát triển có thể sử dụng các thành phần của ReactJS để xây dựng các giao diện người dùng động cho ứng dụng di động của mình. React Native cũng cung cấp các thành phần và tính năng để tương tác với các thành phần khác nhau của hệ thống di động như camera, GPS, và các cảm biến khác [4].

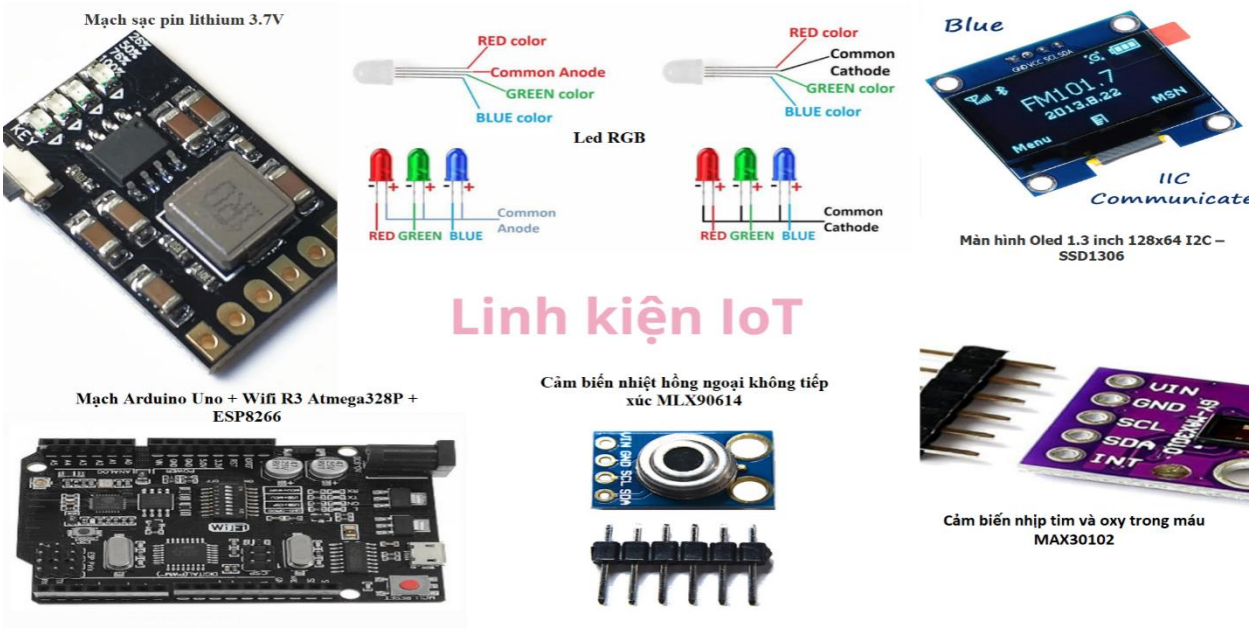
**c) Firebase**

Firebase là một nền tảng dịch vụ đám mây do Google cung cấp, được sử dụng để phát triển ứng dụng web và di động. Firebase cung cấp nhiều dịch vụ khác nhau để giúp các nhà phát triển xây dựng các ứng dụng web và di động nhanh chóng và dễ dàng. Một số công dụng của firebase được sử dụng trong Health Monitoring như: xác thực người dùng cho phép xác thực và quản lý người dùng của ứng dụng một cách dễ dàng, thông báo đẩy cho phép gửi thông báo đẩy đến người dùng của ứng dụng một cách dễ dàng, cloud functions cho phép xây dựng các hàm chạy trên đám mây để xử lý các sự kiện trong ứng dụng [5].


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**d) Thiết bị IoT**

Chúng tôi tạo thiết bị đeo tay cho bệnh nhân bằng những thiết bị sau *Hình 2*: Mạch Arduino Uno + Wifi R3 Atmega328P + ESP8266, mạch được tích hợp module Wifi ESP8266 giúp kết nối mạng không dây và truyền dữ liệu qua mạng Internet. Cảm biến hồng ngoại không tiếp xúc MLX90614, sử dụng công nghệ hồng ngoại đo nhiệt độ của các vật thể mà không cần tiếp xúc trực tiếp. Cảm biến nhịp tim và oxy trong máu MAX30102, cảm biến đo nhịp tim và nồng độ oxy trong máu, sử dụng công nghệ phát quang để đo lường. Led RGB loại đèn led đa sắc có khả năng phát ra ánh sáng với ba màu cơ bản là đỏ (R), xanh lá (G), và xanh dương (B). Màn hình Oled 1.3inch 128x64 I2C – SSD1306 một loại màn hình hiển thị OLED nhỏ gọn có độ phân giải 128x64 pixel và được kết nối thông qua giao thức I2C. Mạch sạc pin lithium 3.7V là một loại mạch điện được sử dụng để sạc các loại pin lithium-ion có điện áp 3.7V [6].




Hình 2: Linh kiện tạo thiết bị IoT [6]

**e) Jenkins**

Jenkins là một công cụ tích hợp liên tục (CI - Continuous Integration) và liên tục triển khai (CD - Continuous Deployment) mã nguồn mở. Jenkins được viết bằng Java và có thể chạy trên nhiều nền tảng khác nhau, bao gồm cả Windows, MacOS và Linux. Jenkins cho phép các nhà phát triển kiểm tra và xây dựng mã nguồn của ứng dụng một cách tự động. Nó tự động lấy mã nguồn từ kho lưu trữ mã nguồn (ví dụ như Git), xây dựng ứng dụng, kiểm tra và thông báo kết quả đến các nhà phát triển. Việc sử dụng Jenkins giúp tăng tốc độ phát triển, giảm thiểu lỗi và đảm bảo tính ổn định của ứng dụng đạt được tính linh hoạt cho các dự án phát triển phần mềm [7].

**f) Cloud Computing với AWS**

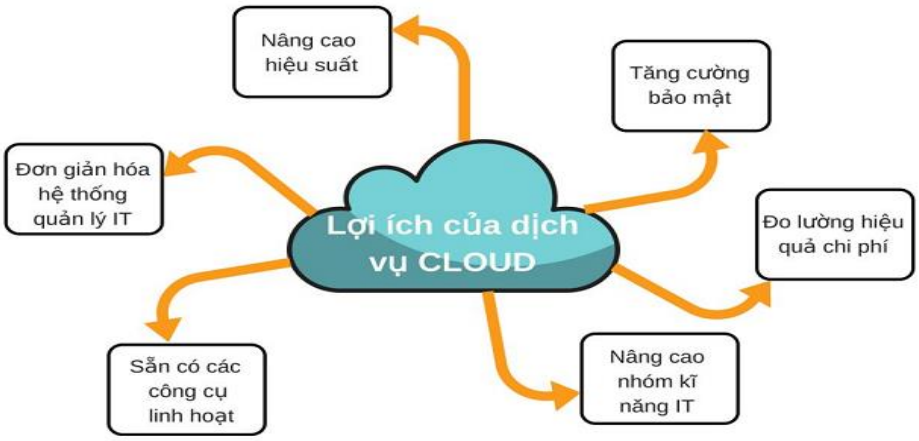
Cloud Computing là một mô hình tính toán dựa trên internet, cho phép người dùng truy cập vào các tài nguyên tính toán, lưu trữ và phần mềm thông qua một mạng lưới máy chủ từ xa. Nó có thể giúp cho doanh nghiệp tiết kiệm chi phí và tăng tính linh hoạt trong việc quản lý tài nguyên. Với AWS, người dùng có thể dễ dàng tạo ra các máy chủ ảo, lưu trữ dữ liệu, chạy ứng dụng và triển khai các dịch vụ trực tuyến. AWS cũng cung cấp các công cụ quản lý tài nguyên như Amazon CloudWatch và AWS CloudFormation, giúp người dùng quản lý các tài nguyên của họ trong một môi trường đám mây [8].

EC2 (Elastic Compute Cloud) là một dịch vụ tính toán đám mây của AWS, cung cấp cho người dùng khả năng tạo và quản lý các máy chủ ảo (instances) trên nền tảng đám mây của AWS. Với EC2, người dùng có thể tạo ra các máy chủ ảo với các tài nguyên như CPU, RAM, ổ cứng, mạng và băng thông mạng tùy chỉnh. Với tính năng tự động mở rộng, EC2 có thể tăng hoặc giảm số lượng các máy chủ ảo tự động dựa trên tài nguyên yêu cầu của ứng dụng [9].

S3 (Simple Storage Service) là một dịch vụ lưu trữ đám mây của AWS, cung cấp cho người dùng khả năng lưu trữ và quản lý dữ liệu trong một môi trường đám mây linh hoạt, bảo mật và tiết kiệm chi phí. Với S3,

người dùng có thể lưu trữ và truy cập dữ liệu từ bất kỳ đâu trên thế giới thông qua internet. Dữ liệu được lưu trữ trên S3 là phi cấu trúc, có thể là các tệp tin, hình ảnh, video hay dữ liệu của ứng dụng. S3 cung cấp khả năng lưu trữ không giới hạn, nghĩa là người dùng có thể lưu trữ bất kỳ lượng dữ liệu nào mà họ cần [10].

CloudFront là một dịch vụ CDN (Content Delivery Network) của AWS, cung cấp khả năng tăng tốc độ truy cập và phân phối nội dung trên toàn cầu. Với CloudFront, người dùng có thể tăng tốc độ tải các tài nguyên của họ, bao gồm các trang web, ứng dụng web, nội dung video và hình ảnh, bằng cách đưa nội dung gần với người dùng. CloudFront sử dụng một mạng lưới các máy chủ đám mây trên toàn cầu để lưu trữ và phân phối nội dung tới người dùng [11].



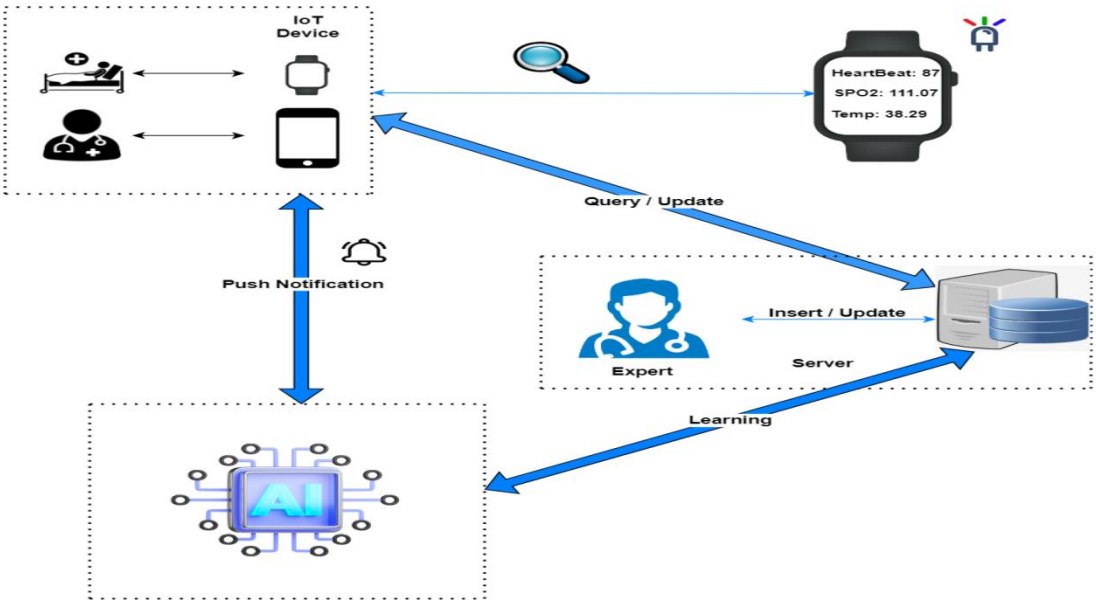

Hình 3: Lợi ích dịch vụ Cloud Computings [8]

## 3 HEALTH MONITORING: CÔNG CỤ CẢNH BÁO CHĂM SÓC SỨC KHỎE DỰA TRÊN NỀN TẢNG IoT

### 3.1 Giới thiệu

Như bài toán đã được trình bày trong phần giới thiệu, chúng tôi đã xây dựng công cụ Health Monitoring với sự kết hợp của nhiều công nghệ tiên tiến. Phần Backend được xây dựng trên nền tảng Spring Boot, đảm nhận vai trò tạo ra các Restful-API và xử lý các request từ phía client, đồng thời trả về các response để phục vụ cho client. Dữ liệu được lưu trữ trong MongoDB giúp tăng tốc độ lấy dữ liệu, phù hợp cho dự án có tính chất real-time. Phần Frontend được xây dựng trên nền tảng ReactJS, chịu trách nhiệm xây dựng giao diện website và gửi các request xuống server, phục vụ cho việc hiển thị đồ thị đường các chỉ số của bệnh nhân cho bác sĩ hoặc chuyên gia theo dõi trong thời gian thực. ReactJS cũng thực hiện các chức năng quản lý hệ thống như quản lý bệnh nhân, bác sĩ, tập luật, mầm bệnh, và nhiều chức năng khác. Phần ứng dụng di động cho Android/IOS được xây dựng trên nền tảng React Native, phục vụ cho chức năng thông báo bệnh nhân có tình trạng bất thường tới bác sĩ chịu trách nhiệm của bệnh nhân đó và các chức năng liên quan của bác sĩ như quản lý bệnh án, xem bệnh nhân đang điều trị (Hình 4). Để đảm bảo ứng dụng áp dụng được vào thực tế, chúng tôi đã sử dụng một số dịch vụ của AWS như Elastic Compute Cloud (EC2) để tạo máy ảo dùng để cài đặt server, S3 để lưu trữ file tài liệu và hình ảnh cho toàn bộ hệ thống và CloudFront để tăng tốc độ tải nguồn cho hệ thống. Đồng thời, chúng tôi cũng sử dụng Jenkins để hỗ trợ cho CI/CD, giúp những cập nhật của hệ thống được cập nhật nhanh chóng và liên tục. Để đảm bảo tính bảo mật và độ tin cậy cho dữ liệu trong hệ thống, chúng tôi đã sử dụng các giải pháp mã hóa dữ liệu, bao gồm mã hóa dữ liệu trong truyền thông và mã hóa dữ liệu trong lưu trữ. Ngoài ra, chúng tôi cũng đã đảm bảo tính khả dụng và hiệu suất của hệ thống bằng cách triển khai các giải pháp như Load Balancer và Auto Scaling Group, giúp tăng khả năng chịu tải và đảm bảo hệ thống luôn hoạt động ổn định, đáp ứng được số lượng lớn người dùng sử dụng đồng thời.

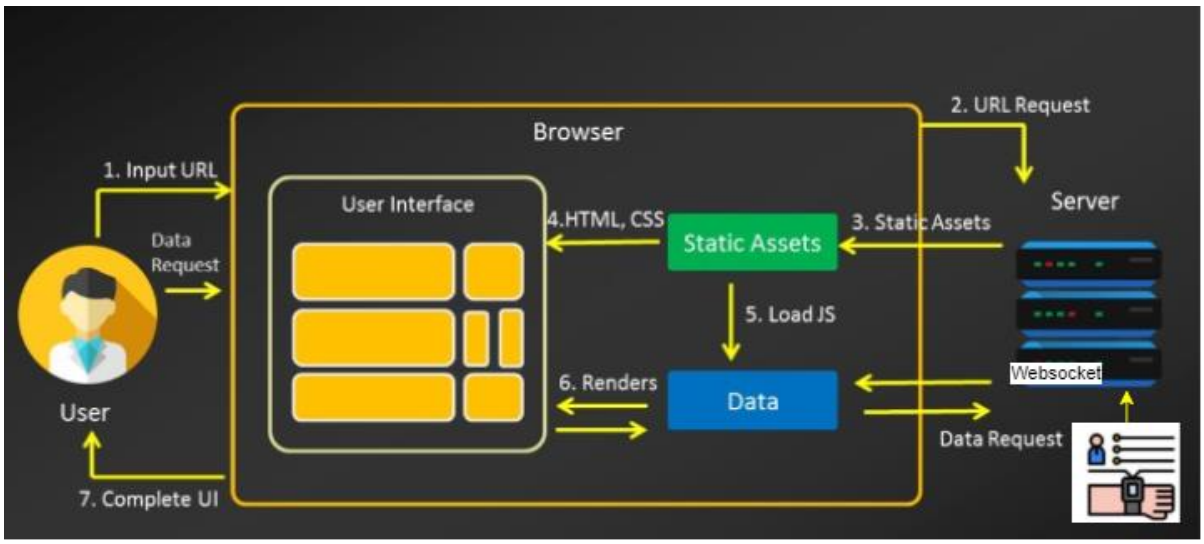
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 4: Sơ đồ hoạt động của hệ thống Health Monitoring

## 3.2 Kiến trúc của hệ thống

Chúng tôi sử dụng kiến trúc Single Page Application (SPA) để xây dựng ứng dụng Health Monitoring. Kiến trúc được sử dụng phổ biến trong các ứng dụng web và app hiện đại. Kiến trúc SPA hoạt động dựa trên việc tải một lần các tài nguyên cần thiết cho ứng dụng, sau đó sử dụng JavaScript để xử lý và hiển thị các nội dung trên cùng một trang web hoặc một màn hình mobile. Cơ chế hoạt động của SPA có thể được mô tả như sau: Khi người dùng truy cập vào ứng dụng, trình duyệt sẽ tải một lần các tài nguyên cần thiết cho ứng dụng, bao gồm các tệp HTML, CSS và JavaScript. Sau khi tải xong, trình duyệt sẽ hiển thị một trang web giao diện đơn giản, thường là một trang chào đón hoặc trang đăng nhập. Khi người dùng thao tác trên trang web, JavaScript sẽ được sử dụng để tải các dữ liệu mới từ máy chủ và hiển thị các phần nội dung mới lên trang web mà không cần tải lại toàn bộ trang. Khi người dùng chuyển đổi giữa các trang, JavaScript sẽ được sử dụng để thay đổi nội dung của trang mà không cần tải lại toàn bộ trang. SPA sử dụng các API của trình duyệt để thay đổi URL của trang mà không cần tải lại toàn bộ trang, giúp cải thiện trải nghiệm người dùng và hỗ trợ việc chia sẻ liên kết (Hình 5).



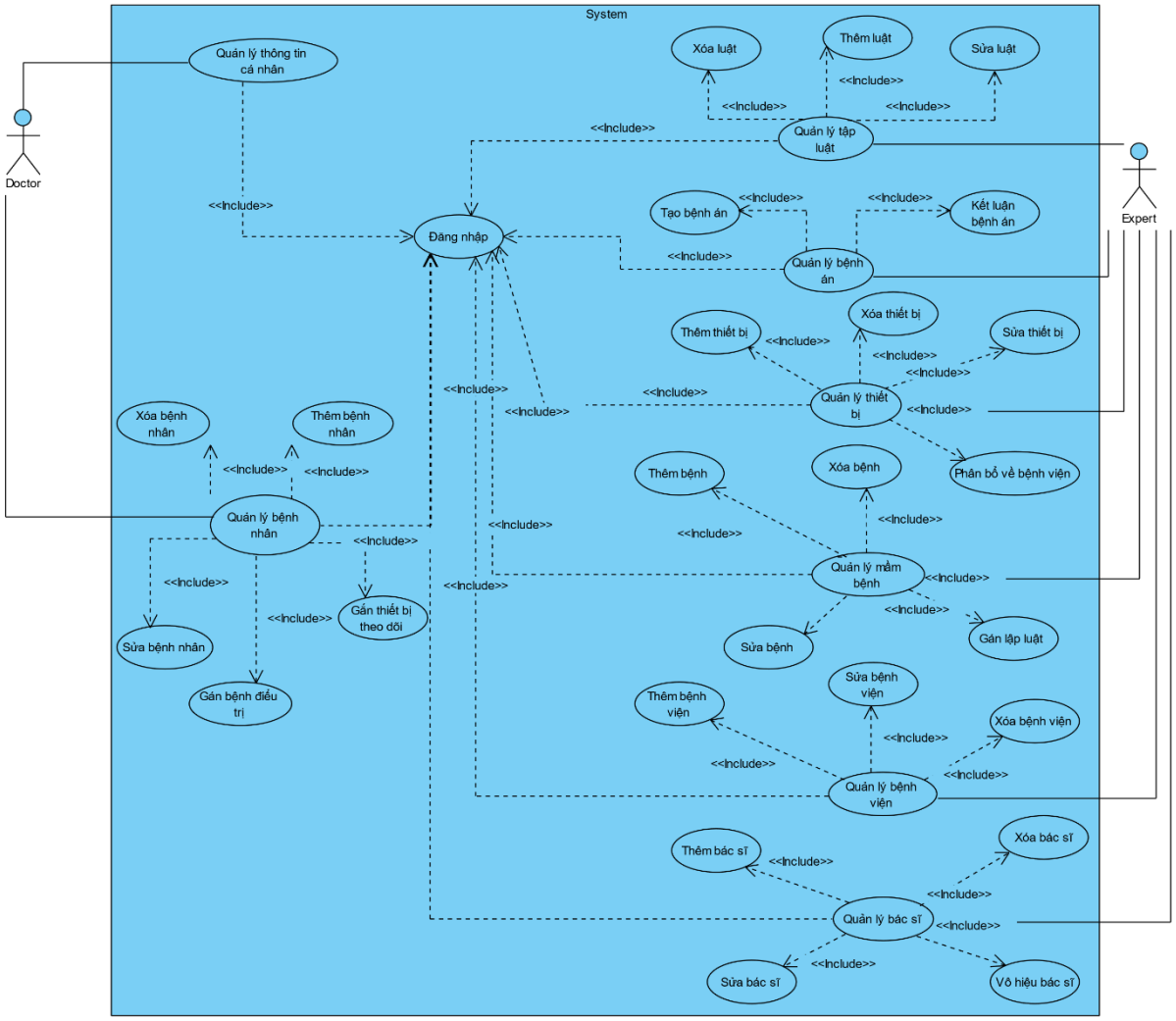

Hình 5: Kiến trúc Single Page Application trong Health Monitoring

### 3.3 Phân tích hệ thống Health Monitoring

Đối tượng người dùng cho hệ thống Health Monitoring gồm 3 tác nhân chính: chuyên gia, bác sĩ, thiết bị IoT. Trong đó, chúng tôi tập trung vào chức năng chính của hệ thống là phát hiện bất thường, xây dựng tập luật hỗ trợ việc cảnh báo, theo dõi tình trạng bệnh nhân cũng như phát hiện bất thường của bệnh nhân trong thời gian thực. Ngoài ra, những chức năng phụ cũng được đưa vào hệ thống để hỗ trợ việc quản lý hệ thống và tăng trải nghiệm người dùng. Từ những chức năng trên, chúng tôi tập trung vào các chức năng chính như: Xem tình trạng bệnh nhân, gửi dữ liệu bệnh nhân lên hệ thống, đồng bộ trạng thái thiết bị, gửi thông báo đến bác sĩ. Sau đó chúng tôi mở rộng các chức năng tương ứng trong Hình 6.

Trong mô hình use-case chúng tôi trình bày, gồm có hai thành phần actor chính mô tả cho hai đối tượng người dùng mà chúng tôi hướng đến (chuyên gia, bác sĩ). Trong đó, những use-case chính và phức tạp về nghiệp vụ đòi hỏi nhiều xử lý cho hệ thống đó là: use-case quản lý thiết bị và use-case quản lí tập luật y tế. Ngoài ra, chúng tôi cũng mở rộng các use-case cho các chức năng của hệ thống nhằm tạo điều kiện thuận lợi cho người dùng là chuyên gia và bác sĩ.

Từ các usecase đã phân tích ở trên thì chúng tôi đã phác thảo ra được lược đồ quan hệ thực thể của Health Monitoring. Trong đó, chúng tôi chú trọng đến bảng Medical Record và bảng Notification hai bảng này giúp khá nhiều dữ liệu cho chức năng chính là xem tình trạng bệnh nhân và gửi thông báo cho bác sĩ khi gặp bất thường. Ngoài ra thì còn các bảng dữ liệu khác bổ trợ cho chức năng chính đồng thời làm rõ các mối quan hệ trong lúc xử lý dữ liệu.

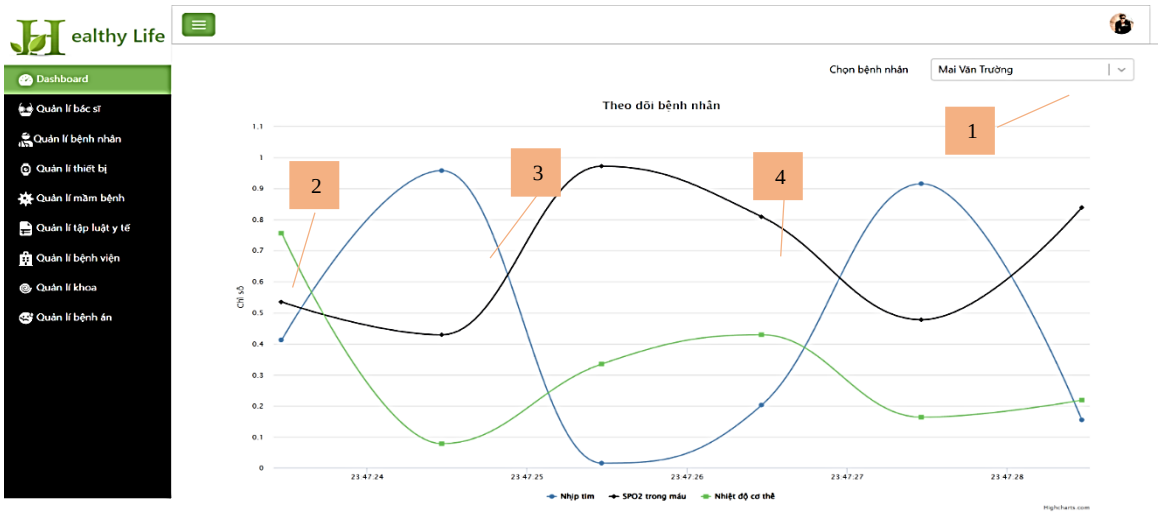



Hình 6: Sơ đồ use case của hệ thống Health Monitoring


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.4 Giao diện đồ họa

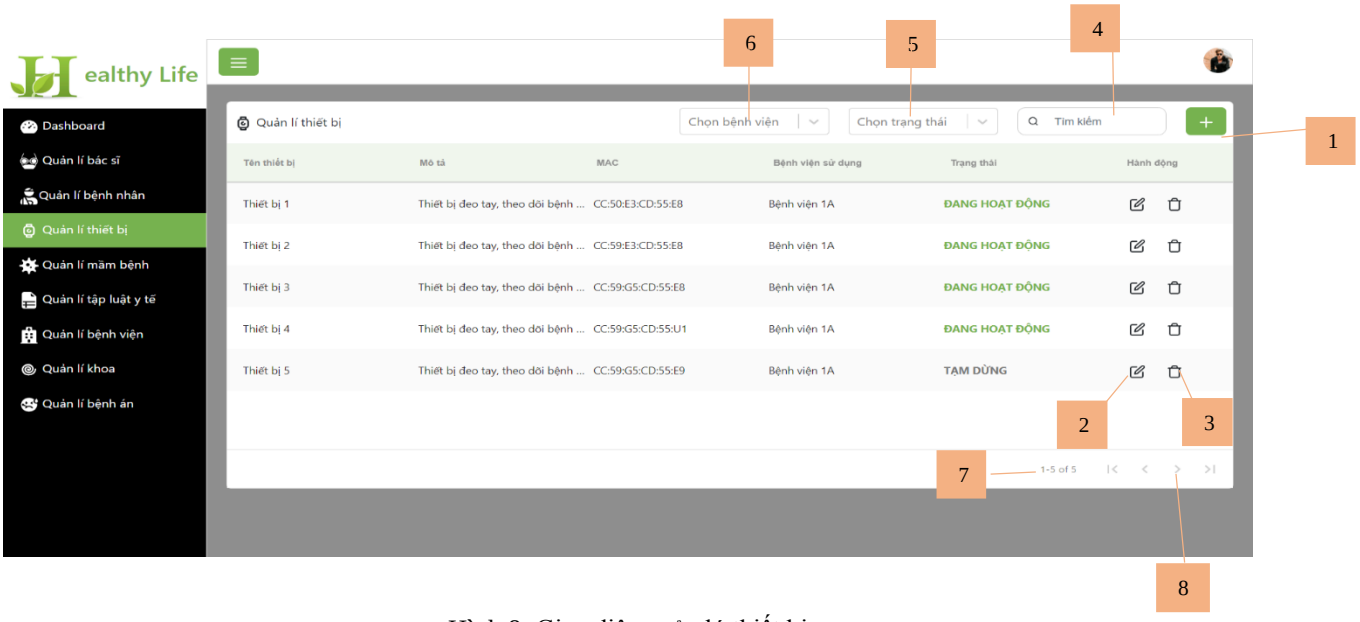
Hệ thống Health Monitoring triển khai nền tảng Web app và Mobile app. Web app được triển khai theo dạng CMS, Mobile app được xây dựng để chạy trên cả hai nền tảng Android/iOS. Công cụ này chúng tôi quan tâm đến trải nghiệm của người dùng, giao diện thân thiện và đảm bảo tính ổn định cho cả website và app. Sau đây chúng tôi xin giới thiệu một vài giao diện và chức năng của Health Monitoring trên Web và App. Sau khi chuyên gia hoặc bác sĩ đăng nhập sẽ hiện ra giao diện dashboard hiển thị biểu đồ đường của một bệnh nhân. Đối với chuyên gia, có thể chọn bất kì bệnh nhân nào có trong hệ thống. Đối với bác sĩ, chọn bệnh nhân mà bác sĩ đó quản lý. Dựa trên bệnh nhân đã chọn website sẽ hiển thị biểu đồ đường gồm 3 đường: đường màu xanh dương đại diện cho nhịp tim, đường màu đen đại diện cho phần trăm SPO2 trong máu và đường màu xanh lá đại diện cho nhiệt độ cơ thể của bệnh nhân. Biểu đồ chạy trong thời gian thực mỗi 5 giây, những dữ liệu bệnh nhân trước đó sẽ được đưa xuống database.




Hình 7: Giao diện dashboard

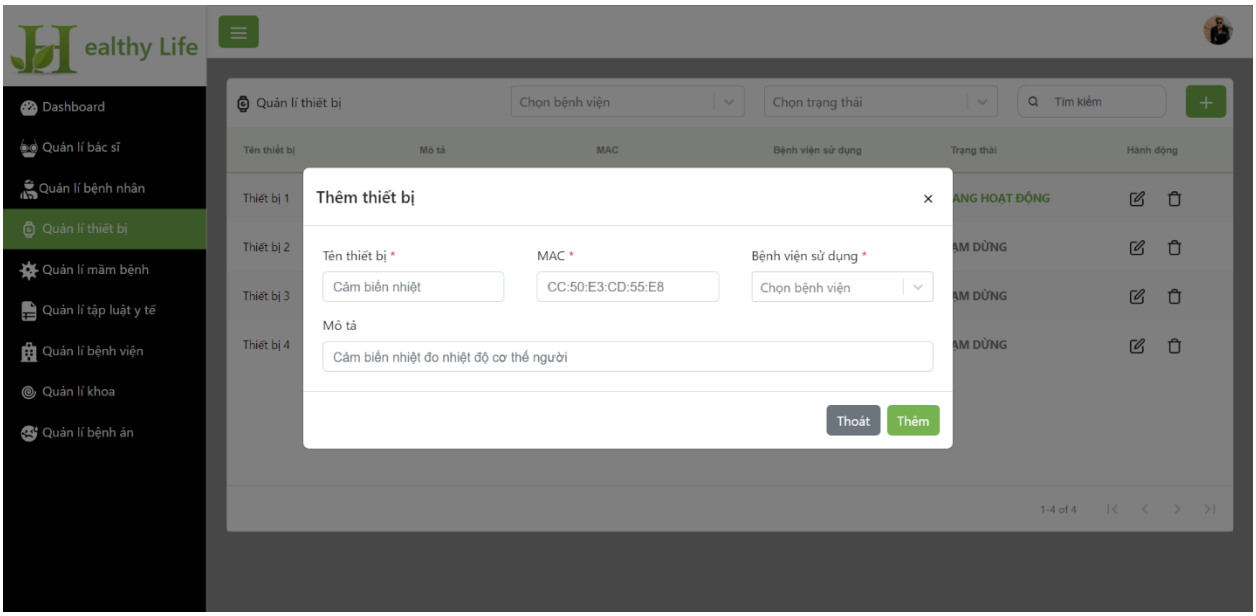
Dashboard là giao diện trang chủ của website. Theo dõi bệnh nhân tùy theo lựa chọn của bác sĩ hoặc chuyên gia. Hiển thị biểu đồ đường với các chỉ số như nhịp tim, SPO2 trong máu, nhiệt độ cơ thể do thiết bị IoT truyền lên. "vùng 1": Chọn bệnh nhân để xem tình trạng của bệnh nhân vừa chọn. Mỗi bệnh nhân sẽ có một bệnh án hiện tại, bệnh án đó sẽ cho biết bệnh nhân đang sử dụng thiết bị nào, từ thiết bị sẽ hiển thị các thông số đã chọn. "vùng 2": Biểu đồ đường thể hiện nhiệt độ cơ thể của bệnh nhân đã chọn. "vùng 3": Biểu đồ đường thể hiện nhịp tim của bệnh nhân đã chọn. "vùng 4": Biểu đồ đường thể hiện SPO2 trong máu của bệnh nhân đã chọn Hình 7.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 8: Giao diện quản lý thiết bị

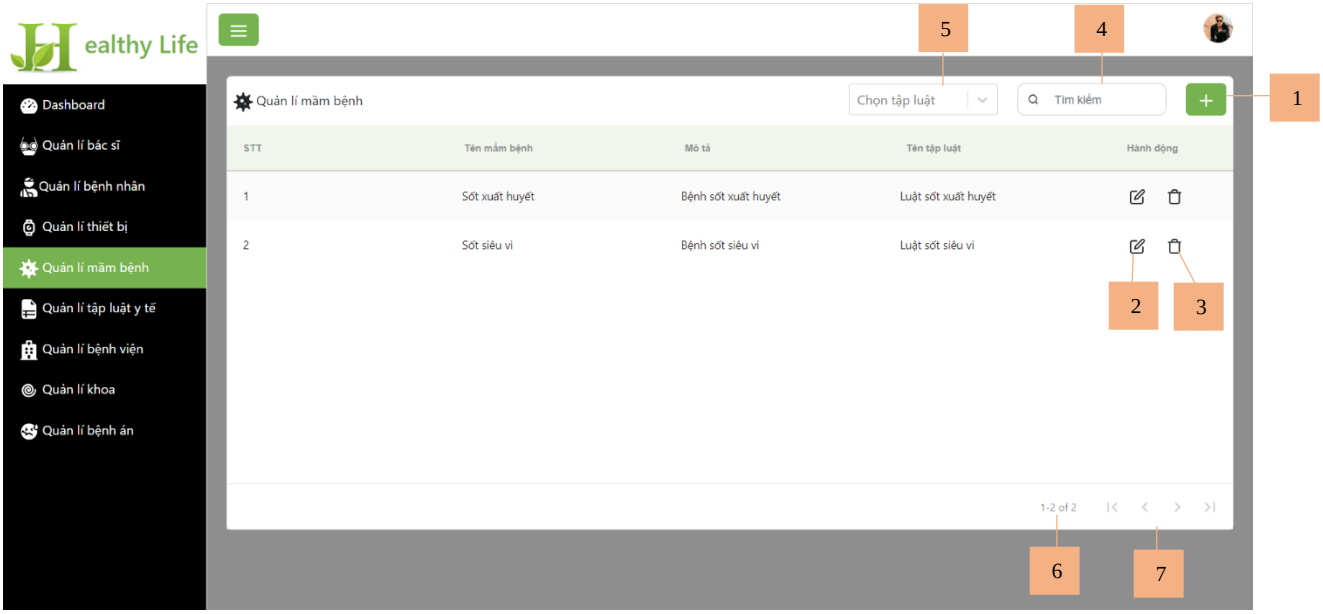
Quản lý thiết bị: giao diện hỗ trợ xem danh sách thiết bị của hệ thống, cũng như tạo, cập nhật và xóa thiết bị. "vùng 1": Nút + dùng để thêm thông tin một thiết bị mới. "vùng 2": Nút dùng để sửa một thông tin thiết bị. "vùng 3": Nút dùng để xóa mềm một thiết bị nào đó. "vùng 4": Ô tìm kiếm thiết bị theo từ khóa. "vùng 5": Filter trạng thái thiết bị theo người dùng chọn. "vùng 6": Filter bệnh viện sử dụng thiết bị theo người dùng chọn. "vùng 7": Thể hiện tổng các dữ liệu thiết bị đang hiển thị của table. "vùng 8": Các nút thao tác chuyển trang Hình 8.




Hình 9: Giao diện thêm thiết bị

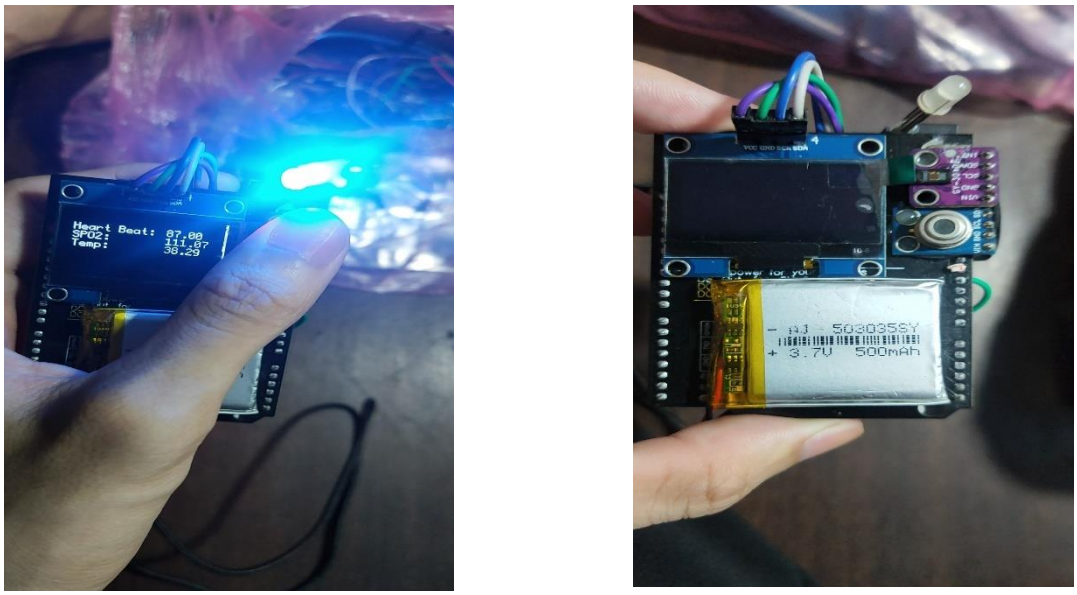
"vùng 1": Nút thêm, khi người dùng nhấn sẽ thực hiện chức năng thêm thiết bị. "vùng 2": Nút thoát, khi người dùng nhấn sẽ thực hiện chức năng thoát khỏi modal và trở về trang chính. "vùng 3": Tương tự "vùng 2" Hình 10.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 10: Giao diện quản lý mầm bệnh

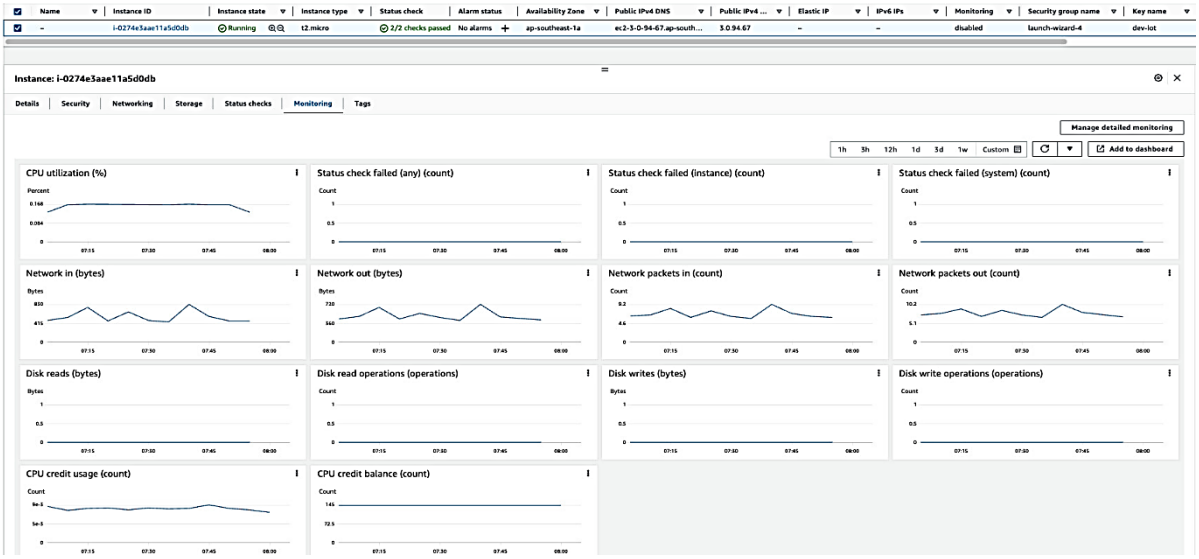
Quản lý mầm bệnh: giao diện hỗ trợ xem danh sách mầm bệnh của hệ thống, cũng như tạo, cập nhật và xóa thiết bị. "vùng 1": Nút dùng để thêm thông tin một mầm bệnh mới vào hệ thống. "vùng 2": Nút dùng để cập nhật một thông tin mầm bệnh đã có trong hệ thống. "vùng 3": Nút dùng để xóa mềm một mầm bệnh nào đó. "vùng 4": Tìm kiếm mầm bệnh theo từ khóa. "vùng 5": Hiển thị tổng số mầm bệnh có trong hệ thống và số dữ liệu đang hiển thị. "vùng 6": Các nút thực hiện chức năng chuyển trang.




Hình 11: Thiết bị đeo tay dùng cho bệnh nhân trong hệ thống Health Monitoring

Thiết bị IoT có cảm biến nhiệt độ, nhịp tim, SPO2 dùng để thu thập dữ liệu. Mạch Arduino Uno + Wifi R3 Atmega328P + ESP8266 trong đó ESP8266 trung tâm để xử lý các dữ liệu và đồng bộ dữ liệu với hệ thống bao gồm: Màn hình OLED hiển thị thông tin các dữ liệu cơ thể; cảm biến nhịp tim và oxy trong máu MAX30102 (sử dụng để đo nhịp tim và oxy trong máu của bệnh nhân khi bệnh nhân chạm vào cảm biến); cảm biến nhiệt hồng ngoại không tiếp xúc MLX90614 (sử dụng đo nhiệt độ cơ thể của bệnh nhân khi bệnh nhân chạm vào cảm biến); đèn Led RGB sẽ bật đèn và thay đổi màu sắc tùy theo trạng thái. (trạng thái bệnh

nhân không chạm vào cảm biến đèn xanh lá, trạng thái bệnh nhân bình thường xanh dương đậm, trạng thái bất thường xanh dương phát quang); Pin hỗ trợ pin cho thiết bị IoT.




Hình 12: ECS Cluster Monitoring trên AWS cho hệ thống Health Monitoring

## 4 KIỂM NGHIỆM THỰC TẾ

Chúng tôi sử dụng dịch vụ ECS Cluster của AWS để triển khai hệ thống, ECS Cluster giúp quản lý và triển khai các ứng dụng trong một môi trường điều khiển đám mây. ECS Cluster cho phép chúng tôi chạy các ứng dụng container trên các máy ảo (EC2 instances) được quản lý bởi ECS. Đồng thời quản lý các tài nguyên của chúng tôi thông qua giao diện quản lý web hoặc các API được cung cấp bởi AWS. Chúng tôi có thể tự động triển khai các ứng dụng container, quản lý các phiên bản container và giám sát các container trong cluster (Hình 12).

| | |
|---|---|
| **CPU Utilization (%)** | Chỉ số đo lường tỷ lệ phần trăm của CPU đang được sử dụng so với tổng khả năng của CPU có sẵn trên các instance. Chỉ số dao động từ 0% đến 100%.<br>Đơn vị: % |
| **Status check failed (count)** | Tổng số lần mà bất kỳ kiểm tra trạng thái nào (cả System Status Checks và Instance Status Checks) thất bại trong một khoảng thời gian nhất định.<br>Đơn vị: count |
| **Network in (bytes)** | Chỉ số này thể hiện lưu lượng dữ liệu đang được gửi đến máy chủ hoặc máy ảo thông qua giao diện mạng. Số liệu này xác định khối lượng lưu lượng mạng đến cho một ứng dụng duy nhất.<br>Đơn vị: byte |
| **Network out (bytes)** | Chỉ số này thể hiện lưu lượng dữ liệu đang được gửi đi từ máy chủ hoặc máy ảo thông qua giao diện mạng. Số liệu này xác định khối lượng lưu lượng mạng đến cho một ứng dụng duy nhất.<br>Đơn vị: byte |
| **Network packets in (count)** | Chỉ số này đo lường số lượng gói tin mạng đã được nhận vào qua giao diện mạng của máy chủ hoặc máy ảo trong một khoảng thời gian cụ thể. Số liệu này xác định khối lượng lưu lượng đến về số lượng gói trên một ứng dụng.<br>Đơn vị: count |
| **Network packets out (count)** | Chỉ số này đo lường số lượng gói tin mạng đã được gửi đi từ máy chủ hoặc máy ảo qua giao diện mạng trong một khoảng thời gian cụ thể.<br>Đơn vị: count |

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| | |
|---|---|
| **Disk reads (bytes)** | Là một thông số trong giám sát hệ thống và lưu lượng I/O (Input/Output) của các ổ đĩa trên máy chủ hoặc máy ảo. Chỉ số này đo lường số lần đọc dữ liệu từ ổ đĩa trong một khoảng thời gian cụ thể.<br>Đơn vị: byte |
| **Disk read operations (operations)** | Là một thông số quan trọng trong giám sát hiệu suất và hoạt động của ổ đĩa (disk) trên máy chủ hoặc máy ảo. Chỉ số này đo lường số lần thực hiện các hoạt động đọc từ ổ đĩa trong một khoảng thời gian cụ thể.<br>Đơn vị: count |
| **Disk writes (bytes)** | Giám sát và đánh giá hiệu suất của ổ đĩa (disk) trên máy chủ hoặc máy ảo. Chỉ số này đo lường khối lượng dữ liệu đã ghi vào ổ đĩa trong một khoảng thời gian cụ thể.<br>Đơn vị: byte |
| **Disk writes operations (operations)** | Thông số giám sát hiệu suất và hoạt động của ổ đĩa (disk) trên máy chủ hoặc máy ảo. Chỉ số này đo lường số lần thực hiện các hoạt động ghi dữ liệu lên ổ đĩa trong một khoảng thời gian cụ thể.<br>Đơn vị: count |
| **CPU credit usage (count)** | Chỉ số này giúp người quản trị đo lường mức tiêu thụ CPU Credits, từ đó đánh giá hiệu suất CPU thực tế của EC2 instance. Theo dõi "CPU credit usage" giúp người quản trị theo dõi việc sử dụng CPU Credits để biết khi nào instance đang tiêu thụ nhiều CPU Credits và có thể tiến hành các biện pháp điều chỉnh nếu cần thiết.<br>Đơn vị: count |
| **CPU credit balance (count)** | Chỉ số này giúp người quản trị biết số lượng CPU Credits còn lại trong tài khoản của EC2 instance. Điều này giúp đảm bảo rằng instance có đủ CPU Credits để đáp ứng yêu cầu xử lý trong thời gian tới. cung cấp thông tin về việc sử dụng và tích lũy CPU Credits, từ đó người quản trị có thể đánh giá hiệu suất CPU thực tế của EC2 instance và xem liệu nó có đang hoạt động ở mức burstable hay không.<br>Đơn vị: count |

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Bài báo chúng tôi đã trình bày giải pháp xây dựng hệ giám sát, cảnh báo thông minh cho bệnh nhân theo thời gian thực có kết hợp các thiết bị đeo tay trong lĩnh vực IoT. Hệ thống chúng tôi đề xuất Health Monitoring đã đáp ứng được hầu hết những chức năng cơ bản của một thiết bị cảnh báo thông minh như thu thập huyết áp, nhịp tim, nhiệt độ cơ thể và các chỉ số khác một các tự động. Dữ liệu này được tự động gửi đến hệ thống xử lý trung tâm thông qua các thiết bị IoT. Hệ thống xử lý trung tâm sử dụng những tập luật (từ chuyên gia) kết hợp với thiết bị IoT đảm bảo sự đồng bộ tình trạng của bệnh nhân với trạng thái của thiết bị theo thời gian thực. Từ đó, bài toán tự động hóa quy trình theo dõi bệnh nhân đã được hệ thống giải quyết một cách gọn gàng và nhanh chóng. Các thông số của bệnh nhân sẽ được hệ thống cập nhật liên tục trong thời gian thực. Health Monitoring được xây dựng trên nền tảng Arduino, sử dụng Mongo Cloud làm database lưu trữ dữ liệu và một số service của firebase, sử dụng mô hình Agile và Scrum để quản lý quá trình phát triển. Website được viết theo hệ thống CMS (Content Management System) sẽ quản lý toàn bộ chức năng của hệ thống, Mobile dành cho bác sĩ với một số chức năng tương tự với website. Hệ thống cũng đã được triển khai trong mô hình thực tế với máy chủ và dữ liệu triển khai trên đám mây của AWS. Chúng tôi cũng đã dùng hệ thống này cho các bệnh nhân Sốt (Sốt xuất huyết, sốt siêu vi) tại Tp.HCM, đồng thời cũng lấy ý kiến về chuyên gia trong lĩnh vực này. Hiện chúng tôi cũng nhận được nhiều ý kiến tích cực về hệ thống Health Monitoring về tính khả thi và hiệu quả. Tuy nhiên, hệ thống vẫn còn một số hạn chế như chưa kết hợp thêm AI (thu thập và phân tích dữ liệu thành các tập luật tự động) trong việc xử lý các thông số của bệnh nhân để đồng bộ với các thành phần trên thiết bị IoT. Chưa triển khai những công nghệ trả lời tự động như chat GPT vào website và mobile. Trong tương lai, chúng tôi sẽ hoàn thiện các chức năng này trong hệ thống. Đồng thời cũng bổ sung một số chức năng tiện ích khác như: với mobile thêm chức năng chat và call video tư vấn trực tiếp với bác sĩ; tối ưu thêm các chức năng filter, search cho các module

trong hệ thống; bổ sung tính năng cảnh báo SOS cho thiết bị IoT cũng như thông tin về GPS của bệnh nhân trong các trường hợp khẩn cấp.

## TÀI LIỆU THAM KHẢO

[1]. Sim, W., Choi, W.: Spring boot-based web application development for providing information on security vulnerabilities and patches for open source software. Journal of Korea Society of Digital Industry and Information Management 17(4) (2021) 77–83

[2]. Spring boot Architecture. [online] https://dzone.com/articles/spring-boot-architecture-andworkflow [Accessed 14/06/2023]

[3]. ReactJS Library Version 18 (Release March 16, 2022), [online], https://react.dev/, Accessed 01/06/2023

[4]. React Native Library Version 0.71 (Release January 12, 2023), [online], https://reactnative.dev/, Accessed 01/06/2023

[5]. Firebase Google Service, [online], https://firebase.google.com/, Accessed 01/06/2023

[6]. Adruino VN "Cộng đồng Adruino Việt Nam", [online], http://arduino.vn/, Accessed 01/06/2023

[7]. Jenkins "Jenkins Build great things at any scale", [online], https://www.jenkins.io/, Accessed 01/06/2023

[8]. "Điện toán đám mây là gì?". [online], https://aws.amazon.com/vi/what-is-cloudcomputing/, Accessed 02/06/2023

[9]. Amazon Web Service – EC2 (2023), [online], Secure and resizable Cloud Compute, https://aws.amazon.com/ec2, Accessed 02/06/2023

[10]. Amazon Web Service – S3 (2023), [online], Cloud Object Storage, https://aws.amazon.com/s3/, Accessed 02/06/2023

[11]. Amazon Web Service – CloudFront (2023), [online], Content Delivery Network, https://aws.amazon.com/cloudfront/, Accessed 02/06/2023

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F021**

# HỆ THỐNG HỖ TRỢ HỌC VỤ CHO SINH VIÊN TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH

TÔN LONG PHƯỚC[1], LÊ VĂN TÙNG[1], PHAN TẤN TÀI[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*tonlongphuoc@iuh.edu.vn, tuny113tungle@gmail.com, taito1doraemon@gmail.com*

**Tóm tắt.** Trường Đại học Công nghiệp TP.HCM được xếp hạng 10 trường đại học hàng đầu Việt Nam năm 2021 theo bảng xếp hạng QS. Đây là một trường đào tạo đa ngành với điều kiện cơ sở hạ tầng tốt và đội ngũ giảng viên có chuyên môn cao. Hiện tại, trường có hơn 36,000 sinh viên, học viên đang học tập và nghiên cứu tại đây. Do đó, trường cần có một công cụ hỗ trợ sinh viên, học viên trong quá trình học tập và nghiên cứu tốt hơn. Hệ thống phải có các chức năng học vụ cơ bản cũng như đa phân hệ đảm bảo cho sinh viên thuận tiện sử dụng, đồng thời hệ thống phải luôn ổn định ở mọi thời điểm. Chúng tôi đề xuất công cụ có tên là "ERUKALEARN", một công cụ có đầy đủ các chức năng về học vụ cho sinh viên, cũng như giảng dạy cho giảng viên, nhân viên tại trường Đại học. Công cụ gồm hai phân hệ chính: ứng dụng Web và ứng dụng trên thiết bị di động. Chúng tôi dùng các công nghệ như NodeJS, ReactNative và đặc biệt là AWS để triển khai hệ thống thay vì dùng chủ vật lý giúp giải quyết vấn đề tắc nghẽn đang cấp thiết. Hiện hệ thống đã được thử nghiệm cho một số sinh viên tại Khoa Công nghệ thông tin tại trường. Trong tương lai, chúng tôi sẽ mở rộng các chức năng cũng như tích hợp các hệ thống khác trong trường nhằm nâng cao tính tiện dụng của hệ thống.

**Từ khoá.** Ứng dụng web, Ứng dụng di động, AWS, Điện toán đám mây.

## A STUDENT SUPPORT SYSTEM FOR STUDENTS OF INDUSTRIAL UNIVERSITY OF HO CHI MINH CITY

**Abstracts.** Industrial University of Ho Chi Minh City is ranked in the top 10 universities in Vietnam in 2021 according to the QS rankings. This is a multidisciplinary training school with good infrastructure conditions and highly qualified teaching staff. Currently, the school has more than 36,000 students and trainees studying and researching here. Therefore, the school needs a tool to support students and learners in the process of studying and researching here and tool need to be stable at all times. The system needs to have academic functions as well as many modules for students and trainees to be convenient during use. We recommend a tool called "ERUKALEARN", a tool with full academic functions for students, trainees as well as teaching for lecturers and staff at the University of Industry. The tool consists of two main modules: Web application and mobile application. We use technologies like NodeJS, ReacteNative and specially AWS instead of physical server to solve the urgent congestion. Currently the system has been tested for some students at the Faculty of Information Technology at the university. In the future, we will expand the functions as well as integrate other systems in the school to improve the usability of the system.

**Keywords.** Web application, Mobile application, AWS, Cloud-Computing.

## 1 GIỚI THIỆU

Trường Đại học Công nghiệp TP.HCM (IUH) là một trong những trường đại học lớn của Bộ Công Thương, cũng là trường có quy mô đào tạo lớn với hơn 36,000 sinh viên, đa ngành, đa bậc học và nhiều chương trình liên kết quốc tế. Trường được đầu tư cơ sở vật chất tốt, đáp ứng nhu cầu của người học và các tiêu chuẩn quốc tế. Bên cạnh, đội ngũ giảng viên có trình độ cao cùng với nhiều chương trình đào tạo đạt chuẩn quốc tế như AUN. Vì vậy, Nhà trường cần có một hệ thống website hỗ trợ đào tạo một cách hiệu quả cho người

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

học, người dạy, người quản lý. Tuy nhiên, hệ thống website hỗ trợ cho người học hiện nay còn nhiều hạn chế. Về mặt hệ thống, khả năng đáp ứng truy cập đồng thời của sinh viên trong một số thời điểm nhất định không được ổn định. Ví dụ như vào thời điểm nhà trường mở đăng ký lớp học phần cho học kỳ mới hoặc đầu tiết học thì nhu cầu truy cập website tăng cao một cách đột ngột. Lúc này hiện tượng trang không phản hồi khi truy cập website của nhà trường xảy ra thường xuyên trong nhiều ngày hoặc không thực hiện các chức năng trên hệ thống. Sinh viên không thể làm gì được trong vòng nhiều giờ dù đã tốn công thức khuya dậy sớm để đăng ký học phần chẳng hạn. Tình trạng này có nhiều nguyên nhân. Nguyên nhân đầu tiên là số lượng sinh viên truy cập website của trường tăng đột biến, từ đó dẫn đến việc số lượng yêu cầu (request) gửi lên máy chủ tăng cao. Thứ hai, xử lý quá nhiều yêu cầu vượt quá giới hạn của máy chủ vật lý gây hiện tượng trang không phản hồi. Ngoài ra, cũng còn có những nguyên nhân khách quan khác như đường truyền và hệ thống hạ tầng của nhà trường chưa đáp ứng đủ mạnh. Chính vì lẽ đó, giải pháp dễ thấy nhất là nâng cấp cấu hình của máy chủ, thậm chí là tăng số lượng máy chủ lên để xử lý được nhiều yêu cầu hơn. Đồng thời, nếu như việc đầu tư này đã đáp ứng, chúng ta cũng cần phải đưa ra các giải pháp này rõ ràng hơn như việc kết nối máy chủ mới vào hệ thống xử lý cũ cũng là một bài toán lớn. Tương tự như với việc nâng cấp cấu hình cũng có nhiều câu hỏi đặt ra: Nâng cấp trong bao lâu?, bao nhiêu? Có xảy ra rủi ro trong lúc nâng cấp không? Nếu phát sinh những vấn đề ngoài ý muốn lúc nâng cấp thì phải ứng cứu kịp thời như thế nào? Và còn rất nhiều vấn đề cần giải quyết nữa. Vấn đề cần bận tâm là một năm chỉ xảy ra hai đến ba đợt mở lớp đăng ký học phần, nghĩa là chỉ có tại thời điểm đó thì nhu cầu mới vượt quá ngưỡng xử lý của máy chủ. Vậy nhà trường có nên tốn kém hay không khi đầu tư một hệ thống với chi phí lớn cho một số sự cố xảy ra không thường xuyên? Về mặt học vụ, việc thanh toán học phí hiện nay ở IUH vẫn còn thực hiện thủ công, dẫn đến việc sinh viên xếp hàng và mất thời gian.

Để giải quyết các vấn đề của các mặt, chúng tôi đã đưa ra 2 giải pháp chính: một là sử dụng điện toán đám mây (cloud computing) về mặt hệ thống và hai là tích hợp thanh toán học phí online bằng các ví điện tử nội địa (momo, vnpay) vào trang sinh viên. Đầu tiên, nói về giải pháp sử dụng các dịch vụ trên điện toán đám mây như cho thuê máy chủ, dùng các dịch vụ web,.. nổi tiếng trên thế giới hiện nay đó là các dịch vụ cung cấp bởi các công ty như Amazon với AWS, Microsoft với Azure, trong nước có các tổ chức như Viettel, FPT, Sao bắc đẩu… Điện toán đám mây là một giải pháp có tính xu hướng toàn cầu cho các cá nhân, tổ chức, doanh nghiệp vừa và nhỏ [1]. Hiện nay các dịch vụ trực tuyến phục vụ cho nhu cầu cá nhân, kinh doanh ngày càng nhiều. Hệ thống triển khai trên điện toán đám mây sẽ có chi phí rẻ, cấu hình đơn giản nhưng vẫn an toàn, bảo mật tốt, khả năng nâng cấp bảo trì hệ thống dễ dàng. Hơn hết trong các ưu điểm trên là khả năng cân bằng tải, đó là ưu điểm tuyệt vời của điện toán đám mây. Cân bằng tải (scale) là dịch vụ có khả năng tự động điều chỉnh số lượng máy chủ để đáp ứng nhu cầu truy cập của khách hàng (tăng khi yêu cầu nhiều và giảm khi yêu cầu ít). Do đó nhà phát triển hệ thống không còn phải đầu tư quá nhiều chi phí chỉ để giải quyết nhu cầu cho việc sử dụng tài nguyên của hệ thống trong một quãng thời gian ngắn trong năm. Đó cũng là lý do chúng tôi chọn giải pháp này cho bài toán xây dựng hệ thống hỗ trợ đào tạo cho sinh viên của IUH.

Riêng giải pháp xây dựng website và ứng dụng trên di động cho sinh viên sử dụng nhằm giải quyết các nguyên nhân được liệt kê sau đây. Nguyên nhân thứ nhất là website của trường IUH thực hiện cơ chế kết xuất máy chủ Server Side Rendering (SSR). Cơ chế này có nhiều ưu điểm tuyệt vời như khả năng Search Engine Optimization (SEO) hiệu quả, chạy được trên phần lớn các trình duyệt thông dụng (kể cả disable JavaScript vẫn chạy tốt), hiệu năng cho lần tải trang đầu tiên tốt... Nhưng lại có một nhược điểm lớn chính là máy chủ phải đảm nhiệm gánh nặng kết xuất trang web, gánh nặng trả về nội dung trang web (tốn nhiều băng thông) và vô tình làm tăng số lượng như cầu phải xử lý cho máy chủ [2]. Nguyên nhân thứ hai, đặt trường hợp khi nhu cầu truy cập website tăng cao đột biến, máy chủ phải thực hiện một số lượng yêu cầu cực lớn và còn phải phụ trách gánh nặng kết xuất trang web và gửi trả về nó càng làm tình hình xấu đi. Hơn nữa, các trang web như trang đăng ký học phần, quản lý thông tin cho sinh viên là những trang web có tính cá nhân nên cần rất ít khả năng SEO, tính chất tương thích với trình duyệt hoàn toàn có thể khắc phục được bởi lập trình viên. Đó là lý do chúng tôi xây dựng ứng dụng cho thiết bị di động nhằm hỗ trợ cá nhân hóa cũng như giảm tải cho máy chủ trong hệ thống.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

# 2 CÁC CÔNG NGHỆ LIÊN QUAN

## 2.1 Cloud Computing với AWS

### 2.1.1 Amazon Relational Database Service (Amazon RDS)

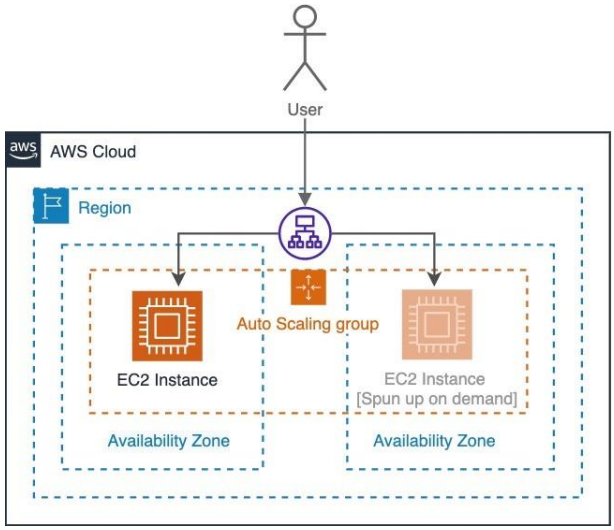
Amazon Relational Database Service (Amazon RDS) là một dịch vụ tổ chức và quản lý dữ liệu quan hệ trên đám mây của AWS, dịch vụ này giúp cho người dùng dễ dàng thiết lập, vận hành và quản trị dữ liệu một cách hiệu quả.

### 2.1.2 Amazon Simple Storage Service (S3)

AWS S3 là dịch vụ lưu trữ đối tượng do AWS cung cấp. Đây có lẽ là dịch vụ lưu trữ được sử dụng phổ biến nhất cho người dùng AWS nhờ các tính năng như tính khả dụng, bảo mật cực cao và kết nối đơn giản với các dịch vụ AWS khác. Chúng tôi dùng AWS S3 để lưu trữ các file với nhiều định dạng khác nhau: lưu trữ ảnh png, jpg 3x4 của sinh viên, file excel nhập điểm của giảng viên tải lên, file backup .bak mà nhân viên thực hiện backup để làm dự phòng, lưu trữ các tập tin tài nguyên của việc phát triển (keys, certificates,..). Sử dụng S3 kết hợp với các service khác của AWS cũng là một lợi thế [4, 5].

### 2.1.3 Amazon Elastic Compute Cloud (Amazon EC2)

Đầu tiên, EC2 là viết tắt của Elastic Compute Cloud. Amazon EC2 là một máy ảo cơ bản với các thành phần phần cứng có thể tùy chỉnh và một hệ điều hành. Elastic Compute Cloud là hệ thống dịch vụ chính và được sử dụng nhiều trong hệ sinh thái AWS khổng lồ. EC2 giúp bạn không cần phải mua phần cứng. Ngoài ra, không có thêm chi phí khi bảo trì phần cứng đã thuê. AWS cho phép bạn khởi chạy nhiều EC2. Nó cũng cung cấp khả năng kiểm soát để mở rộng quy mô hoặc thu nhỏ tương ứng với tốc độ của lưu lượng truy cập trang web [6].

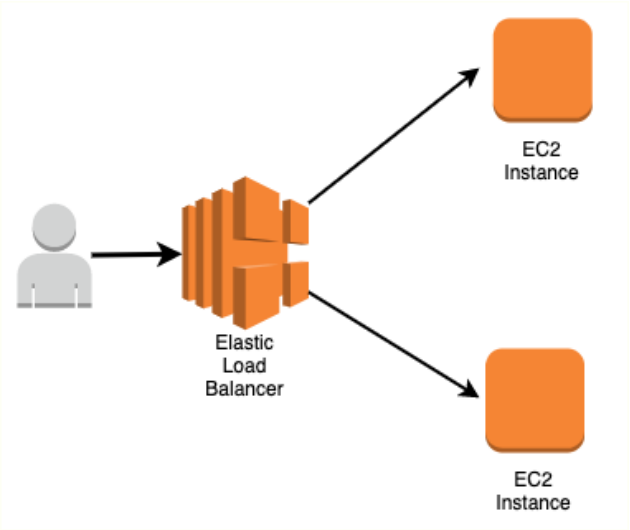



Hình 1: Đặc điểm AutoScale của máy chủ EC2 khắc phục điểm yếu của máy chủ vật lý [5]

Chúng tôi sẽ dùng các máy chủ EC2 này để thay thế các máy chủ vật lý tại Nhà trường. Càng mua nhiều EC2 thì việc chia tải sau khi áp dụng Load Balancer càng hiệu quả. Việc mua nhiều phần mềm máy chủ EC2 trên AWS giúp tiết kiệm chi phí hơn rất nhiều so với việc ta mua thêm phần cứng máy chủ vật lý đời thực. Trong lần này, chúng tôi sẽ tạo ra (mua) khoản 2 con EC2 để tiến hành triển khai máy chủ. Mỗi EC2 sẽ được triển khai hệ thống ErukaLearn giống nhau và đồng bộ hóa truy xuất dữ liệu từ cơ sở dữ liệu RDS.

### 2.1.4 Load Balancers (EC2 Feature – Load Balancing)

Load Balancers – EC2 Feature cũng là một dịch vụ cân bằng tải của AWS nhưng EC2-Feature có nghĩa là nó thuộc về một chức năng của EC2 có khả năng phân phối tải công việc giữa nhiều máy ảo EC2, giúp cân bằng tải và tăng tính sẵn sàng của hệ thống. Chúng tôi sẽ sử dụng loại Load Balancer này (có nhiều cách để load balancing khác nữa) để tiện lợi cấu hình vì nó như là một service con của EC2 service.




Hình 2: Load Balancer sẽ là người điều khiển chia đều tải cho các máy chủ EC2 [6]

## 2.2 Framework React Native Expo

React Native Expo: là một công cụ mã nguồn mở giúp xây dựng ứng dụng di động trên nền tảng IOS và Android. Nó cho phép lập trình viên sử dụng các công nghệ web như HTML, CSS và JavaScript để tạo ứng dụng di động chất lượng cao. Expo cung cấp một loạt các công cụ và tính năng để giúp đơn giản hóa việc phát triển ứng dụng, bao gồm cả thư viện và các hướng dẫn. Với React Native Expo, chúng tôi có thể nhanh chóng tạo ra các ứng dụng di động đa nền tảng chất lượng cao và phát triển nó với tốc độ nhanh chóng và hiệu quả [7, 8]. Chúng tôi chọn Expo thay vì React Native CLI để phát triển app di động bởi vì: Expo build ứng dụng nhanh hơn; Chạy được đa nền tảng (hỗ trợ build web); Được cộng đồng hướng dẫn và đóng góp rộng rãi; Ngoài ra expo còn cho phép deploy ứng dụng lên Expo Store như App Store, CH Play

## 2.3 Thiết kế backend với mô hình MVC

Mô hình MVC (Model-View-Controller) là một kiến trúc phần mềm được sử dụng rộng rãi trong phát triển ứng dụng phần mềm. Chúng tôi dùng kiến trúc này vì nó mang lại những lợi ích sau: Giúp tách biệt logic xử lý dữ liệu (model), giao diện người dùng (view), và luồng điều khiển (controller) để dễ dàng quản lý và bảo trì mã nguồn; Tái sử dụng mã các giao diện người dùng có cùng bố cục; Tích hợp dễ dàng với các công nghệ và thư viện khác được cộng đồng phát triển hỗ trợ. (Ví dụ: phát triển tích hợp thanh toán online); Phù hợp với ứng dụng cho doanh nghiệp có quy mô vừa và nhỏ [9].

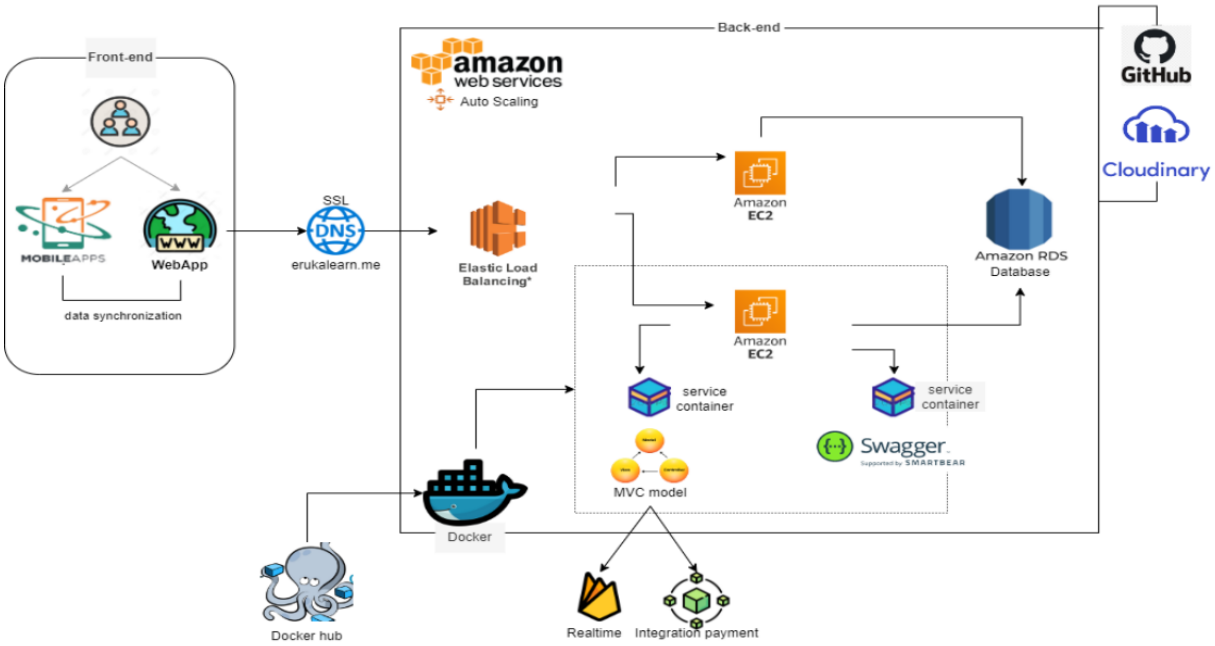
## 2.4 Docker

Docker là một nền tảng ảo hóa cấp phần mềm, cho phép đóng gói ứng dụng và các phụ thuộc của chúng vào các container độc lập. Giúp đơn giản hóa việc chạy ứng dụng trên nhiều môi trường khác nhau mà không cần quan tâm đến cài đặt hệ thống. Docker Compose là công cụ đi kèm với Docker, cho phép định nghĩa và quản lý việc chạy nhiều container cùng lúc. Bằng cách sử dụng tệp cấu hình docker-compose.yml, chúng tôi có thể định nghĩa các ứng dụng nhỏ, mạng và các tùy chọn khác cho ứng dụng giao tiếp với nhau. Docker Compose cho phép các service nhỏ giao tiếp với nhau thông qua mạng nội bộ Docker network. Chúng tôi dùng kỹ thuật Docker Compose vì nhằm mục đích chia nhỏ các trang web dành cho các đối tượng người dùng như: sinh viên, nhân viên, giáo viên, thành từng service (ứng dụng nhỏ). Ngoài ra còn nhiều servce hỗ trợ khác. Điều này giúp việc triển khai ứng dụng sau này tiết kiệm chi phí, chạy đa nền tảng [10].


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

# 3 PHÂN TÍCH HỆ THỐNG

## 3.1 Kiến trúc hệ thống



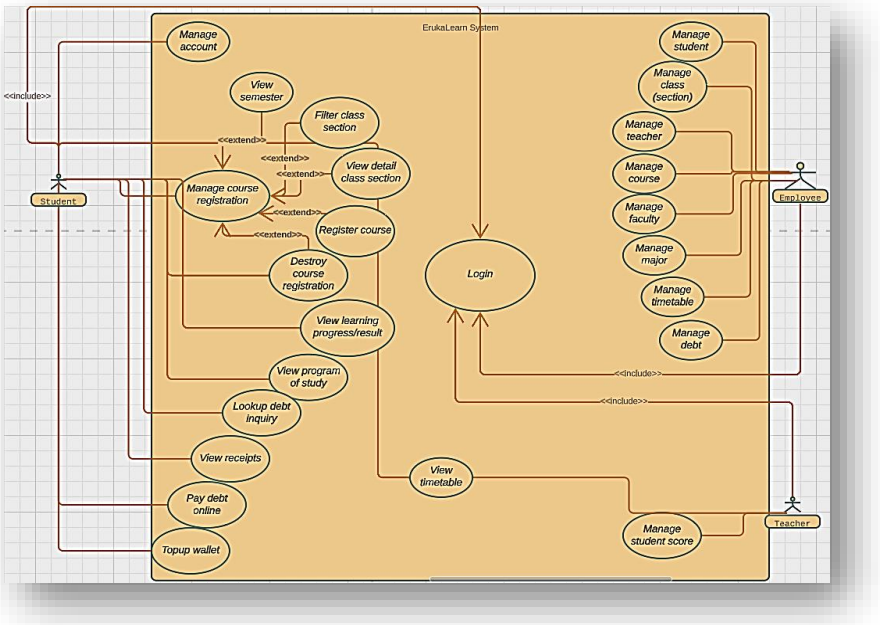

Hình 3: Cấu trúc hệ thống ErukaLearn

Chúng tôi sử dụng kiến trúc Client – Server (chia làm Front-end và Back-end) để xây dựng ứng dụng bằng cách xử lý tất cả request từ người dùng gửi đến máy và hầu hết các công nghệ ở phía máy chủ đều là các service của AWS nhằm mục đích tận dụng các ưu điểm của nó. Với kiến trúc này phần xử lý backend sẽ là quan trọng nhất. Chúng tôi sẽ giải quyết các vấn đề về tắc nghẽn, phân luồng, mở rộng tài nguyên, bảo mật và lưu trữ dữ liệu ở phía server (Hình 3).

## 3.2 Usecase tổng quát

Đối tượng người dùng cho hệ thống ERUKALEARN gồm có 3 actor chính: Sinh viên, Nhân viên, Giáo viên. Tuy nhiên chương trình tập trung vào việc phát triển hỗ trợ cho sinh viên trên nền tảng điện thoại thông minh.. Bởi lý do chính, hiện tại lượng sinh viên IUH có số lượng đông nhưng chưa có nền tảng hỗ trợ trên điện thoại. Bảng use-case thể hiện rõ các chức năng hiện có trong ứng dụng của sinh viên cũng như nhân viên. Nhân viên đóng vai trò là giáo vụ có thể quản lý đào tạo sinh như việc sinh viên nhập học, mở lớp học phần, học phần... Còn sinh viên thực hiện các chức năng cơ bản như theo dõi lịch học, kết quả học tập, đăng ký học phần. Dưới đây là bảng vẽ lược đồ use-case. Những chức năng này chúng tôi trình bày trong Hình 4.

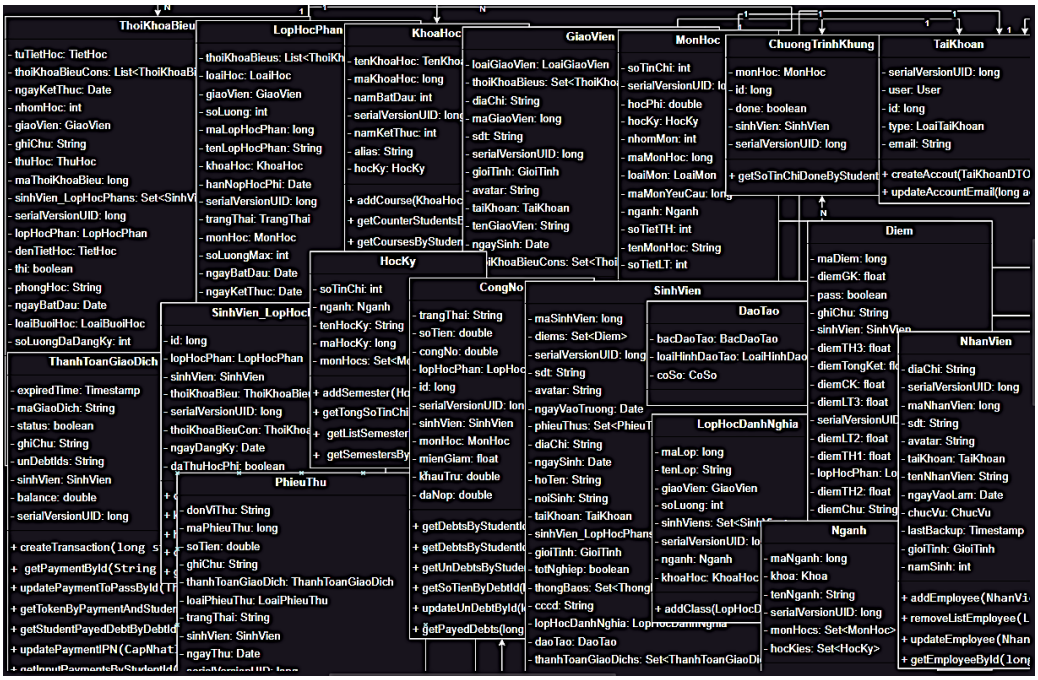
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 4: Sơ đồ Use Case tổng quát của hệ thống ERUKALEARN

## 3.3 Sơ đồ lớp

Từ các usecase đã phân tích ở trên thì chúng tôi đã tạo ra được lược đồ lớp của hệ thống ERUKALEARN. Trong đó, chúng tôi chú trọng đến các lớp như SinhVien, Nhanvien và GiaoVien. Các lớp khác được sinh ra để phục vụ cho 3 đối tượng này. Trong đó chúng tôi tập trung lớp ThoiKhoaBieu để đáp ứng nhu cầu xem lịch của Sinh viên và Giáo viên, tiếp đó lớp ThanhToanGiaoDich nhằm thực hiện các giao dịch của sinh viên, đồng thời cũng hỗ trợ cho đối tượng nhân viên kiểm soát dựa vào đó chứng thực các giao dịch. Ngoài ra, các lớp khác nhằm phục vụ cho việc quản lý, hỗ trợ học vụ, lưu trữ dữ liệu của nhà trường. Chi tiết hơn cho các đối tượng này, chúng tôi trình bày trong lược đồ lớp được thiết kế như Hình 5.

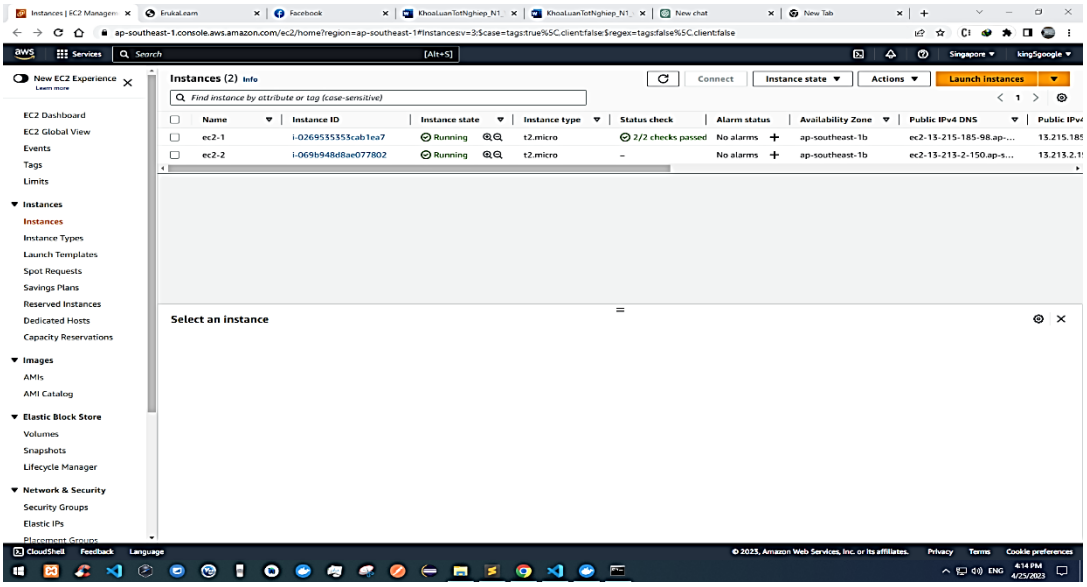



Hình 5: Sơ đồ Class Diagram trong hệ thống ERUKALEARN


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

# 4 KẾT QUẢ HIỆN THỰC

## 4.1 Triển khai Load Balancing

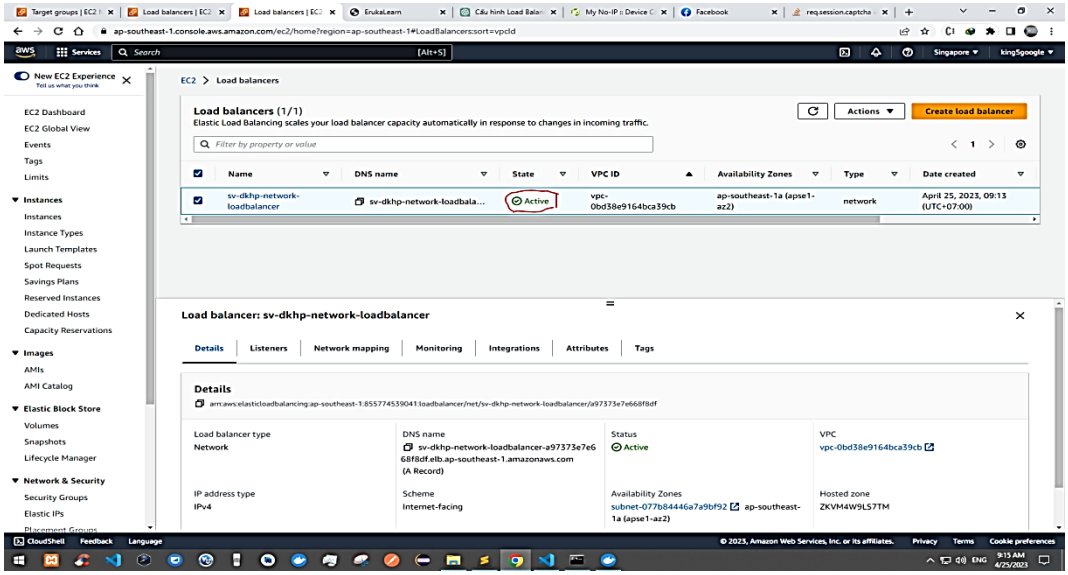



Hình 6: Khởi tạo các máy chủ EC2 trên AWS dùng để chia đều tải

Quy trình triển khai toàn ứng dụng gồm 4 bước chính: i) Khởi tạo và cấu hình số lượng EC2 đủ cần thiết; ii) Triển khai source code ứng dụng lên các EC2 vừa tạo; iii) Tiến hành cấu hình và kích hoạt con Load Balancers để điều khiển tải trên các EC2; iv) Gắn tên miền và chứng chỉ cho con Load Balancers để người dùng truy cập.

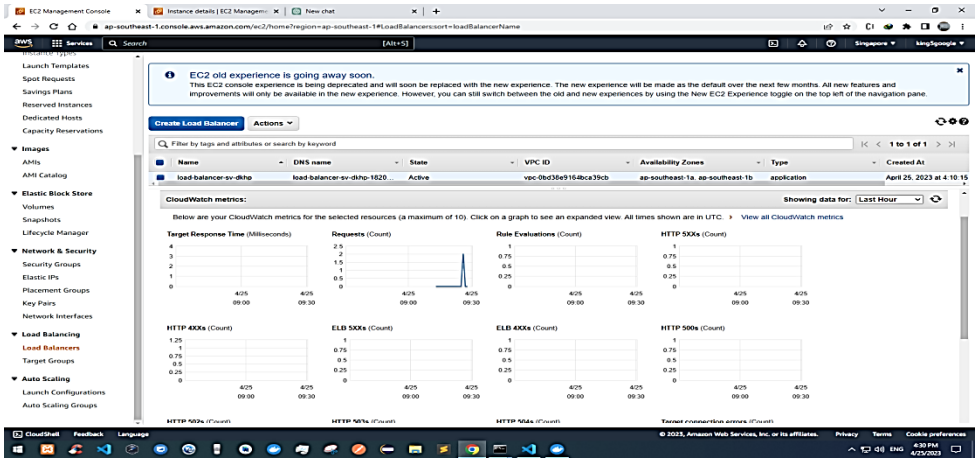
Chúng tôi sẽ thực hiện quy trình từng bước chính thông qua từng quy trình nhỏ: Đầu tiên là cần tạo lượng EC2 để cân bằng tải. Mỗi EC2 hãy cứ hình dung nó nhưng 1 chương trình và được nhân bản ra. Sau đó truy cập vào từng EC2 để kéo (pull) source code ErukaLearn từ Github về. Source code và cách deploy trên mỗi EC2 đều giống nhau lên và chúng được cấu hình để cùng kết nối đến một database RDS. Chi tiết quy trình triển khai lên lên máy chủ EC2: Bước 1: Tạo 2 con EC2 và đặt tên dễ nhớ là ec2_1 và ec2_2 với hệ điều hành linux và cấu hình chúng (mở port, tạo rule,..); Bước 2: Kết nối vào ec2_1 rồi kéo source code từ github về; Bước 3: Thay đổi, cấu hình các biến môi trường kết nối tới RDS, S3, các biến environment; Bước 4: Build ứng dụng bằng docker-compose.yml và ứng dụng ErukaLearn được chạy; Bước 5: Làm tương tự bước 1 đến 4 cho EC2 thứ 2 (Hình 6).

Tiếp theo là đến triển khai một Load Balancer (người điều khiển tải các EC2). Quy trình cấu hình Load Balancer lên 2 con EC2 đã tạo trước đó như sau: Bước 1: Tạo ra một Target Group – EC2 Feature; Bước 2: Gom (đăng ký) nhóm các EC2 lại với nhau trên port 4000 (càng nhiều EC2 thì tải chia ra càng nhiều và càng làm giảm tắc nghẽn); Bước 2: Tạo 1 load balancer, chúng tôi chọn loại Load Balancer Network và cấu hình điều khiển target group. Và load balancer sẽ có trạng thái `Active`; Bước 3: Theo dõi monitor. Theo lý thuyết các luồng request đến server sẽ đi đến qua domain của Load Balancer (Hình 7).



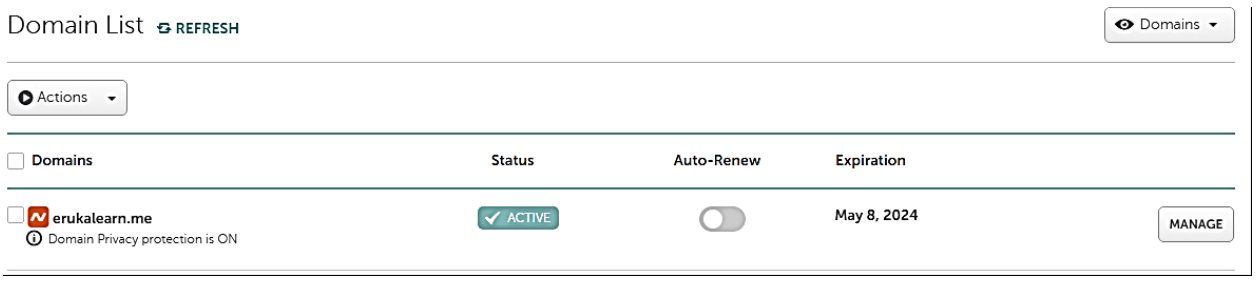

Hình 7: Khởi tạo Load Balancer thành công và cấu hình trạng thái Active

Và đây là một số biểu đồ (monitor) Load Balancers do dịch vụ phân tích trong Hình 8.



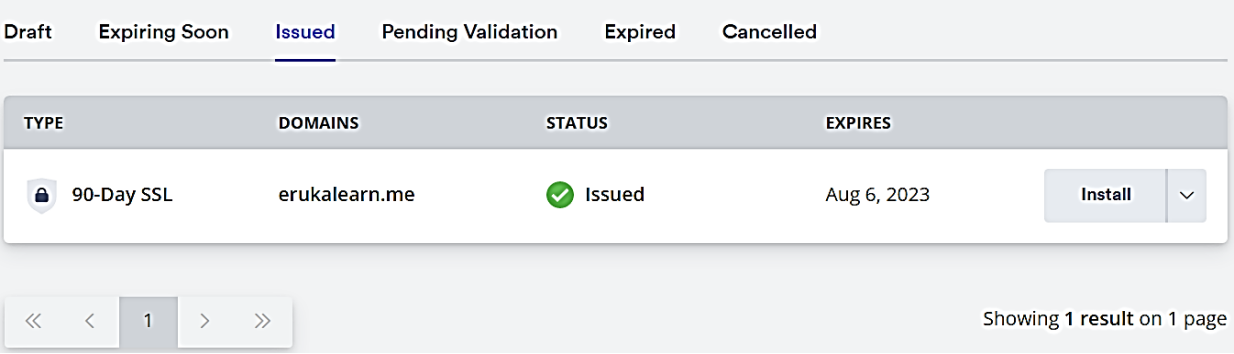

Hình 8: Các biểu đồ Load Balancers của dịch vụ cung cấp

Sau khi đã cấu hình Load Balancer ta sẽ cấp cho nó một tên miền (hiện Namecheap.com tài trợ cho tài khoản sinh viên một domain miễn phí). Với các hiệu năng cơ bản chỉ qua vài bước đăng ký sẽ có kết quả trong Hình 9. Chúng tôi xin phép đặt tên cho ứng dụng mang tên là "ERUKALEARN" và các tên liên quan được đề cập trong ứng dụng, tài liệu, video demo sản phẩm hoặc tên miền đặt cho ứng dụng đều sẽ nhắc đến cụm từ ErukaLearn này để nhất quán dễ gợi nhớ khi nhắc đến chương trình này.




Hình 9: Tên miền erukalearn sau khi mua bằng Namecheap


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

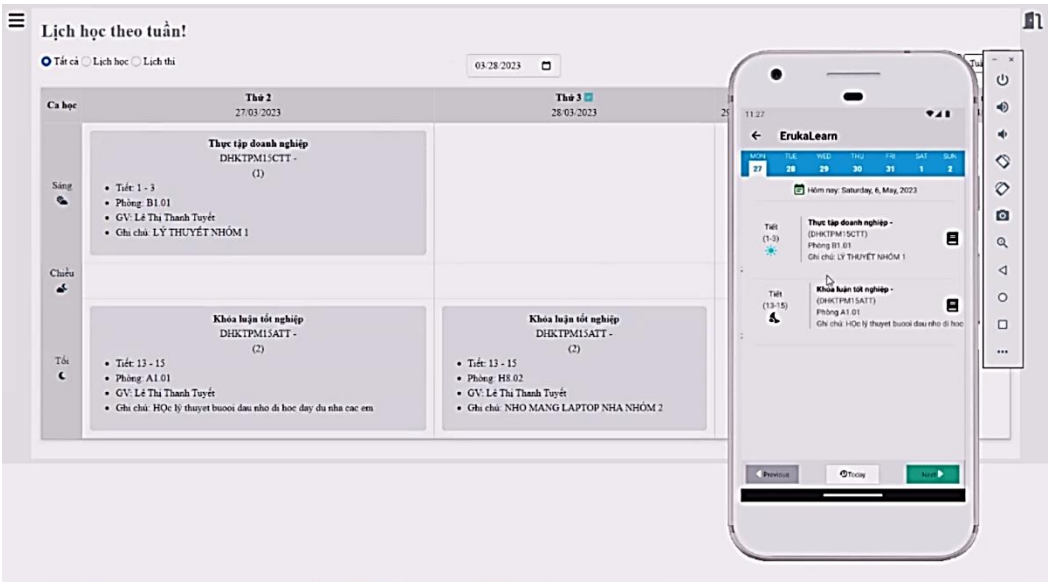



Hình 10: Tên miền Load Balancer được xác thực SSL bằng Namecheap

Ngoài ra ứng dụng còn sử dụng vài tên miền chạy ngầm với các cổng (port) khác nhau như: erukalearn.iuh.edu.vn:8080 - domain của service Springboot; erukalearn.iuh.edu.vn:8000 – domain của service Apache PHP; Để các ứng dụng có thể giao tiếp frontend và backend với nhau. Tiếp theo chúng tôi sẽ tiến hành việc xác thực SSL – Chứng chỉ khóa xanh để đảm bảo trang web thực hiện chứng năng Socket. (Cám ơn Namecheap.com cũng tài trợ một SSL miễn phí 1 năm). Sau khi có domain cho Load Balancer là đã thành công việc cấu hình cân bằng tải và đã thành công một bước giải quyết bài toán lưu lượng truy cập cùng lúc tại một thời điểm tăng cao đột ngột. Load Balancer sẽ là người điều phối tải hoàn hảo giúp giải quyết vấn đề tắc nghẽn ở phía server. Việc còn lại là theo dõi hiệu quả Load Balancer thông qua các biểu đồ monitor có sẵn.

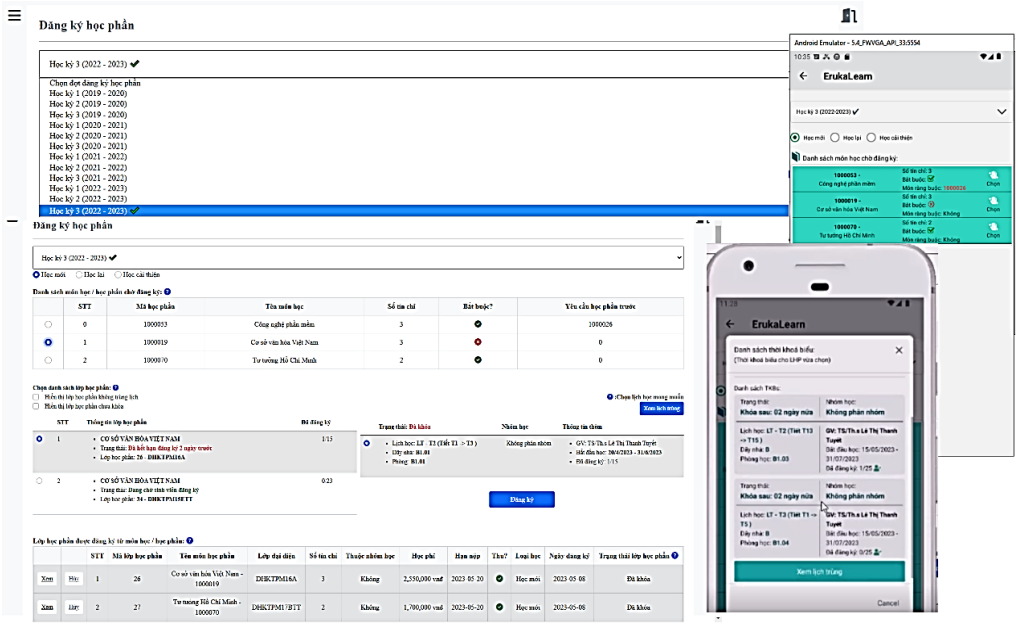
## 4.2 Triển khai trên hai nền tảng Web và App

Ứng dụng hỗ trợ học vụ được triển khai trên hai nền tảng là web và app (ứng dụng web và ứng dụng trên mobile; điện thoại di động cho cả Android và IOS). Trong đó web app sẽ được ưu tiên phát triển và triển khai đầu tiên tập trung vì đơn giản hơn. Ứng dụng web sẽ được triển khai lên cloud computing của AWS và được trỏ một tên miền đại diện vào. Hai nền tảng web và app sẽ được đồng bộ hóa dữ liệu (data) do tất cả nền tảng đều sử dụng chung một cơ sở dữ liệu khi thực hiện truy vấn, đọc ghi thông tin.
Nền tảng di động của ứng dụng này cung cấp tính năng linh hoạt, cho phép sinh viên truy cập thông tin học tập và quản lý nhiệm vụ của mình bất cứ lúc nào và ở bất kỳ đâu. Nền tảng web app cũng cung cấp trải nghiệm sử dụng tốt trên các thiết bị desktop hoặc app sẽ nhắc đến (Hình 11).



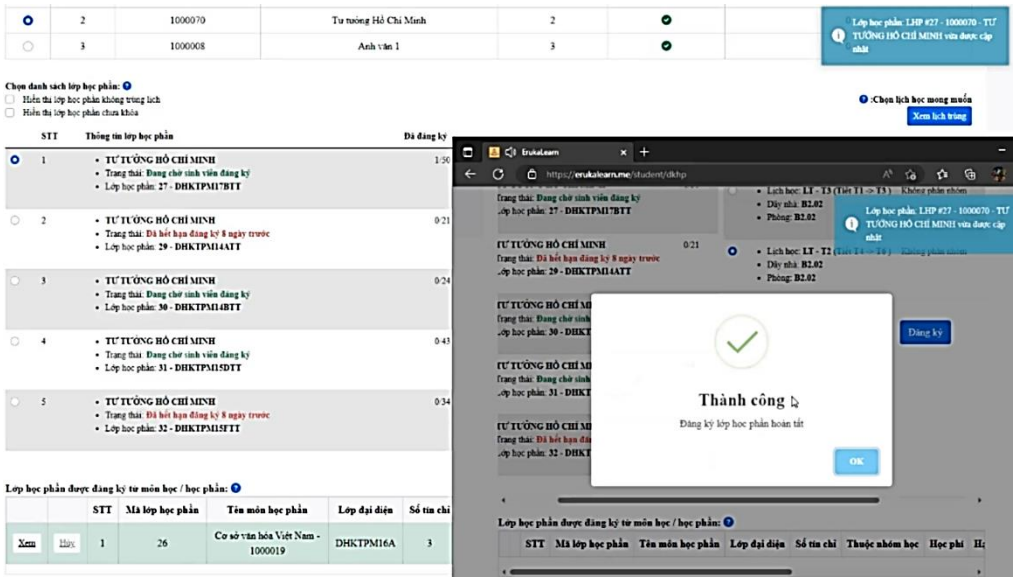

Hình 11: ErukaLearn đã triển khai trên hai flatform web và mobile đồng bộ data trên cloud

Trong giao diện đăng ký học phần, đầu tiên sinh viên sẽ chọn học kỳ cần đăng ký, học kỳ nào vừa được giáo vụ (nhân viên) mở thì sẽ được đánh tick xanh lá để sinh viên biết và chọn. Các hộp thoại của học kỳ cũ sẽ hiển thị các môn học đã mở ở quá khứ và các lớp học phần đã đăng ký tương đương cho học kỳ đó (Hình 12).




Hình 12: Giao diện đăng ký học phần trên 2 nền tảng

Việc đăng ký học phần đối với sinh viên vào những đợt mở lớp cũng là cần "chiến thuật". ErukaLearn cung cấp khả năng cập nhật sĩ số của một lớp học phần theo môn học mà sinh viên đang quan tâm realtime, giúp sinh viên có thể nắm bắt sĩ số mỗi lớp học phần ứng nhanh chóng mà không cần phải tải lại trang và chọn môn. Chúng tôi tin chức năng này sẽ hữu ích với sinh viên đang cần đăng ký nhiều môn và muốn sắp xếp thời gian hợp lý (Hình 13).

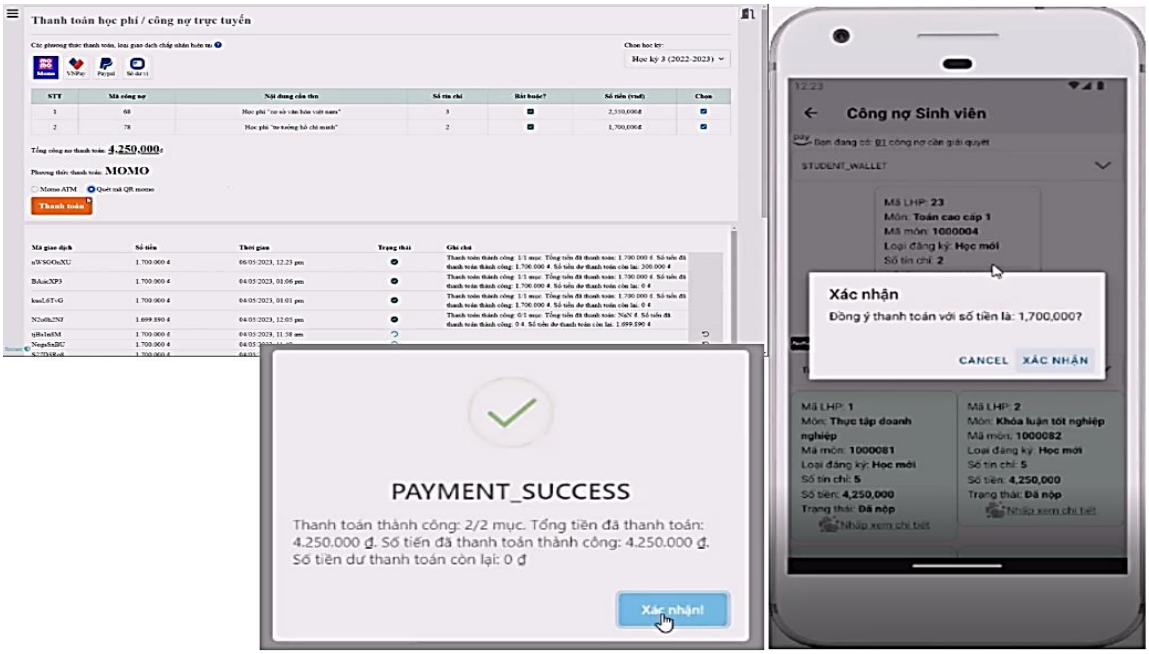



Hình 13: Sinh viên đăng ký học phần cập nhật realtime sĩ số lớp mà không cần tải lại trang

Tính năng thanh toán học phí đơn giản, dễ dàng, có thể thực hiện ở bất kì nơi nào, bất kì thiết bị nào gồm cả máy tính, laptop, điện thoại. Sau khi thực hiện một giao dịch điện tử số tiền lớn ErukaLearn sẽ từng bước

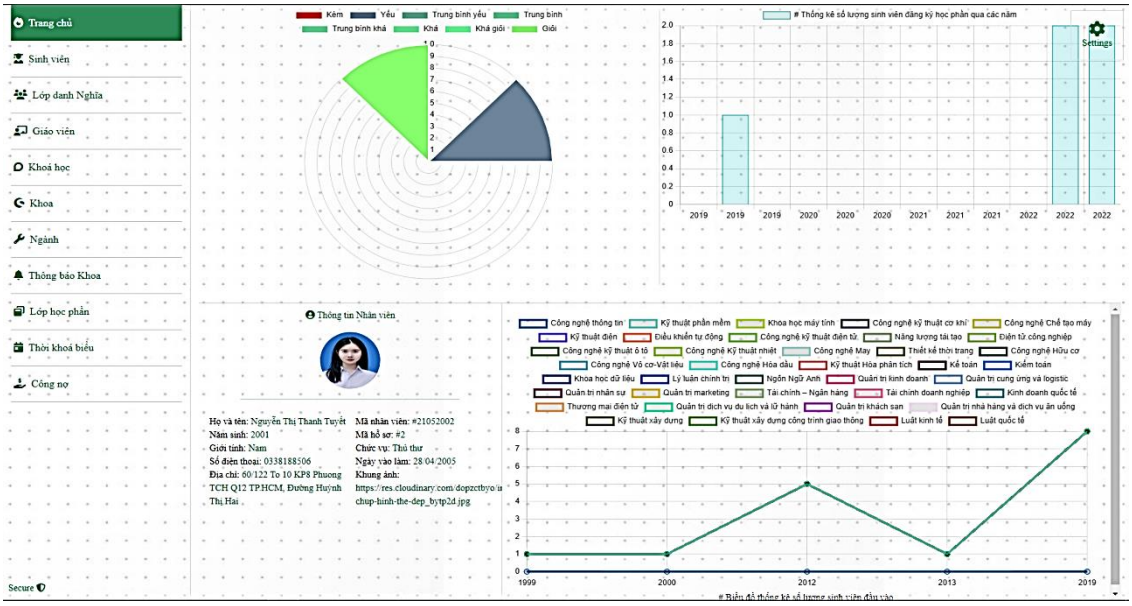
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

tạo ra từng tác vụ giúp người dùng tin tưởng hơn về giao dịch hệ thống như là: ghi nhận lịch sử giao dịch lại (Hình 13), tạo phiếu thu pdf (cho phép download), gửi email hóa đơn điện tử tới email của sinh viên, tạo thông báo giao dịch điện tử cá nhân. Để sinh viên có thể an tâm và tăng sự tin cậy hệ thống (Hình 14).




Hình 14: Giao diện thanh toán học phí online bằng ví MOMO sau khi đăng ký học phần

Đối với đối vai trò người Quản lý Nhân Viên (employee) có thể thống kê dữ liệu học vụ của sinh viên gồm: dữ liệu sinh viên đầu vào, mức đánh giá năng lực học tập chung của sinh viên toàn trường, số các ngành đầu vào sinh viên theo học,.. bằng các biểu đồ được tính toán trước để thu thập dữ liệu chung của toàn sinh viên (Hình 15).




Hình 15: Quản trị viên có thể quan sát dữ liệu học vụ sinh viên bằng các biểu đồ

## 5 KẾT LUẬN

Bài báo này chúng tôi đã trình bày hướng tiếp cận cho việc xây dựng một hệ thống hỗ trợ các nghiệp vụ học vụ cho sinh viên tại trường Đại học Công nghiệp TP.HCM với công nghệ cloud. Trong hệ thống này, chúng tôi xây dựng hệ thống gồm có hai phân hệ Web và ứng dụng di động (trên cả Android và iOS). Ứng dụng được tạo ra bằng cách tối ưu những chức năng cơ bản học vụ cơ bản đồng thời giải quyết vấn đề tắc nghẽn cấp thiết mà hệ thống cũ gặp phải bằng cách áp dụng dịch vụ Load balancer của AWS. Dịch vụ này thay thế cho máy chủ vật lý để xây dựng hệ thống giúp tiết kiệm chi phí, thời gian và nhiều lợi ích khác. Ngoài ra, chúng tôi còn giải quyết được vấn đề thanh toán học phí thủ công bằng cách tích hợp thanh toán online giao dịch điện tử thông qua các ví điện tử nội địa. Tạo ra một đơn vị tiền tệ mới trong hệ thống gọi là "Số Dư Ví" để lưu thông tiền giữa nhà trường và sinh viên. Về mặt kỹ thuật, phía back-end chúng tôi đã sử dụng công nghệ Docker triển khai ứng dụng đa người dùng (các trang sinh viên, giáo viên, quản lý đề là mỗi service của docker) nhằm tiết kiệm chi phí triển khai. Hệ thống cũng đã được triển khai trên tên miền thực với tên miền là ErukaLearn.me. Chúng tôi cũng đã kiểm chứng trên môi trường thực tế và hệ thống đảm bảo các yêu cầu chất lượng, bảo mật của phần mềm. Tuy nhiên hệ thống chỉ mới triển khai trên môi trường kiểm thử đơn giản và vẫn còn rất nhiều hạn chế vì chưa thể triển khai, kiểm thử toàn diện về hiệu năng trên môi trường thực. Trong tương lai, chúng tôi sẽ hoàn thiện các chức năng cũng như kiểm thử các chức năng trong môi trường thực tế tại trường Đại học Công nghiệp TP.HCM.

## TÀI LIỆU THAM KHẢO

[1] Amazon RDS (Sep 2021), https://www.techtarget.com/searchaws/definition/Amazon-Relational-Database-

[2] Service-RDS, accessed 03/4/2023

[3] Amazon Web Service (Jan 2020), https://aws.amazon.com/vi/rds/?nc2=h_ql_prod_db_rds, accessed 16/4/2023

[4] AWS Amplify, https://aws.amazon.com/vi/amplify/, accessed 05/4/2023

[5] Fast and Cost-Effective Image Manipulation with Serverless Image Handler (NOV 2020), https://aws.amazon.com/vi/blogs/architecture/fast-and-cost-effective-image-manipulation-with-serverless-image- handler/, accessed 22/5/2023

[6] Introduction to Amazon EC2 (Nov 2015), https://s3-us-west-2.amazonaws.com/uw-s3-cdn/wp- content/uploads/sites/149/2018/12/28193450/Curtis-Bray_Amazon_Introduction-to-Amazon-EC2.pdf, accessed 05/4/2023

[7] Introduction to AWS Simple Storage Service (Dec 2021), https://www.geeksforgeeks.org/introduction-to-aws-simple-storage-service-aws-s3/, accessed 18/02/2023

[8] Introduction to React Native Expo (Dec 2021), https://docs.expo.dev/tutorial/introduction/, accessed 18/02/2023

[9] Introduction to React Native Expo (Dec 2021), https://www.geeksforgeeks.org/mvc-framework-introduction/, accessed 18/02/2023

[10] Dobrean, Dragos, and Laura Diosan. "An Analysis System for Mobile Applications MVC Software Architectures." ICSOFT. 2019.

[11] ANDERSON, Charles. Docker software engineering. IEEE Software, 2015, vol 32(3).

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F022**

# TIẾP CẬN HỖ TRỢ CHẨN ĐOÁN UNG THƯ TỪ NHŨ ẢNH SỬ DỤNG PHƯƠNG PHÁP HỌC SÂU

LƯU LƯƠNG VI NHÂN[1*], QUÁCH XUÂN NAM[1]

[1]*Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
[*]*nhanvi212@gmail.com*

**Tóm tắt.** Ung thư là một trong những bệnh lý nguy hiểm và phổ biến trên toàn thế giới, và ung thư vú là loại ung thư thường gặp nhất ở nữ giới và cũng là nguyên nhân chính gây tử vong do ung thư. Việc xác định và phân loại chính xác khối u là rất quan trọng trong việc phát hiện ung thư vú ở giai đoạn đầu. Trong các phương pháp sàng lọc để phát hiện sớm tế bào ung thư vú, người ta đã công nhận rằng chụp nhũ ảnh là một phương pháp hiệu quả. Do đó, kỹ thuật phân đoạn hình ảnh và trí tuệ nhân tạo có thể được áp dụng để chẩn đoán ung thư vú giai đoạn đầu. Mô hình học sâu hiện đang được sử dụng để học các đặc trưng của nhũ ảnh, tự điều chỉnh và đạt được độ chính xác đáng kể. Mô hình được xây dựng dựa trên mạng nơ ron tích chập với kiến trúc là ConvNeXt và HRNet. Các kiến trúc này đã được chứng minh là hiệu quả trong nhiều bài toán thị giác máy tính, bao gồm cả phân đoạn hình ảnh và phân loại. ConvNeXt kết hợp kiến trúc hiệu quả của ResNeXt và mô hình Convolutional Neural Network (CNN) để tăng cường khả năng học tương quan giữa các đặc trưng trong ảnh. HRNet tập trung vào việc giữ nguyên độ phân giải cao trong suốt quá trình xử lý hình ảnh để tăng cường khả năng học các đặc trưng của mô hình. Mô hình được huấn luyện trên các bộ dữ liệu nhũ ảnh: RSNA (54706 ảnh), DDSM (7808 ảnh), CMMD (5202 ảnh), INbreast (410 ảnh), MIAS (322 ảnh). Trước khi đào tạo mô hình, chúng tôi đã áp dụng một số thuật toán tiền xử lý như chuẩn hóa dữ liệu, tăng cường ảnh, xoay và lật ảnh,… sau đó áp dụng mô hình và đạt được độ chính xác. Kết quả thử nghiệm chứng minh rằng phương pháp của chúng tôi đạt được kết quả tốt hơn so với các phương pháp dựa trên nền tảng CNN khác.

**Từ khóa.** Ung thư vú, Học sâu, xử lý ảnh, Mạng nơ-ron tích chập, ConvNeXt, HRNets, Nhũ ảnh.

# TOWARDS AN APPROACH TO DETECT BREAST CANCER FROM MAMMOGRAPHY USING DEEP LEARNING

**Abstract.** Cancer is one of the most dangerous and common diseases worldwide, and breast cancer is the most common cancer in women and is also the main cause of cancer death. Accurate tumor identification and classification are very important in detecting breast cancer at an early stage. Among the screening methods for early detection of breast cancer cells, it has been recognized that mammography is an effective method. Therefore, image segmentation techniques and artificial intelligence can be applied to diagnose early breast cancer. Deep learning models are currently being used to learn mammographic features, self-correct, and achieve remarkable accuracy. The model is built on a convolutional neural network with the architectures of ConvNeXt and HRNet. These architectures have been shown to be effective for many computer vision problems, including image segmentation and classification. ConvNeXt combines the efficient architecture of ResNeXt and the Convolutional Neural Network (CNN) model to enhance the ability to learn the correlation between features in the image. HRNet focuses on maintaining high resolution throughout the image processing to enhance the ability to learn the features of the model. The model was trained on the mammography datasets: RSNA (54706 images), DDSM (7808 images), CMMD (5202 images), INbreast (410 images), and MIAS (322 images). Before training the model, we applied some pre-processing algorithms such as data normalization, image enhancement, image rotation and flipping, etc., then applied the model and achieved accuracy. Experimental results demonstrate that our method achieves better results than other CNN-based methods.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**Keywords.** breast cancer, deep learning, image processing, convolutional neural network, ConvNeXt, HRNets, mammography.

## 1 GIỚI THIỆU

Trong lĩnh vực y tế, ung thư là một trong những căn bệnh phức tạp và nguy hiểm nhất đối với con người bởi khả năng xâm lấn, lây lan đến các bộ phận của cơ thể. Theo thống kê toàn cầu về ung thư, ung thư vú chiếm tỷ lệ cao nhất trong số các loại ung thư phát hiện ở phụ nữ trên toàn thế giới. Theo Hiệp hội Ung thư Hoa Kỳ thì năm 2022 có tới 1,919,030 ca ung thư mới và 609,360 ca tử vong tính riêng ở Hoa Kỳ. Ung thư vú chiếm tới 19% trong tổng số ca ung thư và chiếm 30% trong số ca ung thư ở phụ nữ [1]. Ung thư vú phát sinh do sự phát triển bất thường của các tế bào trong mô vú, tạo nên các tổn thương. Các tổn thương lành tính (benign) thường vô hại và không cần đến sinh thiết. Ngược lại các tổn thương ác tính (malignant) thì không ổn định, trong trường hợp di căn, tế bào ung thư trong vú có thể tách khỏi khối u ban đầu và lưu thông qua hệ thống mạch máu hoặc mạch lympho đến các bộ phận khác của cơ thể [2]. Vì vậy việc phát hiện ung thư vú ở giai đoạn sớm giúp tăng khả năng chữa khỏi bệnh, giảm nguy cơ xuất hiện các biến chứng và giảm mức độ ảnh hưởng của bệnh với chất lượng cuộc sống của bệnh nhân.

Mặc dù đã có sự tiến bộ về công nghệ và phương pháp chẩn đoán ung thư vú, nhưng việc đánh giá hình ảnh y tế là một công việc tốn thời gian và tốn sức, và đòi hỏi kiến thức chuyên môn cao. Chúng tôi đã áp dụng các kỹ thuật trí tuệ nhân tạo để hỗ trợ chẩn đoán phân loại ung thư vú. Các mô hình học sâu như CNNs, ConvNeXt, HRNets được xây dựng dựa trên dữ liệu hình ảnh của tuyến vú giúp phát hiện bệnh nhân có ung thư hay không, giúp các bác sĩ xác định những trường hợp cần phải tiếp tục xét nghiệm và chẩn đoán chính xác hơn… Các mô hình này có thể giúp tăng độ chính xác và giảm thời gian kiểm tra, đồng thời giúp giảm thiểu những sai sót trong quá trình chẩn đoán.

## 2 NGHIÊN CỨU LIÊN QUAN

Về cơ bản có hai kỹ thuật được sử dụng để phát hiện ung thư vú, đó là học máy (machine learning) và học sâu (deep learning). Học máy là một lĩnh vực của trí tuệ nhân tạo tuy nhiên học máy hiệu quả đối với lượng dữ liệu nhỏ, khi có nhiều dữ liệu hơn thì kết quả mô hình cũng không cải thiện hơn. Trong khi đó các thuật toán học sâu sẽ cho ra kết quả tốt hơn khi được đào tạo trên dữ liệu chất lượng cao theo số lượng lớn.

- Anusha Bharat và cộng sự [3] đề xuất mô hình dùng bốn thuật toán phân loại là SVM, Decision Tree (CART), KNN, Naive Bayes. Theo tác giả thì KNN cho kết quả tốt hơn. SVM có một hạn chế đó là cho kết quả tốt hơn đối với bài toán phân loại nhị phân. Đó là lý do Multi-SVM được sử dụng.
- Ch. Shravya và cộng sự [4] đề xuất mô hình học máy có giám sát. Nghiên cứu này được thực hiện sử dụng các thuật toán học máy phân loại có giám sát như Logistic Regression, SVM và KNN. Dữ liệu được tải từ UCI và cho thấy thuật toán SVM đạt kết quả cho tới 92.7%, với nền tảng sử dụng là ngôn ngữ python.
- Sivapriya J và cộng sự [5] đề xuất sử dụng học máy sử dụng các thuật toán phân loại là Random Forest, SVM, Logistic và Naive Bayes. Thực hiện trên môi trường anaconda và cho thấy rằng Random Forest là 1 bộ phân loại tốt với độ chính xác tới 99.76%.
- Kalyani Wadkar và cộng sự [6] sử dụng mô hình ANN cơ bản và hiệu suất phân tích bằng với mô hình SVM. Kết quả cho thấy ANN có độ chính xác là 97% còn SVM chỉ 91%. Tác giả kêu rằng nếu ko có SVM thì ANN sẽ cho kết quả tốt hơn.
- Vishal Deshwal và cộng sự [7]: đầu tiên tác giả chỉ nghiên cứu sử dụng SVM và sau đó là SVM kết hợp Grid search. Kết quả so sánh cho thấy nếu có Grid search thì SVM sẽ hiệu quả hơn.
- S. Shamy và cộng sự [8]: sử dụng Kmean - GMM và CNN. Đầu tiên tác giả tìm phần ROI sau đó áp dụng phương pháp trích xuất đặc trưng, cuối cùng là sử dụng CNN để đạt kết quả tốt nhất. Kết quả đạt độ chính xác 95.8% trên bộ MIAS.

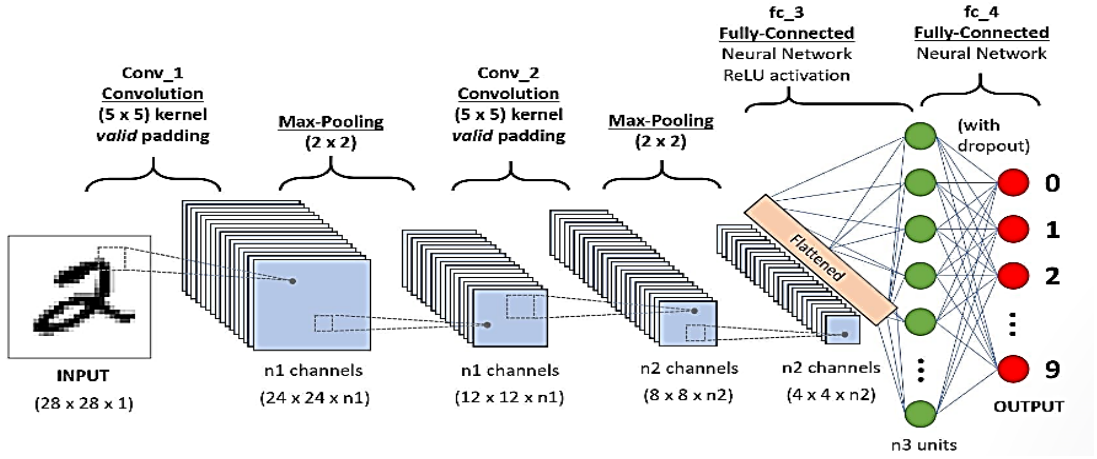
## 3 MÔ HÌNH

### 3.1 Mô hình CNNs

CNNs/ConvNet là một thuật toán học sâu nhận dữ liệu đầu vào (input) là ảnh. Trong mạng thần kinh tích chập có các lớp chuyên dụng để xử lý dữ liệu ảnh như lớp tích chập (Convolutional), lớp tổng hợp (Pooling),

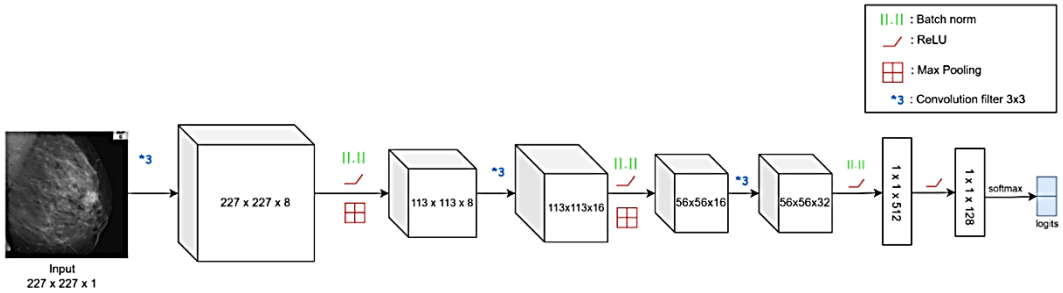
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

lớp kích hoạt (Activation) và lớp kết nối đầy đủ (Fully connected). Lớp tích chập có thể tìm kiếm các đặc trưng quan trọng của ảnh thông qua tính toán nhờ các bộ lọc (filter), lớp Pooling để tổng hợp chỉ lấy các đặc trưng quan trọng nhất và các lớp activation như ReLU có thể giúp tăng tính phi tuyến của mô hình và học được các đặc trưng phức tạp của ảnh [9].




Hình 1: Minh họa kiến trúc CNNs giải quyết bài toán phân loại chữ viết tay.

Trong bài báo này, nhóm chúng tôi sử dụng kiến trúc CNNs được tham khảo từ [10] gồm 3 lớp tích chập, hai lớp max pooling, 5 lớp kích hoạt ReLU, 3 lớp chuẩn hóa và 3 lớp fully connected. Mỗi layer chúng tôi đều sử dụng stride bằng 2, kích thước 2x2 cho mỗi lớp max pooling. Với mỗi lớp convolution, chúng tôi đều sử dụng stride bằng 1, kích thước kernel 3x3 và có padding 'same' để trích xuất đặc trưng của ảnh. Theo sau đó là lớp fully connected với số nút lần lượt là 512, 128 sử dụng hàm kích hoạt ReLU. Cuối cùng là lớp softmax để trả về logits dự đoán các lớp. Kiến trúc này được rút ra từ thực nghiệm của nhóm tác giả, chi tiết được thể hiện dưới hình sau:

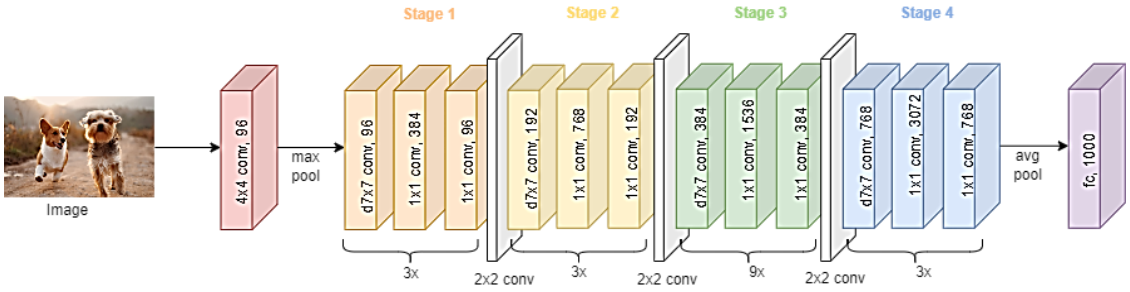



Hình 2: Kiến trúc CNNs sử dụng cho bài toán chẩn đoán nhũ ảnh.

## 3.2 Mô hình ConvNeXt

ConvNeXt đã đạt độ chính xác cao trên tập ImageNet (87,8% - top 1 về độ chính xác) cũng như hiệu quả trong các tác vụ như phát hiện đối tượng trên tập COCO, phân đoạn hình ảnh trên tập ADE20K, đồng thời vẫn giữ được sự đơn giản và hiệu quả của các mô hình ConvNet. Điều này cho thấy ConvNeXt là một giải pháp hữu hiệu để giải quyết các vấn đề trong thị giác máy tính và cải thiện hiệu suất mô hình.

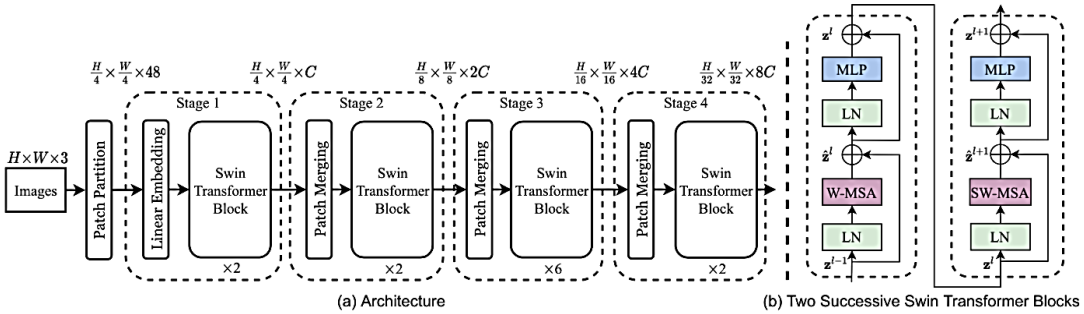
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 3: Kiến trúc ConvNeXt (trong đó bỏ qua không miêu tả các skip connection). [11]

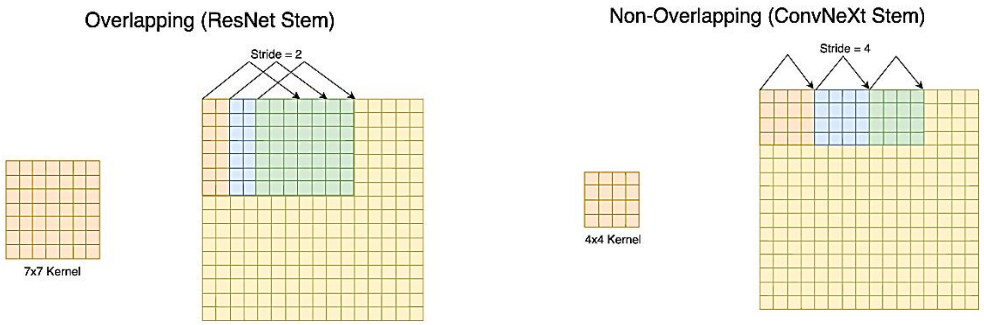
Những cải tiến của ConvNeXt so với ResNet50 khiến mô hình ConvNeXt trở nên vượt trội (tổng quan độ chính xác tăng đến 3.2%):

- Về kĩ thuật training: Mô hình sử dụng hàm tối ưu là AdamW, với các kỹ thuật tăng cường ảnh như Mixup, Cutmix, RandAugment, Random Erasing, regularization schemes bao gồm Stochastic Depth và Label smoothing.
- Điều chỉnh tỉ lệ giữa các stage.




Hình 4: Kiến trúc Swin-Transformer tiêu chuẩn. [12]

- Thay đổi phương pháp "stem" thành "patchify": Patchify là một phương pháp xử lý ảnh trong đó ảnh được chia thành các phần bằng nhau (gọi là các "patch") và mỗi patch được xử lý độc lập. Thay vì sử dụng lớp tích chập với một kernel lớn để xử lý toàn bộ ảnh (stem), Patchify sử dụng các kernel nhỏ hơn để xử lý các patch riêng lẻ.



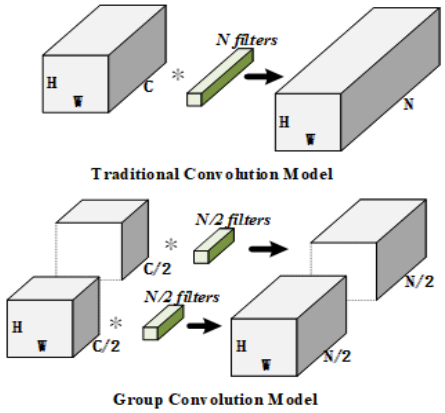

Hình 5: Minh họa về phương pháp 'stem' (Resnet Stem) và patchify (ConvNeXt stem).

Đối với mô hình ResNet50 truyền thống, "stem cell" bao gồm một lớp tích chập 7x7 với stride bằng 2, theo sau là một lớp max pool, khiến cho kích thước ảnh giảm xuống 4 lần. Nhóm tác giả đã thay thế phương pháp stem của ResNet bằng phương pháp patchify với kernel size là 4, stride bằng 4 giống như của Swin Transformer sử dụng. Kết quả cho thấy độ chính xác mô hình thay đổi từ 79.4% lên 79.5%.

- Lấy ý tưởng từ ResNeXt: Nhóm tác giả thay đổi lớp tích chập 3x3 bằng lớp tích chập sâu theo chiều sâu (depthwise convolution). Lớp tích chập sâu theo chiều sâu thực chất là lớp tích chập nhóm

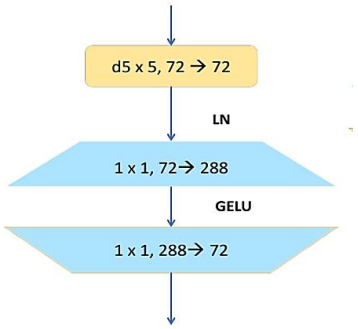
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

(grouped convolution), trong đó số nhóm bằng số kênh của đầu vào. Thay thế lớp tích chập bằng lớp tích chập sâu giúp tăng độ chính xác của mô hình là 80.5% (tăng thêm 1,0%) và tăng GFLOPS thêm 0,9%.



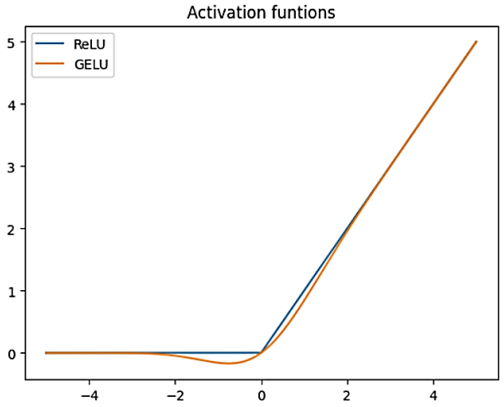

Hình 6: Minh họa về group convolution. [13]

- Inverted bottleneck, nghĩa là thay đổi kích thước của các lớp tích chập (convolutional layers) trong một khối để khối ở giữa có kích thước lớn nhất. Điều thú vị là, điều này giúp tăng độ chính xác của mô hình lên từ 80.5% thành 80.6%.




Hình 7: Minh họa một khối inverted bottleneck. [14]

- Kích thước kernel tăng từ 3x3 thành 7x7 (giống với Swin Transformer).
- Sử dụng hàm kích hoạt GeLU thay vì ReLU. Hàm GELU (Gaussian Error Linear Unit) là một biến thể của hàm ReLU. Hàm GELU khá mượt mà và có tính chất khá tương đồng với hàm sigmoid, với giá trị đầu ra gần với 0 hoặc 1 khi đầu vào rất nhỏ hoặc rất lớn. Điều này giúp cho hàm GELU có thể học các mối quan hệ phức tạp giữa các đặc trưng đầu vào.




Hình 8: Minh họa hàm ReLU và GELU.

Công thức của hàm GELU [15]:

$$g = \frac{1}{2}x\left(1+\frac{2}{\sqrt{\pi}}\int_0^{\frac{x}{\sqrt{2}}} e^{-t^2}dt\right) \quad (1)$$

Có thể thực hiện xấp xỉ GELU bằng công thức sau:

$$g = \frac{1}{2}x\left(1+\tanh\left[\sqrt{\frac{2}{\pi}}\left(x+0.0447155x^3\right)\right]\right) \quad (2)$$

- Giảm số lượng hàm kích hoạt, chỉ giữ lại hàm kích hoạt sau inverted bottleneck. Độ chính xác +0.7%.
- Giảm số lượng lớp chuẩn hóa, chỉ giữ lại một lớp chuẩn hóa trước inverted bottleneck. Độ chính xác tăng 0.1%.
- Thay thế toàn bộ normalization (BN) thành Layer normalization (LN) +0.1%.
- Thêm các lớp downsampling vào giữa các stages của mạng Swin Transformer. Cụ thể, lớp downsampling này gồm một lớp chuẩn hóa (normalization), một lớp convolution với kích thước 2x2, bước nhảy (stride) bằng 2 và một lớp chuẩn hóa nữa. Ngoài ra, thêm lớp chuẩn hóa trước giai đoạn đầu tiên và sau khi áp dụng lớp average pool sau stage 4. Kết quả là độ chính xác của mô hình được cải thiện thêm 0.5%.

Về tổng quan, có các kiến trúc ConvNeXt theo kích thước từ nhỏ tới lớn được giới thiệu dưới đây:

- ConvNeXt-tiny (ConvNeXt-T): C = (96, 192, 384, 768), B = (3, 3, 9, 3).
- ConvNeXt-small (ConvNeXt-S): C = (96, 192, 384, 768), B = (3, 3, 27, 3).
- ConvNeXt-base (ConvNeXt-B): C = (128, 256, 512, 1024), B = (3, 3, 27, 3).
- ConvNeXt-large (ConvNeXt-L): C = (192, 384, 768, 1536), B = (3, 3, 27, 3).
- ConvNeXt-Xtra large (ConvNeXt-XL): C = (256, 512, 1024, 2048), B = (3, 3, 27, 3).

Nhóm đã fine-tune sử dụng trọng số đã qua đào tạo của mô hình "convnext_small.fb_in22k_ft_in1k_384" sử dụng API của HuggingFace. Mô tả thông số mô hình:

Bảng 1: Thông số mô hình convnext_small.fb_in22k_ft_in1k_384

| | |
|---|---|
| **Thông số mô hình** | **50.2 triệu parameters**<br>GMACs: 45.2<br>Size ảnh mặc định: 384x384 |
| **Dữ liệu fine-tune** | ImageNet-1k |
| **Dữ liệu pretrained** | ImageNet-22k |

Vì mô hình sử dụng cho bài toán phân loại 1000 lớp, nên để sử dụng mô hình cho bài toán phân loại nhị phân, nhóm chúng tôi đã sửa đổi lớp Linear ở cuối với đầu ra bằng 2.

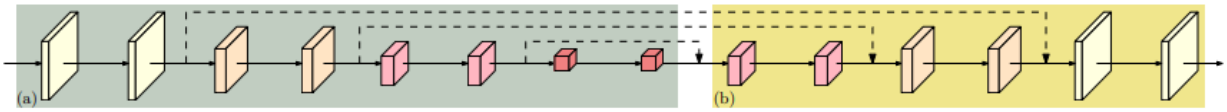
### 3.3 Mô hình High – Resolution Networks (HRNets)

Hầu hết mạng phân loại như Alexnet, VGGNet, GoogleNet, ResNet tuân theo quy tắc của Lenet (mô tả trong hình 9), tức là giảm dần kích thước không gian của các feature map, sau đó kết nối các tích chập từ độ phân giải cao đến độ phân giải thấp, sau đó mới được xử lý thêm bớt để phân loại.

Mạng độ phân giải cao (HRNets) có khả năng duy trì các phản hồi có độ phân giải cao trong toàn bộ quá trình. Bắt đầu từ luồng tích chập có độ phân giải cao, dần dần thêm từng luồng có độ phân giải thấp và kết nối các luồng một cách song song. Do đó HRNets duy trì độ phân giải cao khi chuyển ảnh qua mạng, chứ không phục hồi độ phân giải như ý tưởng của top-down. Trước đây thường sử dụng phép Upsampling đặc trưng độ phân giải thấp lên cho bằng đặc trưng có độ phân giải cao và kết hợp lại bằng các phép cộng hoặc

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

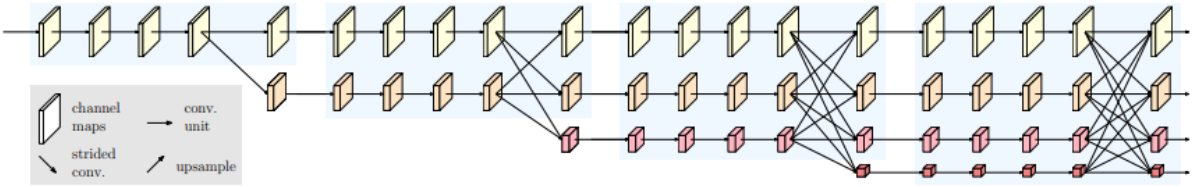
phép nối; HRNets sẽ thực hiện điều này nhiều lần, qua nhiều giai đoạn để các đặc trưng có độ phân giải cao, thấp sẽ bổ sung ngữ nghĩa cho nhau.




Hình 9: Cấu trúc phục hồi độ phân giải cao từ độ phân giải thấp.

**Các tích chập song song đa độ phân giải**

HRNets sẽ duy trì độ phân giải cao xuyên suốt quá trình ảnh truyền qua mạng. Kiến trúc sẽ chia thành nhiều tầng, tầng phía sau sẽ sinh ra đặc trưng có độ phân giải thấp và sâu hơn, tuy nhiên vẫn duy trì đặc trưng có độ phân giải ở tầng trước đó.




Hình 10: Minh họa mạng HRNets.

Quan sát ở luồng số 1, qua nhiều tầng, đặc trưng độ phân giải cao (ký hiệu màu vàng) sẽ được duy trì. Bắt đầu từ tầng số 2, đặc trưng có độ phân giải thấp và sâu hơn (màu cam) sẽ được sinh ra, đặc trưng có độ phân giải cao màu vàng này sẽ được bổ sung ngữ nghĩa bởi các đặc trưng màu cam và ngược lại. Cứ như vậy qua nhiều tầng, các đặc trưng có các độ phân giải khác nhau sẽ được bổ sung ngữ nghĩa cho nhau, dẫn đến cuối cùng ta sẽ thu được đặc trưng độ phân giải cao, tuy nhiên chất lượng vẫn tốt. Hình 10 minh họa mạng HRNets bao gồm 4 giai đoạn. Luồng xử lý chính được mô tả một cách logic bằng công thức sau:

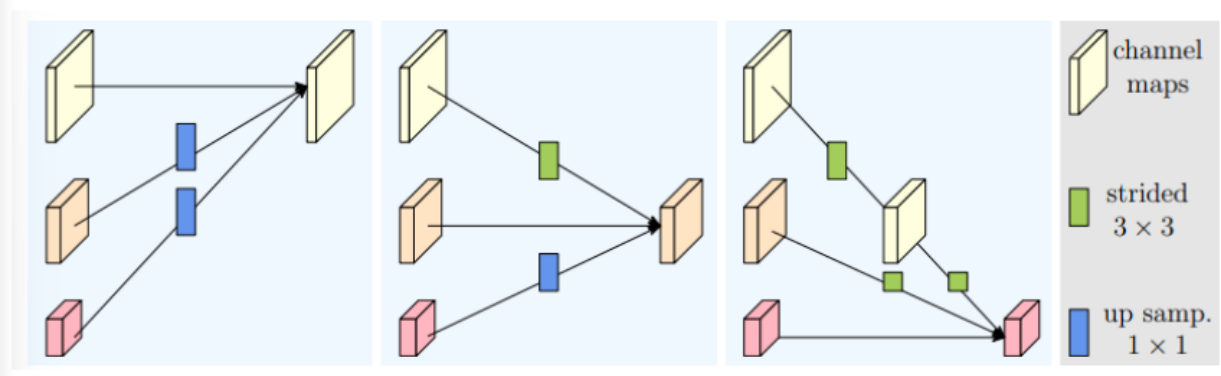
$$\begin{aligned} &\mathcal{N}_{11} \to \mathcal{N}_{21} \to \mathcal{N}_{31} \to \mathcal{N}_{41} \\ &\quad \searrow \mathcal{N}_{22} \to \mathcal{N}_{32} \to \mathcal{N}_{42} \\ &\quad\quad \searrow \mathcal{N}_{33} \to \mathcal{N}_{43} \\ &\quad\quad\quad \searrow \mathcal{N}_{44} \end{aligned}$$

Hình 11: Công thức luồng xử lý mạng HRNets.

Trong đó, $N_{sr}$ là một nhánh đặc trưng tại giai đoạn thứ $s$, và $r$ là chỉ mục độ phân giải. Độ phân giải ở giai đoạn đầu tiên sẽ là $r=1$, độ phân giải thứ $r$ sẽ bằng $\frac{1}{s^r-1}$ độ phân giải ở giai đoạn đầu tiên.

**Các phép kết hợp đặc trưng**

Việc kết hợp các đặc trưng là để trao đổi thông tin ngữ nghĩa giữa các đặc trưng có nhiều độ phân giải khác nhau. Nó được lặp đi, lặp lại nhiều lần. Ví dụ như ở hình 10, việc lặp đi lặp lại được thực hiện 4 lần.




Hình 12: Minh họa cách kết hợp thông tin ngữ nghĩa các đặc trưng độ phân giải cao, trung bình và thấp tương ứng từ trái sang phải.

Quan sát hình 12, bắt đầu từ luồng tích chập có độ phân giải cao làm giai đoạn đầu tiên, dần dần thêm từng luồng có độ phân giải cao xuống thấp, tạo thành các giai đoạn mới và kết nối song song các luồng đa độ phân giải. Do đó, độ phân giải cho các luồng song song của giai đoạn sau bao gồm độ phân giải từ giai đoạn trước và một mức thấp hơn.

**Các dạng biểu diễn**

HRNets có 3 dạng biểu diễn, được gọi là HRNetV1, HRNetV2, HRNetV2p

- HRNetV1: đầu ra là dạng biểu diễn chỉ lấy từ nhánh đặc trưng có độ phân giải cao (high-resolution). Ba đặc trưng của ba độ phân giải còn lại sẽ được lược bỏ.
- HRNetV2: các tác giả rescale các đặc trưng độ phân giải thấp lên độ phân giải cao bằng một phép upsampling song tuyến mà không thay đổi chiều sâu, và nối lại cả bốn đặc trưng. Sau đó đi qua một lớp tích chập 1×1×1 để trộn cả bốn đặc trưng này lại.
- HRNetV2p: các tác giả xây dựng một biểu diễn đa tỷ lệ (multi-level features) bằng cách sử dụng đầu ra từ HRNetV2.
- Đối với bài toán phân loại ảnh ung thư, chúng tôi sử dụng kiến trúc HRNet được tham khảo ở [16]. Ở lớp fully connected, chúng tôi tiến hành thay đổi đầu ra (output features) bằng 2 để phù hợp với bài toán.

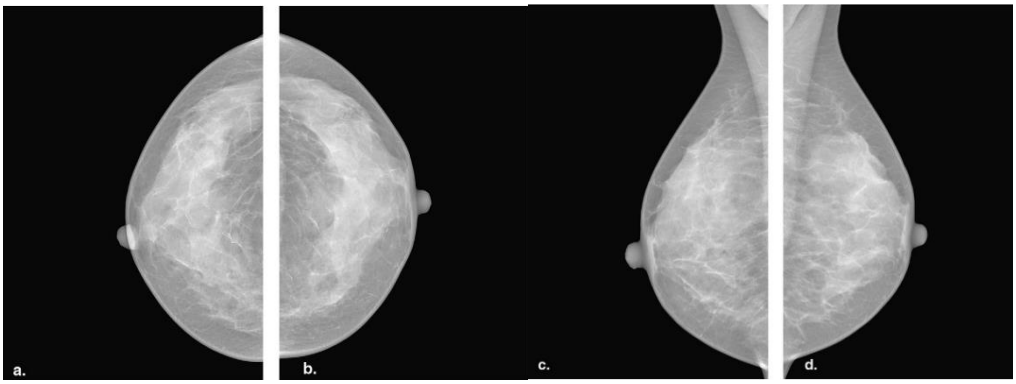
## 4 ĐÁNH GIÁ

### 4.1 Dữ liệu

Chúng tôi sử dụng 5 bộ dữ liệu cho bài toán của mình bao gồm INbreast, Mammogram Image Analysis Society (MIAS), Digital Dataset for Screening Mammography (DDSM), RSNA Screening Mammography Breast Cancer Detection, Chinese Mammography Database (CMMD).

Bộ dữ liệu INbreast có tổng cộng 115 trường hợp (410 ảnh), trong đó có 90 trường hợp được thu thập cả 2 vú (mỗi trường hợp gồm 4 ảnh) và 25 trường hợp là từ bệnh nhân cắt bỏ tuyến vú (2 ảnh cho mỗi trường hợp). Thông tin bao gồm mô tả chi tiết tổn thương (masses), vôi hóa (calcifications), không đối xứng (asymmetries) và biến dạng (distortions). Ma trận hình ảnh có kích thước 3328 x 4084 hoặc 2560 x 3328 pixels. Bộ dữ liệu MIAS bao gồm 322 hình ảnh ở định dạng Portable Gray Map (PGM). Bộ dữ liệu chứa 207 trường hợp bình thường, 63 trường hợp lành tính và 52 trường hợp ác tính. Kích thước của hình ảnh là 1024 × 1024. Bộ dữ liệu RSNA (Radiological Society of North America) là một tập dữ liệu quan trọng được tổ chức và phát triển bởi Hội Nghị Y Học Hình Ảnh Bắc Mỹ (Radiological Society of North America), nhằm hỗ trợ nghiên cứu và phát triển các thuật toán và hệ thống hỗ trợ chẩn đoán ung thư vú từ hình ảnh mammography. Trong đó, các hình ảnh được lưu dưới dạng DICOM, có 53540 ảnh không ung thư và 1158 ảnh ung thư. Bộ dữ liệu DDSM (Digital Database for Screening Mammography) có định dạng DICOM chứa hình ảnh của các khối u (masses), vôi hóa (calcifications) và hình ảnh ROI. Tập dữ liệu có chứa 2716 ảnh ung thư ác tính, 2684 ảnh lành tính và 2408 ảnh bình thường. Bộ dữ liệu CMMD (Chinese Mammography Database) bao gồm các hình ảnh chụp nhũ ảnh được thu thập ở định dạng .TIFF và được chuyển đổi sang DICOM 8 bit, bao gồm 5202 ảnh trong đó có 4094 là ung thư ác tính và 1108 ảnh là ung thư lành tính.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 13: Nhũ ảnh trong bộ dữ liệu INbreast: a) Góc chụp CC vú phải, b) Góc chụp CC vú trái, c) Góc chụp MLO vú phải, d) Góc chụp MLO vú trái.

Đầu tiên, chúng tôi chuyển tất cả hình ảnh trong bộ dữ liệu về chung một định dạng ảnh là png. Tiếp theo, sử dụng mô hình YOLOv8 để cắt riêng vùng vú loại bỏ phần ảnh đen dư thừa gây nhiễu, với độ chính xác của cắt ảnh lên đến 99% theo thang đo mAP. Các ảnh sau khi cắt sẽ được tiền xử lý bằng các phương pháp như: CLAHE (Contrast limited adaptive histogram equalization) tăng độ tương phản ảnh, các kĩ thuật tăng cường ảnh (xoay, lật, làm mờ ảnh). Sau đó đưa vào mô hình để giải quyết bài toán phân loại vú ung thư và vú không ung thư.

Dữ liệu thực nghiệm với mô hình ConvNeXt và CNNs

Trong quá trình thực nghiệm nhiều lần, đối với mô hình ConvNeXt và CNNs, nhóm chúng tôi thực hiện 2 cách chia dữ liệu như sau:

- Cách 1 (TrainSet1): Chỉ sử dụng cố định 4 tập dữ liệu mini-DDSM, Mias, INbreast và CMMD để huấn luyện. Tập dữ liệu huấn luyện bao gồm 13056 ảnh trong đó tỉ lệ ảnh 1: 0 là 0.52: 0.48. Số ảnh validate lấy từ mỗi bộ 5% số ảnh, tỉ lệ tương tự bằng cách lấy sample. Số ảnh test gồm 54703 ảnh của bộ RSNA (bảng 2) và test riêng từng bộ dữ liệu:

Bảng 2: Chia dữ liệu TrainSet1.

| Dataset | Tổng ảnh | Ảnh ung thư | Train | Validate | Test |
|---|---|---|---|---|---|
| **DDSM** | 7808 | 2716 | 7417 | 390 | 7808 |
| **Mias** | 322 | 51 | 306 | 16 | 322 |
| **INbreast** | 410 | 343 | 390 | 20 | 410 |
| **CMMD** | 5202 | 4094 | 4945 | 260 | 5202 |
| **RSNA** | 54703 | 1158 | 0 | 0 | 54703 |
| **Tổng** | 68445 | 7204 | 13056 | 686 | 68445 |

- Cách 2 (TrainSet2): Sử dụng 5 tập dữ liệu, gồm mini-DDSM, Mias, INbreast, CMMD và RSNA để huấn luyện. Riêng với bộ RSNA chỉ sử dụng 600 ảnh mang nhãn 0.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Bảng 3: Chia dữ liệu TrainSet2.

| Dataset | Tổng ảnh | Ảnh ung thư | Train | Validate | Test |
|---|---|---|---|---|---|
| **DDSM** | 7808 | 2716 | 7417 | 390 | 7808 |
| **Mias** | 322 | 51 | 306 | 16 | 322 |
| **INbreast** | 410 | 343 | 390 | 20 | 410 |
| **CMMD** | 5202 | 4094 | 4945 | 260 | 5202 |
| **RSNA** | 54703 | 1158 | 600 | 0 | 54703 |
| **Tổng** | 68445 | 7204 | 13656 | 686 | 68445 |

**Dữ liệu thực nghiệm với mô hình HRNet**
Dữ liệu huấn luyện

Bảng 4: Dữ liệu huấn luyện mô hình HRNet.

| Tên Dataset | Số lượng nhãn Cancer | Số lượng nhãn Normal | Tổng số ảnh |
|---|---|---|---|
| **DDSM** | 2172 | 4073 | 6245 |
| **Mias** | 40 | 216 | 256 |
| **INbreast** | 274 | 53 | 327 |
| **Tổng số** | **2486** | **4342** | **6828** |

Dữ liệu đánh giá

Bảng 5: Dữ liệu đánh giá mô hình HRNet.

| Tên Dataset | Số lượng nhãn Cancer | Số lượng nhãn Normal | Tổng số ảnh |
|---|---|---|---|
| **DDSM** | 554 | 1019 | 1573 |
| **Mias** | 11 | 55 | 66 |
| **INbreast** | 69 | 14 | 83 |
| **Tổng số** | **634** | **1088** | **1722** |

## 4.2 Đánh giá mô hình

a. Kết quả đánh giá chung:

Bảng 6: Bảng kết quả đánh giá các mô hình trên tập huấn luyện và tập kiểm thử.

| Mô hình | Tập dữ liệu huấn luyện | | | | | Tập dữ liệu kiểm thử | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Acc | Precision | F1 score | Recall | loss | Acc | F1 score | Recall | Precision |
| **ConvNeXt** | **92.85%** | **92.73%** | **93.05%** | **93.38%** | **0.3951** | **79.15%** | **80.65%** | **82.32%** | **79.05%** |
| CNN | 91.93% | - | 92.21% | - | 0.3996 | 77.25% | 77.87% | - | - |
| HRNet | 74.37% | 68.91% | 60.51% | 53.94% | 0.04 | 71.2% | 55.78% | 49.83% | 63.34% |


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

b. Kết quả độ chính xác trên từng bộ dữ liệu:

Bảng 7: Bảng kết quả đánh giá độ chính xác các mô hình trên từng tập dữ liệu.

| Tên mô hình | Mias | Inbreast | DDSM | CMMD |
|---|---|---|---|---|
| CNN | 80.74% | 51.46% | 64.34% | 72.16% |
| **ConvNeXt** | **95.03%** | **90.97%** | **93.28%** | **90.00%** |
| HRNet | 66.77% | 41.22% | 65.3% | 43.98% |
| CNN (kết quả bài báo [19]) | 95.95% | 96.3% | 99.39% | - |

## 5 KẾT LUẬN

Trong bài báo này, chúng tôi đã thử nghiệm với 3 mô hình khác nhau là: CNN, HRNet, ConvNeXt để hỗ trợ tiếp cận chẩn đoán ung thư từ nhũ ảnh sử dụng các bộ dữ liệu huấn luyện là Mias, DDSM, INbreast. So với bài báo gốc, phương pháp của chúng tôi đã có sự cải thiện trên 2 bộ dữ liệu là Mias và INbreast. Nhóm chúng tôi nhận thấy rằng, cách tiếp cận này phù hợp với bài toán và có ích trong việc chẩn đoán ung thư, giúp giảm thiểu thời gian và nguồn nhân lực của đội ngũ y tế.

Qua quá trình làm, nhóm chúng tôi cũng đã rút ra được nhiều kinh nghiệm quý báu trong bài toán phân loại ung thư từ nhũ ảnh nói riêng cũng như xử lý ảnh nói chung. Hiện tại, bài báo này chưa đạt được kết quả thực sự tốt. Trong tương lai gần, nhóm chúng tôi sẽ tiếp tục khắc phục và cải tiến đề tài để mang lại kết quả đáng kể nhất.

## TÀI LIỆU THAM KHẢO

[1] D. Tính, "Tỷ lệ mắc ung thư mới và tử vong do ung thư tiếp tục tăng," 1-12, 2022. [Trực tuyến]. Available: Tỷ lệ mắc ung thư mới và tử vong do ung thư tiếp tục tăng (thanhnien.vn)

[2] Tham vấn y khoa: BSCKI. Vũ Thanh Tuấn, Ngày 18/08/2022. Available: https://medlatec.vn/tin-tuc/goc-giai-dap-u-vu-lanh-tinh-va-ac-tinh-khac-nhau-nhu-the-nao-s91-n29587

[3] Anusha bharat, Pooja N, R Anishka Reddy, "Using machine learning algorithms for breast," IEEE 3rd International Conference on Circuits, Control, , pp. 1-4, 2018.

[4] Ch. Shravya, K. Pravalika, ShaikSubhani, "“Prediction of breast cancer using supervised machine learning techniques," International Journal of Innovative Technology and Exploring Engineering, vol. 8, pp. 1106-1110, 2019.

[5] Sivapriya J, Aravind Kumar V, Siddarth Sai S, Sriram S, "Breast cancer prediction using machine learning," International Journal of Recent Technology and Engineering, vol. 8, pp. 4879-4881, 2019.

[6] Kalyani Wadkar, Prashant Pathak, Nikhil Wagh, "Breast cancer detection using ANN network and performance analysis with SVM," International Journal of Computer Engineering and Technology, vol. 8, no. 6S, pp. 501-505, 2019.

[7] Vishal Deshwal, Mukta Sharma, "Breast cancer detection using SVM classifier with grid search techniques,," International Journal of Computer Application,, vol. 178, pp. 18-23, 2019.

[8] V Sansya Vijayan, Lekshmy P L, "Deep learning based prediction of breast cancer in histopathological images,," International Journal of Engineering Research & Technology, vol. 8, pp. 148-152, 2019.

[9] S. Saha, "saturncloud.io," 15 12 2018. [Trực tuyến]. Available: https://saturncloud.io/blog/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way/.

[10] R. Karthiga, K. Narasimhan, Rengarajan Amirtharajan, "Diagnosis of breast cancer for modern mammography using artificial," Mathematics and Computers in Simulation, vol. 202, pp. 316-330, 2022.

[11] Sophia Zell, 05.07.2022, https://tech.bertelsmann.com/en/blog/articles/convnext

[12] Github, https://mmpretrain.readthedocs.io/en/latest/papers/swin_transformer.html

[13] ASTES, https://www.astesj.com/v05/i01/p43/

[14] Arash Ahmadian, Louis S.P. Liu, Yue Fei, Konstantinos N. Plataniotis, Mahdi S. Hosseini, "PSEUDO-INVERTED BOTTLENECK CONVOLUTION FOR DARTS SEARCH SPACE", arXiv:2301.01286v3 [cs.LG] 19 Mar 2023

[15] Dan Hendrycks, Kevin Gimpel, GAUSSIAN ERROR LINEAR UNITS (GELUS), arXiv, 2020.

[16] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu,, "Deep High-Resolution Representation Learning for Visual Recognition for Visual Recognition," IEEE Transactions on Pattern Analysis & Machine Intelligence, vol. 43, pp. 3349-3364, 2021.

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun, "Deep Residual Learning for Image Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016.

[18] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, Saining Xie, "A ConvNet for the 2020s," in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, USA, 2022.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F023**

# XÂY DỰNG HỆ THỐNG QUẢN LÝ KHO TIỆN LỢI

NGUYỄN THỊ HẠNH[1], ĐỖ THỊ TƯỜNG VI[1], HUỲNH TUẤN ANH[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*ngthihanh@gmail.com, dothituongviplic@gmail.com, tuananhitsg@gmail.com*

**Tóm tắt.** Hiện nay, ngành logistics đang phát triển với tốc độ chóng mặt, đặc biệt là trong lĩnh vực quản lý kho hàng. Việc kiểm soát tình trạng các vị trí trong kho hàng đang trở thành một vấn đề quan trọng đối với các doanh nghiệp. Nhận thấy tầm quan trọng, chúng tôi đã phát triển một ứng dụng quản lý kho hàng trên nền tảng điện toán đám mây để giải quyết, sử dụng các công nghệ như Java, Spring Boot, Javascript, React và các dịch vụ AWS (Amazon Web Services). Kiến trúc phần mềm của ứng dụng sử dụng mô hình Client - Server để tối ưu kiến trúc và hiệu suất xử lý trong hệ thống. Ứng dụng giúp người quản lý kho kiểm soát các vị trí hàng hóa trong kho một cách dễ dàng và hiệu quả, đồng thời giảm thiểu thời gian và công sức trong quá trình quản lý kho hàng. Với tính năng tra cứu thông tin và tích hợp các dịch vụ AutoScaling của AWS, ứng dụng giúp đảm bảo tính sẵn có để chứa một lượng số hàng hoá nhất định và điều chỉnh các hoạt động nhập xuất trong kho. Ứng dụng này có thể áp dụng vào nhiều loại doanh nghiệp như các doanh nghiệp sản xuất và phân phối, logistics và vận chuyển, …Triển khai ứng dụng sẽ mang lại hiệu quả và tốc độ cao trong quản lý kho hàng, giảm thiểu thời gian và công sức, đồng thời giảm thiểu sai sót và mất mát hàng hoá so với việc quản lý thủ công.

**Từ khóa.** Quản lý kho hàng, Logistics, Điện toán đám mây, Java, Spring Boot, Javascript, React, AWS, Client - Server, AutoScaling.

## BUILDING A CONVENIENT WAREHOUSE MANAGEMENT SYSTEM

**Abstract.** Nowadays, the logistics industry is rapidly developing, especially in the field of warehouse management. Controlling the status of positions in warehouses has become a crucial issue for businesses. Recognizing this importance, we have developed a cloud-based warehouse management application using technologies such as Java, Spring Boot, JavaScript, React, and AWS (Amazon Web Services). The software architecture of the application adopts a Client-Server model to optimize the system's architecture and processing performance. This application assists warehouse managers in easily and effectively controlling the positions of goods in the warehouse, while reducing time and effort in the management process. With features like information retrieval and integration of AWS's AutoScaling services, the application ensures the availability to contain a certain amount of goods and adjusts import/export operations in the warehouse. This application can be applied to various types of businesses, including manufacturing and distribution enterprises, logistics and transportation companies, and more. Implementing this application will result in high efficiency and speed in warehouse management, minimizing time and effort, as well as reducing errors and loss of goods compared to manual management.

**Key words.** Warehouse management, Logistics, Cloud computing, Java, Spring Boot, Javascript, React, AWS, Client - Server, AutoScaling.

## 1 GIỚI THIỆU

Hiện nay, ngành logistics đang phát triển với tốc độ chóng mặt, đặc biệt là trong lĩnh vực quản lý kho hàng. Việc kiểm soát tình trạng các vị trí trong kho hàng hiện nay đang trở thành một vấn đề quan trọng đối với các doanh nghiệp. Tuy nhiên, vẫn còn một số doanh nghiệp sử dụng phương pháp quản lý thủ công, dễ đến các sai sót trong quá trình nhập, xuất và lưu trữ hàng hóa do sự thiếu sót và khó khăn trong việc kiểm soát

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

và giám sát. Điều này khiến cho việc quản lý kho hàng bị trì trệ, yêu cầu nhân viên phải dành nhiều thời gian và công sức để quản lý hàng hóa, từ việc nhập kho, xuất kho, kiểm kho, lập báo cáo và thống kê.
Trong bối cảnh đó, ngành công nghệ thông tin đang phát triển với rất nhiều tiện ích và lợi ích được áp dụng trong nhiều quy trình và các ngành nghề khác nhau. Vì vậy, việc áp dụng công nghệ thông tin vào quản lý kho sẽ giúp người quản lý kho và nhân viên kiểm soát vị trí hàng hoá một cách dễ dàng và hiệu quả hơn. Hệ thống quản lý kho hàng trên nền tảng công nghệ thông tin giúp giảm thiểu thời gian và công sức trong quá trình quản lý kho hàng, đồng thời đảm bảo tính chính xác và sẵn sàng của thông tin về hàng hóa. Nhờ đó, việc quản lý kho hàng trở nên hiệu quả hơn và tạo thuận lợi cho các hoạt động nhập xuất hàng hoá, tăng tính cạnh tranh và giúp doanh nghiệp vượt qua các đối thủ trong cùng ngành. Ngoài ra, hệ thống quản lý kho còn giúp doanh nghiệp giảm thiểu các chi phí không cần thiết, tăng cường tính linh hoạt và đáp ứng nhanh chóng với các yêu cầu thay đổi của thị trường.
Nắm bắt được các xu hướng và nhu cầu của tổ chức, của các doanh nghiệp, nhóm chúng tôi đã chọn đề tài "Xây dựng hệ thống quản lý kho tiện lợi".
Trong bối cảnh thị trường cạnh tranh khốc liệt, việc áp dụng công nghệ thông tin vào quản lý kho sẽ giúp doanh nghiệp nâng cao chất lượng dịch vụ,Với những lợi ích đó, đề tài "Xây dựng hệ thống quản lý kho tiện lợi" sẽ đóng góp tích cực vào sự phát triển của ngành logistics và đáp ứng nhu cầu của các doanh nghiệp trong việc quản lý kho hàng một cách hiệu quả. Ngoài ra, nghiên cứu còn mang tính ứng dụng cao, có thể áp dụng vào các doanh nghiệp vận chuyển, sản xuất và kinh doanh trong nhiều lĩnh vực khác nhau.
Để thực hiện nghiên cứu này, nhóm chúng tôi sẽ sử dụng các phương pháp nghiên cứu khoa học, phát triển phần mềm và phân tích thiết kế. Đồng thời, chúng tôi sẽ tham khảo các nghiên cứu, tài liệu và quy trình từ các doanh nghiệp đã áp dụng thành công công nghệ thông tin trong quản lý kho hàng.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 Ngôn ngữ lập trình Java

Java được biết đến là ngôn ngữ lập trình bậc cao, hướng đối tượng và giúp bảo mật mạnh mẽ, và còn được định nghĩa là một Platform. Java được phát triển bởi Sun Microsystems, do James Gosling khởi xướng và ra mắt năm 1995. Java hoạt động trên rất nhiều nền tảng như Windows, Mac và các phiên bản khác nhau của UNIX. Nó là ngôn ngữ lập trình có mục đích chung cho phép các nhà phát triển ứng dụng viết một lần, chạy ở mọi nơi nghĩa là mã Java đã biên dịch có thể chạy trên tất cả các nền tảng hỗ trợ Java mà không cần biên dịch lại. Các ứng dụng Java thường được biên dịch thành bytecode có thể chạy trên bất kỳ máy ảo Java (JVM) nào [1].

### 2.2 Framework Spring Boot

Spring Boot là một phần mở rộng của Spring Framework giúp các thành viên lập trình loại bỏ các bước cấu hình phức tạp mà Spring bắt buộc. Spring Boot được dự kiến phát triển bằng ngôn ngữ Java (JAV) trong hệ thống sinh thái Spring Framework. Spring Framework là một framework ứng dụng và bộ chứa đảo ngược điều khiển cho nền tảng Java. Chức năng tính của framework này có thể áp dụng cho bất kỳ ứng dụng Java nào, tuy vậy, nhiều bản mở rộng dành cho việc xây dựng ứng dụng nền web cũng được phát triển trên nền tảng Java EE. Tuy framework không ấn định lên một mô hình lập trình cụ thể nào, Spring Framework đã trở nên phổ biến trong cộng đồng Java như một phương án song hành, thay thế, hoặc thậm chí bổ sung cho mô hình Enterprise JavaBeans (EJB). Đây là framework mã nguồn mở [2].

### 2.3 Ngôn ngữ lập trình JavaScript

JavaScript là một ngôn ngữ lập trình thông dịch được phát triển từ các ý niệm nguyên mẫu và đây cũng là ngôn ngữ căn bản của các lập trình viên học khi mới bắt đầu. Ngôn ngữ này được sử dụng một cách rộng rãi cho các trang web (phía người dùng) cũng như phía máy chủ (với Nodejs). JavaScript có tác dụng xử lý các tương tác của con người lên Website như cuộn chuột, nhấn chuột, ... Ngoài ra nó còn có các chức năng như xử lý, hiển thị dữ liệu và nhiều chức năng khác [3].

### 2.4 Thư viện reactJS

Reactjs là một thư viện Javascript mã nguồn mở hỗ trợ xây dựng các thành phần giao diện nhanh gọn và tiện lợi, được tạo ra bởi sự cộng tác giữa Facebook và Instagram [4].

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 2.5 Cơ sở dữ liệu PostgreSQL

PostgreSQL là một hệ thống quản trị cơ sở dữ liệu quan hệ và đối tượng (object-relational database management system) miễn phí và mã nguồn mở (RDBMS) tiên tiến nhất hiện nay. Khả năng mở rộng cao và tuân thủ các tiêu chuẩn kỹ thuật. Nó được thiết kế để xử lý một loạt các khối lượng công việc lớn, từ các máy tính cá nhân đến kho dữ liệu hoặc dịch vụ Web có nhiều người dùng đồng thời. PostgreSQL thiết kế để chạy trên các nền tảng tương tự UNIX. Tuy nhiên, nhà sản xuất đã điều chỉnh linh động để có thể chạy trên nhiều nền tảng khác như Mac OS X, Solaris hay Windows. PostgreSQL được phát triển bởi PostgreSQL Global Development Group, phát hành vào năm 1996 [5].

### 2.6 Amazon Simple Storage Service (S3)

AWS S3 là dịch vụ lưu trữ đối tượng do AWS cung cấp. Đây có lẽ là dịch vụ lưu trữ được sử dụng phổ biến nhất cho người dùng AWS nhờ các tính năng như tính khả dụng, bảo mật cực cao và kết nối đơn giản với các dịch vụ AWS khác. AWS S3 có thể được sử dụng bởi mọi người với tất cả các trường hợp sử dụng như ứng dụng di động/web, dữ liệu lớn, máy học và nhiều hơn nữa [6].

### 2.7 Amazon Elastic Compute Cloud (Amazon EC2)

Đầu tiên, EC2 là viết tắt của Elastic Compute Cloud. Amazon EC2 là một máy ảo cơ bản với các thành phần phần cứng có thể tùy chỉnh và một hệ điều hành. Hệ thống cho phép bạn chạy nhiều máy tính ảo khác nhau và quản lý giống nhau bằng một phần cứng duy nhất. Elastic Compute Cloud là hệ thống dịch vụ chính và được sử dụng nhiều trong hệ sinh thái AWS khổng lồ. Ví dụ, hệ thống đám mây cung cấp nhiều tính năng, nó tạo điều kiện thuận lợi cho việc tính toán theo yêu cầu và mở rộng khả năng tính toán trong hệ thống đám mây Amazon. EC2 giúp bạn không cần phải mua trước phần cứng. Ngoài ra, không có thêm chi phí khi bảo trì phần cứng đã thuê. Phần cứng ảo tất cả trong một rất dễ sử dụng và cho phép bạn tạo và chạy các ứng dụng với tốc độ cao. Thích ứng với điện toán đám mây trong AWS cho phép bạn khởi chạy nhiều máy chủ ảo. Nó cũng cung cấp khả năng kiểm soát để mở rộng quy mô hoặc thu nhỏ tương ứng với tốc độ của lưu lượng truy cập trang web [7].

## 3 PHÂN TÍCH HỆ THỐNG

Sau khi nghiên cứu và tìm hiểu về quy trình quản lý kho hàng và yêu cầu của khách hàng, nhóm đã tiến hành phân tích và thiết kế hệ thống quản lý kho tiện lợi để xác định chức năng, tương tác giữa các thành phần và quy trình thực hiện trong hệ thống.

### 3.1 Mô tả hệ thống

Chúng tôi tập trung vào việc xây dựng một hệ thống quản lý kho hàng trên nền tảng công nghệ thông tin, giúp tối ưu hóa quy trình quản lý kho hàng, đồng thời giảm thiểu thời gian và công sức cho nhân viên. Hệ thống này sẽ được thiết kế để giám sát và kiểm soát các vị trí hàng hoá trong kho, từ việc nhập kho, xuất kho đến lập báo cáo và thống kê.

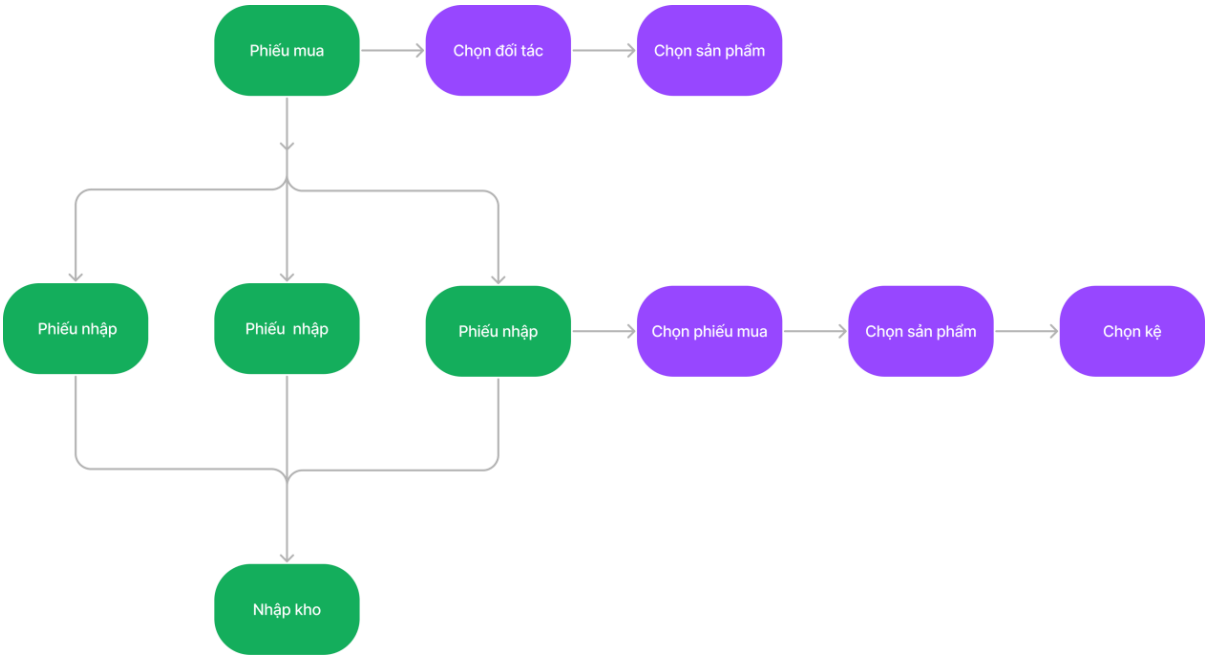
Từ đó, việc quản lý kho hàng trở nên dễ dàng và hiệu quả hơn, giúp doanh nghiệp tiết kiệm thời gian, tối ưu hóa quy trình và tăng cường năng suất sản xuất. Ngoài ra, hệ thống quản lý kho tiện lợi còn giúp đẩy nhanh quá trình xuất nhập hàng hóa, đáp ứng nhanh chóng nhu cầu và yêu cầu của khách hàng.

Hệ thống quản lý kho gồm các thành phần như cơ sở dữ liệu, giao diện người dùng, hệ thống quản lý và hệ thống báo cáo. Cơ sở dữ liệu chứa thông tin về các mặt hàng trong kho, bao gồm tên hàng, số lượng, ngày nhập kho, ngày xuất kho, vị trí lưu trữ và các thông tin khác. Giao diện người dùng cho phép người dùng nhập thông tin vào hệ thống, tìm kiếm và xem thông tin về các mặt hàng trong kho, thực hiện các hoạt động nhập, xuất, tồn kho và kiểm kê hàng hóa. Hệ thống quản lý quản lý, giám sát và điều hành các hoạt động trong kho, đồng thời cung cấp các công cụ để theo dõi số lượng hàng tồn kho. Hệ thống báo cáo cung cấp các báo cáo về tình trạng tồn kho, số lượng xuất, nhập hàng và các chỉ số khác để giúp quản lý kho đưa ra quyết định kinh doanh.

### 3.2 Quy trình

Sau khi tìm hiểu quy trình quản lý kho từ thực tế, nhóm đã xác định được các quy trình chính cho hệ thống.

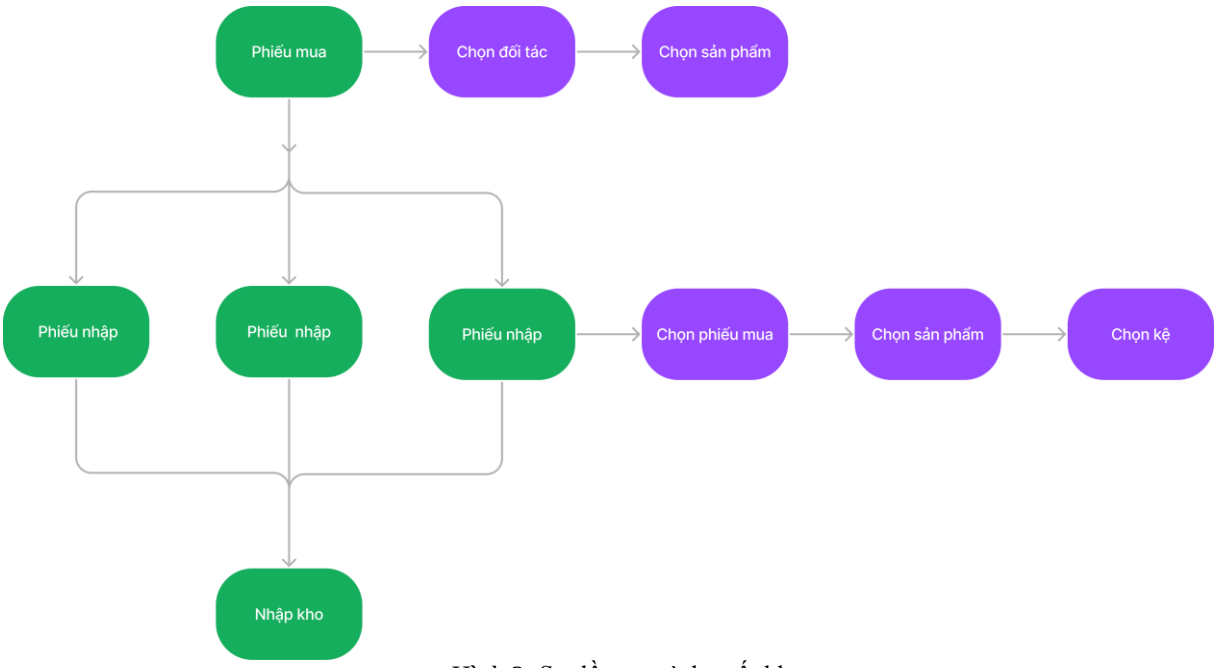
### 3.2.1 Quy trình nhập kho




Hình 1: Sơ đồ quy trình nhập kho

**Bước 1:** Tạo phiếu mua hàng làm đầu vào cho hệ thống. Để tạo được phiếu mua, chúng ta sẽ chọn đối tác và các sản phẩm.
**Bước 2:** Tạo phiếu nhập (một phiếu mua hàng có thể tạo thành nhiều phiếu nhập). Ta sẽ chọn lần lượt từng sản phẩm và số lượng tương ứng để nhập vào kệ trong kho.
**Bước 3:** Nhập kho. Chọn phiếu nhập đã tạo để xác nhận nhập kho.

### 3.2.1 Quy trình xuất kho



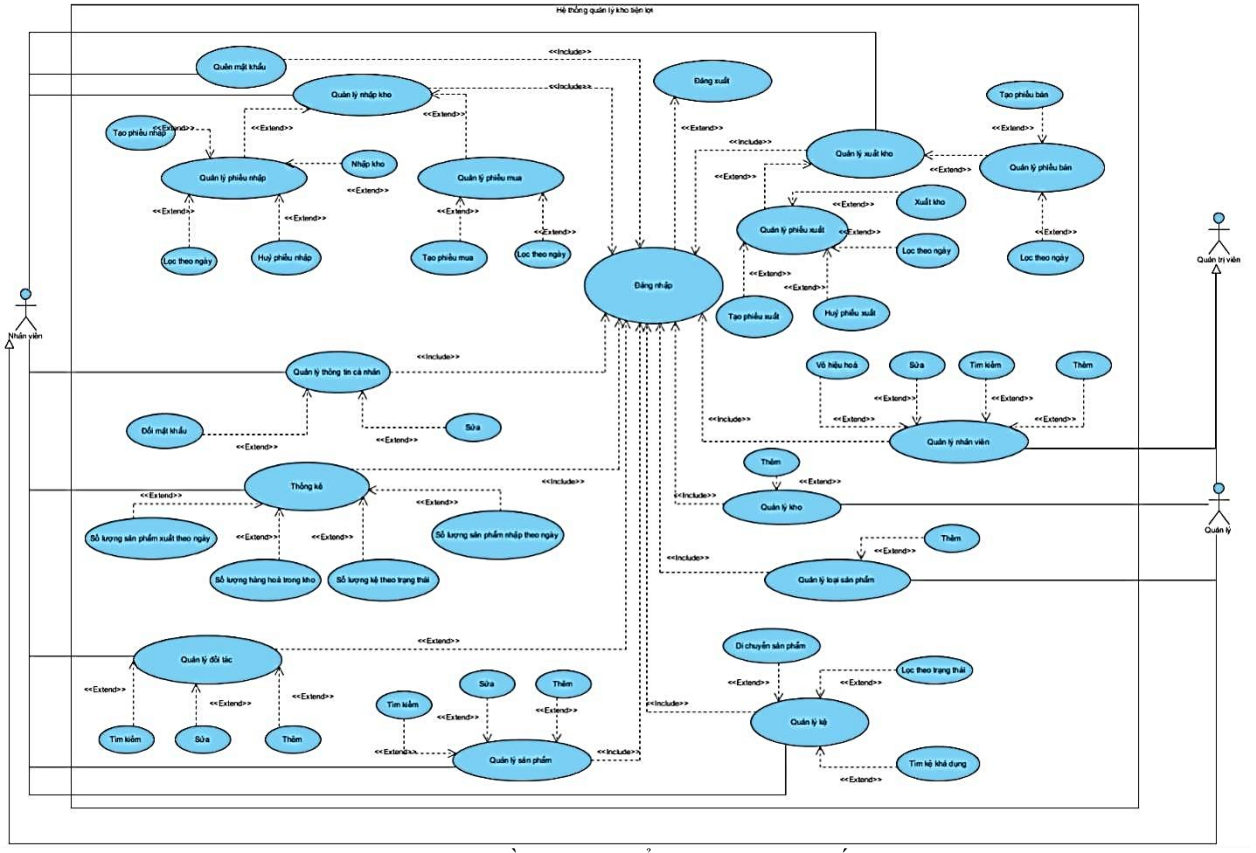

Hình 2: Sơ đồ quy trình xuất kho


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**Bước 1:** Tạo phiếu bán hàng làm đầu ra cho hệ thống. Để tạo được phiếu bán, chúng ta sẽ chọn đối tác và các sản phẩm có sẵn trong kho.
**Bước 2:** Tạo phiếu xuất (một phiếu bán hàng có thể tạo thành nhiều phiếu xuất). Ta sẽ chọn lần lượt từng sản phẩm và số lượng tương ứng để xuất kho)
**Bước 3:** Xuất kho. Chọn phiếu xuất đã tạo để xác nhận xuất kho.

### 3.2.2 Trường hợp sử dụng tổng quát

Đối tượng người dùng cho hệ thống quản lý kho hàng tiện lợi bao gồm Quản trị viên, quản lý, nhân viên. Chúng tôi tập trung vào các chức năng chính như xuất, nhập kho và quản lý kho: Nhân viên thực hiện quản lý nhập kho, quản lý xuất kho, quản lý kệ, quản lý sản phẩm, thống kê và báo cáo. Quản trị viên có vai trò quản lý nhân viên, Quản lý kế thừa các quyền nhân viên
Hình ảnh dưới đây giúp tổng quan các mô tả trên:

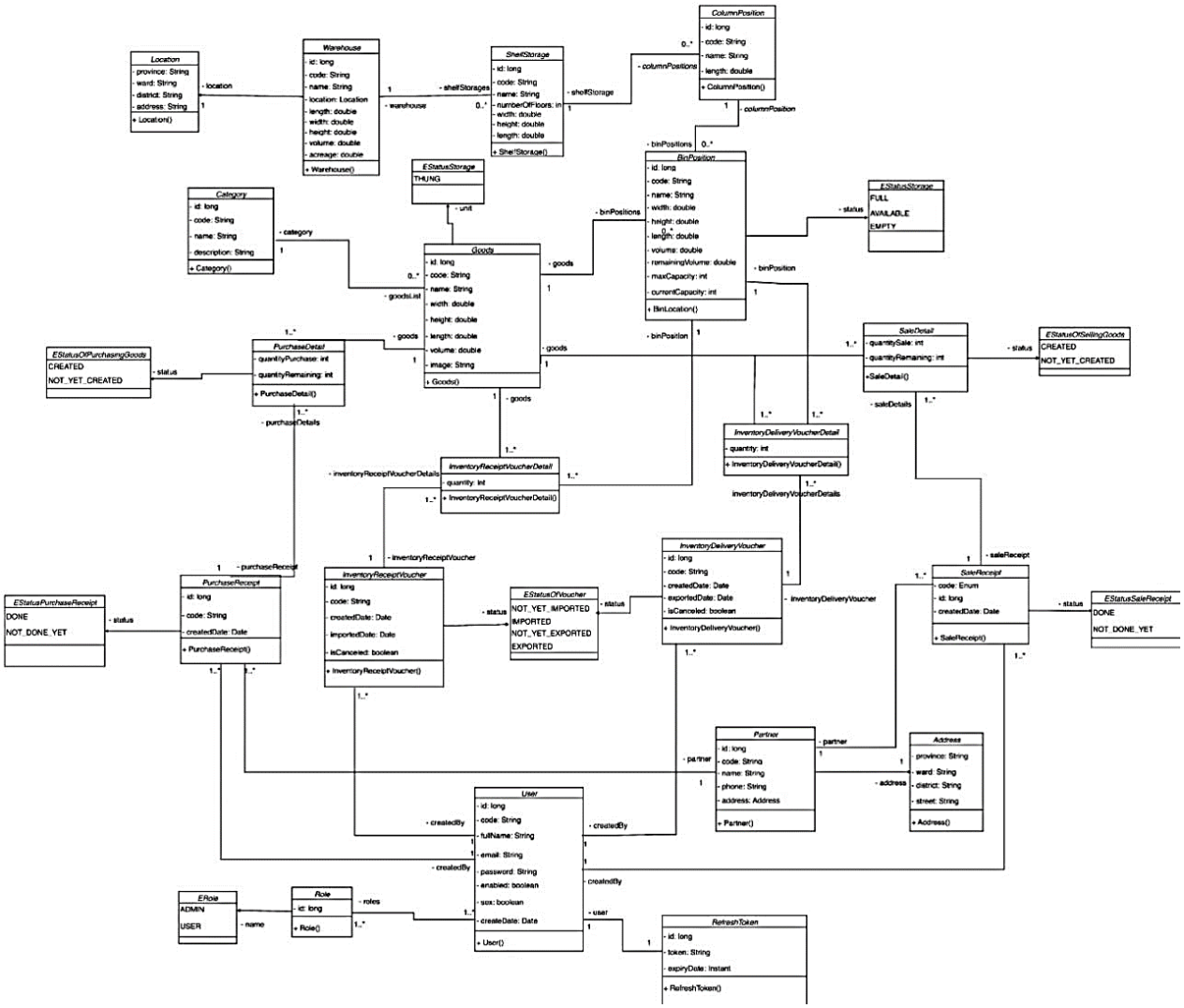



Hình 3: Sơ đồ Use Case tổng quát của hệ thống

### 3.2.3 Sơ đồ lớp

Sau khi phân tích và thiết kế được Trường hợp sử dụng ở trên, chúng tôi thiết kế được sơ đồ lớp của hệ thống. Trong hệ thống quản lý kho hàng tiện lợi, một số lớp chính như sau:

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

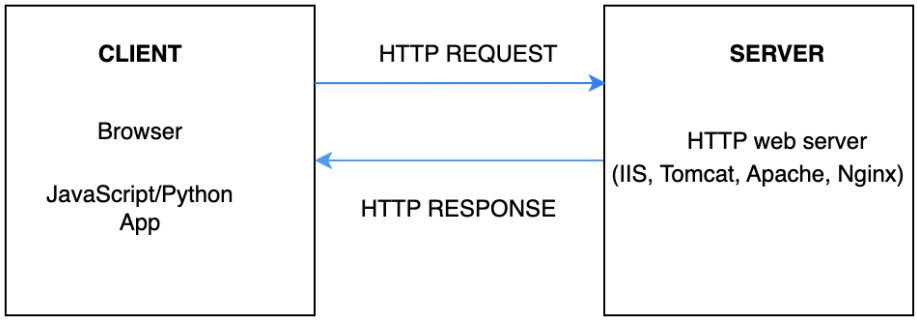



Hình 4: Class Diagram của hệ thống

## 3.3 Kiến trúc phần mềm

Sử dụng mô hình Client-Server vào hệ thống quản lý kho hàng tiện lợi, với các thế mạnh sau:
Tách biệt giữa phần giao diện người dùng và phần xử lý logic: Kiến trúc client-server giúp tách biệt giữa phần giao diện người dùng và phần xử lý logic trên server. Việc này giúp cho việc phát triển và bảo trì hệ thống trở nên dễ dàng hơn, vì các phần của hệ thống có thể được phát triển và bảo trì một cách độc lập.
Phân tán tải: phân tán tải giữa phần client và phần server. Phần client đảm nhận việc hiển thị giao diện người dùng và thực hiện các hoạt động tương tác với người dùng, trong khi phần server đảm nhận việc xử lý logic và lưu trữ dữ liệu của ứng dụng.Việc này giúp giảm thiểu tải cho mỗi phần, giúp cho ứng dụng có thể xử lý được nhiều yêu cầu từ người dùng cùng lúc và tăng hiệu suất của ứng dụng.
Bảo mật: Kiến trúc client-server giúp cải thiện bảo mật của ứng dụng. Các thông tin nhạy cảm được lưu trữ và xử lý trên phần server, giúp giảm thiểu nguy cơ bị tấn công từ phía người dùng.

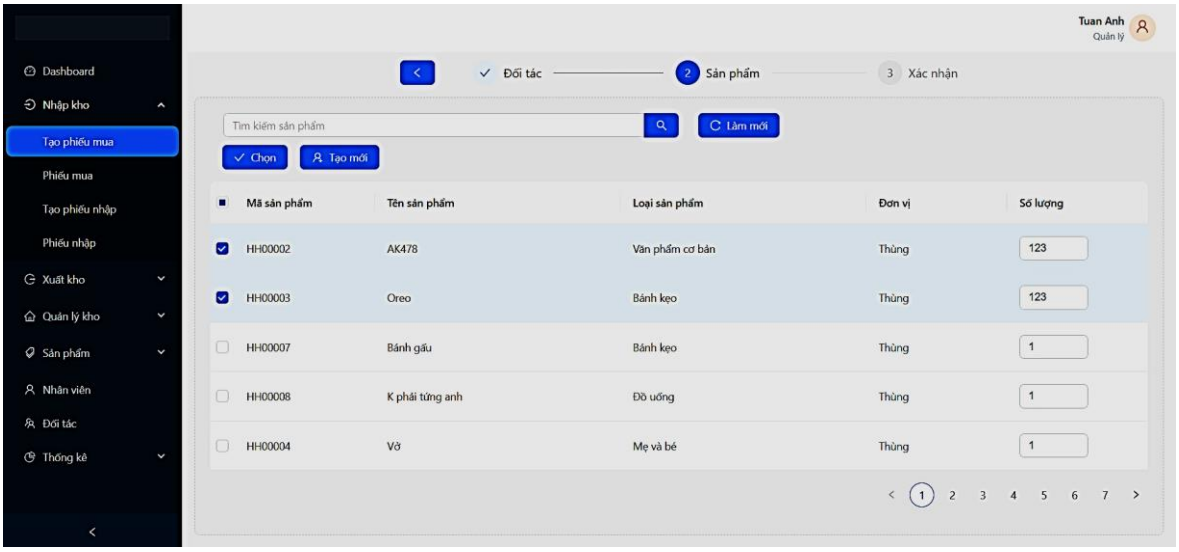
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh



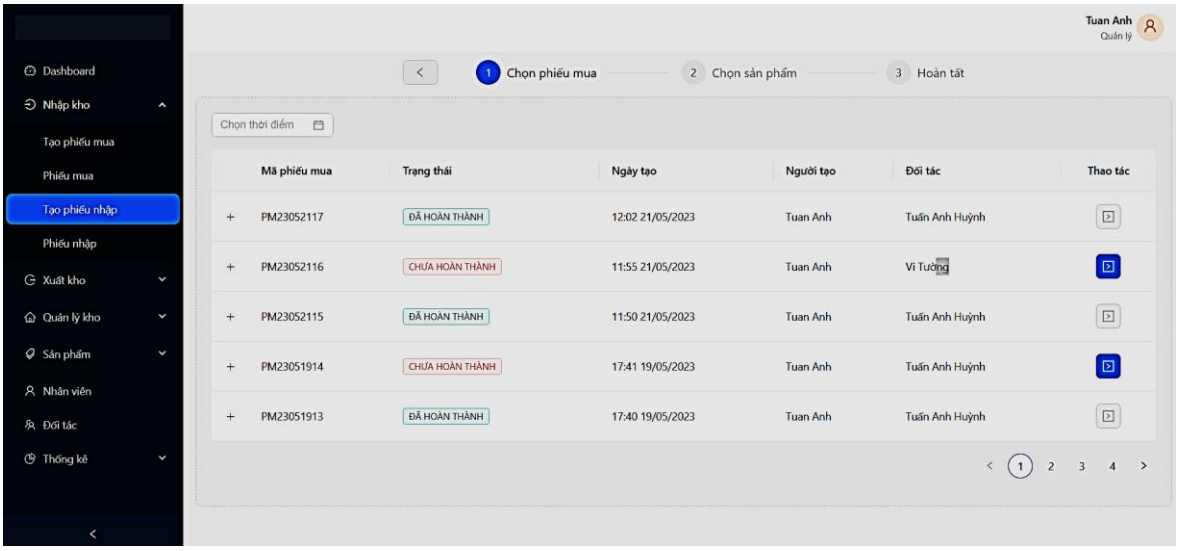

Hình 5: Mô hình kiến trúc client-server.

## 4 KẾT QUẢ HIỆN THỰC

Sau đây là các hình ảnh và các bước về một số chức năng chính của hệ thống quản lý kho hàng tiện lợi. Hệ thống hỗ trợ quá trình xuất, nhập hàng.

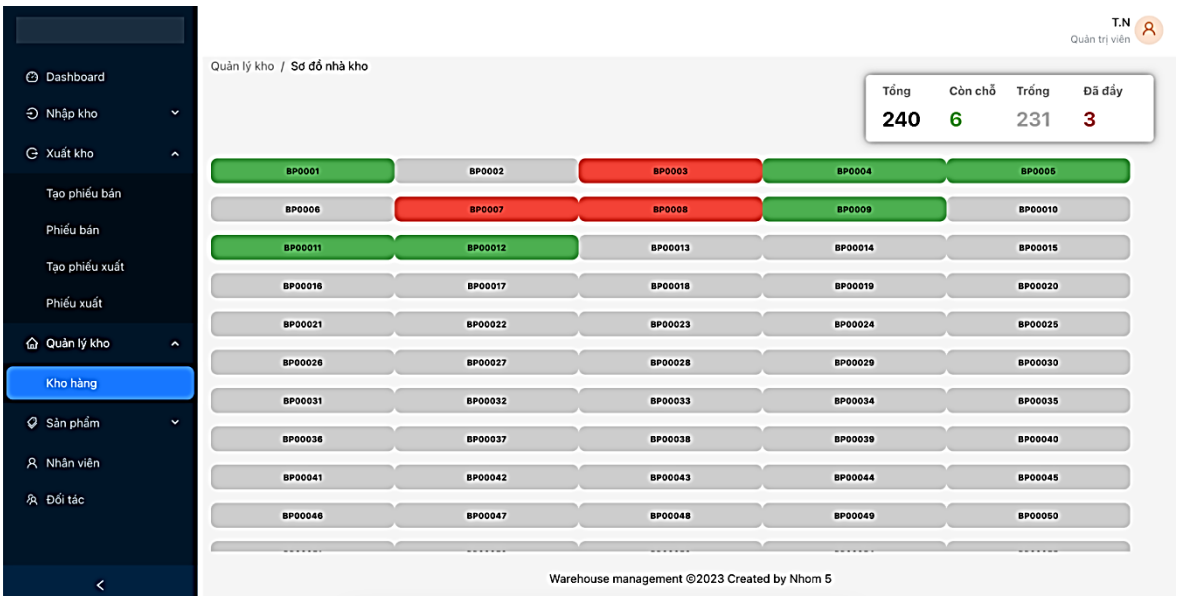



Hình 6: Trang chọn sản phẩm




Hình 7: Trang tạo phiếu nhập

Cũng như quản lý kho theo kệ:




Hình 8: Trang xem các vị trí trong kho

## 5 KẾT LUẬN

Đã thành công trong việc giải quyết các mục tiêu đặt ra, đề tài đã giúp người quản lý có thể quản lý các thông tin liên quan đến quản lý kho, hàng hoá, nhập xuất và các vị trí trong kho một cách hiệu quả hơn. Đồng thời, ứng dụng đã sử dụng các công nghệ tiên tiến như JWT, gửi mail, sử dụng các tài nguyên trên cloud AWS, và Docker để triển khai ứng dụng. Các yêu cầu phi chức năng cũng đã được hoàn thành một cách triệt để.

Trong quá trình triển khai nghiên cứu, các thành viên đã nắm được quy trình, nghiệp vụ của việc quản lý hàng hoá, cách sắp xếp và quản lý các kệ hàng trong kho, cũng như cách tổ chức lưu trữ chúng. Các quy tắc và quy trình nghiệp vụ đã được triển khai và hoàn thành một cách chính xác và hiệu quả. Đề tài đã sử dụng các công nghệ như Java, ReactJS, JavaScript, NodeJS và các kỹ năng đã học từ trường để đạt được kết quả tốt nhất.

Nhờ tính ứng dụng của nghiên cứu này, người quản lý kho có thể tận dụng hệ thống quản lý kho tiện lợi để giảm thiểu thời gian và công sức cho quá trình xuất, nhập và quản lý vị trí hàng hoá trong kho. Đồng thời, hệ thống cũng giúp tăng hiệu quả quản lý kho bằng cách cung cấp các chức năng tự động hóa và thông tin chi tiết về hàng tồn kho. Tất cả các mục tiêu đã được đặt ra đã được giải quyết một cách thành công.

Việc phát triển thêm ứng dụng mobile là một hướng đi của nhóm. Đi kèm với đó, việc tích hợp các công nghệ mới như IoT và gắn chip ở mỗi kệ sẽ giúp cho việc quản lý hàng hoá, vị trí và số lượng trở nên dễ dàng hơn bao giờ hết. Đồng thời, sự thể hiện rõ hơn sơ đồ nhà kho như dãy kệ, cửa ra vào, khoảng cách giữa các dãy kệ, diện tích và thể tích kho, cũng như việc thể hiện sơ đồ kho ở dạng 2d và 3d, sẽ giúp cho quản lý kho hàng trở nên hiệu quả hơn và giảm thiểu rủi ro trong quá trình vận chuyển và lưu trữ hàng hoá. Với sự phát triển này, chắc chắn rằng, các doanh nghiệp sẽ có cơ hội cải thiện và nâng cao chất lượng dịch vụ của mình, đồng thời tối ưu hóa chi phí và gia tăng lợi nhuận của mình.

## TÀI LIỆU THAM KHẢO

[1]. Java Documentation. https://docs.oracle.com/en/java/. Truy cập vào ngày 22/04/2023.

[2]. Sim, W., Choi, W.: Spring boot-based web application development for providing information on security vulnerabilities and patches for open source software. Journal of Korea Society of Digital Industry and Information Management 17(4) (2021) 77–83

[3]. The Modern JavaScript Tutorial. https://javascript.info/. Truy cập vào ngày 25/04/2023.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

[4]. Rawat, P., Mahajan, A.N.: Reactjs: A modern web development framework. International Journal of Innovative Science and Research Technology 5(11) (2020)

[5]. The World's Most Advanced Open-Source Relational Database. https://www.postgresql.org/. Truy cập vào ngày 22/04/2023.

[6]. AWS. Amazon S3. https://aws.amazon.com/vi/s3/. Truy cập vào ngày 22/04/2023.

[7]. AWS. Amazon Elastic Compute Cloud. https://docs.aws.amazon.com/ec2/index.html. Truy cập vào ngày 22/04/2023.

[8]. Tompkins, J. A. (2016). Warehouse Management Handbook.

[9]. Hamilton, S., & Fenstermaker, S. (2014). Warehouse Management using Microsoft Dynamics AX 2012 R3.

[10]. Piasecki, D. J. (2008). Inventory Accuracy: People, Processes, & Technology.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F024**

# XÂY DỰNG ỨNG DỤNG HỖ TRỢ TÌM KIẾM THỢ SỬA KHÓA

NGUYỄN VĂN THẮNG[1*], NGUYỄN MINH VƯƠNG[2], NGUYỄN VĂN TRINH[1]

*Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*[*]nguyenvanthang@iuh.edu.vn*

**Tóm tắt.** Hiện nay, ứng dụng di động không chỉ đơn giản là một công cụ giải trí mà còn hỗ trợ chúng ta trong các hoạt động tìm kiếm dịch vụ. Tuy nhiên, việc tìm kiếm một thợ khóa đáng tin cậy là một thách thức đối với khách hàng. Vì vậy, nghiên cứu tập trung vào phát triển một ứng dụng di động giúp khách hàng tìm kiếm và kết nối các thợ khóa chuyên nghiệp trong khu vực của họ. Ứng dụng cung cấp các tính năng hiển thị vị trí của các thợ khóa gần đó và xếp hạng của họ để giúp khách hàng tìm thợ khóa phù hợp. Ứng dụng cũng tích hợp Google Maps API để thợ khóa có thể cập nhật vị trí hiện tại và tiếp cận vị trí khách hàng nhanh chóng. Sử dụng công nghệ React Native và Firebase để phát triển ứng dụng và tập trung vào việc nghiên cứu các công nghệ mới để cải thiện hiệu năng và tăng sự tin tưởng của khách hàng đối với dịch vụ.

**Từ khóa.** Google Maps API, React Native, Firebase.

## BUILD AN APPLICATION TO SUPPORT FINDING A LOCKSMITH

**Abstract.** Currently, mobile applications are not just a tool for entertainment but also assist us in finding services. However, finding a reliable locksmith is a challenge for customers. Therefore, the research focuses on developing a mobile application that helps customers search for and connect with professional locksmiths in their area. The application provides features to display the location and ratings of nearby locksmiths, helping customers find the right locksmith. The application also integrates Google Maps API, allowing locksmiths to quickly update their current location and access the customer's location. React Native and Firebase technologies are used to develop the application, and research focuses on exploring new technologies to improve performance and increase customer trust in the service.

**Keywords.** Google Maps API, React Native, Firebase.

## 1 LÝ DO CHO SỰ XUẤT HIỆN CỦA ỨNG DỤNG HỖ TRỢ TÌM THỢ SỬA KHÓA?

Trong cuộc sống hiện đại, ứng dụng di động trở thành một phần không thể thiếu của con người, vì nó cung cấp cho chúng ta tiện ích và tiện lợi trong mọi lĩnh vực. Trong lĩnh vực tìm kiếm dịch vụ, việc tìm kiếm một thợ sửa khóa đáng tin cậy là một thách thức đối với khách hàng. Việc tìm thợ sửa khóa truyền thống có thể làm khách hàng mất nhiều thời gian và công sức để tìm một thợ sửa khóa chuyên nghiệp, đáng tin cậy và giá cả hợp lý. Do đó sự xuất hiện của ứng dụng di động hỗ trợ tìm thợ sửa khóa đã giúp giải quyết vấn đề này và tạo ra sự tiện lợi cho người sử dụng.

## 2 LỢI ÍCH VÀ VAI TRÒ CỦA ỨNG DỤNG HỖ TRỢ TÌM THỢ SỬA KHÓA

### 2.1 Lợi ích của khách hàng khi sử dụng ứng dụng

- Tiết kiệm thời gian và công sức: Khách hàng không cần phải tốn nhiều thời gian và công sức để tìm kiếm thợ sửa khóa phù hợp. Thay vào đó, họ có thể sử dụng ứng dụng để tìm kiếm các thợ sửa khóa trong khu vực của mình và chọn thợ sửa khóa phù hợp.
- Tăng tính chính xác: Ứng dụng hỗ trợ tìm thợ sửa khóa giúp khách hàng tìm kiếm và kết nối với các thợ sửa khóa chuyên nghiệp trong khu vực của họ. Điều này đảm bảo tính chính xác và độ tin cậy trong quá trình sửa chữa khóa của khách hàng.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

- Tiết kiệm chi phí: Khách hàng có thể sử dụng ứng dụng để so sánh giá cả và chọn thợ sửa khóa với giá cả hợp lý nhất, giúp họ tiết kiệm chi phí.
- Dịch vụ thân thiện: Khách hàng có thể quản lý được quá trình của thợ sửa khóa từ lúc bắt đầu đơn hàng đến khi kết thúc đơn.

### 2.2 Lợi ích của thợ sửa khóa khi đăng ký sử dụng ứng dụng

- Tăng khả năng tiếp cận khách hàng: Khi sử dụng ứng dụng, thợ sửa khóa có thể nhận được đơn của những khách hàng tiềm năng trong khu vực của mình mà không cần quảng bá.
- Xác định vị trí khách hàng: Khi khách hàng đăng kí yêu cầu dịch vụ, thợ sửa khóa có thể xác định vị trí của khách hàng thông qua ứng dụng và tiếp cận khách hàng một cách nhanh chóng và tiện lợi.
- Đánh giá và tăng khả năng tin cậy: Khách hàng sử dụng ứng dụng có thể đánh giá về dịch vụ của thợ sửa khóa sau khi kết thúc đơn hàng, từ đó giúp tăng khả năng tin cậy và uy tín của thợ sửa khóa.

### 2.3 Vai trò của ứng dụng hỗ trợ tìm thợ sửa khóa

- Tự động hóa tất cả quy trình: Kết nối với khách hàng và thợ sửa khóa, cập nhật trạng thái đơn hàng, thống kê được số lượng đơn yêu cầu trong ngày
- Thợ sửa khóa được hỗ trợ tối đa nhờ hệ thống định vị GPS, giúp tìm ra địa điểm nhanh chóng, chính xác.
- Bảo mật thông tin khách hàng, thông tin đơn hàng và cả thông tin của thợ sửa khóa.

*Tóm lại, Ứng dụng hỗ trợ tìm thợ sửa khóa đóng vai trò quan trọng trong việc khách hàng và thợ sửa khóa, đồng thời mang lại lợi ích cho cả hai bên trong việc tìm kiếm và cung cấp dịch vụ sửa khóa chất lượng.*

## 3 CÁC CÔNG NGHỆ LIÊN QUAN

### 3.1 NodeJS

**Khái niệm NodeJS**

- NodeJS là một nền tảng (platform) phía Server side được xây dựng, vận hành trên V8 JavaScript runtime của Chrome giúp xây dựng và phát triển các ứng dụng mạng một cách nhanh chóng và có khả năng mở rộng dễ dàng.
- Node.js ra mắt vào năm 2009, bởi Ryan Dahl và nhận sự bảo trợ từ công ty Joyent (Hoa Kỳ).
- Phần core bên dưới của Node.js hầu hết được viết bằng C++. Chính vì thế mà tốc độ xử lý và hiệu năng khá cao.
- Bên cạnh đó, Node.js còn tạo ra ứng dụng có tốc độ xử lý nhanh với thời gian thực (real-time).
- Node.js cũng được áp dụng để tạo các sản phẩm có lượng truy cập lớn, mở rộng nhanh và đổi mới công nghệ… hoặc tạo dự án startup tiết kiệm thời gian nhất.

**Đặc điểm của NodeJS**

Nodejs có nhiều đặc điểm nổi bật, vượt trội. Nắm được các đặc điểm này sẽ giúp bạn hiểu sâu hơn về Nodejs, đồng thời đưa ra quyết định sử dụng nền tảng này chính xác.

- Bất đồng bộ và phát sinh sự kiện (Non-blocking and Event Driven): Tất các các APIs của thư viện Node.js đều bất đồng bộ (non-blocking), NodeJS không cần đợi một API trả về dữ liệu. Server chuyển sang một API khác sau khi gọi nó và có cơ chế riêng để gửi thông báo và nhận phản hồi về các hoạt động của Node.js và API đã gọi.
- Tốc độ nhanh: Phần core phía dưới được viết gần như toàn bộ bằng C++ kết hợp Chrome V8 Engine nên tốc độ xử lý công việc của Node.js cực nhanh, nhưng vẫn đảm bảo được tính chuẩn xác.
- Đơn giản – Hiệu năng cao: Node.js sử dụng một mô hình luồng đơn luồng (single thread) và các sự kiện lặp (event-loop). Cơ chế sự kiện cho phép phía Server trả về phản hồi theo non-blocking, đồng thời tăng hiệu quả sử dụng. Các luồng đơn cung cấp dịch vụ cho nhiều request hơn hẳn Server truyền thống.
- Không lưu bộ nhớ đệm (non buffer): Nền tảng Node.js không có vùng nhớ đệm, tức không cung cấp khả năng lưu trữ dữ liệu buffer.




Ảnh minh họa NodeJS

### 3.2 React Native

***React Native là gì?***

- Được phát triển bởi Facebook, React Native là một framework hướng đến phát triển ứng dụng di động đa nền tảng.
- Với sự trợ giúp của React Native, lập trình viên (developer) có thể sử dụng JavaScript để tạo ra mobile apps (ứng dụng di động) hỗ trợ cho cả nền tảng Android và iOS.
- Instagram, Facebook, Skype,… là những ứng dụng nổi bật sử dụng React Native.




Ảnh minh họa React Native

***React Native hoạt động như thế nào?***

React Native hoạt động được nhờ tích hợp 2 thread là Main Thread và JS Thread cho ứng dụng mobile. Trong đó:

- Main Thread: cập nhật giao diện người dùng (UI) và xử lý tương tác người dùng.
- JS Thread: thực thi và xử lý code Javascript.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Hai luồng Main Thread và JS Thread này hoạt động độc lập với nhau. Hai Thread sẽ tương tác với nhau nhờ một Bridge(cầu nối). Chiếc cầu này sẽ chuyển đổi dữ liệu qua lại giữa các Thread.

***Ưu điểm***

- Có thể tái sử dụng code

React Native cho phép các developer có thể tái sử dụng code trong khi phát triển các ứng dụng đa nền tảng. Đặc biệt, developer có thể tái sử dụng hầu như 80-90% các đoạn code thay vì phải viết và tạo các ứng dụng riêng biệt cho các nền tảng khác nhau.

Ưu điểm này giúp người dùng:

- Tiết kiệm thời gian và giảm chi phí phát triển của một ứng dụng.
- Tận dụng nguồn nhân lực tốt hơn
- Duy trì ít code hơn, ít bugs hơn
- Các tính năng trong cả 2 platforms cũng tương tự nhau.




Ảnh minh họa React Native

- Cộng đồng người dùng lớn

React Native được đánh giá là một trong những Framework được yêu thích nhất (khảo sát của stack overflow vào năm 2019).

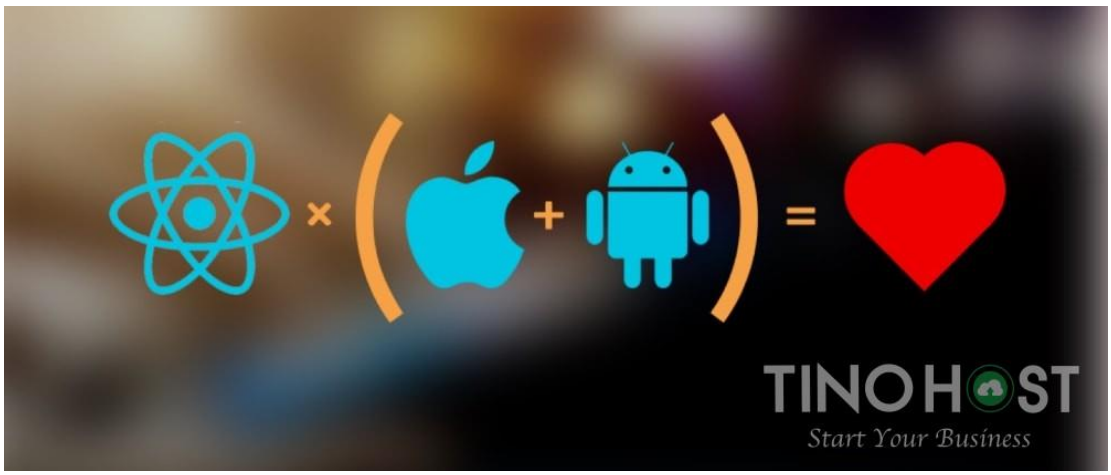
Nhờ cộng đồng người dùng rất lớn trên toàn thế giới, bạn có thể tìm sự hỗ trợ nếu gặp phải bugs.

- Tính ổn định và tối ưu

Được phát triển bởi Facebook, React Native có hiệu năng ổn định khá cao.

- Mã React Native giúp đơn giản hóa quá trình xử lý dữ liệu.
- Đội ngũ phát triển ứng dụng không quá lớn.
- Xây dựng ứng dụng ít native code nhất cho nhiều hệ điều hành khác nhau.
- Trải nghiệm người dùng tốt hơn khi so sánh với ứng dụng Hybrids

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

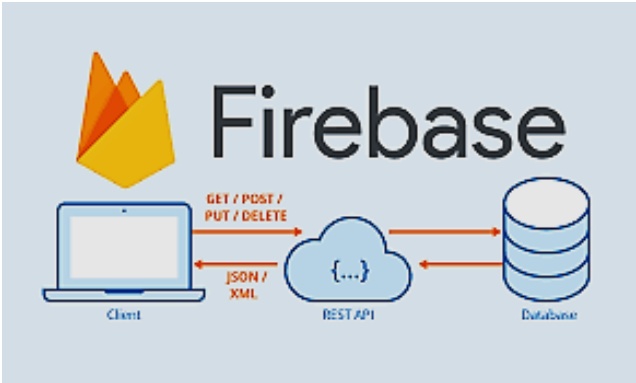



Ảnh minh họa React Native

***Nhược điểm***

- Yêu cầu Native code.
- Hiệu năng kém hơn so với Native App.
- Bảo mật chưa thật sự tốt do dùng JavaScript. Do sử dụng JavaScript, người dùng cũng sẽ bị ảnh hưởng bởi những đặc điểm của JavaScript: dễ làm dễ sai, dẫn đến khó duy trì về sau.
- Quản lý bộ nhớ.
- Tùy biến chưa thật sự tốt ở một số module.
- Không thích hợp cho các app cần năng lực tính toán cao (hash, crypto, etc).

## 3.3 Firebase (NOSQL)




Ảnh minh họa Firebase

● Firebase là gì ?

- Firebase là nền tảng đám mây của Google, cung cấp các dịch vụ và công cụ để xây dựng và phát triển ứng dụng di động và web. Nó bao gồm các tính năng quản lý cơ sở dữ liệu realtime, lưu trữ dữ liệu, xác thực người dùng, push notification, phân tích và thu thập dữ liệu và nhiều tính năng khác. Firebase được thiết kế để hỗ trợ các ứng dụng với lượng người dùng lớn và yêu cầu thời gian thực.
- Firebase cung cấp cho các nhà phát triển một cách tiếp cận dễ dàng và nhanh chóng để tạo ra ứng dụng chất lượng cao mà không cần quá nhiều kiến thức về hạ tầng phía server. Thay vì phải xây dựng các tính năng của ứng dụng như lưu trữ dữ liệu, phân tích và thu thập dữ liệu, Firebase cung cấp cho các nhà phát triển một cách để dễ dàng tích hợp các tính năng này vào ứng dụng của họ bằng cách sử dụng các API và SDK được cung cấp bởi Firebase.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

- Firebase đã trở thành một trong những nền tảng đám mây phổ biến nhất trong cộng đồng phát triển ứng dụng di động và web, với hàng triệu ứng dụng được xây dựng bằng Firebase.

### 3.4 Map4D

Map4D là một dịch vụ địa lý đa nền tảng cung cấp bản đồ và công nghệ không gian 3D. Với Map4D, người dùng có thể khám phá, tìm kiếm và tương tác với địa điểm trên bản đồ thông qua các ứng dụng di động và web. Dịch vụ này cung cấp các công cụ và API cho việc tích hợp dữ liệu địa lý vào ứng dụng của bạn và mang lại trải nghiệm địa lý tương tác và trực quan cho người dùng.

Ảnh minh họa Map4D

## 4 KFIX: ỨNG DỤNG HỖ TRỢ TÌM THỢ SỬA KHÓA

### 4.1 Sơ lược công nghệ sử dụng

- Font-end:

- ReactJS: Tạo giao diện website phục vụ cho các chức năng quản lý.
- React Native: Tạo giao diện app, phục vụ các chức năng của người dùng và thợ khóa.

- **Back-end:**

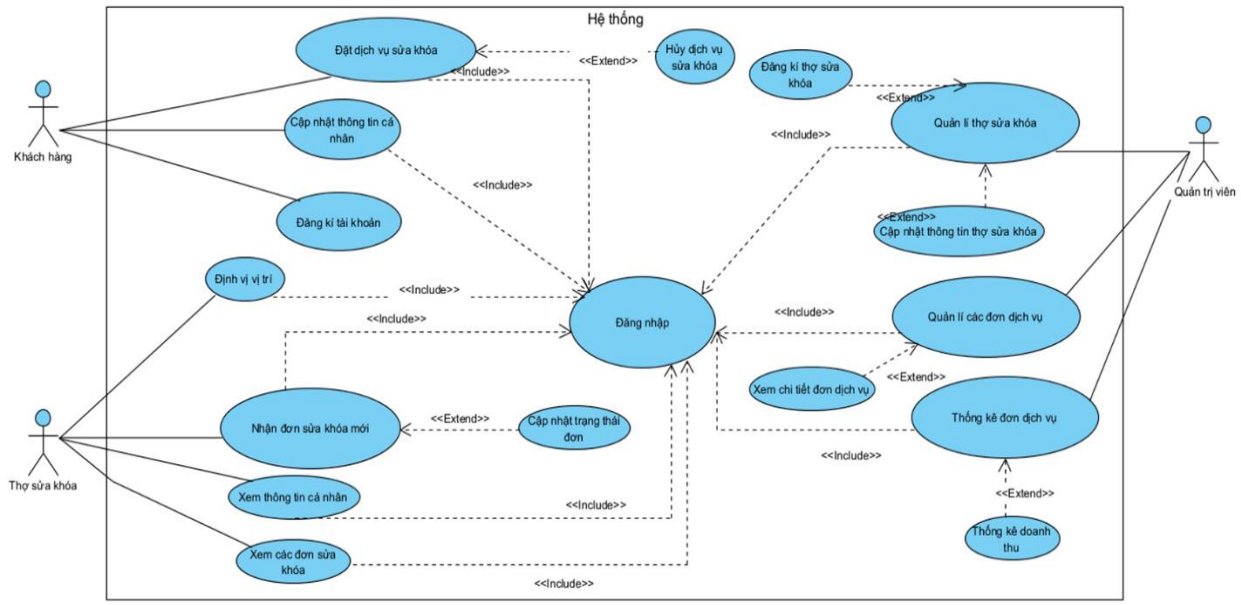
- NodeJS (JavaScript): là framework hiện đại và mới của ngôn ngữ lập trình java, nodejs chịu trách nhiệm tạo ra các API, xử lý các request, đồng thời cũng chịu trách nhiệm xử lý và lưu dữ liệu xuống database.

- **Database:**

- Firebase: Là nơi lưu trữ các dữ liệu của toàn hệ thống như dữ liệu người dùng, thợ khóa, đơn hàng. Ngoài ra hỗ trợ cho các chức năng realtime cho ứng dụng.

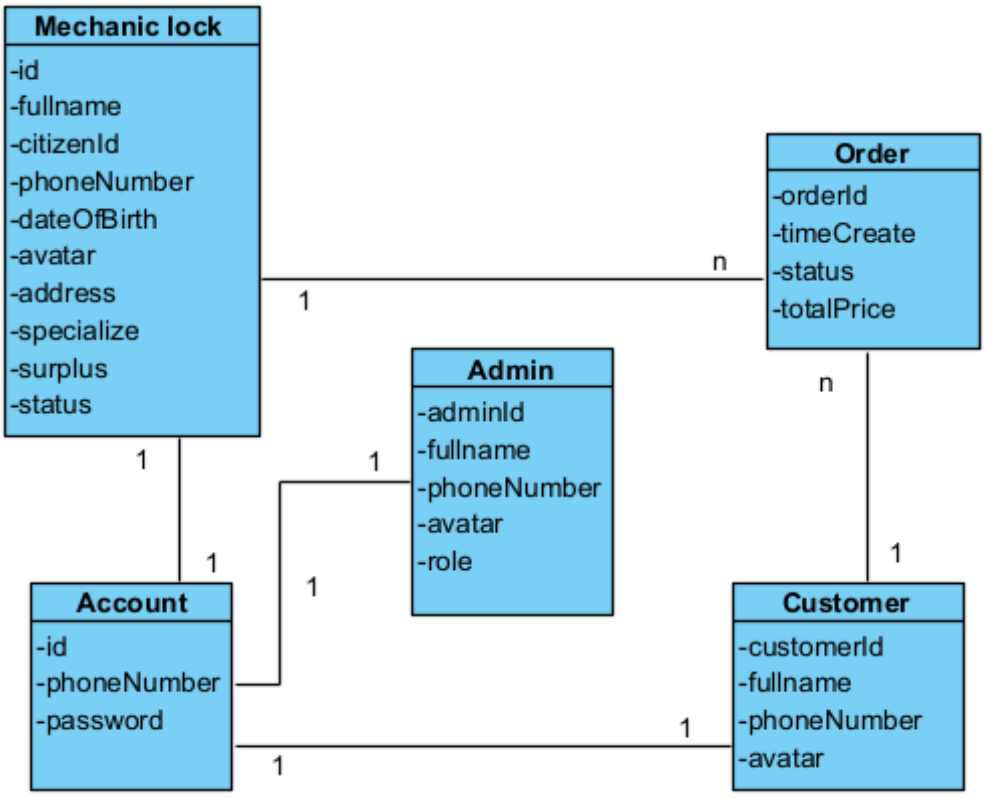
### 4.2 Phân tích thiết kế hệ thống Ứng dụng quản lý shipper

#### 4.2.1 Sơ đồ Use case



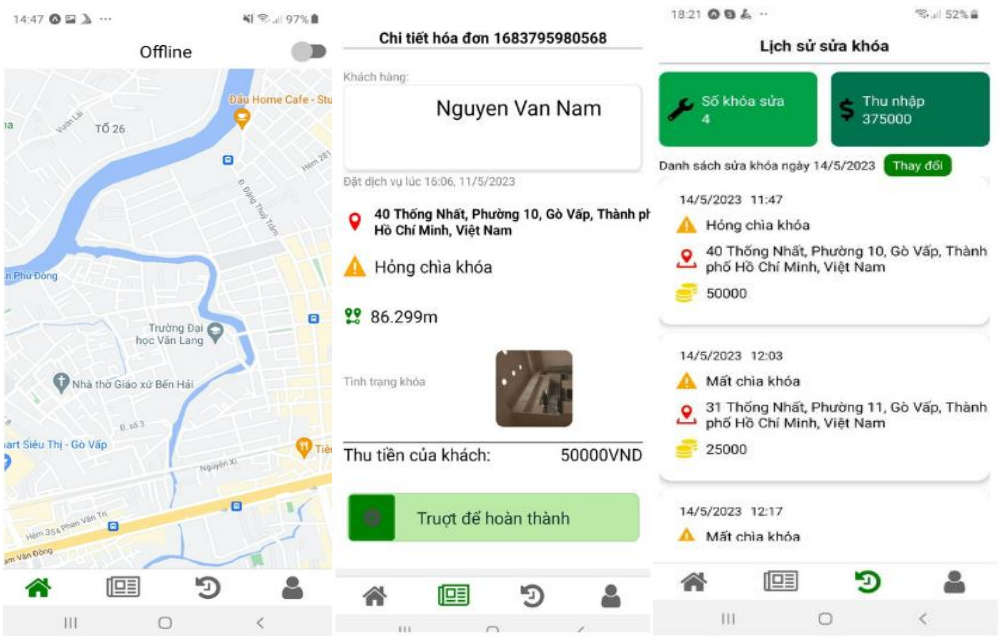

Sơ đồ use case

### 4.2.2 Class diagram



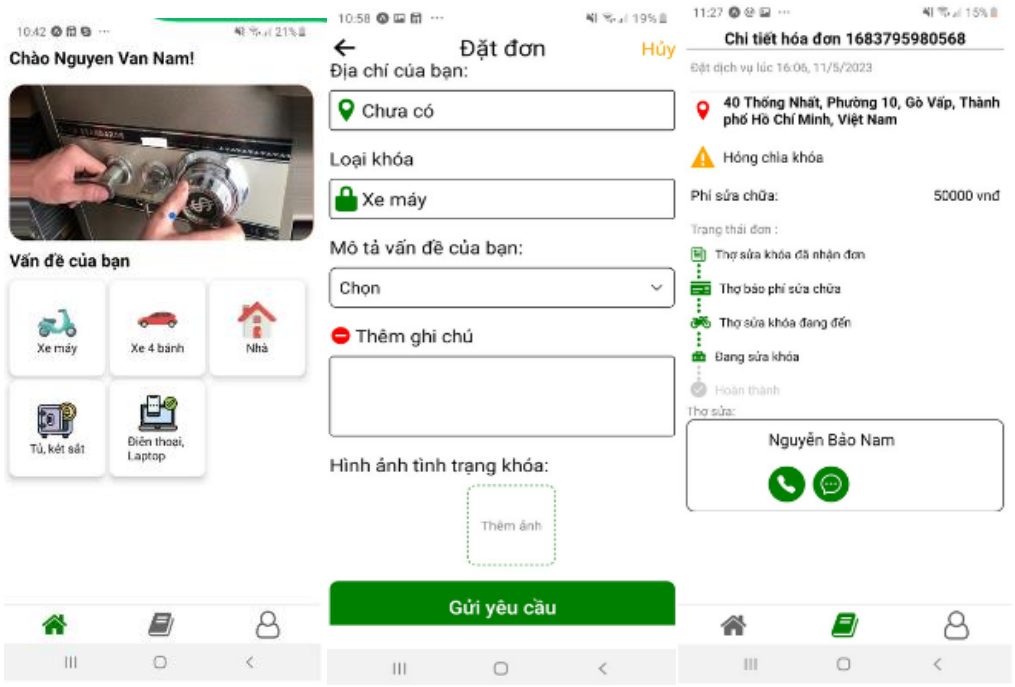

Sơ đồ lớp

### 4.2.3 Giao diện đồ họa (Các màn hình đại diện)




Màn hình đại diện trên ứng dụng dành cho thợ sửa khóa


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Màn hình đại diện trên ứng dụng dành cho khách hàng

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Kết luận

- Xây dựng thành công website, app mobile của ứng dụng.
- Thợ sửa khóa có thể đăng nhập, thay đổi trạng thái làm việc, hiển thị vị trí hiện tại, chỉ đường trên bản đồ, xem lịch sử đơn hàng, xem thông tin cá nhân.
- Về phía khách hàng: Khách hàng có thể đăng ký và sử dụng tài khoản để đăng nhập, có thể tạo một đơn để yêu cầu một dịch vụ sửa khóa. Khách hàng có thể chọn thợ theo mong muốn. Ngoài ra khách hàng có thể đánh giá chất lượng của dịch vụ sửa khóa sau khi đơn hàng kết thúc.
- Về phần quản lý: Nhân viên có thể quản lý các danh sách thợ sửa khóa, đơn hàng, có thể thêm và xóa thợ sửa khóa, cập nhật thông tin thợ, đơn hàng, xem các thống kê doanh thu, lập báo cáo doanh thu.
- Thợ sửa khóa được hỗ trợ tối đa nhờ hệ thống định vị GPS, giúp tìm ra địa điểm nhanh chóng, chính xác.
- Bảo mật thông tin khách hàng, thông tin đơn hàng và cả thông tin thợ sửa khóa.

Hướng phát triển cho ứng dụng:

- Ứng dụng sẽ có thêm các chức năng như: Nhắn tin – gọi điện trao đổi giữa thợ sửa khóa và khách hàng.
- Ứng dụng phải được bổ sung thêm chức năng quản lý các hình ảnh do khách hàng cung cấp
- Tọa độ - Vị trí của thợ sửa khóa phải được cập nhật sau mỗi lần sửa khóa xong.
- Ứng dụng hỗ trợ việc thanh toán trực tuyến.

## TÀI LIỆU THAM KHẢO

[1] React native: https://reactnative.dev/

[2] Firebase: https://firebase.google.com/

[3] ReactJS: https://react.dev/

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F025**

# XÂY DỰNG HỆ THỐNG QUẢN LÝ KHO THUỐC THÔNG MINH

NGUYỄN THỊ HẠNH[1], NGUYỄN HOÀNG SANG[1*], VĂN ANH KHOA[1]

*[1]Khoa Công nghệ thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*ngthihanh@iuh.edu.vn, sangnguyen1596320@gmail.com, anhkhoavan888@gmail.com*

**Tóm tắt.** Quản lý kho thuốc là thách thức lớn trong lĩnh vực quản lý kho hàng, yêu cầu độ chính xác và đáng tin cậy cao để đảm bảo an toàn và hiệu quả. Nhiều doanh nghiệp vẫn đang phải đối mặt với khó khăn và bất cập trong quản lý kho hàng. Để giải quyết bất cập trong quản lý kho thuốc, hệ thống quản lý kho thuốc thông minh là giải pháp hiệu quả cho doanh nghiệp của bạn. Với các ngôn ngữ Java, Servlet, Bootstrap và JavaScript là những ngôn ngữ phổ biến và mạnh mẽ trong việc phát triển phần mềm web. Hệ thống triển khai theo mô hình MVC (Model-View-Controller) cũng giúp tăng tính ổn định và dễ bảo trì của phần mềm, đồng thời cải thiện trải nghiệm người dùng và nâng cao hiệu quả trong công việc quản lý kho thuốc. Hệ thống quản lý kho thuốc thông minh nổi bật với chức năng quản lý nhập và xuất hàng. Bên cạnh đó, hệ thống còn tích hợp tính năng thống kê và báo cáo giúp quản lý kho trở nên dễ dàng hơn. Hệ thống này có thể áp dụng cho các doanh nghiệp trong lĩnh vực kho thuốc và chuỗi cung ứng.

**Từ khóa.** Phần mềm quản lý kho, Logistics thông minh, Quản lý hàng hóa hiệu quả, Lĩnh vực kho thuốc và chuỗi cung ứng, Tối ưu hoá quá trình vận chuyển hàng hóa.

## BUILDING A SMART DRUG WAREHOUSE MANAGEMENT SYSTEM

**Abstract.** Drug warehouse management is a major challenge in the field of warehouse management, requiring high accuracy and reliability to ensure safety and efficiency. Many businesses are still facing difficulties and inadequacies in warehouse management. To solve inadequacies in drug warehouse management, a smart drug warehouse management system is an effective solution for your business. Java, Servlet, Bootstrap, and JavaScript are popular and powerful languages in web software development. The system is deployed according to the MVC model (Model-View-Controller) deployment system also helps to increase the stability and ease of maintenance of the software, while improving user experience and improving efficiency in drug warehouse management. Smart drug warehouse management system stands out with import and export management functions. Besides, the system also integrates statistical and reporting features to make warehouse management easier. This system can be applied to businesses in the field of drug warehouse and supply chain.

**Keywords.** Warehouse Management Software, Smart Logistics, Efficient Goods Management, Pharmaceutical and supply chain industries, Optimizing transportation process.

## 1 GIỚI THIỆU TỔNG QUAN

Ngày nay, với sự phát triển về mọi mặt của đời sống từ văn hóa, giáo dục cho đến công nghệ và đặc biệt, lĩnh vực công nghệ thông tin sẽ có những bước phát triển chóng mặt. Việc ứng dụng công nghệ thông tin vào các hoạt động kinh doanh là một nhu cầu thiết yếu trong cuộc sống. Nhận thấy được tầm quan trọng của việc ứng dụng công nghệ thông tin vào hoạt động kinh doanh và quản lý nhanh chóng và tiện lợi hơn gấp nhiều lần so với việc quản lý truyền thống trước đó. Vì vậy, website quản lý kho thuốc thông minh ra đời với mong muốn thay đổi theo sự phát triển của thời đại nên việc quản lý cần được nâng cao và đáp ứng được nhu cầu cao hơn ứng dụng công nghệ thông tin vào hoạt động kinh doanh quản lý kho thuốc thông minh. Mục tiêu chính của quản lý kho thuốc thông minh là tối ưu hóa quá trình quản lý và điều phối các hoạt động lưu kho của các sản phẩm y tế, đồng thời quản lý một cách dễ dàng, không còn mất thời gian để

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

ghi chép và giảm thiểu lỗi trong quá trình quản lý kho, nâng cao tính nhanh chóng và hiệu quả, tiết kiệm được thời gian, chi phí.

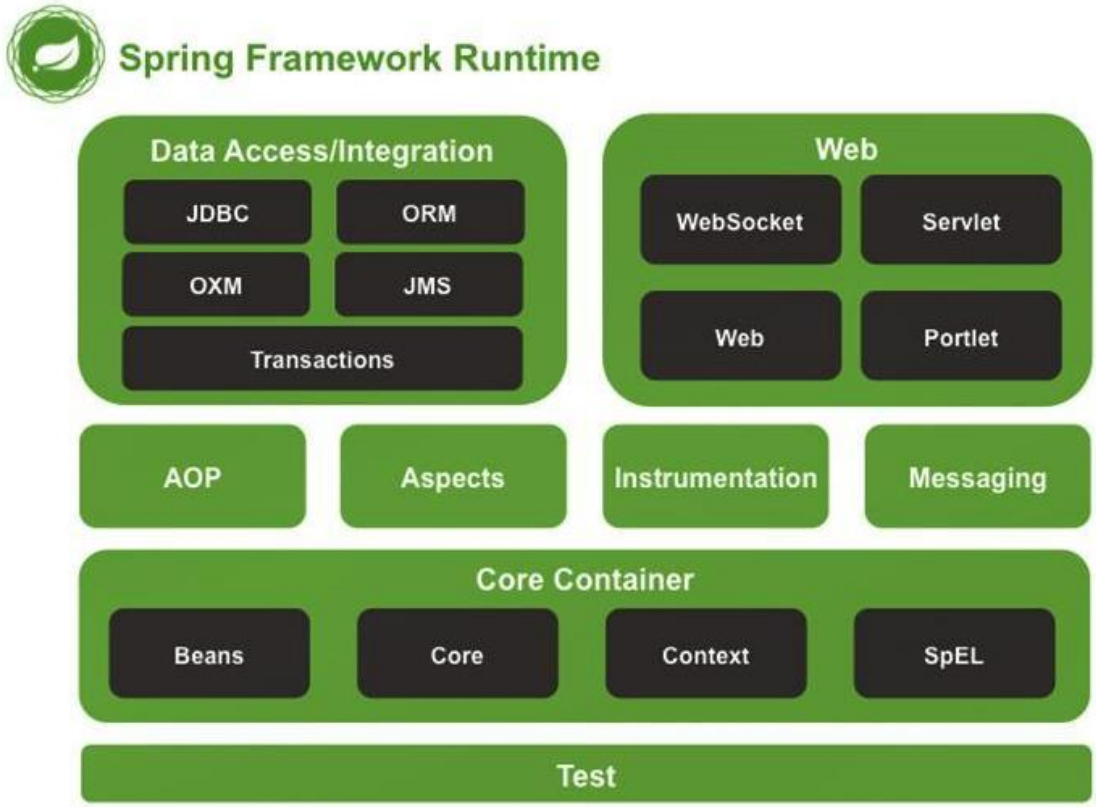
Đầu tiên website sẽ có các thông tin các kho và các phiếu hóa đơn để có thể quản lý các sản phẩm ra vào trong kho như là giá cả, sản phẩm, thời gian, ... về việc quản lý kho thuốc thì sẽ có quản lý nhiều kho khác nhau, trong một kho thì sẽ có nhiều kệ hàng chứa các thùng hàng, mỗi kệ hàng sẽ có các ô để chứa các thùng hàng đó. Và nhập sản phẩm vào thì hệ thống sẽ tự động cho sản phẩm đó vào ô của kệ có danh mục tương ứng với sản phẩm, cập nhật số lượng tồn và sẽ cập nhật tình trạng của ô đó theo trạng thái: chưa đầy, đã đầy. Việc xuất ra thì sẽ chọn kho cần xuất và sau đó hệ thống sẽ tự động lấy hàng ra từ ô của kệ có danh mục tương ứng với sản phẩm cần xuất ra, cập nhật số lượng tồn và sẽ cập nhật tình trạng của ô đó theo hai trạng thái: còn trống, chưa đầy.

Màn hình Dashboard sẽ hiển thị tổng quát các ô của các kệ thuộc kho đó, cập nhật màu sắc tương ứng với ba trạng thái: đã đầy (màu cam), chưa đầy (màu cam nhạt), trống (màu xám) để nhân viên có một cái nhìn cụ thể về kho của mình, giúp cho quá trình giám sát chính xác hơn, cụ thể hơn.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 Spring Servlet Framework

Spring Servlet Framework là một phần trong Spring Framework [1], là một bộ công cụ phát triển ứng dụng Java phổ biến và mạnh mẽ. Nó cung cấp một mô hình phát triển ứng dụng web dựa trên Servlet và JSP, giúp đơn giản hóa việc xây dựng các ứng dụng website.




Hình 1: Kiến trúc MVC

Spring Servlet Framework đặc biệt tập trung vào triển khai mô hình MVC (Model-View-Controller) để phân tách logic xử lý, giao diện và dữ liệu. Nó cung cấp các thành phần quan trọng như DispatcherServlet, HandlerMapping, Controller và ViewResolver để quản lý luồng xử lý trong ứng dụng web [2].

### 2.2 SQL Server 2019

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ phổ biến do Microsoft phát triển [3]. Nó cung cấp một môi trường để lưu trữ, quản lý và truy vấn dữ liệu. SQL Server sử dụng ngôn ngữ truy vấn SQL (Structured Query Language) để tương tác và quản lý cơ sở dữ liệu.

### 2.3 FontAwesome

FontAwesome là một bộ biểu tượng (icon) phổ biến và mạnh mẽ được sử dụng trong phát triển web [4]. Nó cung cấp hơn 7.000+ biểu tượng đẹp và linh hoạt, từ các biểu tượng đơn giản như hình tròn, hình vuông đến các biểu tượng phức tạp như logo mạng xã hội, biểu tượng nút bấm, các biểu tượng hình ảnh và văn bản khác.

FontAwesome cho phép lấy và sử dụng các biểu tượng thông qua CSS hoặc JavaScript. Nó cung cấp một thư viện CSS và một bộ font chữ đặc biệt để hiển thị biểu tượng. Bằng cách thêm các lớp CSS hoặc mã HTML phù hợp, có thể dễ dàng chèn các biểu tượng vào trang web.

### 2.4 Bootstrap

Bootstrap là một framework phát triển web phổ biến và mạnh mẽ [5]. Nó cung cấp một bộ công cụ và tài nguyên giúp xây dựng các giao diện web đáp ứng, tương thích trên nhiều thiết bị và trình duyệt khác nhau.

### 2.5 HTML/CSS

HTML/CSS đều là hai ngôn ngữ lập trình. Trong khi HTML (HyperText Markup Language – ngôn ngữ đánh dấu) được dùng để xây dựng cấu trúc cho từng phần nội dung thì CSS (Cascading Style Sheet- ngôn ngữ định dạng theo từng lớp) được dùng để tạo định dạng hiển thị cho HTML [6]. Nói cách khác, HTML là công cụ để thêm nội dung và mô tả ý nghĩa cho bố cục, còn CSS là công cụ để trang trí, thiết lập cỡ chữ, màu sắc, các kiểu chữ, hiệu ứng đơn giản… cho giao diện trang web.

### 2.6 JavaScript

JavaScript là một ngôn ngữ lập trình thông dịch, phổ biến và mạnh mẽ được sử dụng chủ yếu để tạo ra các trang web tương tác và ứng dụng web động. JavaScript cho phép thêm các tính năng và chức năng động vào trang web bằng cách tương tác với các phần tử HTML, xử lý sự kiện và thay đổi nội dung trang mà không cần tải lại toàn bộ trang [7].

Nói cách khác, HTML là công cụ để thêm nội dung và mô tả ý nghĩa cho bố cục, còn CSS là công cụ để trang trí, thiết lập cỡ chữ, màu sắc, các kiểu chữ, hiệu ứng đơn giản… cho giao diện trang web.

## 3 SK+: QUẢN LÝ KHO THUỐC THÔNG MINH

### 3.1 Kiến trúc phần mềm

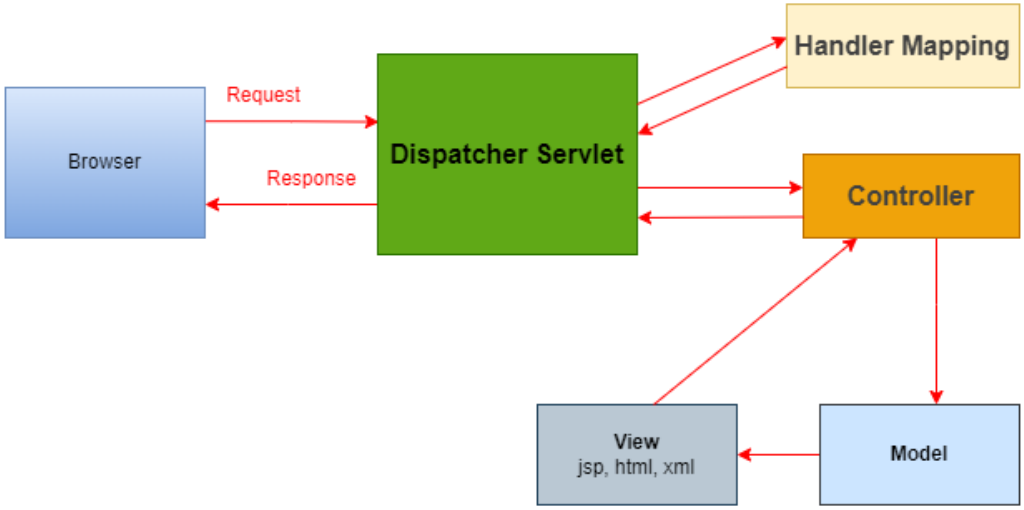
Chúng tôi sử dụng kiến trúc MVC(Model-View-Controller) là một mô hình phát triển phần mềm được sử dụng rộng rãi trong việc xây dựng hệ thống website [8]. Nó tách biệt logic xử lý dữ liệu (Model), giao diện người dùng (View) và điều khiển luồng ứng dụng (Controller) thành ba thành phần độc lập, nhằm tăng tính tái sử dụng, dễ bảo trì và phát triển độc lập giữa các thành phần (Hình 2).

Model (M): Đại diện cho dữ liệu và logic xử lý dữ liệu. Nó có nhiệm vụ truy xuất, xử lý và cung cấp dữ liệu cho các thành phần khác của ứng dụng. Model không phụ thuộc vào View và Controller, giúp tách biệt logic dữ liệu khỏi giao diện người dùng.

View (V): Đại diện cho giao diện người dùng. Nó hiển thị dữ liệu từ Model và cung cấp cách người dùng tương tác với ứng dụng. View không chứa logic xử lý dữ liệu, chỉ đảm nhận nhiệm vụ hiển thị và cập nhật giao diện.

Controller (C): Là thành phần điều khiển luồng của ứng dụng. Nó nhận các yêu cầu từ người dùng thông qua giao diện (View), xử lý và điều hướng yêu cầu đến Model tương ứng, sau đó cập nhật lại View để hiển thị kết quả. Controller là trung gian giữa Model và View, không chứa logic xử lý dữ liệu chi tiết.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 2: Kiến trúc của hệ thống quản lý kho thuốc thông minh SK+

### 3.2 Phân tích hệ thống SK+

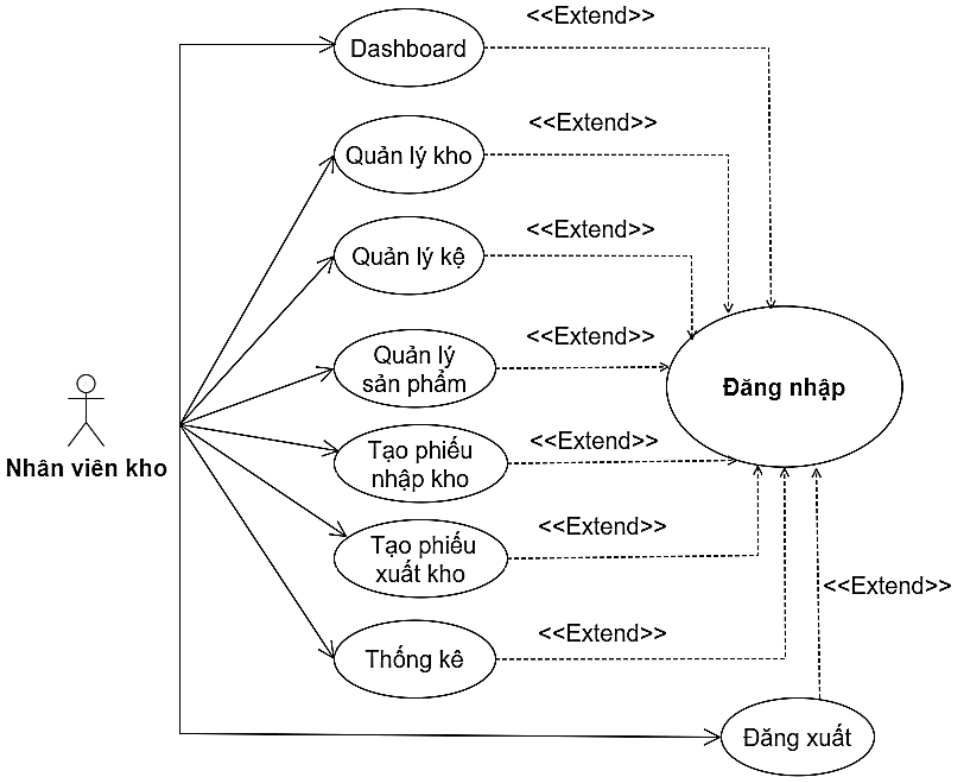
Hệ thống SK+ hoạt động nội bộ, đối tượng chính là nhân viên kho. Các chức năng được liệt kê trong bảng sau (Bảng 1).

Bảng 1: Danh sách các chức năng trong hệ thống SK+

| Đối tượng | Tên chức năng | Mô tả |
|---|---|---|
| Nhân viên kho | Đăng nhập | Cho phép nhân viên kho đăng nhập với tài khoản đã được cấp |
| | Dashboard | Cho phép nhân viên kho quản lý chi tiết các kệ của các kho |
| | Quản lý kho | Cho phép nhân viên kho quản lý các kho |
| | Quản lý kệ | Cho phép nhân viên kho quản lý các kệ |
| | Quản lý sản phẩm | Cho phép nhân viên kho quản lý các sản phẩm |
| | Tạo phiếu nhập | Cho phép nhân viên kho tạo các phiếu nhập hàng vào các kho |
| | Tạo phiếu xuất | Cho phép nhân viên kho tạo các phiếu xuất hàng từ các kho |
| | Thống kê | Cho phép nhân viên kho xem thống kê của các kho |

Trong mô hình use-case chúng tôi trình bày, người dùng chính là nhân viên kho. Trong đó, những use-case chính và phức tạp về nghiệp vụ đòi hỏi nhiều xử lý cho hệ thống đó là: use-case tạo phiếu nhập và use-case tạo phiếu xuất. Dựa trên những chức năng đã nêu, chúng tôi tiến hành thiết kế mô hình quan hệ cho hệ thống (Hình 3).

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 3: Mô hình Use-case của SK+

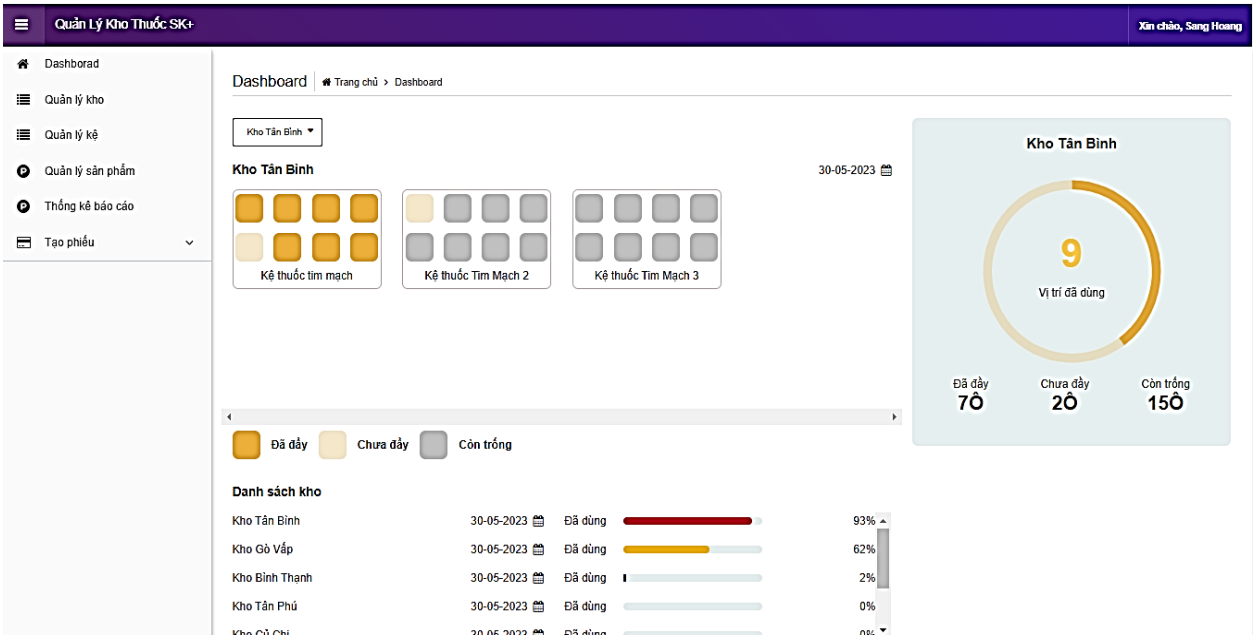
Trong hệ thống quản lý kho thuốc thông minh của chúng tôi cũng nói đến quy trình nhập hàng (Hình 4), xuất hàng (Hình 5).
Quy trình nhập kho bao gồm 6 bước:
+ Bước 1: Khi mua hàng về, nhân viên kho sẽ tiếp nhận đơn hàng đó.
+ Bước 2: Nhân viên kho sẽ tiến hành tạo phiếu nhập kho dựa trên đơn hàng đó.
+ Bước 3: Nhập thông tin sản phẩm tương ứng.
+ Bước 4: Nhập thông tin nhà cung cấp và chọn kho hàng cần nhập hàng.
+ Bước 5: Nhân viên tạo phiếu nhập hàng thành công.
+ Bước 6: Đơn hàng sẽ tự động chuyển đến kệ tương ứng với danh mục sản phẩm tại kho đó.
Quy trình nhập kho bao gồm 7 bước:
+ Bước 1: Nhân viên kho sẽ tiếp nhận đơn hàng xuất kho.
+ Bước 2: Nhân viên kho sẽ tiến hành tạo phiếu xuất kho dựa trên đơn hàng đó.
+ Bước 3: Nhân viên chọn kho tương ứng cần xuất hàng.
+ Bước 4: Nhập thông tin sản phẩm tương ứng.
+ Bước 5: Nhập thông tin khách hàng.
+ Bước 6: Nhân viên tạo phiếu xuất kho thành công.
+ Bước 7: Hệ thống sẽ tự động lấy đơn hàng từ kệ tương ứng với danh mục sản phẩm tại kho đó.

## 3.3 Hiện thực hệ thống

### 3.3.1 Dashboard

Chức năng này giúp cho nhân viên kho có 1 cái nhìn trực quan hơn, dễ kiểm soát và quản lý chi tiết các kệ đó đang chứa sản phẩm nào của kho đó.

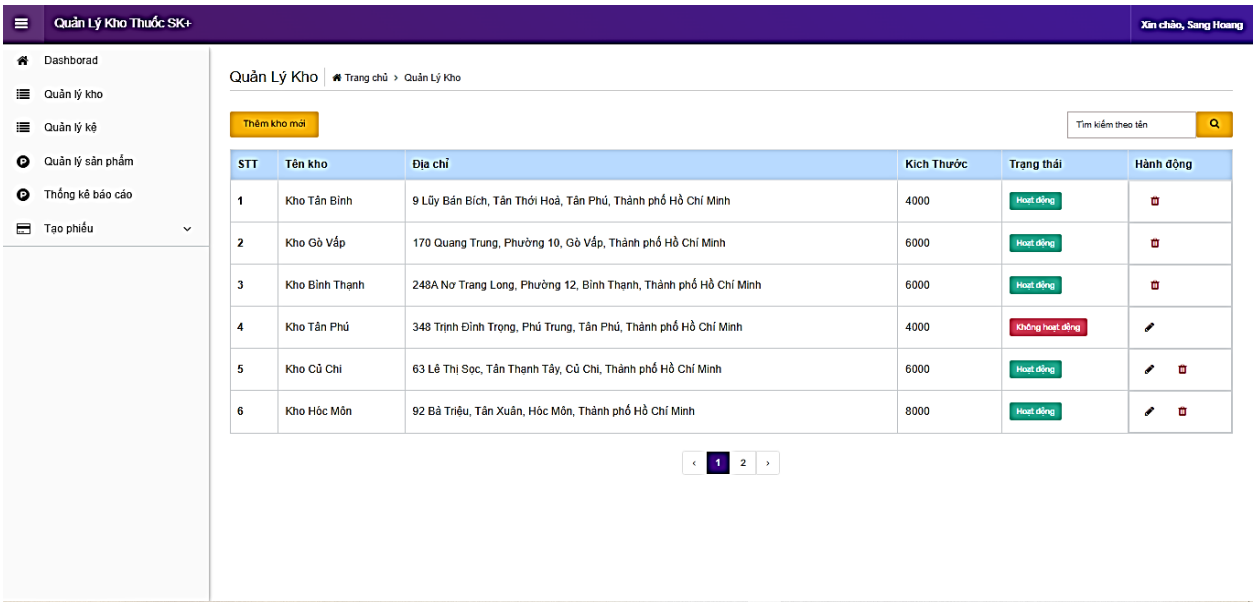
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 4: Giao diện Dashborad

### 3.3.2 Quản lý kho

Chức năng này giúp cho nhân viên kho có thể quản lý các thông tin của kho và có thể thêm sửa xóa kho. Ngoài ra có thể tìm kiếm kho.

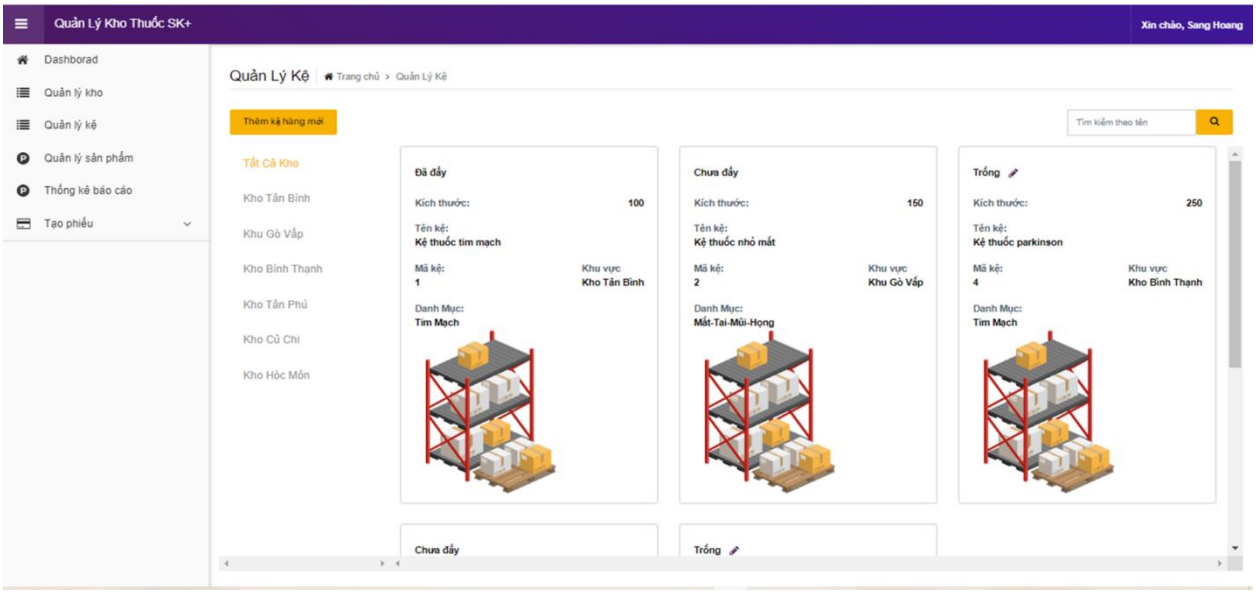



Hình 5: Màn hình quản lý kho

### 3.3.3 Quản lý kệ

Chức năng này giúp cho nhân viên kho có thể quản lý các thông tin của kệ và có thể thêm sửa xóa kho. Ngoài ra có thể tìm kiếm các kệ thuộc kho nào.

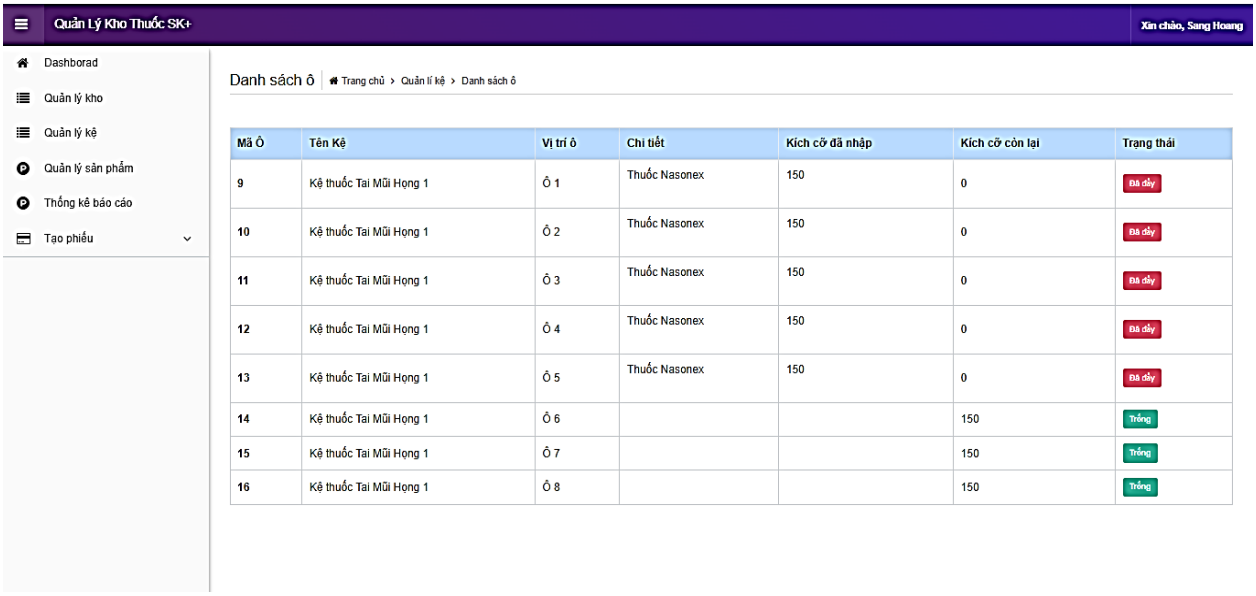
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 6: Màn hình quản lý kệ

### 3.3.4 Quản lý chi tiết ô

Chức năng này giúp cho nhân viên kho có thể quản lý chi tiết thông tin của kệ đó đang chứa sản phẩm của kho đó.

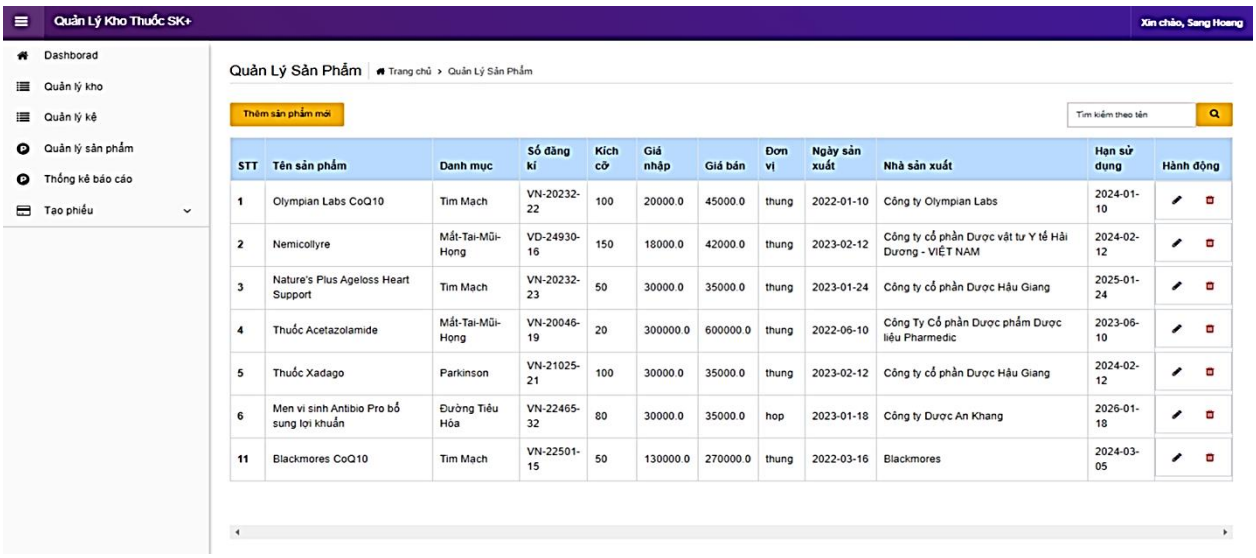



Hình 7: Màn hình chi tiết ô

### 3.3.5 Quản lý sản phẩm

Chức năng này giúp cho nhân viên kho có thể quản lý các sản phẩm, có thể thêm sửa xóa hoặc tìm kiếm sản phẩm.

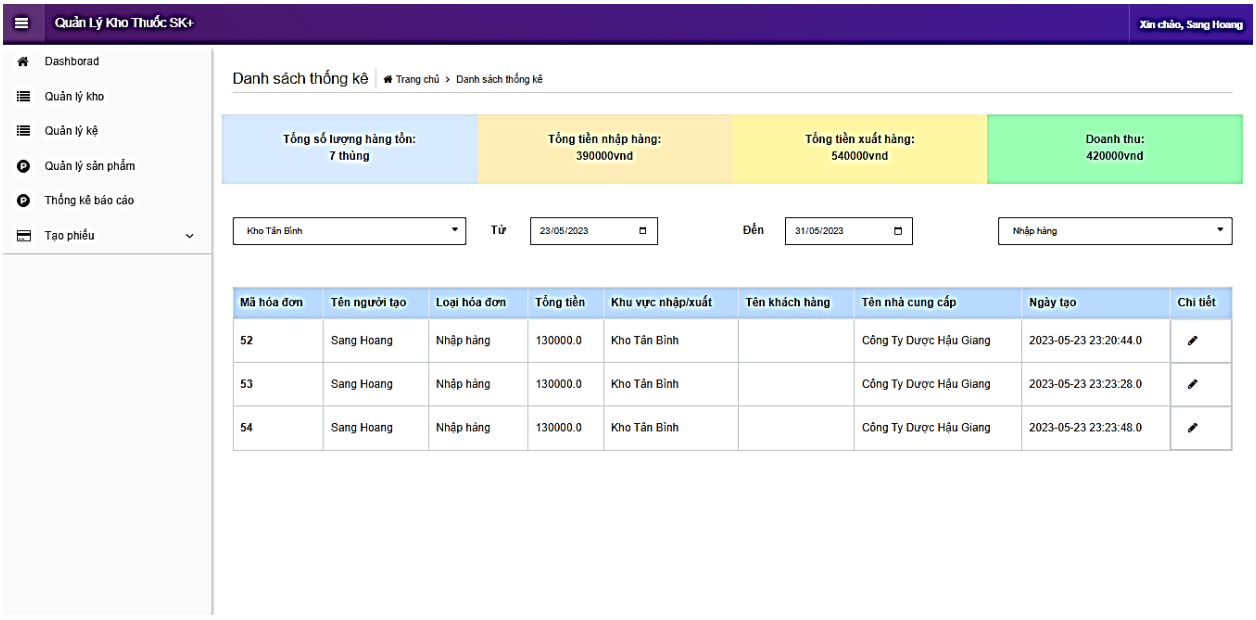

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 8: Màn hình quản lý sản phẩm

### 3.3.6 Thống kê báo cáo

Chức năng này giúp cho nhân viên kho có thể xem thống kê của các kho.

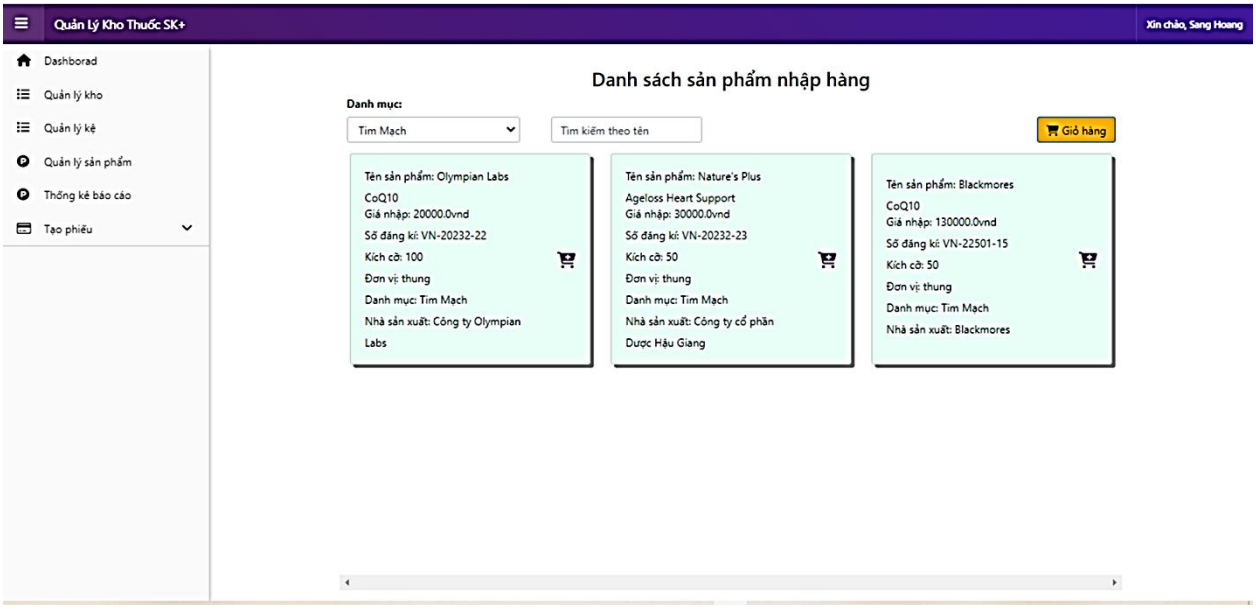



Hình 9: Màn hình danh sách thống kê

### 3.3.7 Tạo phiếu nhập

Chức năng này giúp cho nhân viên kho có thể tạo phiếu nhập hàng gồm thông tin sản phẩm và đơn vị nhập hàng đến kho cần nhập hàng.

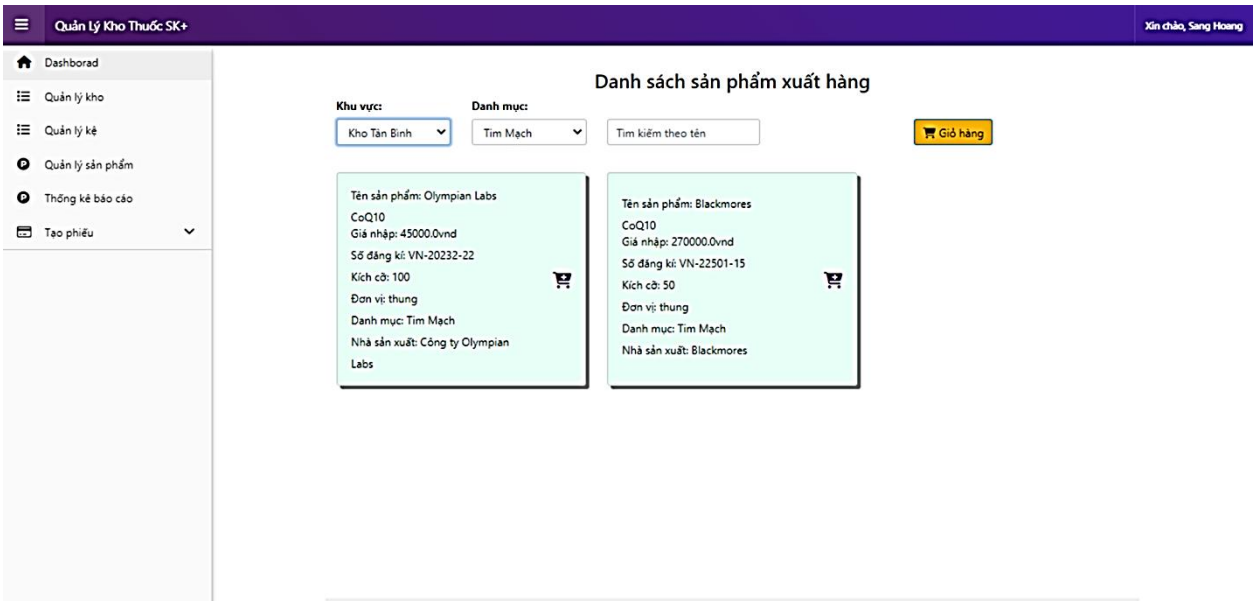
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 10: Màn hình phiếu nhập

### 3.3.7 Tạo phiếu xuất

Chức năng này giúp cho nhân viên kho có thể tạo phiếu xuất hàng gồm thông tin sản phẩm và đơn vị xuất hàng từ kho cần xuất hàng.




Hình 11: Màn hình phiếu xuất

## 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Bài báo này chúng tôi đã đề xuất giải pháp cải thiện quy trình quản lý kho thông thường bằng hệ thống quản lý kho thuốc thông minh SK+. Trong bài báo này, chúng tôi đã cơ bản giải quyết được hai vấn đề trong việc quản lý kho theo cách thông thường. Đầu tiên là hệ thống có thể quản lý từ xa các kệ đó đang chứa sản phẩm và hiển thị màu tương ứng với trạng thái kệ đó, có thể xem các kệ của các kho khác nhau. Thứ hai là hệ thống nhập hàng và xuất hàng 1 cách tự động, nó sẽ tự động nhập hàng hoặc xuất hàng tương ứng với kệ sau khi tạo phiếu nhập và xuất thành công. Và chúng tôi cũng đã trình bày các giải pháp bao gồm chức năng dashboard, quản lý kho, quản lý kệ, quản lý sản phẩm, thống kê, tạo phiếu nhập và tạo

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

phiếu xuất nhằm hỗ trợ nghiệp vụ cho nhân viên kho. Chúng tôi cũng tìm hiểu thêm được nhiều ngôn ngữ lập trình như Servlet, Javascript, Java. Tuy nhiên, hệ thống quản lý kho thuốc thông minh SK+ vẫn còn nhiều hạn chế về mặt chức năng và quy mô. Ngoài những cải tiến đã được đề cập, tương lai hệ thống có thể được mở rộng để áp dụng cho các hoạt động mua hàng và bán hàng của doanh nghiệp. Đầu tiên có thể sử dụng mã QR để quản lý sản phẩm. Mỗi sản phẩm có thể được gắn mã QR độc nhất, cho phép nhân viên dễ dàng quét mã QR để truy xuất thông tin chi tiết về sản phẩm và quản lý tồn kho một cách chính xác. Thứ hai là tích hợp mã QR vào quy trình thanh toán: Khách hàng có thể quét mã QR để thực hiện thanh toán, giúp giảm thời gian và đơn giản hóa quy trình thanh toán. Một hướng mở rộng tiềm năng là tích hợp hệ thống mua hàng tự động, cho phép hệ thống SK+ tự động đặt hàng từ các nhà cung cấp dựa trên dữ liệu tồn kho và mức tiêu thụ hàng hóa của kho đó. Điều này giúp tối ưu hóa quy trình mua hàng và đảm bảo rằng sản phẩm luôn có sẵn khi cần thiết. Bên cạnh đó có thể tích hợp thêm thông báo tự động: Khi mức tồn kho của một sản phẩm xuống dưới mức quy định, hệ thống có thể tự động gửi thông báo đến nhân viên để đặt hàng mới. Thêm vào đó, một hướng mở rộng khác là tích hợp thanh toán trực tuyến vào hệ thống, cho phép khách hàng thanh toán mua hàng trực tiếp thông qua ứng dụng hoặc trang web của doanh nghiệp. Điều này mang lại sự tiện lợi và tăng cường trải nghiệm mua hàng cho khách hàng, đồng thời giúp doanh nghiệp nâng cao hiệu suất và tăng cường doanh thu. Với những hướng mở rộng này, tương lai hệ thống quản lý kho thuốc thông minh SK+ đang hướng tới việc tối ưu hóa quy trình mua hàng và bán hàng của doanh nghiệp, cung cấp sự linh hoạt, hiệu quả và tạo ra một trải nghiệm tốt hơn cho cả khách hàng và doanh nghiệp.

## TÀI LIỆU THAM KHẢO

[1]. Jason Hunter (2001), "Java Servlet Programming", O'Reilly Media, Sebastopol, California

[2]. "Java Servlet Specification, version 4.0", Oracle Corporation

[3]. SQL Server: [online] https://www.w3schools.com/sql/default.asp [Accessed 08/01/2023]

[4]. FontAwesome: [online] https://fontawesome.com/ [Accessed 10/01/2023]

[5]. Bootstrap: [online] https://www.w3schools.com/bootstrap/bootstrap_get_started.asp [Accessed 05/01/2023]

[6]. HTML/CSS: [online] https://www.w3schools.com/html/html_css.asp [Accessed 25/03/2023]

[7]. JavaScript: [online] https://www.w3schools.com/js/default.asp [Accessed 02/04/2023]

[8]. Mô hình MVC: [online] https://vietnix.vn/tim-hieu-mo-hinh-mvc-la-gi/ [Accessed 09/02/2023]

[9]. Quy trình kho: [online] https://www.topcv.vn/quy-trinh-quan-ly-kho-can-biet [Accessed 09/02/2023]

[10]. Tài liệu Servlet: [online] https://vietjack.com/servlets/tai_lieu_tham_khao_ve_servlet.jsp [Accessed 24/02/2023]

[11]. Kho hàng: [online] https://gulfshipping.com.vn/kho-hang-la-gi-vai-tro-va-chuc-nang-cua-kho-hang/?lang=vi [Accessed 13/02/2023]

[12]. Tổng quan về kho: [online] https://kekhotrungtai.com/kho-hang [Accessed 18/02/2023]

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F026**

# TÌM LẠI NỤ CƯỜI XƯA

HỒ VÕ HOÀNG DUY[1], VÕ QUỐC HUY[1], NGUYỄN XUÂN GIANG[1], NGUYỄN QUANG BẢO[1], DƯƠNG QUANG HUY[1], ĐỖ THU ĐÔNG[1], BÙI THANH HÙNG[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*

*h.hoangduy2002@gmail.com, huyvo8500@gmail.com,nxgiang2202@gmail.com, nguyenquangbao08@gmail.com,dhuy9954@gmail.com, buithanhhung@iuh.edu.vn*

**Tóm tắt.** Tạo video từ hình ảnh khuôn mặt có ý nghĩa quan trọng trong việc tạo ra các video tự nhiên từ một bức ảnh, được ứng dụng rộng rãi trong nhiều lĩnh vực như làm phim ảnh hay truyền thông xã hội. Các phương pháp trước đó, còn tồn tại một số hạn chế như tạo ra các video có chất lượng tốt nhưng thiếu tự nhiên trong việc tái tạo lại cử chỉ của nhân vật, một số nghiên cứu chỉ tập trung vào việc tạo các video chất lượng cao mà làm mất đi sự đa dạng về nội dung và phong cách của bức ảnh. Trong nghiên cứu này chúng tôi đề xuất phương pháp tạo ra một video ngắn có chuyển động tự nhiên của môi, mắt và các phần có liên quan trên khuôn mặt bằng cách sử dụng kỹ thuật học sâu, mạng nơ ron tích chập CNN, phép chuyển đổi Hidden Affine kết hợp với Conditional Generative Adversarial Network (cGAN) và một số phương pháp về xử lý ảnh và thị giác máy tính. Chúng tôi sẽ đánh giá trên các bộ dữ liệu CK-Mixed và so sánh với các phương pháp nghiên cứu khác. Dựa trên những kết quả đạt được chúng tôi sẽ tiến hành xây dựng ứng dụng có thể tạo một video có chuyển động khuôn mặt từ một bức ảnh đầu vào và đưa vào thử nghiệm ứng dụng trong thực tiễn.

**Từ khóa.** Tạo video khuôn mặt cười, Deep learning, CNN, Hidden Affine transformation, cGAN, Thị giác máy tính.

## FINDING A SMILE OF THE PAST

**Abstract.** Creating a video from facial images plays a significant role in generating natural-looking videos from a single image. This technique is widely used in various fields such as filmmaking and social media. Previous methods have had limitations, such as creating high-quality videos lacking naturalness in reproducing character movements. Some studies have focused solely on producing high-quality videos, resulting in a loss of diversity in the content and style of the image. In this study, we propose a method to create a short video with natural facial movements of the lips, eyes, and related facial parts using deep learning techniques, convolutional neural networks (CNN), Hidden Affine transformation combined with Conditional Generative Adversarial Network (cGAN), image processing technique and computer vision methods. We will evaluate this method on CK-Mixed datasets and compare it with other research methods. Based on the results, we will develop an application that can create a facial motion video from a single input image and test its practicality.

**Keywords.** Smiling face video, Deep learning, CNN, Hidden Affine transformation, cGAN, Computer vision.

## 1 GIỚI THIỆU

Trong những năm gần đây, lĩnh vực thị giác máy tính đã chứng kiến những tiến bộ đáng kể, cho phép máy tính nhận biết và hiểu thông tin hình ảnh với độ chính xác ngày càng cao. Một ứng dụng hấp dẫn của công nghệ thị giác máy tính là khả năng chuyển đổi hình ảnh tĩnh thành video động.

Chuyển đổi hình ảnh tĩnh thành video động mang lại những tiềm năng ứng dụng rộng rãi trong lĩnh vực tạo nội dung đa phương tiện, thực tế ảo, chỉnh sửa video và các công nghệ hỗ trợ, làm giàu trải nghiệm người

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

xem và mở ra những khả năng sáng tạo mới. Đồng thời, cũng góp phần mở ra những cánh cửa mới trong việc khai thác tiềm năng của công nghệ thị giác máy tính để tạo ra nội dung sáng tạo và thu hút.
Trong nghiên cứu này, chúng tôi tập trung vào việc tạo ra một video từ một bức ảnh duy nhất, bao gồm các chuyển động của môi, mắt và các phần có liên quan trên khuôn mặt. Chúng tôi đề xuất một phương pháp sáng tạo và hiệu quả để tổng hợp video động từ một bức ảnh tĩnh, tích hợp chuyển động môi, mắt và các phần có liên quan trên khuôn mặt vào video tổng hợp để tạo ra sự tự nhiên và chân thực. Các đóng góp của chúng tôi trong nghiên cứu này bao gồm các nội dung sau:
- Chúng tôi phát triển một phương pháp dựa trên kỹ thuật học sâu kết hợp phương pháp biến đổi Hidden Affine Transformation và kỹ thuật xử lý ảnh để có thể tổng hợp một chuỗi video động từ một bức ảnh tĩnh duy nhất, trong đó tích hợp một cách mượt mà các chuyển động của môi, mắt và các phần có liên quan trên khuôn mặt để tạo ra video từ một hình ảnh tĩnh một cách chân thực và sinh động.
- Chúng tôi sử dụng mô hình học sâu cGAN để học các phép biến đổi Affine, các tham số Affine này được tính toán dựa trên các thông tin từ ảnh đầu vào như: ảnh khuôn mặt và được sử dụng để tạo ra các khung hình sau đó tổng hợp thành video có chuyển động môi, mắt và các phần có liên quan trên khuôn mặt.
- Chúng tôi tiến hành các thực nghiệm và đánh giá đầy đủ trên các tập dữ liệu đa dạng, nhằm chứng minh tính hiệu quả và độ ổn định của mô hình đề xuất của mình.
- Chúng tôi so sánh phương pháp của chúng tôi với các phương pháp hiện có và đánh giá kết quả theo chất lượng hình ảnh, tính thực tế và sự liên kết.
Ngoài phần giới thiệu, phần còn lại của bài báo bao gồm các nội dung sau. Phần 2 trình bày về các nghiên cứu có liên quan. Mô hình đề xuất của chúng tôi được trình bày ở Phần 3. Phần 4 trình bày về thực nghiệm và đánh giá so sánh kết quả của chúng tôi với các phương pháp khác. Phần 5 trình bày về kết luận về nghiên cứu của chúng tôi và hướng phát triển trong tương lai.

## 2 CÁC NGHIÊN CỨU LIÊN QUAN

Tạo video từ hình ảnh là quá trình biến đổi một loạt các ảnh tĩnh thành một chuỗi các khung hình liên tiếp nhằm tạo ra một video chuyển động của các vật thể có trong bức ảnh. Có nhiều phương pháp để giải quyết bài toán này, đa số các phương pháp sử dụng phương pháp học sâu vì tính hiệu quả của phương pháp này. Sử dụng mô hình cINN và các phương pháp khác như phép biến đổi Affine, học sâu phi tuyến và mô hình autoregressive để tổng hợp video từ các hình ảnh đầu vào [1]. Tuy nhiên phương pháp trên tồn tại một số hạn chế như khả năng giới hạn của mô hình ước lượng, và khó khăn trong việc xử lý các phân bố lớn, phức tạp.
Yang Zhou và cộng sự [2] đã giới thiệu một phương pháp sử dụng học sâu để tạo ra video động của người nói với môi chuyển động mượt mà và diễn đạt hình ảnh chân dung tự nhiên. Mô hình có thể tạo ra hoạt hình biểu diễn sự đồng bộ của miệng, biểu cảm khuôn mặt cá nhân và động tác đầu một cách tốt hơn so với công nghệ tiên tiến hiện tại, song mô hình trên vẫn còn tồn tại một số điểm hạn chế không thể sinh ra các hình ảnh người nói chuyện với độ phân giải cao.
Guangyao Shen và cộng sự [3] đề xuất một phương pháp để chuyển đổi ảnh khuôn mặt đến video bằng cách sử dụng một phép biến đổi affine ẩn. Phương pháp này được thực hiện bằng cách sử dụng một mô hình học sâu có tên là AffineGAN, với kiến trúc dựa trên mô hình GAN (Generative Adversarial Networks). Phương pháp này có thể tạo ra các video mới với nhiều biểu cảm khác nhau trên khuôn mặt, và tạo ra các video mượt hơn, có độ chân thực cao hơn. Hạn chế của phương pháp này là cần sự chuẩn bị dữ liệu cẩn thận; giới hạn của biến đổi affine; phụ thuộc vào chất lượng dữ liệu đầu vào; khả năng tổng quát hóa và mất mát thông tin.
Haomiao Ni và cộng sự [4] đề xuất mô hình Latent Flow Diffusion Models (LFDMs) là một lớp mô hình sinh dựa trên dòng latent (latent flow), được sử dụng để mô phỏng phân phối dữ liệu phức tạp như ảnh và video. Mô hình đề xuất có thể tạo ra video bằng cách biến đổi hình ảnh cho trước bằng các chuỗi flow được tạo ra trong không gian ẩn dựa trên điều kiện lớp. Hạn chế của mô hình này là LFDM giới hạn trong việc xử lý video chứa một đối tượng di chuyển duy nhất; LFDM hiện tại được điều kiện dựa trên nhãn lớp thay vì mô tả tự nhiên bằng văn bản; so với các mô hình GAN, LFDM chậm hơn rất nhiều khi lấy mẫu với DDPM 1000 bước.
Ming-Yu Liu và cộng sự [5] đề xuất mô hình tạo ra video từ các hình ảnh đơn lẻ. Quá trình này có thể bao gồm việc sinh ra các khung hình trung gian giữa các khung hình có sẵn, để tạo ra dự báo liên tục về chuyển động. Ưu điểm của phương pháp này là cung cấp khả năng tạo ra video động từ các hình ảnh đơn lẻ, mở

rộng khả năng tạo ra video từ dữ liệu hình ảnh thưa; tạo ra các khung hình video có tính chân thực và liên tục về chuyển động. Tuy nhiên, phương pháp này tồn tại một số hạn chế như đòi hỏi lượng dữ liệu huấn luyện lớn và tính toán phức tạp; cần tinh chỉnh tham số và quá trình huấn luyện kỹ lưỡng để đạt được chất lượng video tốt; có thể gặp khó khăn trong việc tái tạo các chi tiết và chuyển động tự nhiên trong video.
Long Zhao và cộng sự [6] giới thiệu một phương pháp sinh video từ ảnh đầu vào bằng cách dự đoán và cải tiến chuyển động dư thừa. Mô hình trên có thể dự đoán chuyển động dư thừa, có khả năng tạo ra video chất lượng cao. Điểm hạn chế của phương pháp này là phụ thuộc vào khung hình gốc; độ phức tạp tính toán lớn, giới hạn trong việc xử lý các trường hợp phức tạp như đối tượng di chuyển nhanh, chuyển động phức tạp hoặc sự thay đổi nhanh về ánh sáng trong video.
Dựa trên những nghiên cứu trước đây, trong bài nghiên cứu này chúng tôi đề xuất và sử dụng mô hình hiệu quả để tổng hợp video động từ một bức ảnh tĩnh, tích hợp chuyển động môi, mắt và các phần có liên quan trên khuôn mặt vào video tổng hợp để tạo ra sự tự nhiên và chân thực dựa trên kỹ thuật học sâu kết hợp phương pháp biến đổi Hidden Affine Transformation và kỹ thuật xử lý ảnh. Chúng tôi sử dụng mạng nơ ron tích chập CNN cho việc trích xuất các đặc trưng hình ảnh khuôn mặt và conditional Generative Adversarial Network (cGAN) để tổng hợp video từ các đặc trưng hình ảnh. Mô hình tổng quát sẽ được trình bày trong phần tiếp theo.

# 3 MÔ HÌNH ĐỀ XUẤT

## 3.1 Mô hình tổng quát

Từ tập dữ liệu thô ban đầu chúng tôi sử dụng các phương pháp tiền xử lý dữ liệu sau đó đưa qua mô hình học sâu để tiến hành tổng hợp video từ hình ảnh khuôn mặt. Tiếp đến sẽ lưu lại các tham số của mô hình và tối ưu các tham số đó. Để mô hình cho ra video chuyển động chân thực và sinh động hơn. Mô hình tổng quát của chúng tôi được trình bày trong Hình 1.

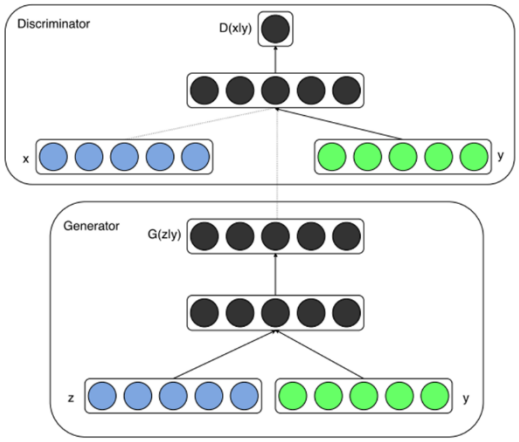



Hình 1: Sơ đồ mô hình tổng quát

## 3.2 Các thành phần của mô hình

### 3.2.1 Conditional Generative Adversarial Network (cGAN)

Generative Adversarial Networks (cGAN) được đề xuất bởi Mirza và Osindero vào năm 2014[7]. cGAN là một dạng mạng Generative Adversarial Network (GAN) trong lĩnh vực học sâu có khả năng tạo ra dữ liệu mới dựa trên một điều kiện đầu vào cụ thể.




Hình 2: Cấu trúc cGAN [7]

Cấu trúc cGAN gồm 2 phần chính: mạng sinh (generator) và mạng phân biệt (discriminator).

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Generator:
Mạng sinh nhận đầu vào là một nhãn và một hình ảnh khuôn mặt. Nhãn chứa các thông tin về biểu cảm hoặc các thuộc tính khác của khuôn mặt. Mạng sinh sử dụng kiến trúc dựa trên các lớp convolutional và deconvolutional để ánh xạ từ hình ảnh khuôn mặt đến các khung hình video. Lớp convolutional giúp trích xuất đặc trưng từ ảnh khuôn mặt và phép biến đổi affine ẩn, trong khi các lớp deconvolutional giúp tạo ra các khung video chi tiết.
Chúng tôi sử dụng CNN trong mạng sinh (generator) để phân tích và rút trích các đặc trưng quan trọng từ hình ảnh khuôn mặt, bao gồm các đặc trưng hình dạng và cấu trúc khuôn mặt, các chi tiết như mắt, mũi, miệng và các đặc trưng diễn cảm. Các đặc trưng này sau đó được sử dụng để thực hiện phép biến đổi Hidden Affine và tạo ra video từ hình ảnh khuôn mặt đầu vào.
Discriminator:
Mạng phân biệt có nhiệm vụ phân loại các khung hình video là thật hay giả. Khác với mạng phân biệt truyền thống, mạng phân biệt trong cGAN nhận thêm một nhãn làm đầu vào, giúp phân biệt dựa trên cả nội dung hình ảnh và nhãn. Mạng phân biệt nhận đầu vào là khung hình video và nhãn, sau đó đi qua các lớp convolutional để trích xuất đặc trưng.
Chúng tôi sử dụng cGAN để chuyển đổi hình ảnh khuôn mặt thành video thông qua quá trình tạo ra các khung hình tiếp theo trong chuỗi video.
Mô hình cGAN được sử dụng như sau:
Generative Network:
Mạng sinh nhận vào một hình ảnh khuôn mặt đầu vào và cố gắng tạo ra video tương ứng với hình ảnh đó.
Mô hình cGAN sử dụng mạng sinh có cấu trúc Encoder-Decoder, trong đó:
Encoder làm nhiệm vụ trích xuất các đặc trưng từ hình ảnh khuôn mặt đầu vào.
Sau đó, mạng sinh sử dụng một Hidden Affine Transformation để ánh xạ các đặc trưng từ không gian hình ảnh sang không gian video. Hidden Affine Transformation là một biến đổi tuyến tính bí mật ánh xạ các đặc trưng từ không gian khuôn mặt sang không gian video. Điều này giúp tạo ra sự chuyển động tự nhiên trong video kết quả.
Decoder: Cuối cùng, các đặc trưng được ánh xạ qua mạng decoder để tạo ra video đầu ra.
Discriminative Network:
Mạng phân biệt được đào tạo để phân biệt giữa các video được tạo ra bởi mạng sinh và các video thực tế. Mạng phân biệt sử dụng một kiến trúc mạng phân loại thông thường, được huấn luyện để phân biệt đúng giữa video khuôn mặt tạo ra và video thực tế. Nhiệm vụ của mạng sinh là tạo ra các video mà mạng phân biệt không thể phân biệt được với video thực tế.

### 3.2.2 Phép biến đổi Hidden Affine

Phép biến đổi affine ẩn (hidden affine transformation) là một phương pháp để áp dụng các phép biến đổi affine trên dữ liệu học sâu mà không cần xác định rõ ràng các tham số affine. Thay vào đó, mô hình học sâu sẽ học tự động các phép biến đổi affine dựa trên dữ liệu huấn luyện.
Phép biến đổi affine ẩn cho phép mô hình học cách tạo ra các khung hình video từ ảnh khuôn mặt ban đầu bằng cách thay đổi tỷ lệ, xoay và dịch chuyển các đặc trưng trên khuôn mặt.
Cách thực hiện phép biến đổi affine ẩn trong mô hình học sâu thường bao gồm sử dụng mạng nơ-ron để học và dự đoán các tham số affine tương ứng với từng khung hình video. Các tham số affine này được tính toán dựa trên thông tin đầu vào, chẳng hạn như ảnh khuôn mặt, và được sử dụng để tạo ra các khung hình video mới.
Trong nghiên cứu này chúng tôi sử dụng phép biến đổi Hidden Affine để cải thiện quá trình biến đổi hình ảnh khuôn mặt thành video bằng cách áp dụng các biến đổi tuyến tính ẩn để tái tạo các biểu cảm và chuyển động tự nhiên trong khuôn mặt.

## 3.3 Phương pháp đánh giá

### 3.3.1 Các độ đo và các phương pháp so sánh

Để đánh giá mô hình một cách chi tiết và định lượng, chúng tôi sử dụng các độ đo sau:
PSNR (Peak Signal-to-Noise Ratio) [8]: Đánh giá chất lượng hình ảnh hoặc video. Độ đo này được sử dụng để đo độ tương đồng giữa hình ảnh (hoặc video) gốc và hình ảnh (hoặc video) tái tạo.

SSIM (Structural Similarity Index Measure) [8]: Đánh giá sự tương đồng cấu trúc giữa hai hình ảnh hoặc video. Đây là độ đo được sử dụng để đo mức độ tương tự về cấu trúc, thông tin và hình dạng giữa hình ảnh (hoặc video) gốc và hình ảnh (hoặc video) tái tạo.
ACD (Average Content Distance) [9]: Đánh giá tính nhất quán về nội dung trong các video được tạo ra. Độ đo này được tính bằng cách lấy trung bình của tất cả các khoảng cách L2 giữa các đặc trưng khuôn mặt của hai khung hình liên tiếp trong một video được tạo ra.
ACD-I (Average Content Distance - Identity) [10] là một phiên bản mở rộng của tiêu chí đánh giá ACD (Average Content Distance) để đo tính bảo tồn định danh của khuôn mặt đầu vào trong video được tạo ra. ACD-I tính trung bình khoảng cách L2 giữa các đặc trưng khuôn mặt trong các khung hình của video và hình ảnh đầu vào. Nó đo lường mức độ bảo tồn định danh của khuôn mặt gốc trong quá trình chuyển đổi thành video. Trong nghiên cứu này, chúng tôi sử dụng OpenFace[11], một mô hình học sâu được huấn luyện để nhận dạng khuôn mặt và có khả năng vượt qua hiệu suất của con người, để trích xuất các đặc trưng khuôn mặt.
Các điểm số SSIM và PSNR càng cao thì chất lượng video được tạo ra càng tốt, các điểm số ACD thấp hơn cho thấy sự tương tự giữa các khuôn mặt trong các khung video liên tiếp được tạo ra, và các giá trị ACD-I thấp hơn cho thấy sự tương đồng cao hơn giữa các khuôn mặt trong ảnh đầu vào và các video được tạo ra.
Để đánh giá mô hình của chúng tôi với các mô hình khác, chúng tôi lựa chọn 5 mô hình tiên tiến, đạt kết quả tốt cho bài toán này được xây dựng dựa trên mô hình GAN để so sánh đó là: VGAN[12], MoCoGAN[13], GANimation[14], ImaGINator[15], và CwGAN[16].

### 3.3.2 Hàm mất mát

MSE (Mean Square Error) [17]:
Được sử dụng trong LSGAN (Least Squares GAN).
Hàm mất mát này tính toán sai số trung bình giữa đầu ra của Discriminator và nhãn mục tiêu.
Thích hợp cho bài toán GAN khi muốn tối thiểu hóa sai số bình phương giữa đầu ra và nhãn.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \tilde{y}_i)^2 \quad (1)$$

BCE (Binary Cross-Entropy) [18]:
Một hàm mất mát (loss function) phổ biến trong các bài toán phân loại nhị phân trong học máy. Hàm BCE được sử dụng để đo lường sai khác giữa các dự đoán nhị phân và nhãn thực tế trong bài toán phân loại nhị phân. Thông thường, đầu ra của một mô hình phân loại nhị phân được ánh xạ vào khoảng giá trị [0, 1], ví dụ như sử dụng hàm kích hoạt Sigmoid. BCE tính toán sai số chéo (cross-entropy) giữa dự đoán nhị phân và nhãn thực tế dưới dạng một giá trị số.
Hàm mất mát này tính toán cross entropy (mất mát entropy chéo) giữa đầu ra của Discriminator và nhãn mục tiêu. Thích hợp cho bài toán GAN khi muốn tối thiểu hóa sự khác biệt giữa phân bố xác suất của đầu ra và nhãn.

$$\mathrm{BCE} = -\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=0}^{\mathrm{N}} \mathrm{y_i} \cdot \log(\hat{\mathrm{y}}) + (1 - \mathrm{y_i}) \cdot \log(1 - \hat{\mathrm{y}}_\mathrm{i}) \quad (2)$$

## 4 THỰC NGHIỆM

### 4.1 Dữ liệu

Trong nghiên cứu này chúng tôi sử dụng tập dữ liệu CK-Mixed. CK-Mixed là bộ dữ liệu Cohn-Kanade (CK+)[19] phổ biến ở phân tích biểu cảm khuôn mặt, trong đó mỗi biểu cảm được mở ra từ trạng thái trung lập ban đầu một cách đơn điệu. Bộ dữ liệu này gồm 593 video của 6 loại cảm xúc, trong đó hầu hết các video đều ở thang độ xám. Sau khi tiến hành xử lý dữ liệu video chúng tôi trích xuất các khung hình từ video thu được tập dữ liệu ảnh gồm 8000 ảnh. Chúng tôi tiến hành chia tập dữ liệu theo tỉ lệ 8:2. Bảng 1 trình bày khái quát về Bộ dữ liệu CK-Mixed, một số hình ảnh về bộ dữ liệu được trình bày trong Hình 3.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Bảng 1: Bộ dữ liệu

| Datasets | Số lượng |
|---|---|
| Train | 5400 ảnh |
| Test | 1600 ảnh |




Hình 3: Một số hình ảnh trong tập dữ liệu CK-Mixed

### 4.1.1 Dlib

Để xác định vị trí phần môi trên khuôn mặt trong ảnh chúng tôi sử dụng Dlib [20] là một thư viện mã nguồn mở được phát triển bởi Davis E. King.
Sau khi xác định được vị trí môi, chúng tôi áp dụng thư viện OpenCV [21] để cắt phần môi đó ra từ ảnh gốc, sau đó tiến hành chuyển đổi phần môi này sang ảnh nhị phân để tạo patch dùng để vá lỗi cho video khi chuyển động. Hình 4 mô tả chi tiết patch phần môi sau khi được xử lý của chúng tôi.



Hình 4: Patch phần môi

## 4.2 Kết quả

Dựa trên phương pháp đề xuất trên, chúng tôi sẽ tạo video từ một hình ảnh đầu vào duy nhất. Phương pháp này có khả năng tạo ra video chuyển động mượt mà và chân thực từ hình ảnh khuôn mặt đầu vào. Các kết quả đã được đánh giá dựa trên các tiêu chí như sự tương đồng với hình ảnh gốc, sự chân thực và tính đa dạng của video kết quả. Hình 5 trình bày một ví dụ về kết quả dự đoán sau khi thực hiện mô hình.



Hình 5: Kết quả dự đoán sau khi thực hiện mô hình

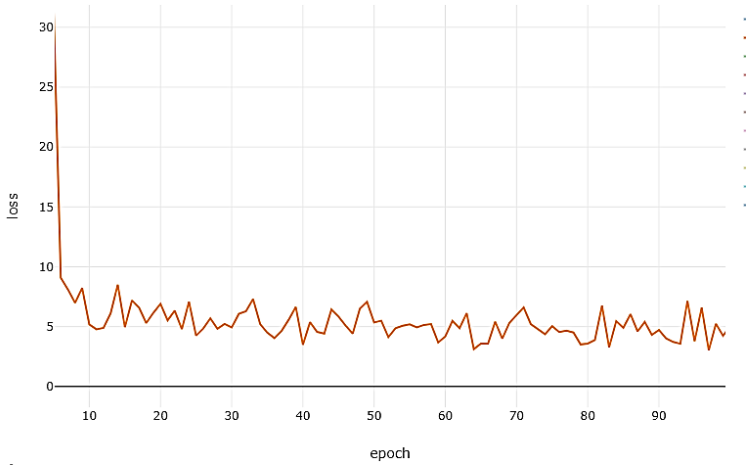
Như phần trước chúng tôi đã trình bày, chúng tôi lựa chọn 5 mô hình so sánh đó là: VGAN[12], MoCoGAN[13], GANimation[14], ImaGINator[15], và CwGAN[16].
Về các mô hình dùng để so sánh chúng tôi đã thực nghiệm lại trên cơ sở mã nguồn công khai của VGAN và ImaGINAtor do các tác giả cung cấp. Để áp dụng GANimation, chúng tôi căn chỉnh khuôn mặt và sau đó trích xuất các đặc điểm biểu hiện (AUs) thông qua OpenFace [11]. Trong giai đoạn dự đoán, chúng tôi

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

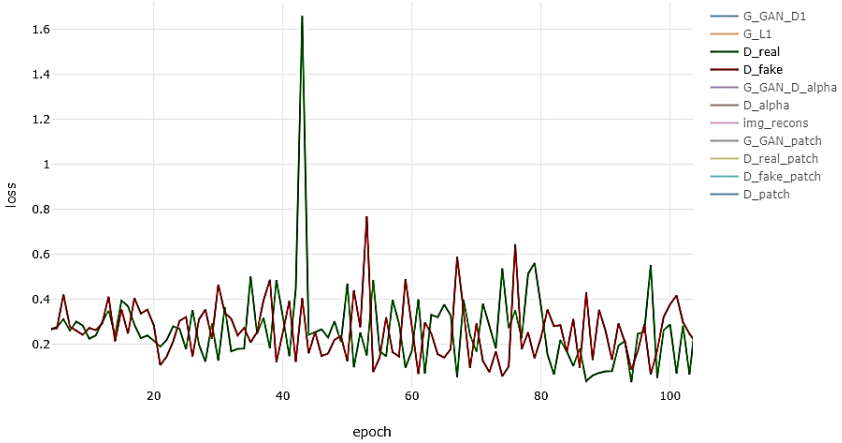
chọn AU biểu hiện phong phú nhất từ tập dữ liệu huấn luyện làm nhãn cho mỗi biểu cảm mục tiêu. Đối với MoCoGAN chúng tôi sử dụng kết quả được đưa ra bởi Wang et al. (2020b)[15]. Đối với CwGAN chúng tôi sử dụng kết quả trong bài báo gốc.
Kết quả so sánh giữa 5 phương pháp với mô hình đề xuất của chúng tôi được trình bày trong Bảng 2. Kết quả về hàm mất mát đánh giá sự khác biệt giữa ảnh được tạo (fake image) và ảnh thật (real image) được trình bày trong Hình 6 và Hình 7 trình bày về kết quả của Discriminator Loss.

Bảng 2: Bảng so sánh giữa các phương pháp

| **Phương pháp** | **PSNR** | **SSIM** | **ACD** | **ACD-I** |
|---|---|---|---|---|
| VGAN | 16.32 | 0.41 | 0.14 | 1.55 |
| MoCoGAN | 18.16 | 0.58 | 0.15 | 0.9 |
| GANimation | 26.54 | 0.89 | 0.10 | 0.35 |
| ImaGINator | 20.29 | 0.85 | 0.08 | 0.29 |
| CwGAN | 25.90 | 0.90 | 0.11 | 0.12 |
| **AffineGAN** | **35.50** | **0.91** | **0.06** | **0.16** |




Hình 6: Hàm mất mát đánh giá sự khác biệt giữa ảnh được tạo (fake image) và ảnh thật (real image)




Hình 7: Discriminator Loss

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Nhìn vào kết quả trong Bảng 2, chúng ta thấy được rằng phương pháp đề xuất của chúng tôi đạt kết quả tốt nhất so với các phương pháp so sánh khác ở cả 4 độ đo PSNR, SSIM, ACD và ACD-I. Kết quả này cho thấy phương pháp học sâu cGAN kết hợp với Hidden Affine Transformation cho kết quả tốt hơn so với các phương pháp GAN khác.

Chúng tôi đánh giá sự khác biệt giữa ảnh được tạo (`fake_image`) và ảnh thật (`real_image`) dựa trên L1Loss [22]. Mục tiêu là giảm thiểu độ sai lệch giữa ảnh được tạo và ảnh thật. Ở Hình 6, ta thấy loss giảm dần theo thời gian khi Generator cải thiện khả năng tạo ảnh. Mục tiêu cuối cùng là đạt được giá trị G_L1 thấp nhất, khi Generator tạo ra các ảnh có sự tương đồng cao so với ảnh thật, đồng nghĩa với việc tạo ra các ảnh chất lượng cao và khó phân biệt với ảnh thật.

Đối với Hình 7, D_real và D_fake là kết quả đầu ra của bộ phân biệt (Discriminator) khi đưa vào dữ liệu thật (real) và dữ liệu giả tạo (fake) tương ứng. D_real là kết quả đầu ra của bộ phân biệt khi đưa vào dữ liệu thật. D_fake là kết quả đầu ra của bộ phân biệt khi đưa vào dữ liệu giả tạo. Mục tiêu của mô hình là tạo ra dữ liệu giả tạo (fake) mà bộ phân biệt khó phân biệt được với dữ liệu thật (real). Bằng cách tối ưu hóa hàm mất mát, mô hình cố gắng làm cho D_real gần với 1 và D_fake gần với 0, từ đó tạo ra dữ liệu giả tạo chất lượng cao.

Tuy nhiên, khi phân tích chi tiết chúng tôi thấy rằng mô hình đề xuất của chúng tôi cũng có những hạn chế như: vẫn còn một số trường hợp đặc biệt khi tổng hợp video gặp khó khăn, ví dụ như khi bức ảnh ban đầu có chất lượng thấp hoặc không rõ ràng, hoặc khi các chuyển động trên khuôn mặt rất phức tạp và tinh tế. Các hạn chế này có thể gây ra các vấn đề như nhiễu hình ảnh, video không tự nhiên hoặc video không chính xác.

## 5 KẾT LUẬN

Kết quả thực nghiệm của nghiên cứu đã chứng minh tính hiệu quả và tiềm năng của phương pháp đề xuất trong tạo ra video động từ một bức ảnh duy nhất. Bằng cách sử dụng mô hình CNN để trích xuất đặc trưng và mô hình cGAN để tổng hợp video, phương pháp đã đạt được kết quả ấn tượng. Các video tổng hợp có chứa chuyển động tự nhiên và chân thực của môi, mắt và các phần khác trên khuôn mặt. Thực nghiệm trên các tập dữ liệu đa dạng đã chứng minh tính ổn định và khả năng tái tạo tốt của phương pháp trong nhiều trường hợp khác nhau.

Để cải thiện và phát triển phương pháp trong tương lai, chúng tôi có đề xuất một số hướng tiếp cận để có thể tiếp tục nghiên cứu chuyên sâu hơn. Đầu tiên, mở rộng kiến trúc mô hình để xử lý các trường hợp đặc biệt và khó khăn hơn, bao gồm chất lượng hình ảnh thấp, đa dạng chuyển động, và sự biến đổi khuôn mặt lớn. Thứ hai, nghiên cứu có thể tập trung vào cải thiện tính thực tế của video tổng hợp, đảm bảo rằng các chuyển động và chi tiết trên khuôn mặt được phản ánh chính xác. Thứ ba, khám phá khả năng ứng dụng của phương pháp trong các lĩnh vực khác nhau, chẳng hạn như điện ảnh, truyền thông và giải trí.

Với sự phát triển tiếp tục và nghiên cứu sâu hơn, phương pháp đề xuất có thể trở thành một công cụ quan trọng và mạnh mẽ trong việc tạo ra video động từ một bức ảnh duy nhất, với nhiều ứng dụng tiềm năng trong nhiều lĩnh vực khác nhau.

## TÀI LIỆU THAM KHẢO

[1] M. Dorkenwald, T. Milbich, A. Blattmann, R. Rombach, K. G. Derpanis and B. Ommer. (2021) Stochastic Image-to-Video Synthesis using cINNs.[Online]. Available: https://arxiv.org/pdf/2105.04551.pdf

[2] Y. Zhou, X. Han, E. Shechtman, J. Echevarria, E. Kalogerakis And D. Li. (2021) MakeItTalk: Speaker-Aware Talking-Head Animation. [Online]. Available: https://arxiv.org/pdf/2004.12992.pdf

[3] G. Shen, W. Huang, C. Gan, M. Tan, J. Huang, W. Zhu, and B. Gong. (2019) Facial Image-to-Video Translation by a Hidden Aine Transformation. [Online]. Available: https://doi.org/10.1145/3343031

[4] H. Ni, C. Shi, K. Li, S. X. Huang and M. R. Min. (2023) Conditional Image-to-Video Generation with Latent Flow Diffusion Models. [Online]. Available: https://arxiv.org/pdf/2303.13744v1.pdf

[5] M.-Y. Liu, X. Huang, J. Yu, T.-C. Wang and A. Mallya. (2020) Generative Adversarial Networks for Image and Video Synthesis: Algorithms and Applications. [Online]. Available: https://arxiv.org/pdf/2008.02793.pdf

[6] L. Zhao, X. Peng, Y. Tian, M. Kapadia and D. Metaxas. (2018) Learning to Forecast and Refine Residual Motion for Image-to-Video Generation. [Online]. Available: https://arxiv.org/pdf/1807.09951v1.pdf

[7] M. Mirza and S. Osindero. (2014) Conditional Generative Adversarial Nets. [Online]. Available: https://arxiv.org/pdf/1411.1784.pdf

[8] A. Horé and D. Ziou, "Image Quality Metrics: PSNR vs. SSIM," 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 2010, pp. 2366-2369, doi: 10.1109/ICPR.2010.579.

[9] S. Tulyakov, M.-Y. Liu, X. Yang, J. Kautz. "Mocogan: Decomposing motion ancontent for video generation", 2018. Available:https://ieeexplore.ieee.org/document/8578263

[10] L. Zhao, X. Peng, Y. Tian, M. Kapadia, D. Metaxas. "Learning to forecast and refine residual motion for image-to-video generation", 2018. Available: https://arxiv.org/abs/1807.09951v1

[11] B. Amos, B. Ludwiczuk, M. Satyanarayanan. "OpenFace: A general-purpose face recognition library with mobile applications", 2016. Available:http://cmusatyalab.github.io/openface/

[12] Vondrick, Carl, H. Pirsiavash, and A. Torralba. Generating videos with scene dynamics.Advances in neural information processing systems 29. 2016

[13] Tulyakov, Sergey, et al. "Mocogan: Decomposing motion and content for video generation.Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

[14] Pumarola, Albert, et al. "Ganimation: Anatomically-aware facial animation from a single image.Proceedings of the European conference on computer vision (ECCV). 2018.

[15] Wang, Yaohui, et al. "Imaginator: Conditional spatio-temporal gan for video generation.Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. 2020.

[16] Otberdout, Naima, et al. Dynamic facial expression generation on hilbert hypersphere with conditional wasserstein generative adversarial nets. IEEE Transactions on Pattern Analysis and Machine Intelligence 44.2 (2020): 848-863.

[17] Schluchter, Mark D. "Mean square error." Encyclopedia of Biostatistics 5 (2005).

[18] Li, Li, Miloš Doroslovački, and Murray H. Loew. "Approximating the gradient of cross-entropy loss function." IEEE Access 8 (2020): 111626-111635.

[19] Lucey, Patrick, et al. "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression." 2010 ieee computer society conference on computer vision and pattern recognition-workshops. IEEE, 2010.

[20] DLIB: http://dlib.net/

[21] OpenCV: https://opencv.org/

[22] Chai, Tianfeng, and Roland R. Draxler. "Root mean square error (RMSE) or mean absolute error (MAE)." Geoscientific model development discussions 7.1 (2014): 1525-1534.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F027**

# PTASK: XÂY DỰNG WEBSITE QUẢN LÝ DỰ ÁN TÙY CHỈNH CHO DOANH NGHIỆP

VÕ MINH PHƯƠNG[1], NGUYỄN ĐỨC HUY[1], NGUYỄN THỊ HẠNH[1*]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
*[*]ngthihanh@gmail.com*

**Tóm tắt.** Làm việc theo nhóm là điều cần thiết trong thế giới ngày nay, ảnh hưởng trực tiếp đến sự thành công hay thất bại của các dự án và nhiệm vụ. Để quản lý làm việc nhóm hiệu quả, nhóm chúng tôi xin giới thiệu PTASK - một hệ thống quản lý dự án tùy chỉnh dành cho doanh nghiệp. PTASK chia dự án thành các nhiệm vụ nhỏ hơn, cho phép các thành viên hoặc nhóm riêng lẻ hoàn thành chúng trong khung thời gian cụ thể, đảm bảo tổ chức hiệu quả và kiểm soát tiến độ. Được phát triển bằng các công nghệ mới nhất như NodeJS, ASP.NET Core, AWS, Cyclic, Github và MongoDB Cloud, PTASK là một nền tảng dựa trên web giúp tối ưu hóa việc quản lý dự án. Nó tạo điều kiện cộng tác, phân bổ nhiệm vụ và theo dõi tiến độ, nâng cao năng suất và sự phối hợp tổng thể. Với PTASK, doanh nghiệp được hưởng lợi từ khả năng hiển thị nhiệm vụ được cải thiện, quy trình hợp lý hóa và giao tiếp nhóm nâng cao. Bằng cách giao nhiệm vụ, đặt thời hạn và theo dõi tiến độ, PTASK đảm bảo mọi người đều được liên kết và chịu trách nhiệm. Trải nghiệm sức mạnh của PTASK để tối ưu hóa việc quản lý dự án của bạn, tăng năng suất và đạt được kết quả thành công.

**Từ khóa.** Quản lý dự án, Doanh nghiệp, Quản lý, Quá trình làm việc nhóm.

## PTASK: BUILDING A CUSTOM PROJECT MANAGEMENT WEBSITE FOR YOUR BUSINESS

**Abstract.** Teamwork is essential in today's world, directly influencing the success or failure of projects and tasks. To effectively manage teamwork, our team introduces PTASK - a customized project management system for businesses. PTASK breaks down projects into smaller tasks, allowing individual members or groups to complete them within specific timeframes, ensuring efficient organization and progress control. Developed using the latest technologies like NodeJS, ASP.NET Core, AWS, Cyclic, Github, and MongoDB Cloud, PTASK is a web-based platform that optimizes project management. It facilitates collaboration, task allocation, and progress tracking, enhancing overall productivity and coordination. With PTASK, businesses benefit from improved task visibility, streamlined processes, and enhanced team communication. By assigning tasks, setting deadlines, and monitoring progress, PTASK ensures everyone is aligned and accountable. Experience the power of PTASK to optimize your project management, increase productivity, and achieve successful outcomes.

**Keywords.** Project management, Business, manage, teamwork process.

## 1 GIỚI THIỆU TỔNG QUAN

Trong môi trường quản lý doanh nghiệp ngày nay, quản lý dự án đóng vai trò then chốt trong việc đạt được sự thành công và phát triển bền vững cho doanh nghiệp. Việc theo dõi, tổ chức và điều phối các hoạt động dự án một cách hiệu quả là một thách thức lớn. Để đáp ứng nhu cầu này, Website Quản lý Dự án Tùy chỉnh đã trở thành một công cụ quan trọng và không thể thiếu cho các doanh nghiệp.

Website Quản lý Dự án Tùy chỉnh là một hệ thống phần mềm dự án được phát triển dựa trên công nghệ web và tùy chỉnh theo yêu cầu của từng doanh nghiệp. Nó cung cấp một nền tảng trực tuyến cho phép các thành viên trong dự án tương tác, trao đổi thông tin và quản lý công việc một cách thuận tiện và hiệu quả.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Website Quản lý Dự án Tùy chỉnh cung cấp một loạt các tính năng giúp doanh nghiệp quản lý dự án một cách toàn diện. Một trong những tính năng quan trọng là theo dõi tiến độ dự án, cho phép người dùng xem tổng quan về các công việc đang diễn ra, các mốc quan trọng và thời gian hoàn thành. Điều này giúp nhà quản lý dự án và các thành viên liên quan có cái nhìn tổng quan về tình hình dự án và có thể đưa ra các quyết định kịp thời để đảm bảo tiến độ và chất lượng.
Website Quản lý Dự án Tùy chỉnh cho phép người dùng phân công nhiệm vụ cho các thành viên trong dự án và ghi chú các chỉ định cụ thể cho từng công việc. Điều này giúp tăng cường sự rõ ràng và đồng nhất trong việc phân công và thực hiện nhiệm vụ. Ngoài ra, nền tảng này cũng cung cấp các công cụ giao tiếp nội bộ như tin nhắn, hệ thống thông báo và diễn đàn, tạo điều kiện thuận lợi cho việc trao đổi thông tin, giải quyết vấn đề và phối hợp làm việc giữa các thành viên trong dự án.
Website Quản lý Dự án Tùy chỉnh cho phép lưu trữ và quản lý tài liệu dự án một cách trực tuyến. Người dùng có thể tải lên, chia sẻ và truy cập tài liệu từ mọi nơi và bất kỳ thiết bị nào có kết nối internet. Điều này giúp đảm bảo tính toàn vẹn và sẵn sàng của thông tin dự án, từ các tài liệu kỹ thuật, báo cáo tiến độ cho đến tài liệu liên quan khác.
Website Quản lý Dự án Tùy chỉnh cung cấp khả năng đồng bộ hóa dữ liệu và cập nhật thời gian thực. Điều này cho phép các thành viên trong dự án cùng làm việc trên cùng một phiên bản dự án, đồng bộ hóa thông tin và cập nhật thay đổi một cách nhanh chóng. Nhờ đó, mọi người có thể làm việc cùng nhau trên dự án một cách hiệu quả và tránh những sự không phù hợp hoặc mất thông tin quan trọng.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 Ngôn ngữ lập trình C# (C Sharp)

C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.
C# với sự hỗ trợ mạnh mẽ của .NET Framework giúp cho việc tạo một ứng dụng Windows Forms hay WPF (Windows Presentation Foundation), phát triển game, ứng dụng Web, ứng dụng Mobile trở nên rất dễ dàng.

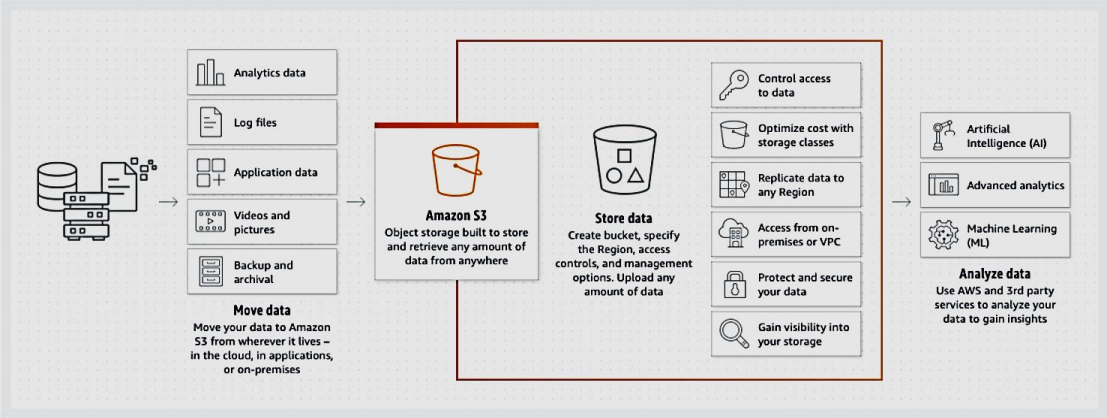



Hình 1: Microsoft .NET [2]

### 2.2 S3 (Amazon Simple Storage Service)

Amazon Simple Storage Service (Amazon S3) là vùng lưu trữ trên Internet. Dùng Amazon S3 để lưu trữ và truy xuất bất kỳ dữ liệu tại bất kỳ thời điểm nào, từ bất cứ nơi đâu trên web. Bạn có thể tương tác với Amazon S3 dùng AWS Management Console, giao diện web đơn giản nhưng trực quan. Tài liệu này hướng dẫn dùng AWS Management Console để tương tác với Amazon S3.

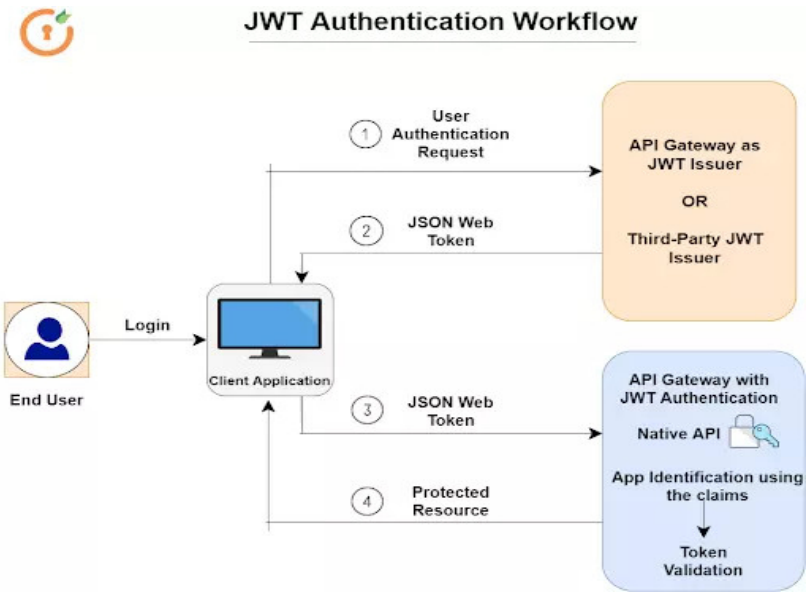
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 2: S3 (Amazon Simple Storage Service) [3] [4]

## 2.3 Json Web Token (JWT)

JSON Web Token (JWT) là một chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và tự chứa để truyền thông tin an toàn giữa các bên dưới dạng một đối tượng JSON [24]. Khi người dùng đăng nhập vào ứng dụng thì ứng dụng sẽ gửi một request  user cho api gateway as issuer sao đó api gateway sẽ trả về cho ứng dụng một mã token. Tiếp theo muốn sử dụng token đó thì người dùng phải giải mã token. Để giải mã token thì  nhà phát triển phải gửi token đó đến api gateway with JWT Authentication và Api gateway with JWT Authentication sẽ xác thực và giải mã token.




Hình 3: JWT [8] [9]

## 2.4 Cyclic

Cyclic được xây dựng và triển khai vào AWS. Cyclic cung cấp trước một ứng dụng không có máy chủ bằng cách sử dụng cấu trúc đám mây. Ở lần khởi chạy đầu tiên, Cyclic chọn một ngăn xếp có sẵn và triển khai mã cho ứng dụng vào lambda hiện có. Đối với các lần khởi chạy tiếp theo, Cyclic cập nhật lambda để sử dụng mã mới nhất.




Hình 4: Cyclic [7]

## 2.5 Nodejs

- ✓ NodeJS là một nền tảng (Platform) được xây dựng trên nền tảng Javascript V8 Engine. Được xây dựng để phát triển những ứng dụng server side.
- ✓ Phần core sử dụng Javascript và C++ cho phép xử lý với hiệu năng cao.
- ✓ Phù hợp với các ứng dụng xử lý nhanh, real time hoặc những ứng dụng cần thay đổi công nghệ nhanh.
- ✓ Chạy sigle thread nhưng có cơ chế non blocking giúp chạy bất đồng bộ, tăng khả năng xử lý
- ✓ Thường phát triển theo hướng module hóa thành các phần nhỏ của ứng dụng và có thể thêm module có sẵn bên ngoài thông qua NPM, yarn.…




Hình 5: Node Js [1]

## 2.6 MongoDb

- ✓ MongoDB là một hệ quản trị cơ sở dữ liệu mã nguồn mở, là CSDL thuộc NoSql và được hàng triệu người sử dụng.
- ✓ MongoDB là một database hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu JSON thay vì dạng bảng như CSDL quan hệ nên truy vấn sẽ rất nhanh.
- ✓ Với CSDL quan hệ chúng ta có khái niệm bảng, các cơ sở dữ liệu quan hệ (như MySQL hay SQL Server...) sử dụng các bảng để lưu dữ liệu thì với MongoDB chúng ta sẽ dùng khái niệm là collection thay vì bảng
- ✓ So với RDBMS thì trong MongoDB collection ứng với table, còn document sẽ ứng với row , MongoDB sẽ dùng các document thay cho row trong RDBMS.
- ✓ Các collection trong MongoDB được cấu trúc rất linh hoạt, cho phép các dữ liệu lưu trữ không cần tuân theo một cấu trúc nhất định.
- ✓ Thông tin liên quan được lưu trữ cùng nhau để truy cập truy vấn nhanh thông qua ngôn ngữ truy vấn MongoDB

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Hình 6: Mongo DB [5][6]

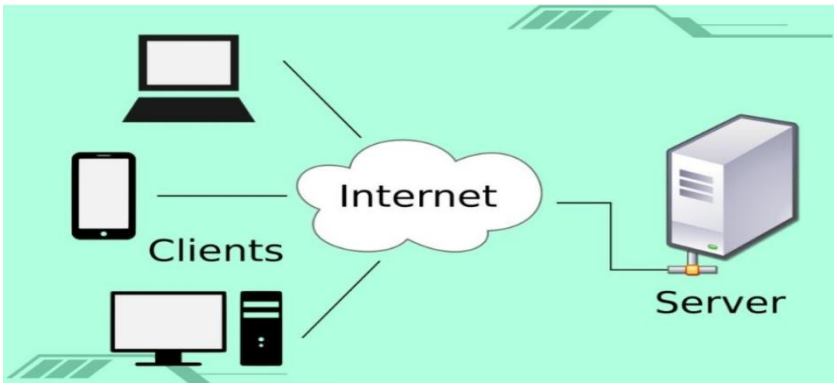
# 3 CÔNG CỤ PTASK

## 3.1 Giới thiệu

Hệ thống của chúng tôi được xây dựng bằng một số công nghệ chính như ASP.NET, NodeJs, JavaScript, JQuery, Boostrapt, HTML/CSS, Json Chart, JSON Web Token (JWT) và cơ sở dữ liệu MongoDB. Ngoài ra, chúng tôi đã tích hợp một số công nghệ bên ngoài như AWS, Firebase Authentication, Cyclic. Hệ thống được triển khai trên hai nền tảng là website. NodeJS được sử dụng để xây dựng máy chủ cho hệ thống, phối hợp với MongoDB để lưu trữ dữ liệu. ASP.NET, HTML/CSS, Boostrapt được sử dụng để xây dựng website. Cả ba đều có các chức năng tương tự và được thiết kế theo yêu cầu của bài toán. JWT được sử dụng để mã hóa dữ liệu dựa trên JSON, giúp xác thực và truyền dữ liệu an toàn giữa các bên. Json Chart được sử dụng để xây dựng biểu đồ, giúp người dùng có cái nhìn trực quan hơn về dữ liệu. Firebase Authentication là một dịch vụ điện toán đám mây nổi tiếng, hỗ trợ đăng ký và xác thực thông qua số điện thoại và email. Chúng tôi sử dụng AWS (Amazon Web Services) và dịch vụ S3 của nó để lưu trữ hình ảnh, video và các tài nguyên khác. Cyclic là một công cụ hỗ trợ việc đưa website lên host để người dùng có thể truy cập online bất cứ khi nào.

3.2 Kiến trúc hệ thống

Ứng dụng của chúng tôi áp dụng kiến trúc Client-Server để xây dựng và thiết kế. Người dùng có thể tương tác với ứng dụng thông qua giao diện web. Khi người dùng thực hiện các hành động như đăng nhập, đăng ký hoặc thực hiện các tác vụ khác trên giao diện, yêu cầu của người dùng sẽ được gửi đến máy chủ. Máy chủ sẽ xử lý yêu cầu của người dùng, truy xuất dữ liệu từ cơ sở dữ liệu và trả về kết quả tương ứng cho người dùng. Cấu trúc này cho phép các thành phần của ứng dụng hoạt động độc lập và tương tác với nhau để cung cấp cho người dùng các tính năng và trải nghiệm tốt nhất.




Hình 7: Kiến trúc Client-Server [10]

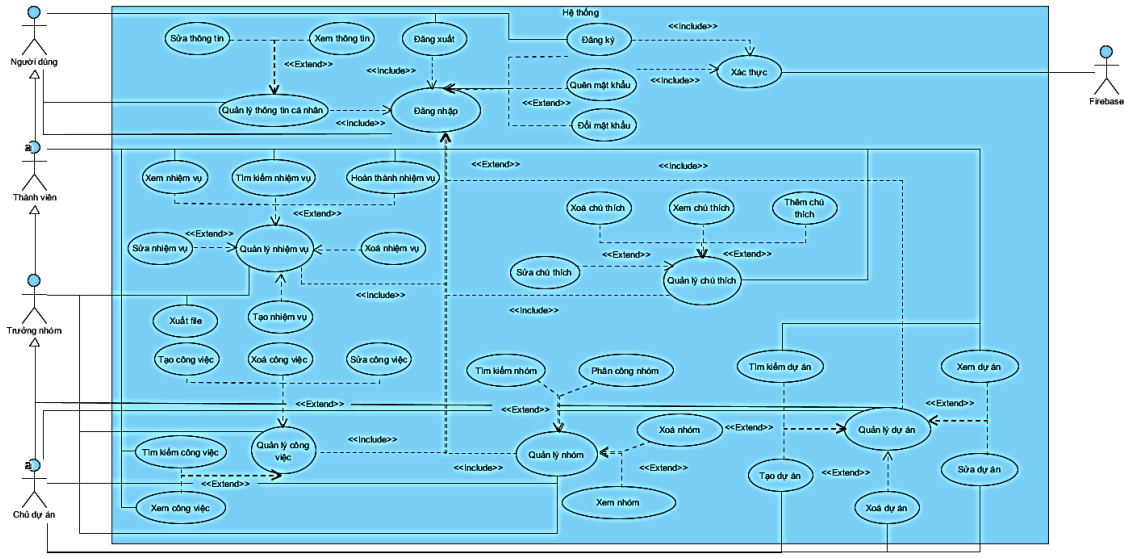
## 3.2 Phân tích hệ thống PTASK

Đối tượng của PTASK là các doanh nghiệp, cá nhân có nhu cầu về việc quản lý dự án to và nhỏ. PTASK cung cấp việc quản lý dự án từ nhỏ đến lớn nên các cá nhân hay tổ chức có nhu cầu quản lý dự án đều có thể sử dụng website quản lý dự án PTASK.

Bảng 1: Danh sách các tác nhân và mô tả

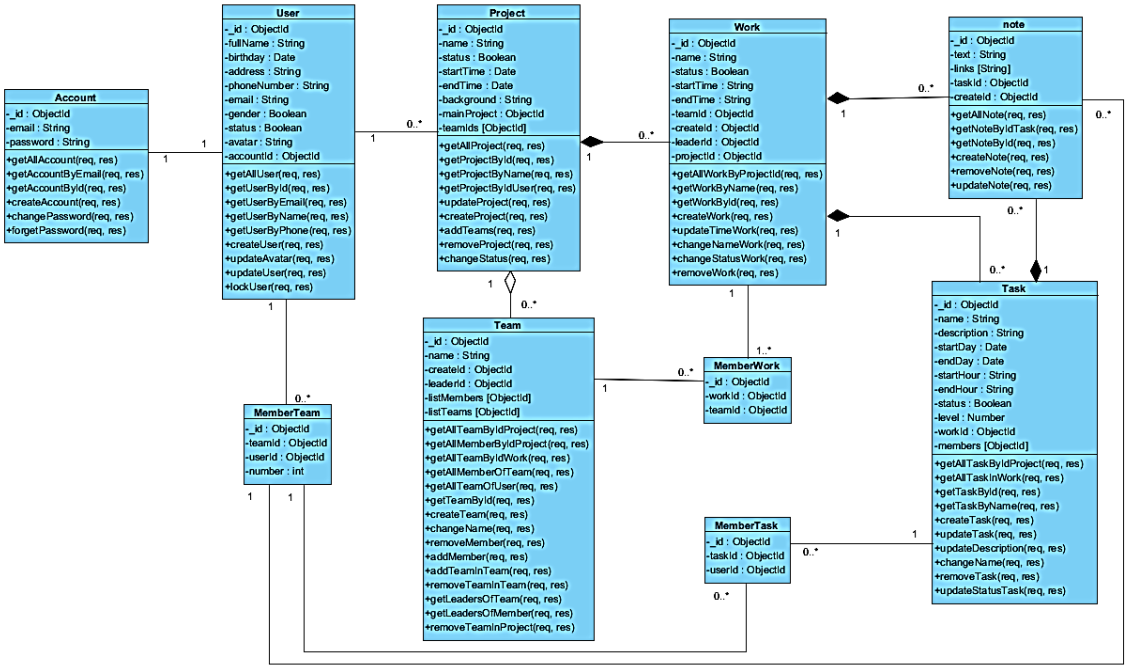
| Tác nhân | Mô tả tác nhân |
|---|---|
| Chủ dự án | Chủ dự án sau khi đăng nhập vào website có thể tạo dự án mới và có thể thực hiện các chức năng như: Quản lý về dự án: có thể xem thông tin, chỉnh sửa và xoá dự án. Quản lý về nhân sự của dự án: có thể xem các nhóm tham gia, các thành viên tham gia vào dự án, có thể chỉnh sửa quyền của thành viên cũng như thêm xoá các thành viên cũng như các nhóm trong dự án. Quản lý về công việc trong dự án: có thể xem chi tiết mọi hoạt động của các công việc trong dự án, có thể thêm xoá sửa các công việc, phân công đội nhóm, phân công thành viên vào các nhiệm vụ phù hợp và xem biểu đồ thống kê của dự án, xuất file excel về thông tin của dự án |
| Nhóm trưởng | Nhóm trưởng sau khi đăng nhập vào website có thể vào xem thông tin dự án và có thể thực hiện các chức năng quản lý công việc trong dự án nhóm trưởng có thể thêm, xoá, chỉnh sửa và xem các công việc, cùng với đó, nhóm trưởng có thể quản lý các thành viên có trong nhóm của mình, phân công các thành viên vào các công việc phù hợp cũng như loại bỏ những thành viên chưa phù hợp ở công việc đó |
| Thành viên | Thành viên sau khi đăng nhập vào có thể xem thông tin của dự án, thông tin công việc mà mình tham gia, thành viên có thể chọn hoàn thành công việc hay thêm các ghi chú cho các công việc của mình |

Sơ đồ usecase của PTASK bao gồm các tác nhân chính là người dùng, thành viên, trưởng nhóm, chủ dự án và 1 actor phụ là Firebase.




Hình 8: Sơ đồ usecase của PTASK


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 9: Sơ đồ Class Diagram của PTASK

## 3.4 Giao diện phác họa

Trong giao diện website sử dụng các công nghệ Cyclic hỗ trợ việc đưa website lên host, công nghệ AWS sử dụng để lưu trữ các hình ảnh, Json chart dùng để vẽ biểu đồ thống kê dự án.

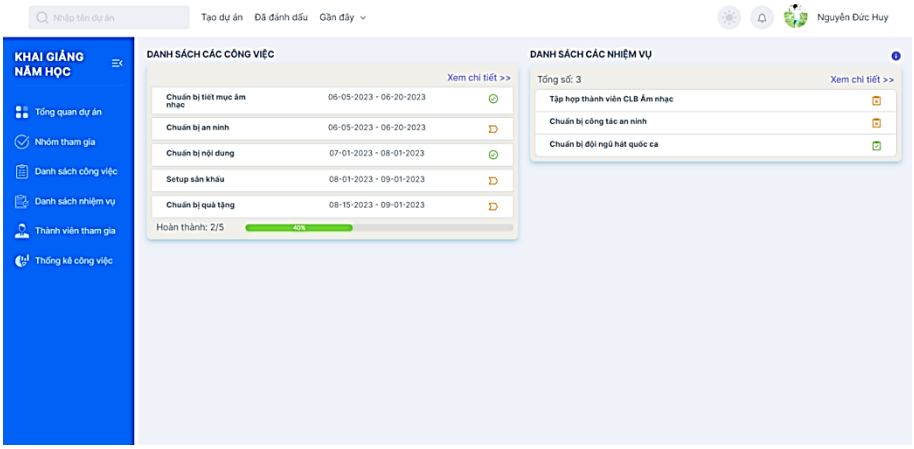
### 3.4.1 Một số giao diện website




Hình 10: Giao diện quản lý dự án

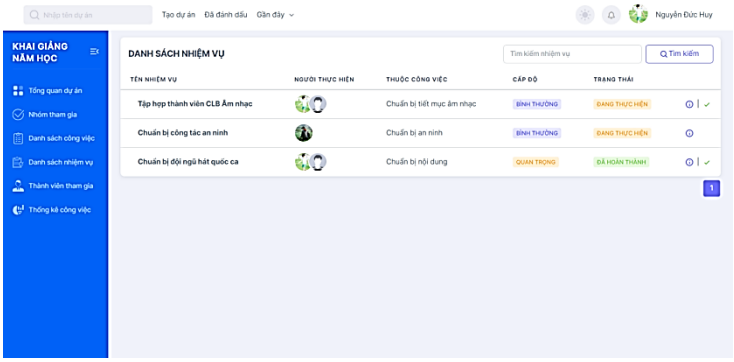
Màn hình chính với các chức năng

- ✓ Chọn vào image của dự án để truy cập vào trang của dự án
- ✓ Chọn "Tạo dự án" để tạo dự án mới
- ✓ Nhập tên dự án để tìm kiếm dự án
- ✓ Chọn vào avatar để truy cập xem thông tin người dùng hoặc đăng xuất khỏi hệ thống
- ✓ Chọn vào hình mặt trang để chuyển sang chế độ sáng hoặc tối
- ✓ Chọn vào hình chuông để xem thông báo




Hình 11 Giao diện tổng quan dự án của thành viên

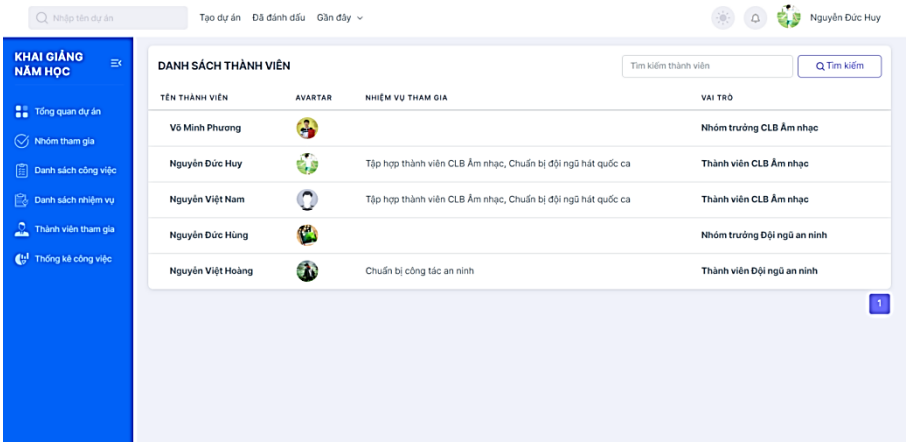
Màn hình với các chức năng
- Nhập tên công việc cần tìm vào ô textview và chọn "Tìm kiếm" để tìm kiếm công việc
- Chọn vào biểu tượng "!" để chỉnh sửa công việc
- Chọn "Tổng quan dự án" để xem tổng quan thông tin dự án
- Chọn "Danh sách công việc" để xem danh sách các công việc trong dự án
- Chọn "Danh sách nhiệm vụ" để xem danh sách nhiệm vụ trong dự án
- Chọn "Nhóm tham gia" để xem danh sách các nhóm có trong dự án
- Chọn "Thành viên tham gia" để xem danh sách tất cả các thành viên có trong dự án
- Chọn "Thống kê công việc" để xem biểu đồ thống kê công việc của dự án




Hình 12: Giao diện danh sách nhiệm vụ của thành viên

Màn hình với các chức năng
- Nhập tên nhiệm vụ cần tìm vào ô textview và chọn "Tìm kiếm" để tìm kiếm nhiệm vụ
- Chọn vào biểu tượng cây bút để xem thông tin và tạo ghi chú nhiệm vụ
- Chọn vào thùng rác để xoá nhiệm vụ
- Chọn "v" để hoàn thành nhiệm vụ
- Chọn "Tổng quan dự án" để xem tổng quan thông tin dự án
- Chọn "Danh sách công việc" để xem danh sách các công việc trong dự án
- Chọn "Danh sách nhiệm vụ" để xem danh sách nhiệm vụ trong dự án
- Chọn "Nhóm tham gia" để xem danh sách các nhóm có trong dự án
- Chọn "Thành viên tham gia" để xem danh sách tất cả các thành viên có trong dự án
- Chọn "Thống kê công việc" để xem biểu đồ thống kê công việc của dự án

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 13: Giao diện danh sách thành viên của thành viên

Màn hình với các chức năng
- Nhập tên thành viên cần tìm vào ô textview và chọn "Tìm kiếm" để tìm kiếm thành viên
- Chọn "Tổng quan dự án" để xem tổng quan thông tin dự án
- Chọn "Danh sách công việc" để xem danh sách các công việc trong dự án
- Chọn "Danh sách nhiệm vụ" để xem danh sách nhiệm vụ trong dự án
- Chọn "Nhóm tham gia" để xem danh sách các nhóm có trong dự án
- Chọn "Thành viên tham gia" để xem danh sách tất cả các thành viên có trong dự án
- Chọn "Thống kê công việc" để xem biểu đồ thống kê công việc của dự án




Hình 14: Giao diện thống kê công việc

Màn hình với các chức năng
Chọn vào "Xuất excel" để xuất file excel
Chọn "Tổng quan dự án" để xem tổng quan thông tin dự án
Chọn "Danh sách công việc" để xem danh sách các công việc trong dự án
Chọn "Danh sách nhiệm vụ" để xem danh sách nhiệm vụ trong dự án
Chọn "Nhóm tham gia" để xem danh sách các nhóm có trong dự án
Chọn "Thành viên tham gia" để xem danh sách tất cả các thành viên có trong dự án
Chọn "Thống kê công việc" để xem biểu đồ thống kê công việc của dự án

# 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## 4.1 Kết quả đạt được

Website đã hoàn thành, đáp ứng được nhu cầu đặt ra ban đầu của khách hàng

- Hiện thực đầy đủ các chức năng chính được đề ra ban đầu
- Hệ thống duy trì ổn định, ít gặp các vấn đề làm gián đoạn
- Giao diện đạt tiêu chuẩn 2 dễ: dễ nhìn, dễ sử dụng

## 4.2 Hướng phát triển

- Tích hợp công nghệ realtime
- Hoàn thiện các chức năng còn thiếu
- Phát triển thêm các tính năng mới
- Đưa vào sử dụng trực tiếp tại các doanh nghiệp
- Kết nối đăng nhập với các tài khoản google, facebook, …
- Kết nối với các công cụ hỗ trợ khác như: Github, …

## 4.3 Bài học kinh nghiệm

- Từ dự án trên, nhóm đã rút ra được những kinh nghiệm quý báu cho bản thân để giúp cho công việc sau này như sau:
- Việc sắp xếp lịch trình triển khai sao cho có khoa học, lên kế hoạch chi tiết cho các công việc một cách phù hợp.
- Chuẩn bị thật hoàn chỉnh và chu đáo các nghiệp vụ, nắm rõ phần mềm, thao tác, quy trình,… để triển khai cho doanh nghiệp.

## 4.4 Hạn chế

Bên cạnh những việc đã làm được, hệ thống vẫn còn xảy ra một vài hạn chế chưa được khắc phục do thời gian hạn hẹp cũng như trình độ chuyên môn chưa cao. Kỹ năng nghiệp vụ của nhóm còn yếu, chưa nắm được kỹ các quy trình nghiệp vụ từ đó dẫn đến việc chưa bao quát hết được các nghiệp vụ thực tế phát sinh ở doanh nghiệp.

## REFERENCES

[1] https://nodejs.org/en/docs

[2] https://dotnet.microsoft.com/en-us/apps/aspnet

[3] https://aws.amazon.com/vi/s3/

[4] https://viblo.asia/p/tim-hieu-s3-aws-amazon-simple-storage-service-gDVK2QGm5L

[5] https://www.mongodb.com/docs

[6] https://mongoosejs.com/docs/guide.htm

[7] https://docs.cyclic.sh

[8] https://jwt.io/

[9] https://viblo.asia/p/tim-hieu-ve-json-web-token-jwt-7rVRqp73v4bP

[10] Smith, J. (2022). Client-Server Architecture: Concepts and Implementation. ABC Publishing.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F028**

# XÂY DỰNG HỆ THỐNG QUẢN LÝ HỌC PHẦN KHÓA LUẬN TỐT NGHIỆP ÁP DỤNG TẠI TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH

NGUYỄN THỊ HẠNH[1], NGUYỄN THỊ MINH CHÂU[1*], NGUYỄN THANH SƠN[1], HOÀNG MINH TRÍ[1], NGUYỄN CHI QUỐC ĐANG[1]

*[1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*

*ngthihanh@iuh.edu.vn,chaunguyen.141201@gmail.com, nguyenthanhson162001@gmail.com, 2964139hoangchi@gmail.com, ndang1336@gmail.com*

**Tóm tắt.** Triển khai và quản lý cho các sinh viên thực hiện học phần khóa luận tốt nghiệp gồm nhiều công đoạn rất phức tạp mà bất kỳ một đơn vị đào tạo bậc đại học nào cũng phải thực hiện. Hiện nay, một số đơn vị vẫn còn thực hiện công việc này bằng cách thủ công, điều này tốn nhiều thời gian và không thuận tiện cho người quản lý cũng như giảng viên và sinh viên. Nhóm chúng tôi thực hiện bài báo: "Xây dựng hệ thống quản lý học phần khóa luận tốt nghiệp áp dụng tại Trường Đại học Công nghiệp Thành phố Hồ Chí Minh". Bài báo trình bày về Xây dựng hệ thống quản lý học phần khóa luận tốt nghiệp áp dụng tại Trường Đại học Công nghiệp Thành phố Hồ Chí Minh. Hệ thống này hỗ trợ cho người quản lý, giảng viên và sinh viên thực hiện các công việc liên quan đến khóa luận tốt nghiệp. Hệ thống được xây dựng bằng các công cụ: ReactJS và React Native cho front-end. Back-end được xây dựng bằng NodeJS và triển khai trên máy ảo của Google Cloud. Hệ thống đang được thử nghiệm cho chuyên ngành kỹ thuật phần mềm - khoa Công nghệ Thông tin. Trong tương lai hy vọng hệ thống sẽ được sử dụng cho tất cả các chuyên ngành trong toàn trường. Nhằm nâng cao tính hiệu quả, chính xác và tiết kiệm thời gian đồng thời giúp nâng cao chất lượng giảng dạy và đào tạo của trường.

**Từ khóa.** Ứng dụng web, Ứng dụng di động, Google Cloud, Quản lý khóa luận.

## DEVELOPING A SYSTEM FOR MANAGING THE FINAL THESIS COURSE AT INDUSTRIAL UNIVERSITY OF HO CHI MINH CITY

**Abstract.** Implementing and managing the completion of the thesis project module for institutions students involves many complex tasks that any institution must perform. Currently, some institutions still carry out this work manually, which is time-consuming and inconvenient for both managers, lecturers, and students. Our team has implemented the project: "Developing a system for managing the final thesis course at Industrial University of Ho Chi Minh City". An Article Introducing the Development of a Thesis Management System Applied at Industrial University of Ho Chi Minh City. This system aids administrators, instructors, and students in carrying out tasks related to thesis management. The system is built using ReactJS and React Native for the front-end, while the back-end is developed with NodeJS and deployed on Google Cloud virtual machines. The system is currently undergoing testing for the Software Engineering major - Faculty of Information Technology. In the future, we hope the system will be used for all departments across the institutions. The goal is to improve efficiency, accuracy, and save time while enhancing the quality of teaching and training at the institutions.

**Keywords.** Web application, Mobile application, Google Console, Thesis management.

## 1 GIỚI THIỆU TỔNG QUAN

Hiện nay, thực hiện khóa luận tốt nghiệp đang trở thành một phần không thể thiếu trong quá trình hoàn tất việc học tập tại đại học. Tuy nhiên, việc quản lý và đánh giá hoạt động của các sinh viên trong quá trình

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

thực hiện và nghiên cứu khóa luận vẫn đang gặp nhiều khó khăn, bất cập. Để giải quyết vấn đề này, hệ thống quản lý học phần khóa luận tốt nghiệp được xem là một giải pháp hiệu quả. Nhiều trường đại học đã xây dựng các hệ thống quản lý khóa luận tốt nghiệp. Các hệ thống này giúp quản lý thông tin và đánh giá khóa luận tốt nghiệp một cách hiệu quả. Một số hệ thống quản lý khóa luận tốt nghiệp phổ biến hiện nay bao gồm: Moodle là một hệ thống quản lý học tập (LMS) mã nguồn mở. Moodle có thể được sử dụng để quản lý khóa luận tốt nghiệp bằng cách tạo các khóa học, quản lý các bài luận của sinh viên và đánh giá các bài luận. Hay Blackboard là một hệ thống quản lý học tập khác cũng có thể được sử dụng để quản lý khóa luận tốt nghiệp. Blackboard có nhiều tính năng hơn Moodle, chẳng hạn như khả năng tạo các câu hỏi trắc nghiệm và bài kiểm tra. Với sự phát triển của công nghệ thông tin, ứng dụng hệ thống quản lý học phần khóa luận trở nên cần thiết và hiệu quả hơn bao giờ hết. Hệ thống này sẽ giúp cho việc quản lý, theo dõi và đánh giá hoạt động của sinh viên trong quá trình thực hiện khóa luận trở nên dễ dàng hơn. Mục tiêu của hệ thống sẽ chú trọng vào việc hỗ trợ việc quản lý các thông tin liên quan đến khóa luận, bao gồm danh sách sinh viên, giảng viên hướng dẫn, tiến độ thực hiện, đánh giá kết quả và lưu trữ tài liệu liên quan hay cung cấp cho sinh viên các công cụ giúp cho việc quản lý tiến độ thực hiện khóa luận, giao tiếp với giảng viên hướng dẫn và đánh giá kết quả cuối cùng là tăng cường tính minh bạch trong quá trình thực hiện khóa luận, giúp cho các bên liên quan có thể theo dõi tiến độ và đánh giá kết quả một cách chính xác và kịp thời. Với các tính năng và lợi ích của nó, hệ thống này sẽ giúp cho sinh viên, giảng viên và trường đại học có thể thực hiện quản lý, theo dõi và đánh giá hoạt động của sinh viên trong quá trình thực hiện khóa luận một cách dễ dàng và hiệu quả hơn. Việc xây dựng hệ thống quản lý học phần khóa luận tốt nghiệp sẽ giúp cho Trường Đại học Công nghiệp Thành phố Hồ Chí Minh nâng cao chất lượng đào tạo và giảm bớt các rủi ro có thể xảy ra trong suốt quá trình quản lý học phần khóa luận tốt nghiệp.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 React

React là một thư viện JavaScript mã nguồn mở được phát triển bởi Facebook để xây dựng giao diện người dùng với các thành phần có thể tái sử dụng. Nhằm tạo ra các ứng dụng web nhanh, hiệu quả và hấp dẫn với nỗ lực viết mã tối thiểu. Với tính chất là một framework vì vậy nó không có các mô hình và bộ điều khiển và khi sử dụng phải kết hợp với các thư viện khác ví dụ: ant design, material ui,… Giống như các framework khác React sử dụng DOM ảo để thay đổi nhưng không phải trực tiếp. Ngoài việc hỗ trợ xây dựng giao diện nhanh chóng, hạn chế lỗi trong quá trình viết mã, cải thiện hiệu suất trang web nó còn thân thiện với SEO trang web của bạn xuất hiện cao hơn trong các tìm kiếm của Google. Sử dụng ít state chỉ dùng khi cần thiết và phản ánh đúng trạng thái của thành phần kết hợp với redux để quản lý state hiệu quả hơn. Các components nhỏ gọn tuỳ theo thành phần chức năng thuận tiện cho việc bảo trì hơn. React đã trở thành một công cụ phổ biến cho việc phát triển ứng dụng web và di động, được sử dụng rộng rãi trong cộng đồng phát triển phần mềm. Nó có cộng đồng lớn, nhiều tài liệu học tập và hỗ trợ từ cộng đồng, giúp người dùng nhanh chóng tìm hiểu và phát triển ứng dụng một cách hiệu quả [1].

### 2.2 React Native

React Native là 1 framework mã nguồn mở và đã được sáng tạo nên bởi Facebook, được dùng để có thể phát triển ứng dụng Web, iOS và Android. Không giống như ReactJS, React Native không thao tác với DOM qua DOM ảo mà nó chạy một quá trình xử lý nền trực tiếp. với cơ chế gọi "bridge" để giao tiếp giữa mã JavaScript và kết hợp nó với các thành phần UI cấp cao được cung cấp bởi React Native. Điều này giúp cải thiện hiệu suất so với các ứng dụng hybrid truyền thống. React Native cho phép chia sẻ mã nguồn giữa các nền tảng khác nhau, giúp tiết kiệm thời gian và công sức trong việc phát triển ứng dụng di động. Một cộng đồng lớn phát triển đông đảo và năng động, cung cấp nhiều tài liệu, thư viện và công cụ hỗ trợ. Chính vì vậy nhiều vấn đề được giải quyết nhanh chóng, không tốn nhiều thời gian để nghiên cứu. React Native đang là một giải pháp vô cùng tuyệt vời dành cho phát triển ứng dụng ngay trên điện thoại di động. Nhưng trên thực tế tới thời điểm hiện tại thì vẫn còn tồn tại 1 số những khuyết điểm nhất định mà bạn cần phải lưu ý [2].

### 2.3 NodeJs

Node.js là một nền tảng mã nguồn mở, đơn luồng và đa nền tảng dành cho ứng dụng phía máy chủ (server-side). Nó được xây dựng trên "V8 JavaScript engine", được viết bằng C++ và JavaScript. Node.js cung cấp

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

hiệu suất cao và tốc độ xử lý nhanh chóng. Nó cũng có một hệ thống mô-đun phong phú và một cộng đồng phát triển lớn. Nền tảng này đã được sử dụng bởi nhiều ứng dụng nổi tiếng như LinkedIn, Microsoft, Yahoo! và PayPal để phát triển các ứng dụng phía máy chủ [3].

### 2.4 Google Compute Engine

Google Compute Engine (GCE) là một dịch vụ điện toán đám mây hạng IaaS (Infrastructure as a Service) được cung cấp bởi Google Cloud Platform (GCP). Cung cấp một máy chủ kỹ thuật số của một máy tính vật lý có thể chạy chương trình và hệ điều hành, lưu trữ dữ liệu, kết nối với mạng và thực hiện các chức năng tính toán khác. Hỗ trợ khả năng mở rộng phần cứng linh hoạt và đảm bảo được hiệu suất, mức độ tin cậy và bảo mật cao [4].

### 2.5 Docker

Docker là một nền tảng phần mềm mã nguồn mở cho phép dựng, kiểm thử và triển khai ứng dụng nhanh chóng bằng cách đóng gói phần mềm và các thành phần liên quan vào container - các đơn vị tiêu chuẩn hóa. Mỗi container bao gồm tất cả những gì cần thiết để ứng dụng chạy, từ thư viện, công cụ hệ thống, mã và thời gian chạy. Các container được tạo ra từ hình ảnh, là các gói đã được xây dựng trước chứa các thành phần và cấu hình sẵn sàng. Docker giúp gia tăng tốc độ phát triển và triển khai ứng dụng bằng cách tạo môi trường cô lập và đồng nhất giúp nhanh chóng triển khai và thay đổi quy mô ứng dụng trên môi trường hỗ trợ Docker, đảm bảo ứng dụng chạy ổn định và giảm xung đột liên quan đến môi trường. Docker mang lại tính nhất quán và độ tin cậy cao cho quá trình triển khai ứng dụng của bạn [5].

### 2.6 CI/CD

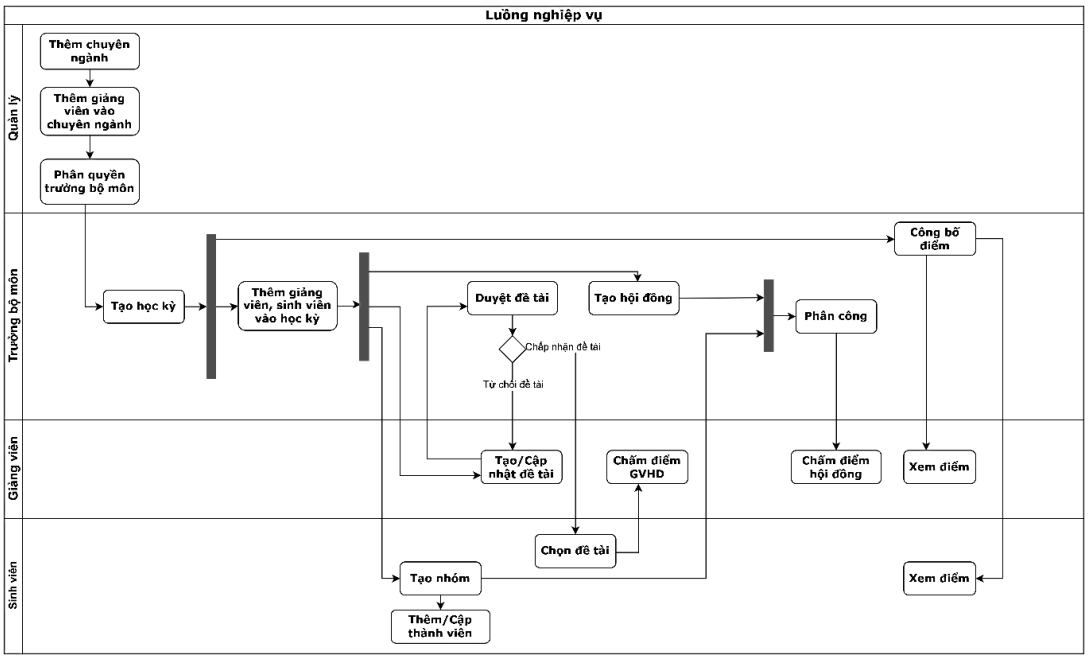
CI là viết tắt của Continuous Integration (tích hợp liên tục) và CD có thể là Continuous Delivery (chuyển giao liên tục) hoặc Continuous Deployment (triển khai liên tục). CI/CD đề cập đến việc tự động hóa trong quy trình phát triển phần mềm và chuyển giao sản phẩm. Nó giúp tích hợp diễn ra nhanh chóng và sản phẩm hoàn thiện được chuyển đến người dùng trong thời gian ngắn nhất. Chúng tôi sử dụng công cụ GitLab CI/CD là một trong những công cụ phổ biến được sử dụng để triển khai CI/CD trong quy trình phát triển phần mềm [6].

## 3 XÂY DỰNG HỆ THỐNG QUẢN LÝ HỌC PHẦN KHÓA LUẬN TỐT NGHIỆP ÁP DỤNG TẠI TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH

### 3.1 Giới thiệu

Thông qua việc thu thập thông tin về các yêu cầu cơ bản của hệ thống hỗ trợ triển khai học phần khóa luận tốt nghiệp. Xác định các chức năng chính mà hệ thống cần cung cấp, như tạo và quản lý hồ sơ sinh viên, theo dõi tiến độ triển khai học phần, quản lý và đánh giá. Để triển khai ứng dụng "Xây dựng hệ thống quản lý học phần khóa luận tốt nghiệp áp dụng tại Trường Đại học Công nghiệp Thành phố Hồ Chí Minh" hệ thống đã sử dụng các công nghệ như React deploy trên Vercel thông qua Github, React Native, NodeJS kết hợp với Express và lưu trữ dữ liệu bằng MySQL và quá trình triển khai tự động được thực hiện thông qua công cụ Gitlab CI/CD, sử dụng Docker để triển khai lên Google Compute Engine. Bên cạnh đó là sự kết hợp với JWT, Redis, Bcrypt nhằm hỗ trợ tương tác giữa client và server. Việc sử dụng các công nghệ trên vì chúng được sử dụng rộng rãi có nhiều tài nguyên hỗ trợ trong việc hiện thực hệ thống. Ngoài ra những hệ thống có thể dùng được các công nghệ linh hoạt và hiệu quả hơn do chúng đề là những công nghệ mới và có thể đáp ứng tốt yêu cầu của ứng dụng. Cuối cùng là việc dùng các công cụ có mã nguồn mở sẽ tiết kiệm chi phí khá lớn cho hệ thống.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 1: Sơ đồ mô tả tổng quan nghiệp vụ của hệ thống

Với sơ đồ mô tả tổng quan của ứng dụng cho thấy người quản lý là người thực hiện công việc cung cấp tài khoản cho các vai trò phía dưới như: trưởng bộ môn, giảng viên và sinh viên. Trưởng bộ môn là người quản lý các mốc thời gian và các công việc, Giảng viên và sinh viên thực hiện các chức năng của mình theo các mốc thời gian.

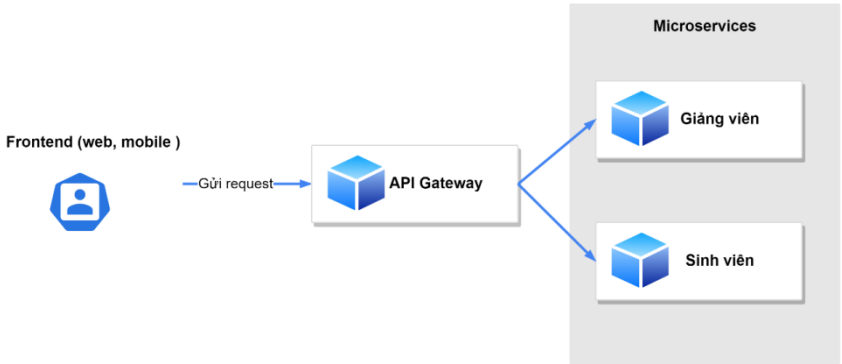
## 3.2 Kiến trúc hệ thống

Hệ thống quản lý học phần khóa luận tốt nghiệp áp dụng tại Trường Đại học Công nghiệp thành phố Hồ Chí Minh, được xây dựng trên mô hình Client-Server và cả 2 nền tảng website và ứng dụng di động nhằm mang lại nhiều lợi ích về hiệu năng, tương thích cụ thể:

- Trang web dành cho người quản lý, trưởng bộ môn và giảng viên.
- Ứng dụng di động dành cho sinh viên.

Triển khai Backend bằng NodeJS và sử dụng database là MySQL sẽ cho hệ thống được khả năng xử lý nhanh chóng và chặt chẽ về dữ liệu, xây dựng hệ thống theo kiến trúc microservice được tách thành ba dịch vụ nhỏ: dịch vụ giảng viên, dịch vụ sinh viên và dịch vụ điều hướng đồng thời áp dụng công nghệ như redis để xử lý được lượng người dùng đồng thời cao.

Triển khai Frontend bằng React JS cho website và React Native cho ứng dụng di động mang lại nhiều lợi ích về hiệu năng, tương thích cao và trải nghiệm người dùng tốt.




Hình 2: Sơ đồ kiến trúc của hệ thống


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.3 Phân tích hệ thống

Đối tượng dùng trong hệ thống bao gồm giảng viên, trưởng bộ môn, sinh viên và quản lý. Các chức năng thuộc cấp giảng viên được kế thừa theo các cấp quản lý là người có quyền cao nhất, sau đó là trưởng bộ môn và cuối cùng là giảng viên.

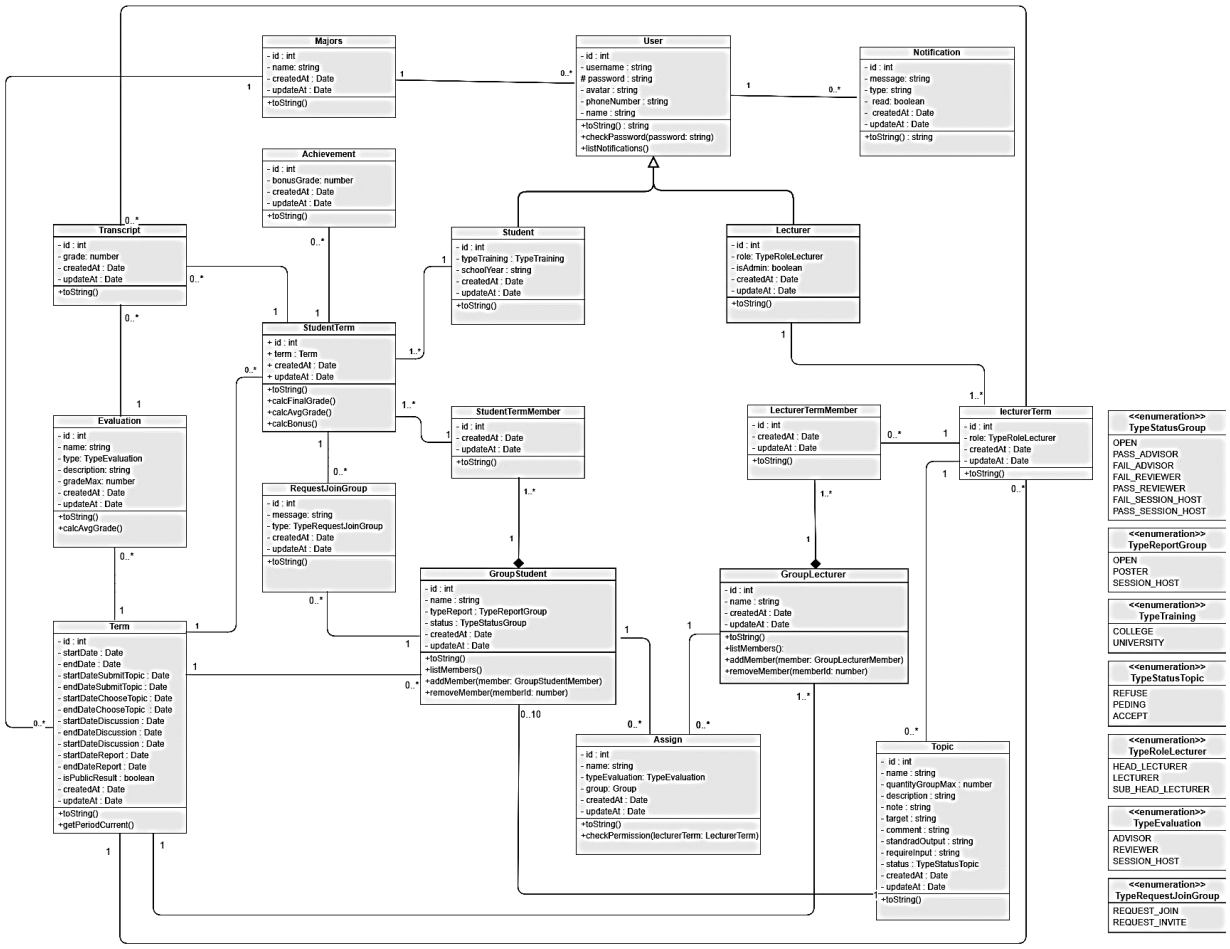
Bảng 1: Danh sách các chức năng trong hệ thống

| Đối tượng | Tên chức năng | Mô tả |
|---|---|---|
| **Khách** | Đăng nhập | Đăng nhập tài khoản là mã số sinh viên hoặc giảng viên vào hệ thống |
| | Quên mật khẩu | Đường dẫn đặt lại mật khẩu sẽ được gửi qua email người dùng |
| **Người dùng xác thực** | Nhận các thông báo | Nhận các thông báo như duyệt đề tài, phân công chấm điểm… trên ứng dụng và email |
| | Xem thông tin học kỳ | Xem được các thông tin học kỳ như: Ngày bắt đầu, ngày kết thúc, ngày chọn đề tài… ở học kỳ hiện tại |
| | Xem các thông tin trong hệ thống | Xem được các thông tin như danh sách đề tài, danh sách sinh viên, danh sách giảng viên, thông tin nhóm sinh viên, thông tin chuyên ngành, điểm số |
| **Sinh viên** | Quản lý nhóm sinh viên | Tạo, cập nhật thông tin nhóm, thoát nhóm và gửi và nhận lời mời hoặc yêu cầu tham gia nhóm |
| | Đăng ký đề tài | Sinh viên đã có nhóm sẽ chọn đề tài thực hiện khóa luận trên các đề tài đã được duyệt |
| | Xem kết quả khóa luận | Xem được các kết quả khóa luận của mình như: điểm tổng kết, điểm phản biện, điểm giáo viên hướng dẫn khi trưởng bộ môn công bố |
| **Giảng viên** | Quản lý nhóm sinh viên thực hiện khóa luận | Xem các thông tin như điểm, danh sách sinh viên trong nhóm, trạng thái đậu hoặc rớt, điểm số… của các nhóm sinh viên đã chọn đề tài của mình làm khóa luận |
| | Tạo và cập nhật đề tài | Tạo, cập nhật các đề tài khóa luận |
| | Chấm điểm | Chấm điểm cho sinh viên mình quản lý hoặc được phân công |
| **Trưởng bộ môn** | Quản lý học kỳ | Tạo và cập nhật các thông tin trong học kỳ |
| | Quản lý giảng viên | Thêm giảng viên vào trong học kỳ bằng cách nhập thủ công hoặc import excel |
| | Quản lý sinh viên | Thêm sinh viên vào trong học kỳ bằng cách nhập thủ công hoặc import excel |
| | Quản lý nhóm hội đồng | Tạo, cập nhật, xóa các nhóm hội đồng |
| | Quản lý tiêu chí đánh giá | Tạo, cập nhật. xóa các tiêu chí đánh giá chấm điểm theo từng học kỳ |
| | Duyệt đề tài | Chấp nhận hoặc từ chối các đề tài của giảng viên |
| | Phân công chấm điểm | Phân công nhóm hội đồng chấm điểm cho các nhóm sinh viên |
| **Quản lý** | Quản lý chuyên ngành | Tạo và cập nhật chuyên ngành |
| | Phân công Trưởng bộ môn | Phân công giảng viên làm Trưởng bộ môn |

Từ những chức năng trên, chúng tôi tập trung vào các chức năng chính như: Đăng ký đề tài, tạo và duyệt đề tài, phân công chấm điểm. Sau đó chúng tôi mở rộng các chức năng tương ứng như trong mô hình use case.

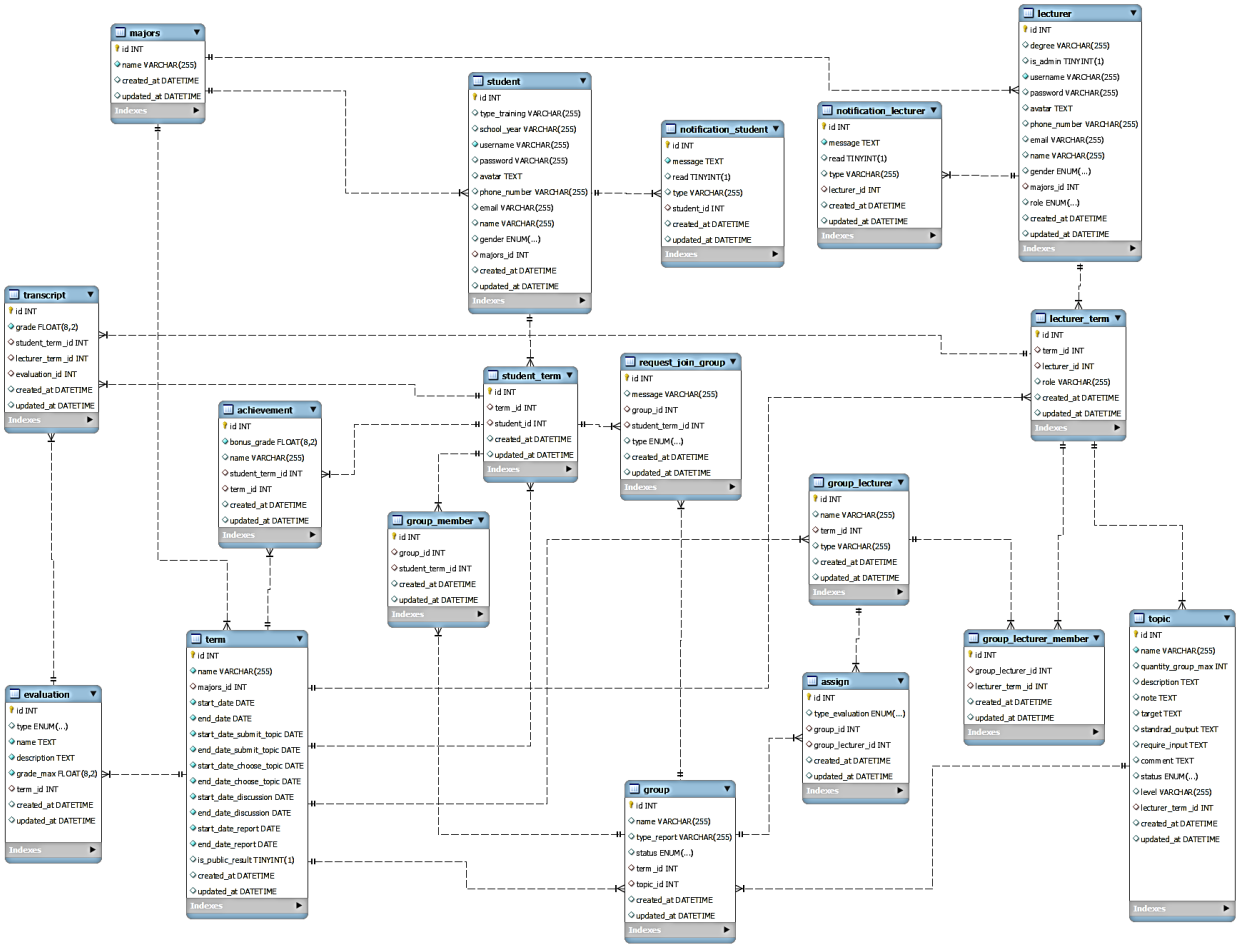
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Mô hình use case của chúng tôi bao gồm sáu thành phần actor chính. Đầu tiên là "khách" - đại diện cho người dùng mới truy cập vào hệ thống. Tiếp theo, chúng tôi có "Người dùng xác thực" - đại diện cho người dùng đã đăng nhập vào hệ thống. "Sinh viên" và "giảng viên" là các người dùng kế thừa từ "Người dùng xác thực". "Trưởng bộ môn" là người dùng kế thừa từ "giảng viên". Cuối cùng là "quản lý".
Dựa trên các use case đã phân tích ở trên, chúng tôi đã tạo ra sơ đồ lớp (Hình 3) của hệ thống. Trong sơ đồ này, chúng tôi đặc biệt quan tâm đến các lớp bao gồm "Bảng đề tài", "Sinh viên", "Giảng viên" và "Bảng điểm", vì chúng đảm nhiệm vai trò trong việc quản lý dữ liệu của hệ thống. Ngoài ra, các lớp khác cũng được sử dụng để làm rõ các mối quan hệ, chức năng chính và ràng buộc dữ liệu trong hệ thống.




Hình 3: Sơ đồ lớp

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

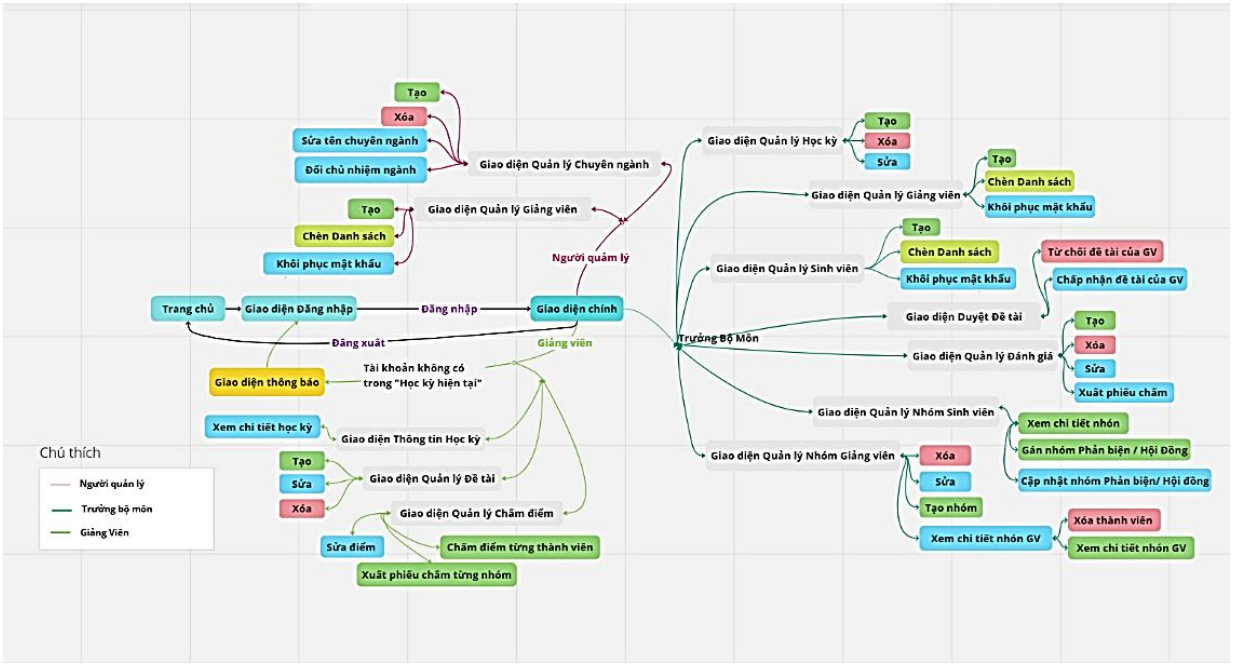



Hình 4: Sơ đồ cơ sở dữ liệu

## 3.4 Giao diện

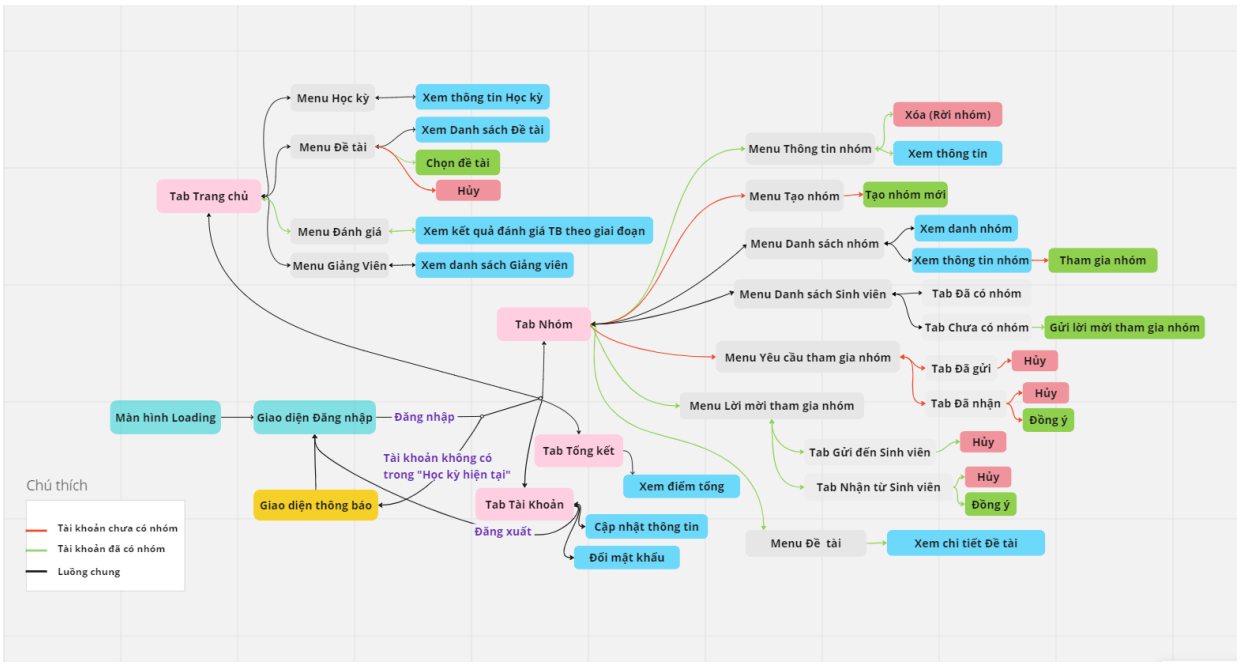
Hệ thống quản lý học phần khóa luận tốt nghiệp áp dụng tại Trường Đại học Công nghiệp Thành phố Hồ Chí Minh được xây dựng trên hai nền tảng trang web và ứng dụng di động. Gồm các chức năng quản lý liên quan đến: học kỳ, đề tài, giảng viên, sinh viên, chọn đề tài của sinh viên và phân công chấm điểm,... Hệ thống có 2 chức năng chính: thứ nhất là quá trình tạo duyệt đề tài, thứ hai là phân công chấm điểm.

**Sơ đồ luồng màn hình trang web:**




Hình 5: Sơ đồ luồng màn hình trang web

**Sơ đồ luồng màn hình ứng dụng di động:**




Hình 6: Sơ đồ luồng màn hình ứng dụng di động

### 3.4.1 Giao diện ứng dụng di động dành cho sinh viên

**Quá trình tạo nhóm, thêm thành viên và đăng ký đề tài:** Sinh viên chọn đề tài trong điều kiện đã có nhóm (Hình 7) và số lượng sinh viên trong nhóm phải có từ 2 thành viên. Trường hợp sinh viên chưa có nhóm có thể tạo nhóm mới  (Hình 7), [2] sẽ là nút “Tham gia nhóm". Sinh viên chọn đề tài nhấn vào nút “Chọn đề tài" (Hình 8).


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

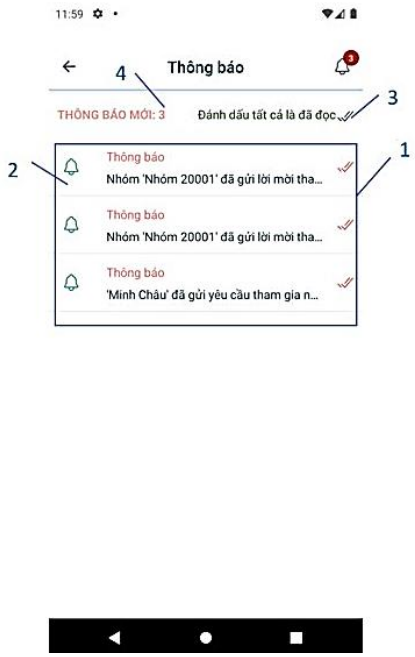



Hình 7: Giao diện thông tin nhóm




Hình 8: Giao diện chọn đề tài

**Nhận thông báo và xem điểm:** Tất cả thông báo liên quan đến nhóm, chọn đề tài, chấm điểm sẽ được thông báo đến sinh viên (Hình 9). Sau khi sinh viên được giảng viên chấm điểm kết quả của các giai đoạn sẽ được hiển thị tương ứng (Hình 10).




Hình 9: Giao diện thông báo



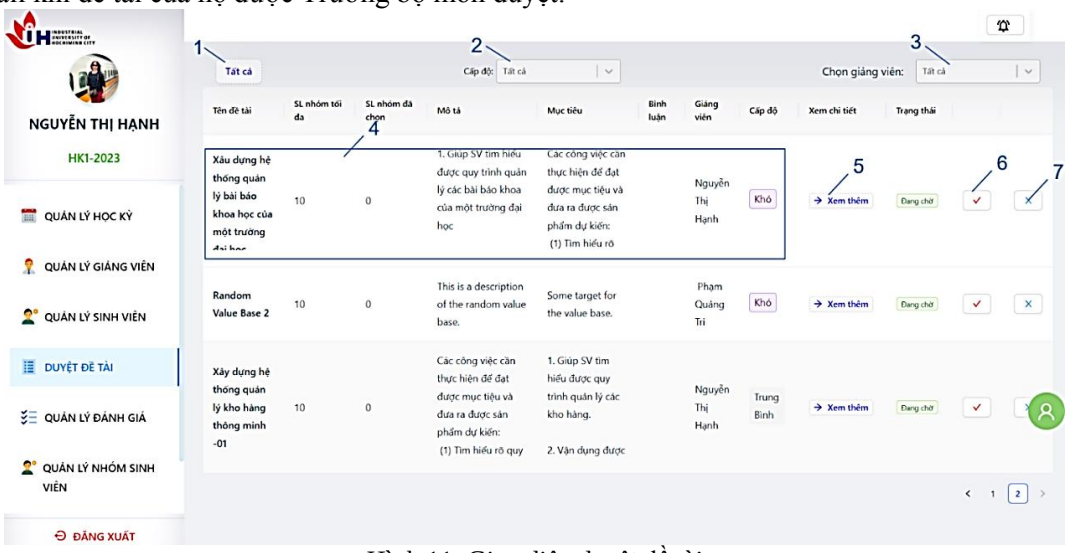

Hình 10: Giao diện hiển thị kết quả

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Hình 9: [1] Hiển thị tất cả thông báo, [2] Nhấn vào để đọc thông báo, thông báo sẽ dẫn đến màn hình chứa nội dung tương ứng. [3] Đánh dấu tất cả thông báo đã đọc. [4] Số lượng thông báo mới chưa đọc.
Hình 10: Điểm trung bình: [1] hướng dẫn, [2] phản biện [3] hội đồng. [4] Điểm trung bình tổng, [5] Hiển thị trạng thái sau khi có 'Điểm trung bình" (Đậu/Rớt).

### 3.4.2 Giao diện trang web dành cho giảng viên

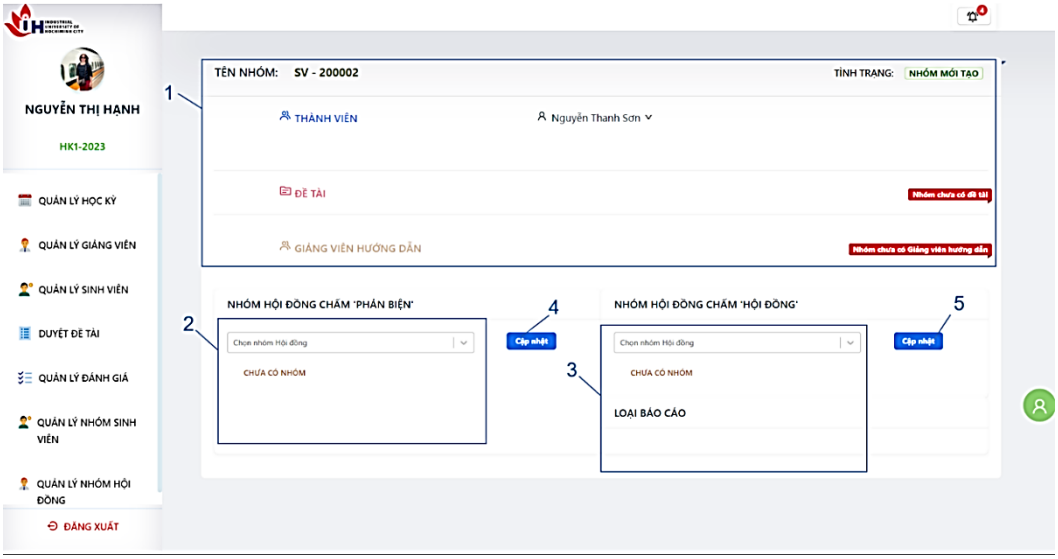
**Quá trình tạo duyệt đề tài:** Giảng viên trong danh sách học kỳ sẽ đăng nhập vào hệ thống và tạo đề tài trong mục quản lý đề tài. Sau đó, Trưởng bộ môn sẽ vào mục duyệt đề tài (Hình 11) và có thể chấp nhận hoặc từ chối đề tài, đồng thời bình luận về đề tài đó. Giảng viên sẽ nhận được thông báo trên hệ thống và qua email khi đề tài của họ được Trưởng bộ môn duyệt.



Hình 11: Giao diện duyệt đề tài

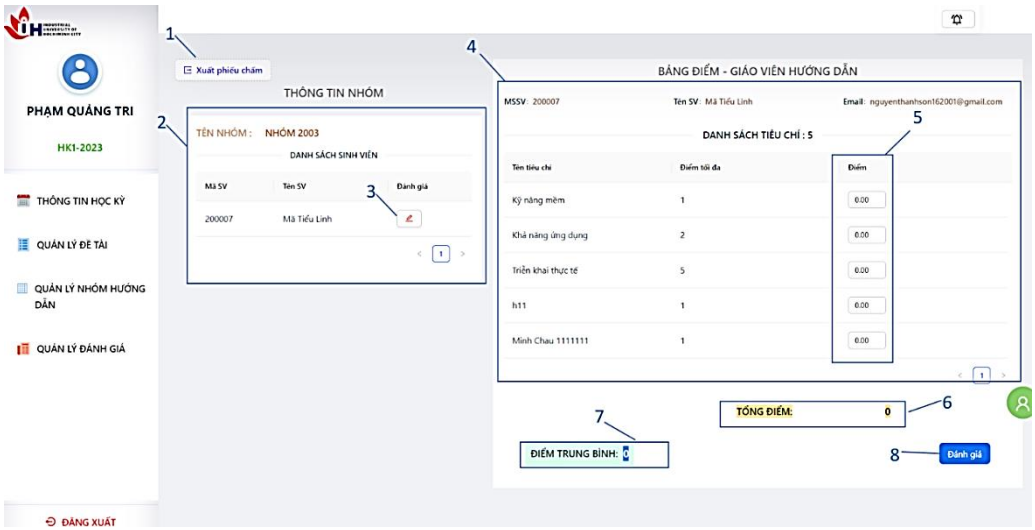
Giao diện hiển thị với các mục: [1] nút hiển thị danh sách các đề tài trong học kỳ, [2] nút chọn cấp độ đề tài (Khó, Rất khó, Bình thường), [3] lọc đề tài theo giảng viên, [4] thông tin đề tài, [5] nút hiển thị chi tiết đề tài, [6] nút chấp nhận đề tài, [7] nút từ chối đề tài.
**Quá trình phân công chấm điểm:** Trưởng bộ môn sẽ đăng nhập vào hệ thống và tạo các nhóm hội đồng giảng viên. Sau đó vào nhóm của sinh viên cần phân công (Hình 3.4.2.2) và phân công nhóm hội đồng phản biện hoặc hội đồng báo cáo cho nhóm sinh viên. Các giảng viên được phân công chấm điểm sẽ nhận được thông báo thông qua ứng dụng và email đồng thời giảng viên sẽ chọn nhóm đã được phân công và tiến hành chấm điểm cho từng sinh viên trong nhóm đó (Hình 3.4.2.3). Khi tất cả giảng viên hoàn tất quá trình chấm điểm, trưởng bộ môn có thể công bố điểm cho sinh viên, và sinh viên có thể xem điểm thông qua xuất danh sách dạng Excel.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 12: Giao diện chi tiết nhóm sinh viên

Giao diện hiển thị với các mục: [1] thông tin nhóm, [2] thông tin nhóm hội đồng chấm phản biện, [3] nút phân công nhóm hội đồng phản biện, [4] thông tin nhóm hội đồng chấm hội đồng báo cáo, [5] nút phân công nhóm hội đồng báo cáo.




Hình 13: Giao diện chấm điểm cho sinh viên của giảng viên

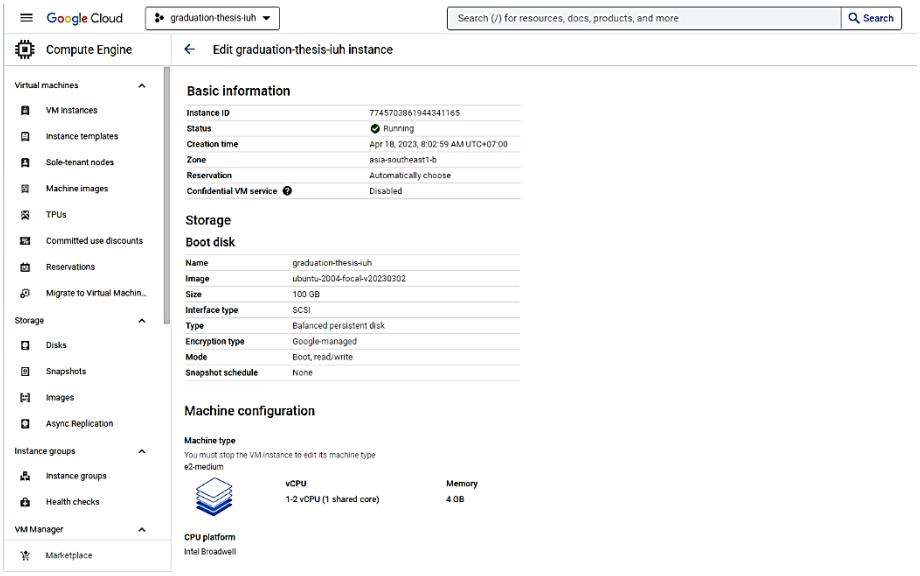
Giao diện hiển thị với các mục: [1] nút xuất phiếu chấm, [2] thông tin các thành viên nhóm sinh viên, [3] nút hiển thị thông tin đánh giá của từng sinh viên, [4] Thông tin tiêu chí đánh giá, [5] ô chấm điểm cho từng tiêu chí, [6] Tổng điểm của sinh viên đang được chấm, [7] điểm trung bình của tất cả các giáo viên đối với sinh viên đang được chấm, [8] nút xác nhận chấm điểm.

## 4 Kiểm nghiệm thực tế

### 4.1 Triển khai

Chúng tôi đã tiến hành triển khai hệ thống trên VPS của Google Compute Engine. Trong giai đoạn này chúng tôi sử dụng một VPS chạy hệ điều hành Ubuntu với cấu hình 100GB SSD, 2 Core vCPU và 4 GB RAM. Với cấu hình tương đối mạnh mẽ như trên thì hệ thống có sẽ đáp ứng được lượng người dùng truy cập đồng thời cao.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 14: Thông tin cấu hình chi tiết VPS của Google Compute Engine.

Đồng thời phiên bản mobile của hệ thống cũng sẽ được triển khai trên Google Play và AppStore để thuận tiện hơn cho sinh viên sử dụng.

### 4.2 Đánh giá chung của người dùng

Thử nghiệm trên khoa Công nghệ thông tin trường Đại học Công nghiệp Tp Hồ Chí Minh nhận được đánh giá tích cực từ người dùng về giao diện đẹp, thuận tiện và hệ thống đáp ứng nhanh. Tuy nhiên, còn hạn chế về đồng bộ hóa giữa ứng dụng di động và website, cũng như việc cho phép giảng viên chấp nhận/từ chối yêu cầu chọn đề tài của sinh viên.

Nhìn nhận những hạn chế này, chúng tôi hoàn toàn nhận thức và thấu hiểu tính hợp lý cũng như khả thi của những đánh giá từ người dùng. Chúng tôi sẽ tiếp tục cải tiến và phát triển ứng dụng để khắc phục những hạn chế này và cung cấp một trải nghiệm tốt hơn cho người dùng trong tương lai.

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Trong bài bào này chúng tôi đã đưa ra giải pháp cho việc quản lý khóa luận tốt nghiệp của trường Đại học Công nghiệp Tp Hồ Chí Minh một cách tiết kiệm thời gian cách thuận lợi, nhanh chóng và chính xác. Hệ thống được phát triển nhằm cung cấp một giao diện dễ sử dụng và thuận tiện cho giảng viên và sinh viên trong việc quản lý các mốc thời gian và công việc liên quan đến khóa luận. Giảng viên có thể tạo và quản lý các mốc thời gian trong học kỳ, tạo và duyệt đề tài khóa luận, theo dõi tiến độ của nhóm thực hiện khóa luận và phân công chấm điểm cho sinh viên. Sinh viên dễ dàng tạo nhóm làm việc, chọn đề tài khóa luận và theo dõi các mốc thời gian, cho phép sinh viên mời thành viên khác vào nhóm hoặc gửi yêu cầu tham gia nhóm, đồng thời cung cấp thông tin về tiến độ và kết quả của khóa luận. Hệ thống được triển khai trên cả nền tảng di động và trang web, tăng tính khả thi và phạm vi sử dụng. Đồng thời giao diện người dùng thân thiện, thông báo trên hệ thống và qua email giúp người dùng dễ dàng kiểm soát tiến độ thực hiện khóa luận. Triển khai hệ thống sử dụng phương pháp CI/CD giúp tối ưu quá trình cập nhật và phát triển sau này và được thiết kế để chịu được lượng người dùng đồng thời cao bằng cách sử dụng công nghệ Redis và triển khai theo kiến trúc microservice. Điều này giúp hệ thống đáp ứng tốt yêu cầu của người dùng và đảm bảo hiệu suất cao.

Trong tương lai hệ thống sẽ được thử nghiệm tại khoa Công nghệ thông tin của trường Đại học Công nghiệp Tp Hồ Chí Minh và sẽ tiếp tục phát triển và mở rộng với quy mô toàn trường. Hơn nữa chúng tôi sẽ tích hợp AI để đưa ra các dự đoán, gợi ý chọn đề tài cho sinh viên dựa trên điểm số các môn, bên cạnh đó đưa ra các thống kê giá trị hơn cho giảng viên và sinh viên.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## TÀI LIỆU THAM KHẢO

[1]. 2023), [online], https://legacy.reactjs.org

[2]. Eric Windmill, Exploring Cross-Platform Development with Flutter, React Native, and Xamarin, (2020).

[3]. TopDev (6, 2023), [online], https://nodejs.org/en/about

[4]. Google (6, 2023), [online], https://cloud.google.com/compute.

[5]. Docker (6, 2023), [online], https://aws.amazon.com/vi/docker/.

[6]. Sander Rossel, continuous Integration, Delivery, and Deployment (2017)

[7]. Barbara Haley Wixom, Systems Analysis and Design (2018)

[8]. Sam Newman, Building Microservices: Designing Fine-Grained Systems (2015)

[9]. David Herron, NodeJS Web Development (2016)

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F029**

# XÂY DỰNG ỨNG DỤNG VẬN CHUYỂN HÀNG BẰNG XE TẢI

NGUYỄN THỊ HẠNH, VƯƠNG ÁNH LÂM, NGUYỄN TRUNG QUỐC

*Khoa Công nghệ thông tin – Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*

*ngthihanh@iuh.edu.vn, lamdavid821@gmail.com, quoc21921@gmail.com*

**Tóm tắt.** Vận chuyển hàng hóa là một hoạt động vô cùng quan trọng trong đời sống hàng ngày của con người. Tuy nhiên, việc quản lý và tối ưu hóa quy trình vận chuyển thường đòi hỏi nhiều thời gian, công sức và chi phí. Vì vậy, chúng tôi đã xây dựng ứng dụng quản lý vận chuyển hàng hóa bằng xe tải với mục đích giảm thiểu tối đa các khó khăn này và giúp các tài xế xe tải có thêm đơn hàng vận chuyển hàng hóa khi dùng ứng dụng. Ứng dụng sử dụng React để xây dựng ứng dụng tích hợp với Map4D API để cung cấp thông tin về địa điểm giao nhận hàng hóa, tìm đường đi ngắn nhất và dùng MongoDB Atlas để lưu trữ dữ liệu. Ứng dụng còn giúp quản lý và theo dõi đơn hàng theo thời gian thực. Với ứng dụng này, chúng tôi hy vọng sẽ giúp đỡ các doanh nghiệp và người dùng tiết kiệm thời gian và chi phí, đồng thời nâng cao hiệu quả vận chuyển hàng hóa.

**Từ khóa.** React, Map4D API, MongoDB Atlas, realtime

## BUILDING A TRUCK TRANSPORTATION APP

**Abstract.** Transportation is an important activity in our daily life. However, managing and optimizing the shipping process often requires a lot of time, effort and cost. Therefore, we have built a trucking management application with the aim of minimizing these difficulties and helping truckers have more freight orders when using the application. The application uses React to build an application that integrates with the Map4D API to provide information about freight forwarding locations, find the shortest route, and uses MongoDB Atlas to store data. The application also helps manage and track orders in real time. We hope to help businesses and users optimize their time and cost and improve their freight efficiency with this application.

**Keyword.** React, Map4D API, MongoDB Atlas, realtime

## 1 GIỚI THIỆU TỔNG QUAN

Trong thời đại hiện nay, sự phát triển mạnh mẽ của công nghệ thông tin và nhu cầu ngày càng cao của xã hội đã tạo ra một sự phụ thuộc không thể thiếu vào việc áp dụng công nghệ thông tin vào các hoạt động sinh hoạt và hoạt động sản xuất. Trong lĩnh vực vận chuyển hàng hóa, việc sử dụng xe tải đóng vai trò quan trọng và không thể thiếu trong nền kinh tế hiện đại, đặc biệt là đối với những khách hàng có nhu cầu vận chuyển hàng hóa với khối lượng lớn.

Tuy nhiên, việc quản lý và tối ưu hóa quá trình vận chuyển hàng hóa vẫn đang gặp phải nhiều khó khăn và thách thức. Một số khó khăn đáng kể bao gồm việc tìm kiếm dịch vụ vận chuyển phù hợp với nhu cầu, thiếu sự liên kết và chia sẻ thông tin giữa các bên liên quan, khó khăn trong việc theo dõi quá trình vận chuyển hàng hóa, chi phí vận chuyển chưa được tối ưu và hiệu quả, cũng như việc người dùng gặp vấn đề và cần giải quyết khiếu nại với đơn vị vận chuyển mà không có đủ thông tin để xác minh.

Để giải quyết những vấn đề này, việc áp dụng phần mềm vào quá trình vận chuyển hàng hóa là rất cần thiết. Tuy nhiên, trên thị trường hiện nay vẫn chưa có nhiều phần mềm đáp ứng đầy đủ nhu cầu này, cũng như chưa xuất hiện những hệ thống tích hợp có khả năng bao quát toàn bộ quá trình vận chuyển hàng hóa.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Nhận thấy tình hình này, chúng tôi nhận thức rằng cần phát triển một hệ thống quản lý quá trình vận chuyển hàng hóa bằng xe tải, nhằm giúp người dùng quản lý tốt hơn quá trình vận chuyển hàng hóa, tối ưu chi phí và hiệu quả vận chuyển, đồng thời góp phần số hóa ngành công nghiệp vận tải đường bộ.
Với mục tiêu đó, chúng tôi đã chọn đề tài "Xây dựng ứng dụng vận chuyển hàng bằng xe tải" nhằm giúp khách hàng có thể tạo đơn hàng, theo dõi quá trình vận chuyển hàng hóa, và cung cấp một cơ chế cho các đối tác tài xế để nhận các đơn hàng phù hợp và kiếm thêm thu nhập. Ngoài ra, hệ thống còn cung cấp các tính năng đánh giá chất lượng dịch vụ, giúp đảm bảo chất lượng và sự hài lòng của khách hàng.
Hệ thống quản lý quá trình vận chuyển hàng hóa bằng xe tải cũng hỗ trợ tối ưu hóa chi phí vận chuyển và tăng hiệu quả vận chuyển. Hệ thống sẽ tự động tìm kiếm các đối tác tài xế phù hợp để vận chuyển hàng hóa, giúp giảm thiểu chi phí vận chuyển và tăng cường hiệu quả vận chuyển.
Bằng cách kết hợp công nghệ thông tin và ngành vận tải, chúng tôi hy vọng rằng hệ thống này sẽ mang lại sự thuận tiện và tiết kiệm thời gian cho khách hàng, đồng thời cung cấp một giải pháp toàn diện để quản lý và theo dõi quá trình vận chuyển hàng hóa bằng xe tải. Qua đó, hệ thống có thể đóng góp tích cực vào việc phát triển và cải tiến ngành công nghiệp vận tải, đồng thời nâng cao hiệu quả và sự tin cậy trong lĩnh vực này.

## 2 CÁC CÔNG NGHỆ LIÊN QUAN

### 2.1 NodeJS

Node JS được phát triển từ Javascript V8 Engine từ năm 2009 bởi Ryan Dahl với mục đích đơn giản hóa việc xây dựng cho các ứng dụng web [1]. Nền tảng này được phát triển bởi Ryan Lienhart Dahl vào năm 2009. Node.js ra đời khi các developer đời đầu của JavaScript mở rộng nó từ một thứ bạn chỉ chạy được trên trình duyệt thành một thứ bạn có thể chạy trên máy của mình dưới dạng ứng dụng độc lập (thực thi mã JavaScript bên ngoài trình duyệt web) [2] .

### 2.2 MongoDB Alats

MongoDB Atlas là một dịch vụ quản lý NoSQL có khả năng đáp ứng hiệu suất cao và nhanh kèm theo khả năng mở rộng [3]. Nếu bạn là một nhà phát triển, bạn có thể sử dụng MongoDB Atlas để tạo ra một bảng có khả năng lưu trữ và truy xuất bất kỳ số lượng dữ liệu, mà vẫn có thể phục vụ cho bất kỳ mức độ lưu lượng truy cập.

### 2.3 Firebase

Firebase là nền tảng di động của Google giúp bạn nhanh chóng phát triển các ứng dụng chất lượng cao và phát triển doanh nghiệp của mình [4]. Chức năng chính là giúp người dùng lập trình ứng dụng bằng cách đơn giản hóa các thao tác với cơ sở dữ liệu. Cụ thể là những giao diện lập trình ứng dụng API đơn giản. Mục đích nhằm tăng số lượng người dùng và thu lại nhiều lợi nhuận hơn. Đặc biệt, còn là dịch vụ đa năng và bảo mật cực tốt. Firebase hỗ trợ cả hai nền tảng Android và IOS. Không có gì khó hiểu khi nhiều lập trình viên chọn Firebase làm nền tảng đầu tiên để xây dựng ứng dụng cho hàng triệu người dùng trên toàn thế giới.

### 2.4 ReactJS

ReactJS là một thư viện của ngôn ngữ lập trình Javascript [5] do Facebook xây dựng và duy trì. Theo nhà sáng lập ReactJS, Jordan Walke, React là một thư viện JavaScript với mã nguồn mở hiệu quả, dễ khai báo và linh hoạt để xây dựng giao diện người dùng đơn giản, nhanh chóng và có thể mở rộng của các ứng dụng web [6].

### 2.5 React Native

Được phát triển bởi Facebook, React Native là một framework hướng đến phát triển ứng dụng di động đa nền tảng. Với sự trợ giúp của React Native, lập trình viên (developer) có thể sử dụng JavaScript để tạo ra mobile apps (ứng dụng di động) hỗ trợ cho cả nền tảng Android và iOS. Instagram, Facebook, Skype,… là những ứng dụng nổi bật sử dụng React Native. React Native hoạt động được nhờ tích hợp hai thread là Main Thread và JS Thread cho ứng dụng mobile. Trong đó: Main Thread: cập nhật giao diện người dùng (UI) và xử lý tương tác người dùng; JS Thread: thực thi và xử lý code Javascript. Hai luồng Main Thread

và JS Thread này hoạt động độc lập với nhau. Hai Thread sẽ tương tác với nhau nhờ một Bridge (cầu nối). Chiếc cầu này sẽ chuyển đổi dữ liệu qua lại giữa các Thread [7].

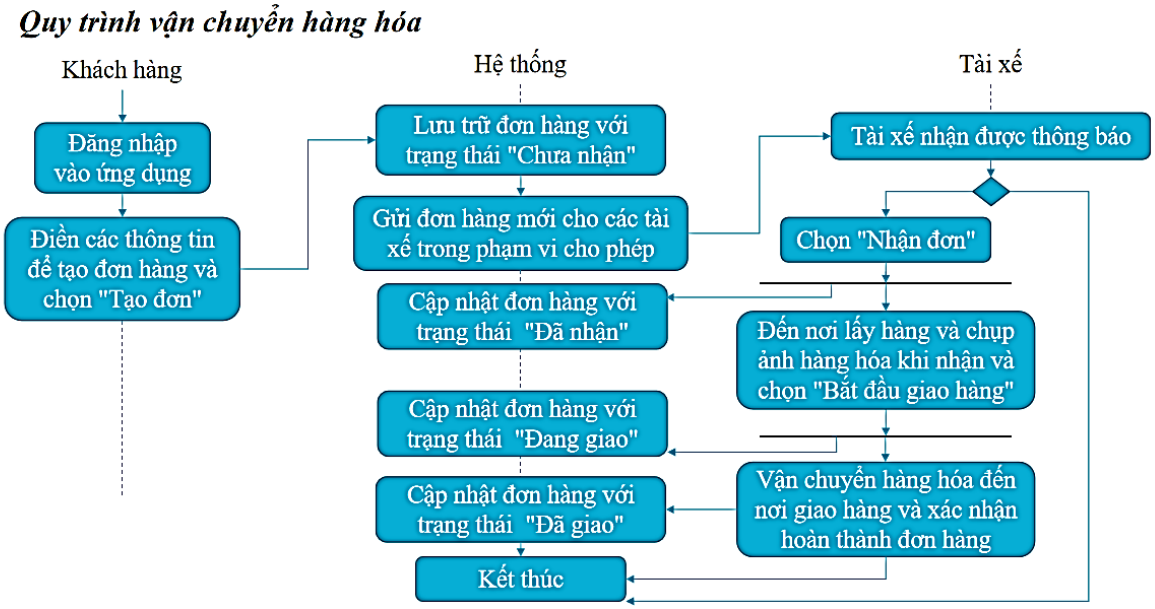
### 2.6. Map4D

Map4D là một sản phẩm của VCCorp, một công ty công nghệ hàng đầu tại Việt Nam. Map4D cung cấp một nền tảng định vị và bản đồ 3D mạnh mẽ, cung cấp giải pháp tương tác và trực quan cho việc hiển thị thông tin địa lý. Nền tảng này hỗ trợ nhiều tính năng tiện ích như tìm kiếm địa điểm, điều hướng, hiển thị thông tin chi tiết, định vị thời gian thực, và tạo ra các trải nghiệm ảo trên màn hình. Có thể tận dụng Map4D để xây dựng ứng dụng di động, trang web hoặc hệ thống để mang đến cho người dùng trải nghiệm địa lý tốt nhất và giải quyết các vấn đề liên quan đến vị trí và dữ liệu địa lý [8].

## 3 GOTRUCK: XÂY DỰNG ỨNG DỤNG VẬN CHUYỂN HÀNG HÓA BẰNG XE TẢI

### 3.1. Giới thiệu

Vận chuyển hàng hóa thực sự có nhiều quy trình khác nhau, từ giai đoạn, đóng gói hàng hóa, gửi hàng, theo dõi vận chuyển cho đến khi hàng đến nơi. Dưới đây là quy trình vận chuyển hàng hóa được áp dụng:

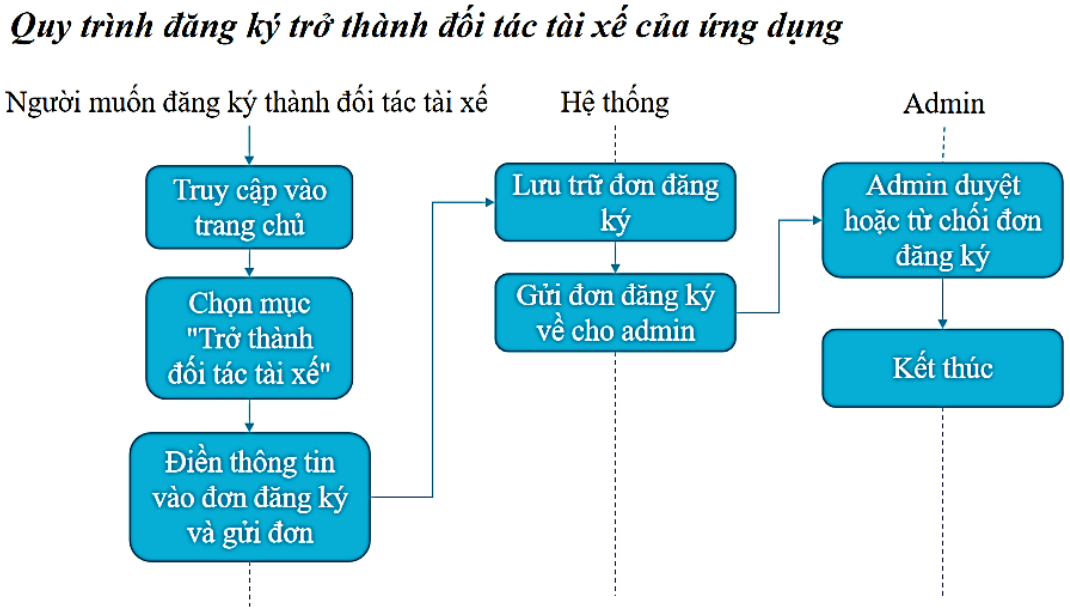



Hình 1. Sơ đồ quy trình vận chuyển hàng hóa

Khách hàng điền đầy đủ thông tin theo yêu cầu của đơn hàng như: địa chỉ lấy hàng, giao hàng, kích thước, trọng lượng, thông tin cá nhân… và tiến hành tạo đơn. Hệ thống sẽ gửi thông báo cho tất cả tài xế trong phạm vi nhất định gần nơi nhận hàng. Khi tài xế nhận được thông báo thì có thể xem chi tiết đơn hàng và nhận đơn hoặc từ chối, khi chọn nhận đơn và tài xế đến nơi lấy hàng để nhận hàng. Tài xế vận chuyển đơn hàng đến nơi giao hàng và kết thúc quá trình vận chuyển hàng hóa.

Tuy nhiên, để có thể trở thành tài xế vận chuyển hàng hóa thì người đăng ký phải truy cập trang đăng ký của ứng dụng và điền các thông tin của đơn đăng ký và gửi về cho quản trị viên, quy trình đó như sau:


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 2. Sơ đồ quy trình đăng ký trở thành đối tác tài xế

Người muốn trở thành đối tác tài xế phải truy cập vào trang đăng ký của ứng dụng và chọn chức năng "Trở thành đối tác tài xế" và điền đầy đủ thông tin vào đơn đăng ký và chọn gửi đơn. Hệ thống sẽ lưu trữ đơn đăng ký và gửi cho quản trị viên. Quản trị viên sẽ duyệt hoặc từ chối đơn đăng ký, khi duyệt hoặc từ chối đơn thì hệ thống sẽ gửi email cho người đăng ký.

Nhóm chúng tôi đã dựa trên quy trình vận chuyển hàng hóa này kết hợp với các công nghệ ReactJS, React Native, MongoDB Altas, Firebase, Map4D, … để xây dựng hệ thống giúp giải quyết các khó khăn trong quá trình vận chuyển hàng hóa. Hệ thống sẽ tập trung vào các tính năng như: tạo đơn hàng với đường đi ngắn nhất giúp tiết kiệm chi phí cho khách hàng, theo dõi quá trình vận chuyển hàng hóa của tài xế, tài xế nhận và vận chuyển đơn hàng phù hợp để tối ưu thời gian và tăng lợi nhuận cho tài xế. Các tài xế nhận vận chuyển hàng hóa trong hệ thống đều được xác minh danh tính đảm bảo an toàn và tin cậy cho khách hàng. Hệ thống hiện thực gồm 2 app Android và 2 trang web sau:

- Một app Android dành cho khách hàng cung cấp các tính năng như tạo đơn hàng, nhắn tin với tài xế,… đặc biệt là khách hàng có thể theo dõi được vị trí tài xế đang vận chuyển hàng hóa.
- Một app Android dành cho tài xế cung cấp các tính năng như nhận và vận chuyển hàng hóa với gợi ý đường đi giao hàng ngắn nhất giúp tài xế tiết kiệm thời gian và chi phí vận chuyển, tài xế còn có thể nhắn tin với khách hàng để đảm bảo việc giao nhận hàng hóa được diễn ra thuận lợi và nhanh chóng.
- Một website dành cho quản trị viên dùng để quản lý tài xế và khách hàng; xử lý các đơn của người dùng gửi lên như: đơn đăng ký của tài xế, đơn góp ý, khiếu nại của khách hàng; quản trị viên còn có thể điều chỉnh giá vận chuyển, phí dịch vụ, bán kính cho phép tài xế nhận đơn hàng,…
- Một website đăng ký của ứng dụng: cho phép những người dùng đăng ký trở thành đối tác tài xế của GoTruck. Ngoài ra, website còn có chứa các thông tin cơ bản cũng như chính sách của GoTruck.

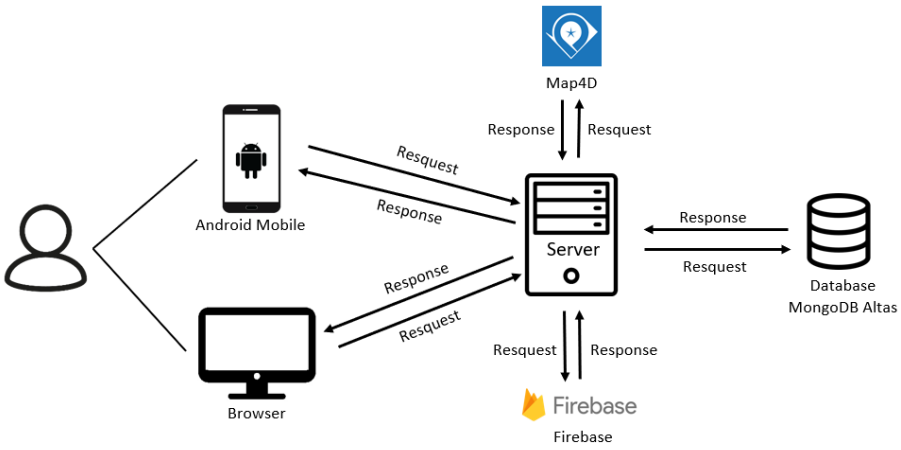
## 3.2. Ưu điểm của hệ thống

Về phía khách hàng, hệ thống có nhiều ưu điểm như gợi ý đường đi giao hàng ngắn nhất và nhanh nhất giúp cho hàng hóa đến tay khách hàng nhanh nhất với chi phí vận chuyển ít nhất. Khách hàng có thể theo dõi toàn bộ quá trình từ lúc tài xế nhận hàng đến lúc tài xế giao hàng đảm bảo luôn biết được tình trạng hàng hóa cùng với việc nhắn tin trao đổi với tài xế đảm bảo cho việc vận chuyển xảy ra rủi ro ít nhất, đảm bảo tính an toàn và bảo mật của hàng hóa. Ứng dụng còn hỗ trợ vận chuyển toàn quốc và hỗ trợ tất cả các loại hàng hóa cho các khách hàng có nhu cầu. Về phía tài xế, ứng dụng hỗ trợ tìm các đơn hàng gần nhất cho tài xế đảm bảo tài xế không cần di chuyển khoảng cách xa để nhận hàng. Tài xế còn có thể xem lịch sử đơn hàng đã vận chuyển, quản lý thông tin của phương tiện tham gia vào hệ thống và các chức năng khác.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

### 3.3. Kiến trúc hệ thống

Chúng tôi sử dụng kiến trúc Client – Server để xây dựng ứng dụng, bởi vì mô hình này hoạt động theo cách là khi máy client gửi yêu cầu đến server, lúc này server sẽ xử lý yêu cầu và trả kết quả về cho client. Client sẽ không chia sẻ bất kỳ tài nguyên nào của họ. Mô hình Client Server có một số ưu điểm chính để lựa chọn như tính tập trung, tính bảo mật, có khả năng mở rộng và khả năng truy cập.




Hình 3. Kiến trúc hệ thống GoTruck

Trong đó, chúng tôi sử dụng React Native để xây dựng hai ứng dụng cho thiết bị di động Android, phục vụ các chức năng của khách hàng như tạo đơn hàng, quản lý đơn hàng, trao đổi với tài xế vận chuyển,… và cũng như các chức năng của tài xế như vận chuyển đơn hàng, trao đổi với khách hàng, cập nhật thông tin phương tiện vận chuyển,… ReactJs được sử dụng để xây dựng hệ thống website gồm một website phục vụ quản trị viên để thuận tiện quản lý hệ thống và một website đăng ký của ứng dụng. Server dùng công nghệ NodeJS kết hợp với framework Expressjs để tạo ra các API, xử lý các request trả về response để phục vụ cho các ứng dụng website và mobile. MongoDB được sử dụng để lưu trữ toàn bộ dữ liệu của hệ thống như đơn hàng, thông tin tài xế và khách hàng, tin nhắn giữa tài xế và khách hàng.… Firebase dùng để gửi OTP xác minh và lưu trữ hình ảnh. Map4D cung cấp các API dùng để tìm kiếm địa chỉ, tìm đường đi ngắn nhất, ước tính thời gian và khoảng cách vận chuyển.

### 3.4 Phân tích hệ thống GoTruck

Đối tượng người dùng cho hệ thống gồm có khách hàng, tài xế, quản trị viên của GoTruck. Các chức năng của từng đối tượng được mô tả trong Bảng 1.

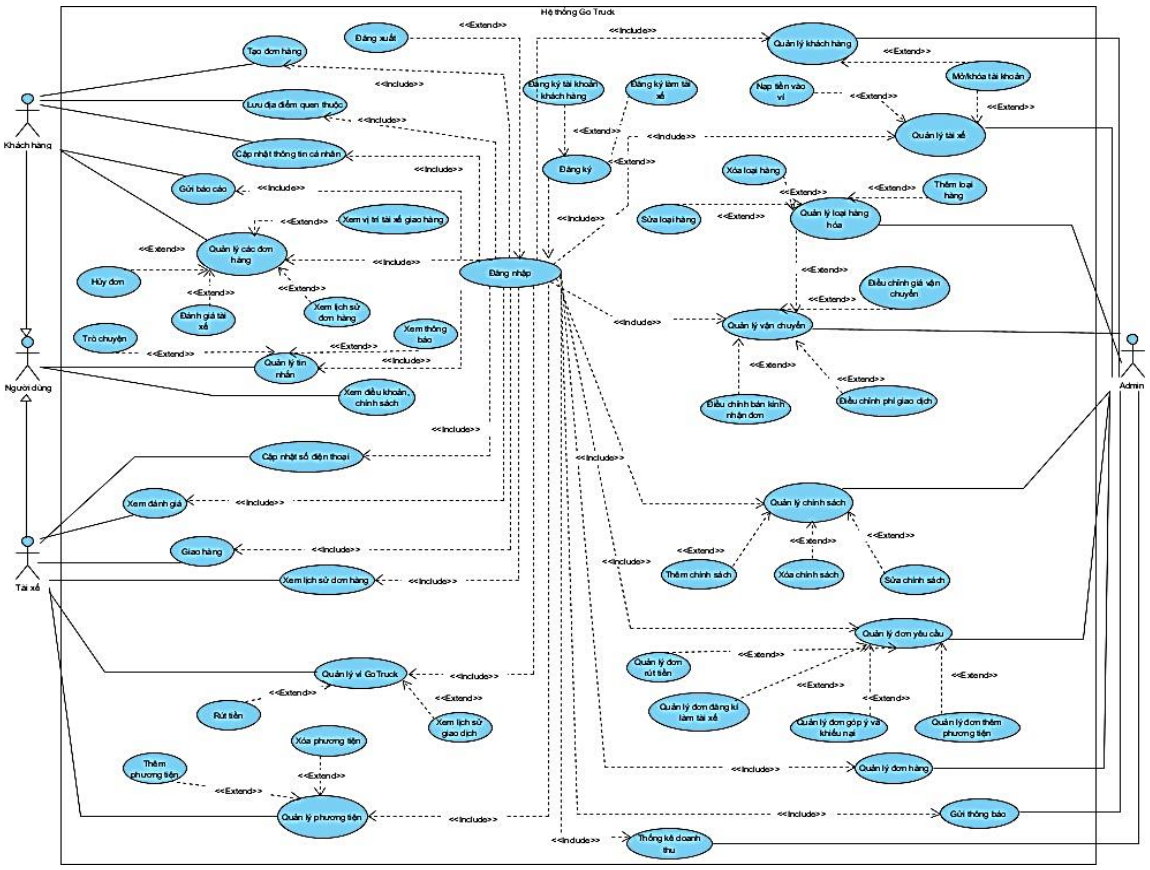
Bảng 1. Danh sách các chức năng của hệ thống GoTruck

| Actor | Tên chức năng | Mô tả |
|---|---|---|
| Khách hàng | Đăng ký tài khoản | Đăng ký tài khoản làm khách hàng |
| | Tạo đơn hàng | Tạo đơn hàng |
| | Lưu địa điểm quen thuộc | Lưu các địa điểm quen thuộc để tạo đơn hàng thuận tiện hơn |
| | Gửi góp ý, khiếu nại | Gửi góp ý, khiếu nại cho quản trị viên |
| | Đánh giá tài xế | Đánh giá tài xế vận chuyển hàng hóa |
| | Nhắn tin với tài xế | Nhắn tin với tài xế đang vận chuyển hàng hóa |
| | Xem lịch sử đơn hàng | Xem lịch sử đơn hàng |
| | Xem vị trí tài xế giao hàng | Xem vị trí tài xế đang giao hàng |
| | Cập nhật thông tin cá nhân | Cập nhật số điện thoại, tên, ảnh đại diện trong hệ thống |


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| | | |
|---|---|---|
| Tài xế | Đăng ký tài khoản | Đăng ký làm tài xế |
| | Vận chuyển hàng hóa | Nhận đơn hàng và vận chuyển đến nơi giao hàng |
| | Xem lịch sử đơn hàng | Xem lịch sử đơn hàng đã vận chuyển |
| | Nạp/rút tiền | Nạp/rút tiền từ ví GoTruck |
| | Xem lịch sử giao dịch | Xem lịch sử giao dịch của ví GoTruck |
| | Quản lý phương tiện | Thêm, xóa phương tiện vận chuyển |
| | Cập nhật thông tin cá nhân | Cập nhật số điện thoại |
| Quản trị viên | Xem thông tin đơn hàng | Xem thông tin các đơn hàng đã vận chuyển |
| | Xem thông tin khách hàng và tài xế | Xem thông tin khách hàng và tài xế |
| | Khóa tài khoản | Khóa tài khoản của khách hàng hoặc tài xế |
| | Điều chỉnh giá vận chuyển | Điều chỉnh giá vận chuyển |
| | Điều chỉnh khoảng cách tài xế nhận đơn hàng | Điều chỉnh khoảng cách tài xế nhận đơn hàng |
| | Quản lý chính sách của GoTruck | Thêm, xóa sửa chính sách của GoTruck |
| | Thống kê doanh thu | Thống kê doanh thu |
| | Gửi thông báo | Gửi thông báo cho khách hàng và tài xế |
| | Xử lý đơn đăng ký làm tài xế | Xử lý đơn đăng ký làm tài xế |
| | Xử lý đơn góp ý, khiếu nại | Xử lý đơn góp ý, khiếu nại của khách hàng |
| | Xử lý đơn thêm phương tiện | Xử lý đơn thêm phương tiện của tài xế |

Sau đây là đặc tả các use case của hệ thống giúp người dùng hiểu rõ hơn về các chức năng của hệ thống bằng các mô tả sự tương tác giữa người dùng với hệ thống

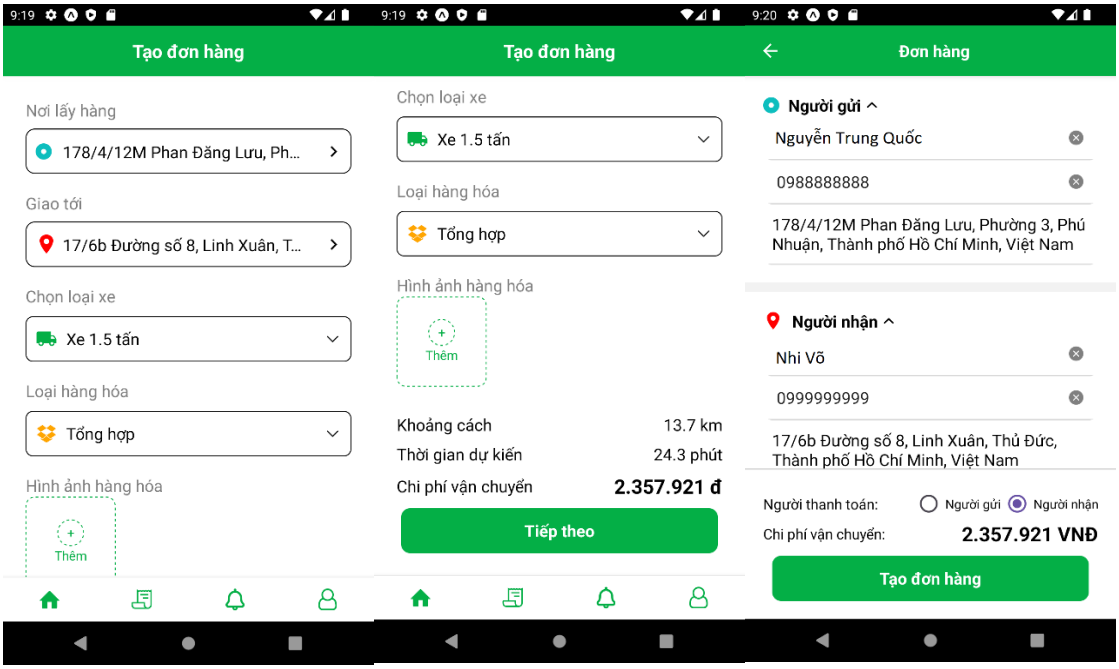



Hình 4. Sơ đồ use case của hệ thống [9]

## 3.5 Giao diện đồ họa

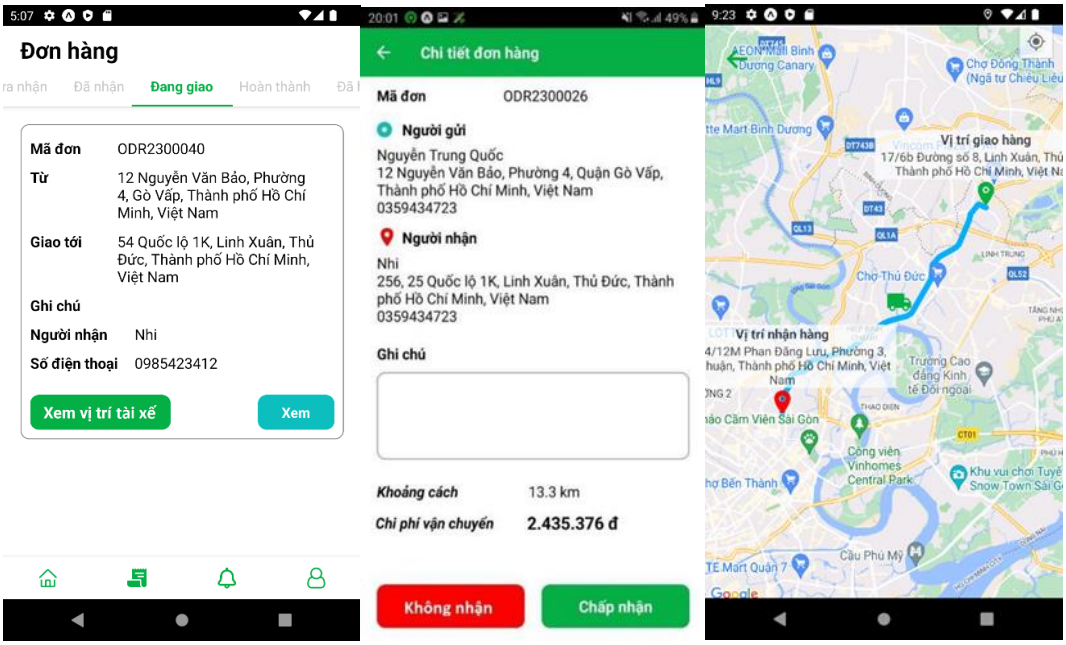
### 3.5.1 Ứng dụng Android dành cho khách hàng

Ứng dụng hỗ trợ khách hàng tạo đơn hàng mới trên ứng dụng của mình, sau khi nhập đầy đủ thì ứng dụng sẽ tính ra khoảng cách giữa nơi nhận hàng và nơi giao hàng, cùng với số tiền vận chuyển và thời gian giao hàng dự kiến, khách hàng xác nhận tạo đơn và chờ cho đến khi có tài xế nhận đơn.




Hình 5. Các màn hình tạo đơn hàng của khách hàng

Sau khi có tài xế nhận hàng và bắt đầu vận chuyển đến nơi giao hàng thì khách hàng có thể theo dõi được quá trình vận chuyển của tài xế cho đến khi đơn hàng hoàn thành.



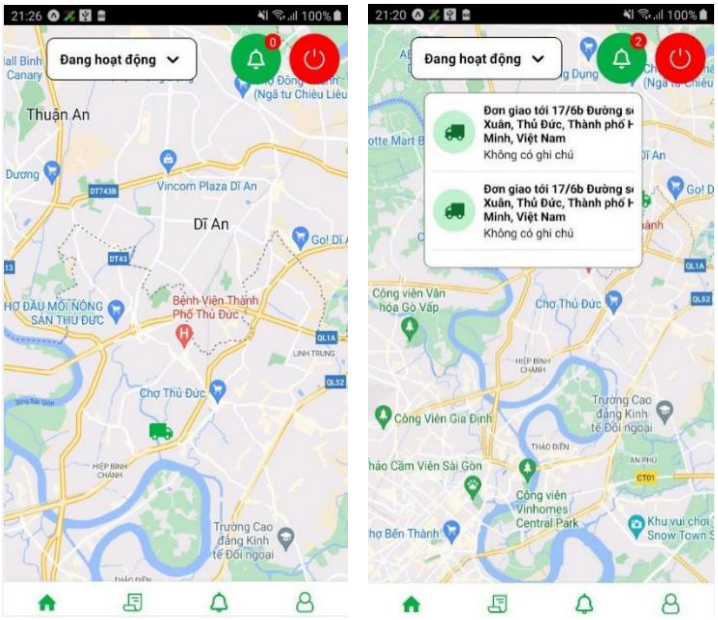

Hình 6. Màn hình xem lịch sử đơn hàng và theo dõi quá trình vận chuyển của tài xế


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Ngoài ra, các chức năng nhắn tin với tài xế, cập nhật thông tin cá nhân, thêm địa điểm quen thuộc, gửi đơn góp ý và khiếu nại và nhiều chức năng khác nữa cũng đã được hiện thực trong ứng dụng.

### 3.5.2 Ứng dụng Android dành cho tài xế

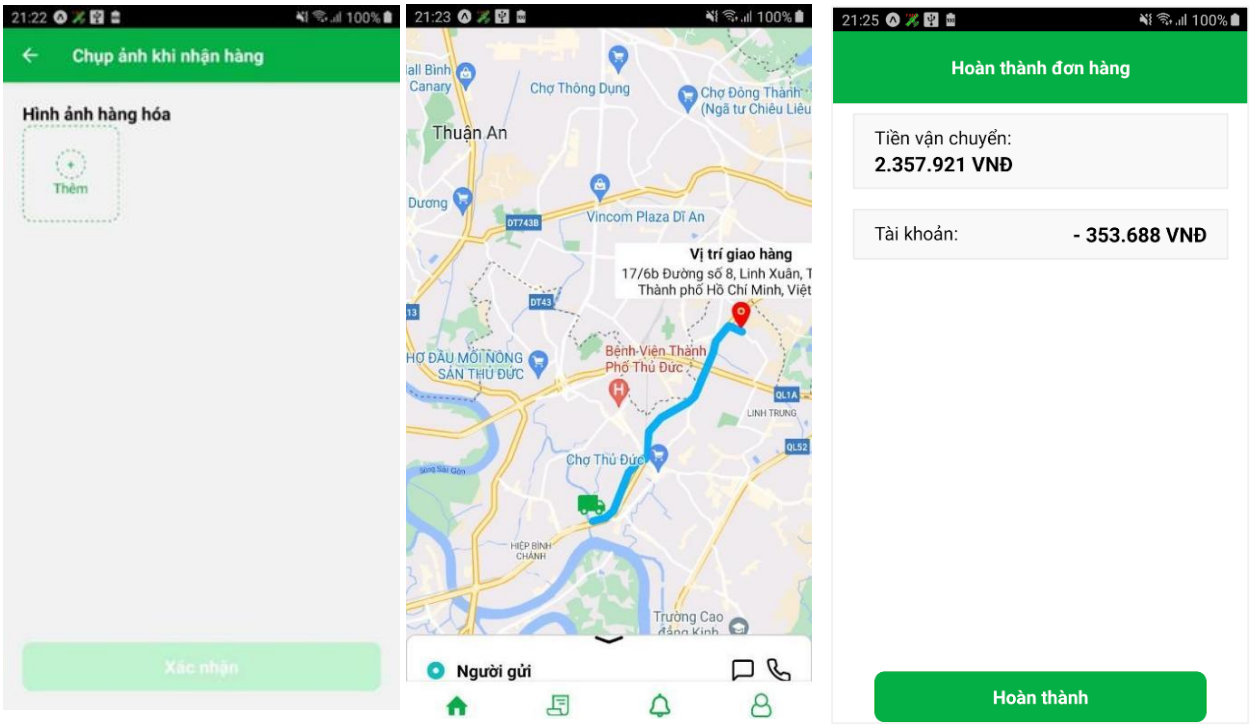
Để giúp tài xế sử dụng những thời gian rảnh rỗi của mình để kiếm thêm thu nhập thì tài xế có thể vận chuyển hàng hóa trên ứng dụng GoTruck. Tài xế khi ở gần nơi nhận hàng của một đơn hàng bất kỳ trong bán kính nhất định thì sẽ nhận được thông báo có đơn hàng, lúc này tài xế có thể chọn đơn hàng để xem thông tin chi tiết và quyết định nhận đơn hoặc không nhận.




Hình 7. Màn hình chính của tài xế và thông báo khi có đơn hàng

Sau khi nhận đơn hàng thì tài xế sẽ vận chuyển đến nơi nhận hàng để lấy hàng và chụp ảnh hàng hóa lúc nhận, sau đó tài xế vận chuyển đơn hàng đến nơi giao hàng. Sau khi hoàn thành đơn hàng thì tài xế sẽ nhận được số tiền vận chuyển và bị trừ một phần phí giao dịch.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

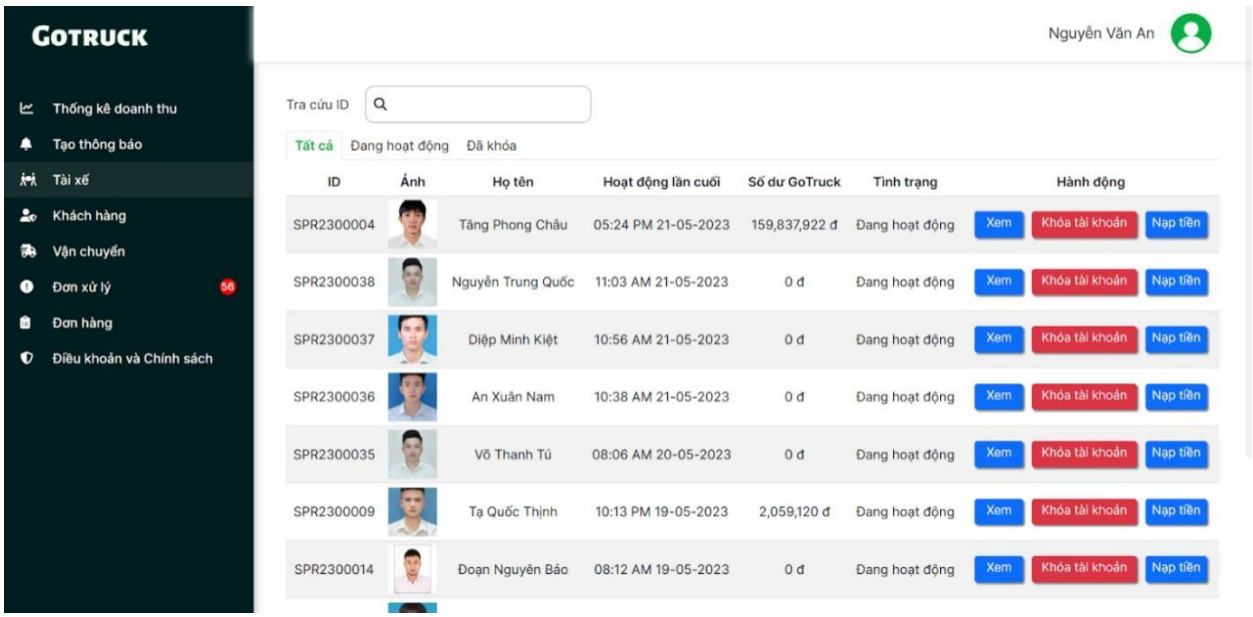



Hình 8. Màn hình chụp ảnh khi nhận hàng và bắt đầu quá trình giao hàng của tài xế

Ứng dụng cũng đã hiện thực được những chức năng khác như: cập nhật thông tin cá nhân, thêm hoặc cập nhật phương tiện vận chuyển, xem lịch sử giao hàng, xem đánh giá của khách hàng, …

### 3.5.3 Website dành cho quản trị viên

Để thuận tiện cho việc quản lý người dùng cũng như quản lý đơn hàng, website của quản trị viên có đầy đủ thông tin của người dùng và đơn hàng, bên cạnh đó còn có các tính năng khác như thống kê doanh thu, tạo thông báo, điều chỉnh giá vận chuyển, bán kính nhận đơn của tài xế,…

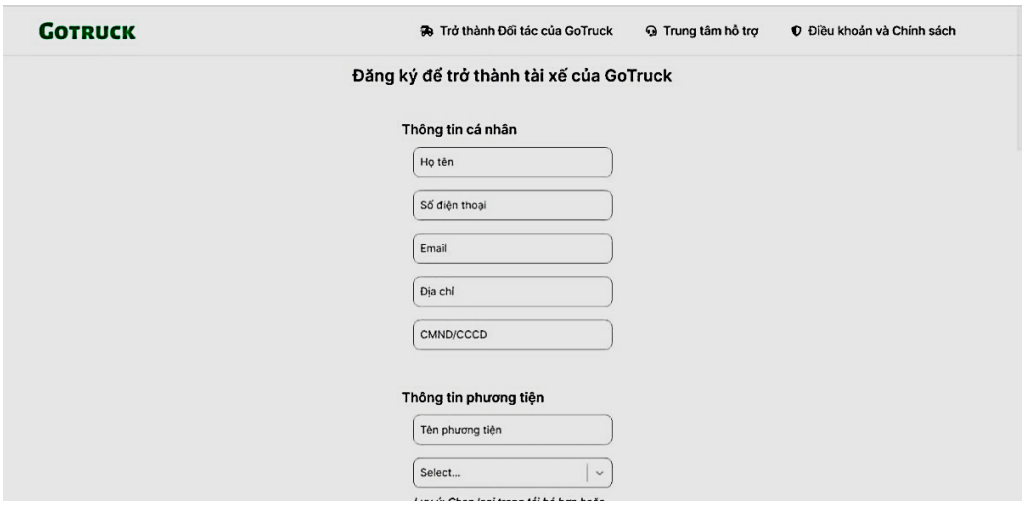



Hình 9. Trang quản lý tài xế của ứng dụng GoTruck

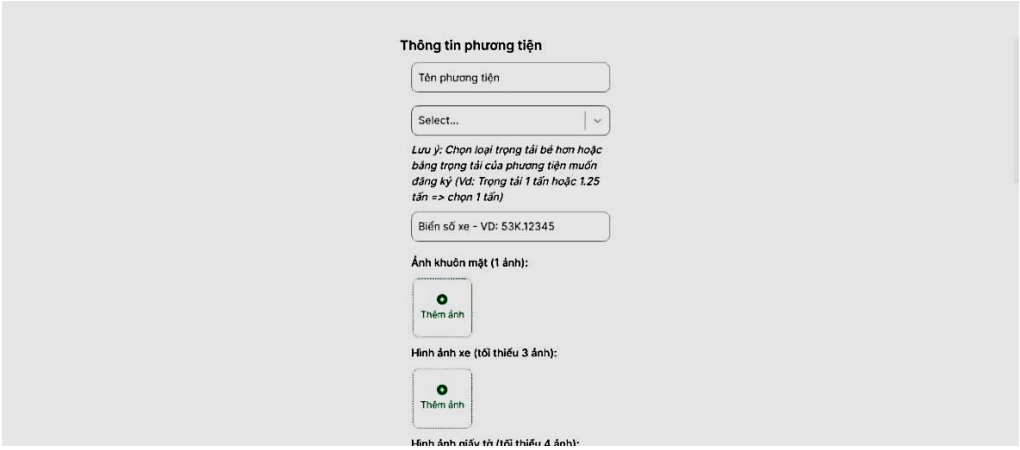

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.5.4 Website đăng ký của ứng dụng

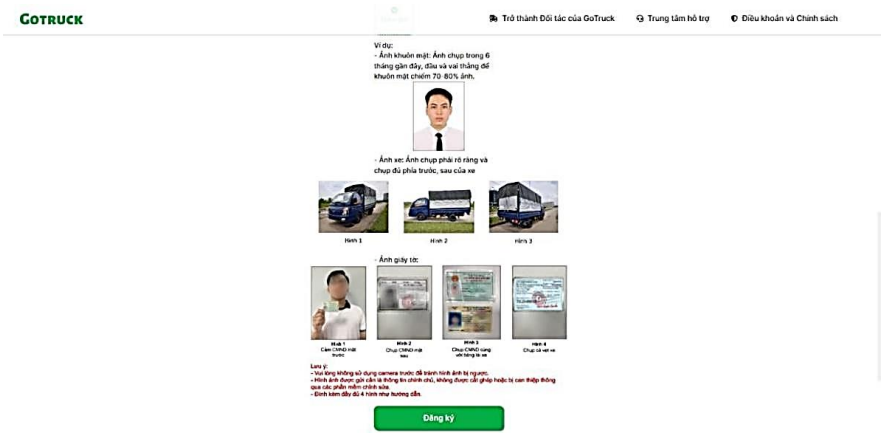
Ở trang chủ, mọi người có thêm tham khảo về GoTruck và nếu ai muốn trở thành đối tác tài xế của GoTruck cũng có thể đăng ký tại đây, chỉ cần chọn vào chức năng trở thành đối tác tài xế của GoTruck, hệ thống sẽ hiển thị đơn đăng ký và người dùng chỉ cần nhập thông tin cá nhân kèm theo những hình ảnh minh chứng về thông tin cá nhân, thông tin phương tiện đăng ký và gửi đơn, quản trị viên sẽ nhận được đơn đăng ký và duyệt hoặc từ chối nếu đơn không đủ tính xác thực.




Hình 10. Nơi điền thông tin cá nhân của đơn đăng ký




Hình 11. Nơi điền thông tin phương tiện của đơn đăng ký




Hình 12. Các ví dụ về hình ảnh minh chứng của đơn đăng ký

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

## 4 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Chúng tôi đã triển khai thành công dự án quản lý vận chuyển hàng hóa bằng xe tải, mang lại nhiều lợi ích đáng kể cho quy trình vận chuyển. Hệ thống đã tăng tính hiệu quả và giảm thiểu thời gian và chi phí liên quan. Chúng tôi đã đạt được những kết quả đáng mừng trong việc đáp ứng các mục tiêu đã đề ra và hệ thống đang hoạt động đúng theo kỳ vọng ban đầu.

Hệ thống đã không gặp tình trạng tắc nghẽn hay quá tải, cho thấy khả năng xử lý lớn và sự linh hoạt trong việc đáp ứng yêu cầu. Các yêu cầu từ khách hàng được xử lý gần như ngay lập tức, tạo ra sự hài lòng và tin tưởng trong quá trình vận chuyển hàng hóa.

Các chức năng của hệ thống đáp ứng tốt các yêu cầu đã đề ra. Khách hàng có thể dễ dàng tạo đơn hàng với chi phí hợp lý và thời gian cụ thể. Họ cũng có thể sử dụng ứng dụng để theo dõi và xem trạng thái vận chuyển hàng hóa của mình. Sự tương tác và giao tiếp giữa tài xế và khách hàng được đơn giản hóa và thuận tiện hơn bao giờ hết. Điều này giúp cải thiện đáng kể sự chính xác và tương tác trong quá trình vận chuyển hàng hóa.

Tài xế nhận được thông tin chi tiết về mỗi đơn hàng, địa điểm và lộ trình, giúp họ tiếp cận các đơn hàng phù hợp và điều hướng một cách thuận tiện. Điều này giúp giảm thiểu thời gian rảnh rỗi và tăng thu nhập cho tài xế, tạo ra một môi trường làm việc tốt hơn cho họ.

Chúng tôi sẽ không ngừng phát triển hệ thống và đáp ứng nhu cầu ngày càng cao của người dùng. Trong tương lai, GoTruck mong muốn cung cấp nhiều tính năng hơn nữa để nâng cao trải nghiệm của người dùng. Điều này bao gồm tích hợp các phương thức thanh toán tiện lợi như qua ngân hàng hoặc ví điện tử như Momo, nâng cấp giao diện và trải nghiệm người dùng, cùng với các chương trình khuyến mãi hấp dẫn. Ngoài ra, chúng tôi sẽ tối ưu mã nguồn và tích hợp các tính năng bảo mật để tăng tốc độ và cải thiện khả năng bảo mật cho người dùng.

Chúng tôi sẽ tiếp tục đồng hành cùng ngành công nghiệp vận tải và mang lại những giải pháp đột phá, từ đó thúc đẩy sự phát triển và tối ưu hóa quá trình vận chuyển hàng hóa bằng xe tải.

## TÀI LIỆU THAM KHẢO

[1]. Ongo, G., Kusuma, G.P. Hybrid database system of mysql and mongodb in web application development. In Proceeding of the 2018 International Conference on Information Management and Technology (ICIMTech), IEEE (2018) 256–260

[2]. Techvify. (2023). What Is Node.js?. https://techvify-software.com/outsource- node-js-development-services/. Truy cập 19/06/2023

[3]. Ronald Caldwell. (2023). What is MongoDB?. https://www.liquidweb.com/kb/what-is-mongodb/. Truy cập 19/06/2023

[4]. Google team. (2023). What is Firebase?. https://firebase.google.com. Truy cập 19/06/2023

[5]. Rawat, P., Mahajan, A.N.: Reactjs: A modern web development framework. International Journal of Innovative Science and Research Technology 5(11) (2020)

[6].ThinkInfo. (2023). React JS Services. https://thinkinfoservices.com/react- development/. Truy cập 19/06/2023

[7]. "Tìm hiểu về React Native". https://wiki.tino.org/react-native-la-gi/. Truy cập 19/06/2023

[8]."Giới thiệu về Map4D". https://iotlink.com.vn. Truy cập 19/06/2023

[9].Doug Rosenberg, Matt Stephens. (2007). Use Case Driven Object Modeling withUML. Springer-Verlag New York, Inc (Tham khảo vẽ UML Diagram)

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F030**

# PHÂN LOẠI BI-RADS SỬ DỤNG PHƯƠNG PHÁP HỌC SÂU

TRỊNH THỊ BẢO BẢO

*Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
** ttbb08022001@gmail.com*

**Tóm tắt.** Trong lĩnh vực y tế, BI-RADS được xem là một chỉ số sinh học quan trọng trong việc đánh giá rủi ro ung thư vú. Tuy nhiên, việc đánh giá phân loại BI-RADS đòi hỏi sự can thiệp từ các chuyên gia xét nghiệm, đội ngũ y bác sĩ để đánh giá sự khác biệt. Để hỗ trợ phần nào vấn đề này, các thuật toán học sâu đã được phát triển với mục đích phân loại BI-RADS. Trong nghiên cứu này, chúng tôi đề xuất phương pháp tiền xử lý ảnh nhũ bằng sử dụng kĩ thuật windowing sau đó sử dụng thuật toán cutmix nhằm giảm tình trạng mất cân bằng dữ liệu… Kế đến chúng tôi cắt vùng nhũ bằng mô hình YoloX rồi đưa tập dữ liệu vào các mô hình học sâu để phân loại nhũ. Các thí nghiệm của chúng tôi trên tập dữ liệu VinDr-mammo và CDD-CESM cho thấy rằng phương pháp đề xuất cải thiện đáng kể hiệu suất phân loại BI-RADS với độ chính xác ở tập kiểm thử là 89.85% và F1-score là 85.16%.

**Từ khóa.** BI-RADS, tăng cường ảnh, học sâu, mô hình ResNet, mô hình HRNet.

## BI-RADS CLASSIFICATION USING DEEP LEARNING

**Abstract.** In the field of healthcare, BI-RADS is considered an important biological indicator for assessing the risk of breast cancer. However, evaluating BI-RADS requires intervention from expert examiners and medical teams to assess the differences. To partially address this issue, deep learning algorithms have been developed to classify BI-RADS. In this study, we propose a preprocessing method for mammogram images using windowing, followed by the utilization of the cutmix algorithm to mitigate data imbalance issues. Subsequently, we segment the breast regions using the YoloX model and feed the dataset into deep learning models for BI-RADS classification. Our experiments on datasets, VinDr-mammo and CDD-CESM, demonstrate that the proposed method significantly improves the performance of BI-RADS classification, achieving an accuracy of 89.85% and F1 score of 85.16% on the test set.

**Keywords.** BI-RADS, image augmentation, deep learning, ResNet model, HRNet model.

## 1 GIỚI THIỆU

Tình trạng ung thư vú là một trong những vấn đề y tế nghiêm trọng trên toàn thế giới. Theo Tổ chức Y tế Thế giới, ung thư vú là loại ung thư phổ biến nhất ở phụ nữ, với khoảng 2,3 triệu trường hợp được chẩn đoán mới mỗi năm và gần 685.000 trường hợp tử vong mỗi năm. Tuy nhiên, tỷ lệ sống sót sau 5 năm của bệnh nhân ung thư vú đã vượt quá 80% ở các quốc gia có thu nhập cao, trong khi tỷ lệ này ở các nước kém phát triển chỉ là 40% đến 60%. Tại Việt Nam, ung thư vú cũng đang trở thành một vấn đề y tế đáng lo ngại. Theo báo cáo của Bộ Y tế năm 2020, ung thư vú đã trở thành loại ung thư phổ biến thứ hai ở Việt Nam, chỉ sau ung thư phổi. Mỗi năm có khoảng 10.000 trường hợp ung thư vú được chẩn đoán mới và khoảng 3.000 trường hợp tử vong do ung thư vú.

Điều này đặt ra nhu cầu cấp thiết trong việc nghiên cứu và phát triển các phương pháp phân loại hỗ trợ chẩn đoán tình trạng ung thư vú, từ đó hỗ trợ cho các chuyên gia y tế. Việc phân loại BI-RADS đóng vai trò quan trọng trong việc tăng cơ hội sống sót và cải thiện chất lượng cuộc sống của bệnh nhân, đồng thời giảm thiểu tối đa tác động xấu của bệnh lên sức khỏe và cuộc sống của người bệnh và gia đình, những thông tin chính xác và hiệu quả để đưa ra quyết định điều trị phù hợp nhất. Và với sự phát triển ngày càng mạnh mẽ của các mô hình học máy, chúng ta cũng có thể áp các mô hình đó trong quá trình huấn luyện nhằm mang lại một mô hình phân loại thật sự có hiệu quả cho việc phân loại tình trạng ung thư vú ở phụ nữ cũng như có sự hỗ trợ cho đội ngũ y bác sĩ phát hiện tình trạng bệnh một cách chính xác nhất. Tuy nhiên, hiện nay phần

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

lớn các tập dữ liệu đều đánh dấu ảnh nhũ ở mức độ ung thư: bình thường hoặc bất thường (lành tính, ác tính). Vì thế sẽ không đủ chi tiết để chẩn đoán bệnh cũng như sẽ rất ít nguồn dữ liệu mô tả tình trạng bệnh bằng cách xác định BI-RADS của nhũ ảnh. Nhiều nghiên cứu cũng đã chỉ ra rằng, việc phân loại BI-RADS đang là một chủ đề được quan tâm trong lĩnh vực y tế và cũng là yếu tố quan trọng trong việc đánh giá nguy cơ ung thư vú. Vì vậy, ở nghiên cứu này tôi đề xuất áp dụng quá trình tiền xử lý ảnh nhằm loại bỏ phần nhiễu của nhũ ảnh, nhận dạng và cắt phần nhũ sau đó áp dụng phương pháp tăng cường ảnh như thuật toán cutmix để làm giảm sự ảnh hưởng của việc mất cân bằng dữ liệu lên mô hình. Kế đến, tập dữ liệu sau khi xử lý được đưa vào mô hình học sâu để học và phân loại 5 loại BI-RADS. Đánh giá độ hiệu quả từ các lần thực nghiệm, ta có thể thấy đề xuất đạt hiệu quả đáng kể, mang lại hiệu suất ấn tượng so với các nghiên cứu đã có trước đây.

## 2 CÁC NGHIÊN CỨU LIÊN QUAN

Bài báo "A deep learning framework to classify breast density with noisy labels regularization" của Hector Lopez-Almazan cùng cộng sự (2022) [1] trình bày về phương pháp RegL để đánh giá mật độ vú từ chụp quang tuyến vú kỹ thuật số và phân loại vú thành các loại BI-RADS. Phương pháp kết hợp các quá trình tiền xử lý hình ảnh và mạng CNN để đạt được kết quả tốt nhất trong việc phân loại chụp quang tuyến vú. Kết quả thử nghiệm trên tập dữ liệu bao gồm 892 hình ảnh làm tập đào tạo, 224 hình ảnh cho tập đánh giá và 279 hình ảnh cho tập thử nghiệm cho thấy phương pháp này có hiệu suất tương tự như các bác sĩ X quang có kinh nghiệm và ước tính tốt hơn các nhãn sự thật cơ bản chưa biết. Mô hình hợp được đánh giá với độ chính xác là 0,85 và chỉ số kappa là 0,71. Điều này chỉ ra rằng phương pháp đề xuất có hiệu suất tương tự như các bác sĩ X quang có kinh nghiệm trong việc phân loại chụp quang tuyến vú thành các loại BI-RADS.

Zhen Yu Lee và cộng sự đã trình bày các mô hình dự đoán khác nhau bao gồm Decision Tree, Support Vector Model - SVM và k-Nearest Neighbour, hiệu quả của các mô hình dự đoán khác nhau trong việc phân loại hình ảnh chụp quang tuyến vú đã được kiểm tra bằng đường cong ROC (Receiver Operator Characteristics) và so sánh với phân loại BI-RADS là tiêu chuẩn vàng trong bài báo "Classification of mammographic breast density and its correlation with BI-RADS in elde¬r women using machine learning approach" – 2022 [2]. Kết quả cho thấy phương pháp học máy có tiềm năng để phân loại mật độ vú trong các hình ảnh chụp quang tuyến vú, mang lại tiện ích trong lâm sàng.

Mô hình phân loại mật độ nhũ ảnh dựa trên đa góc nhìn (multi-view) được các tác giả Huyen T.X.Nguyen cùng với cộng sự giới thiệu trong bài báo "A novel multi-view deep learning approach for BI-RADS and density assessment of mammograms" -2022 [3]. Mô hình thực hiện trên 2 bộ dữ liệu là bộ dữ liệu DDSM và bộ dữ liệu riêng tư của nhóm tác giả. Kết quả thử nghiệm cho thấy mô hình phát hiện hiệu quả vượt trội so với phân loại dựa trên một chế độ xem với điểm macro-F1 là 0.5063 ở mô hình ResNet-34 và 0.5759 ở mô hình EfficientNet-B2.

Cũng trong năm 2022, nhóm tác giả Huyen T.X.Nguyen đề cập đến vấn đề tăng cường hình ảnh để cải thiện hiệu suất của các mô hình học sâu về nhiệm vụ phân loại chụp nhũ ảnh ở bài báo "A Novel Transparency Strategy-based Data Augmentation Approach for BI-RADS Classification of Mammograms" [4]. Từ đó, một phương pháp tăng cường mới là làm rõ vùng tổn thương để tăng cường hình ảnh. Phương pháp đề xuất sử dụng khu vực trung tâm - ROI để tạo ra các ví dụ đào tạo rủi ro cao cho BI-RADS 3,4,5. Cuối cùng, phương pháp đề xuất cải thiện đáng kể việc phân loại mật độ nhũ ảnh.

Trong bài báo "A High-Performance Deep Neural Network Model for BI-RADS Classification of Screening Mammography", nhóm tác giả Kuen-Jang [5] giới thiệu mô hình dựa trên mạng lưới thần kinh sâu (DNN) để phân loại BIRADS. Mô hình được đào tạo bằng cách sử dụng các hình ảnh dựa trên khối được phân đoạn từ mô dữ liệu chụp quang tuyến vú riêng tư của tác giả. Hiệu suất của mô hình được thể hiện bằng độ chính xác tổng thể là 94.22%, độ nhạy là 95.31%, độ đặc hiệu trung bình 99.15%. Tuy nhiên, phân loại BI-RADS 1,2,3 vẫn còn chưa tốt đặc biệt là BI-RADS 1 vì các hình ảnh khối không tổn thương được phân loại là BI-RADS 1 dẫn đến các đặc điểm không đặc biệt sẽ khó chuẩn đoán.

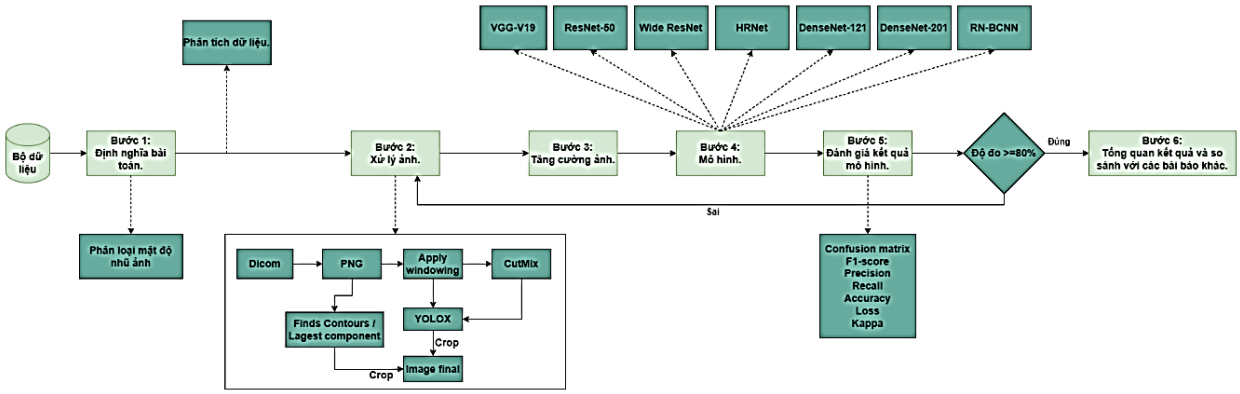
Năm 2021, tác giả S.Siddeq cùng cộng sự của mình đã đề xuất hướng tiếp cận mới trong bài báo "Deep Learning RN-BCNN Model for Breast Cancer BI-RADS Classification" [6]. Hướng tiếp cận này sử dụng mô hình RN-BCNN và tăng cường phân loại ảnh bằng cách sử dụng phương pháp biến dạng đàn hồi. Hiệu suất của mô hình với điểm F1 trung bình 0.859 và độ chính xác trên tập dữ liệu INBreast kiểm thử đạt 85.9%.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Qua các nghiên cứu đã nêu trên cho chúng ta cái nhìn tổng quan về các phương pháp tiếp cận của tác giả dùng để phân loại mật độ nhũ ảnh. Tuy nhiên, phần lớn các bài báo đều giới thiệu các mô hình huấn luyện không quá đi sâu vào việc tiền xử lý ảnh nên hiệu suất bài toán còn hạn chế. Nghiên cứu này đề xuất một phương pháp tiếp cận mới dựa trên việc tiền xử lý nhũ ảnh, phát hiện vùng nhũ sau đó cắt vùng nhũ bằng mô hình YoLoX. Cuối cùng, đưa bộ dữ liệu sau khi xử lý vào mô hình phân loại để đưa ra kết quả mong muốn. Các thực nghiệm và đánh giá được trình bày chi tiết ở các phần kế.

## 3 PHƯƠNG PHÁP TIẾP CẬN

Nghiên cứu được chia làm sáu bước với ba giai đoạn chính là: xử lý ảnh và tăng cường ảnh (bước 2 và bước 3), phân tích vùng nhũ (bước 2) sau đó là đưa vào mô hình phân loại (bước 4). Trong giai đoạn tiền xử lý ảnh, ảnh đầu vào được chuyển từ ảnh DICOM sang ảnh PNG. Đầu tiên, tôi đọc ảnh và chuyển đổi ảnh thành dạng pixel. Tiếp đến xử dụng windowing để chọn dải pixel đặc biệt từ ảnh gốc trước khi chuẩn hóa. Điều này, có tác dụng làm tăng độ tương phản cao hơn giữa các mô mềm và mô đặc biệt. Kế đến, chúng tôi áp dụng ba loại thuật toán để tìm ra vùng nhũ và cắt vùng nhũ. Tuy nhiên, với các trường hợp đặc biệt, ảnh nhiều nhiễu thì cách cắt ảnh bằng tìm biên hoặc tìm bằng liên thông lớn nhất đều không đạt hiệu quả như mong đợi nên tôi tiếp cận đến phương pháp thứ ba là sử dụng mô hình YOLOX và các kết quả sẽ được trình bày chi tiết ở phần kế. Đối với giai đoạn phân loại 5 lớp BI-RADS 1 đến BI-RADS 5, dữ liệu sau khi được tiền xử lý và sử dụng thuật toán tăng cường ảnh để làm giảm ảnh hưởng của việc mất cân bằng dữ liệu sẽ được huấn luyện trên các mô hình học sâu khác nhau như: VGG-19, ResNet50, Wide ResNet, HRNet, DenseNet và đánh giá chúng trên các độ đo để chọn ra mô hình phân loại đạt hiệu suất tốt nhất.



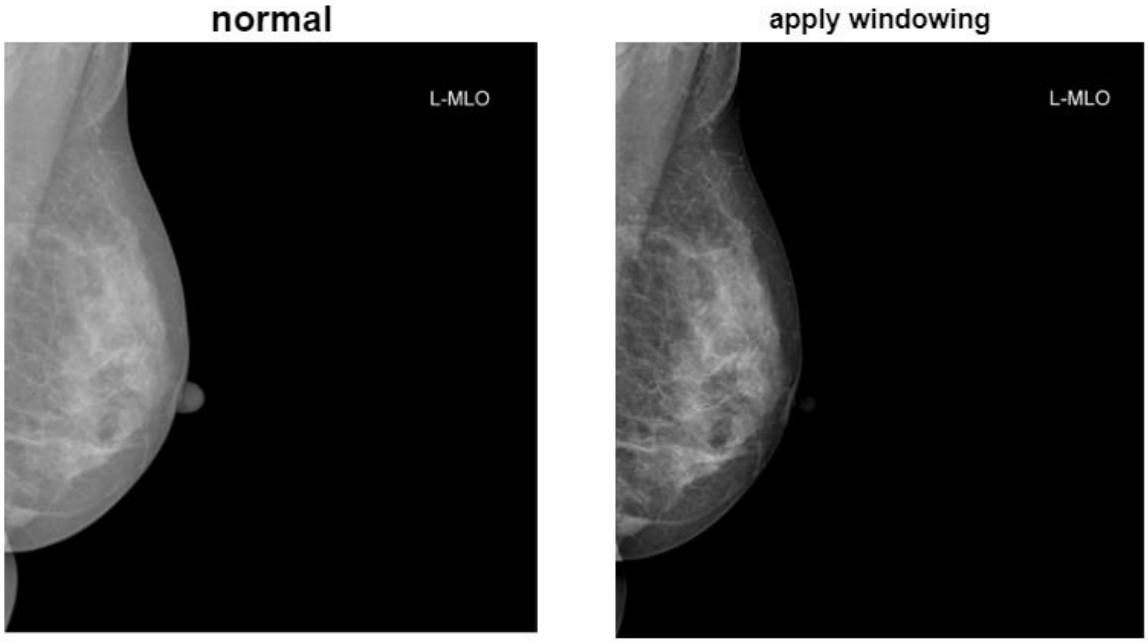

Hình 1. Phương pháp tiếp cận

### 3.1 Tiền xử lý

#### 3.1.1 Chuyển ảnh DICOM sang PNG.

Ảnh Digital Imaging and Communications in Medicine - Dicom là một dạng ảnh tiêu chuẩn sử dụng trong lĩnh vực y tế để trao đổi và lưu trữ các hình ảnh CT, MRI, siêu âm và X-quang. Bên cạnh đó, ảnh này còn chưa các thông tin chi tiết của bệnh nhân nên dung lượng tệp rất lớn. Chính vì vậy để giảm dung lượng và dễ dàng xử lý tác giả đã chuyển sang định dạng png. Đầu tiên, nhóm đọc ảnh và chuyển đổi thành dạng mảng các pixel. Ảnh dicom thường chứa các giá trị lớn hơn 8 bit vì thế việc chuẩn hóa các pixel dẫn đến việc mất thông tin ảnh. Windowing cho phép chúng ta chọn các pixel đặc biệt từ ảnh gốc trước khi chuẩn hóa. Làm tăng độ tương bản cao hơn giữa các mô mềm và mô đặc biệt. Kỹ thuật này được áp dụng cho việc xuất hình ảnh dưới dạng PNG/JPG.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

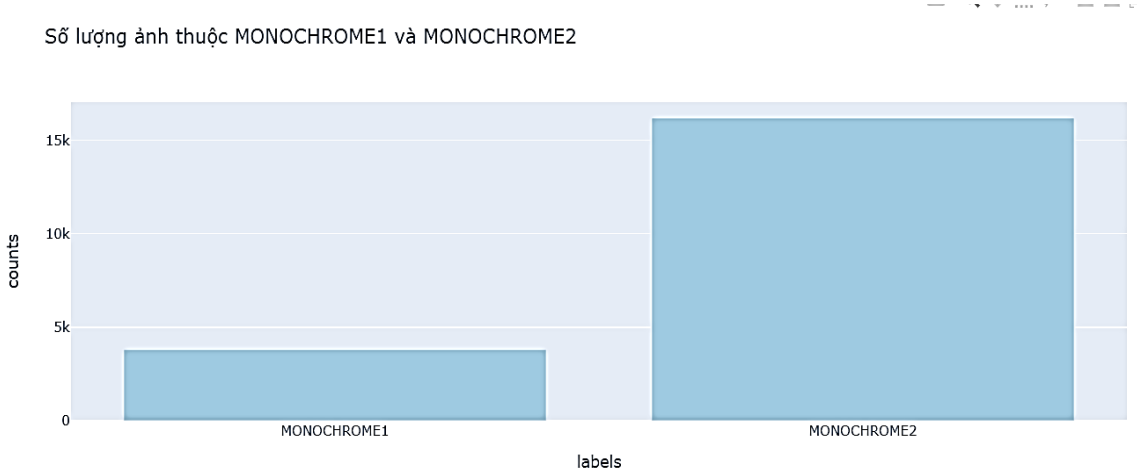



Hình 2. Minh họa ảnh bình thường và ảnh sử dụng windowing.

Kế đến, ảnh mới được tạo ra bằng công thức sau để chuẩn hóa lại mức xám.

$$Image_{new} = \frac{image_i - \min(image)}{\max(image) - \min(image)} \quad (1)$$

Bên cạnh đó, dữ liệu dicom có 2 dạng MONOCHROME-1 và MONOCHROME-2 nó dùng để xác định cách hiển thị các pixel dưới dạng đơn sắc. Hai thông số này được lưu vào PhotometricInterpretation. Với MONOCHROME-1 nghĩa là các pixel cao được hiển thị dưới dạng đen hơn, ngược lại với MONOCHROME-2 các pixel cao được hiển thị trắng hơn.




Hình 3. Số lượng ảnh thuộc MONOCHROME1 và MONOCHROME2.

Vì lượng ảnh thuộc MONOCHROME-2 nhiều hơn nên nhóm tác giả chọn nó làm dạng chính với các ảnh thuộc MONOCHROME-1 thì sẽ lấy $1 - pixel$ ảnh đã được nén chuyển đổi thành MONOCHROME-2.

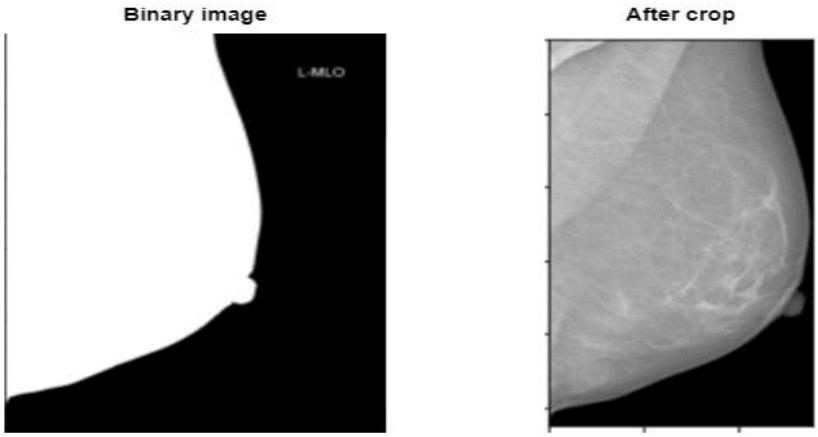

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.1.2 Cắt ảnh

#### 3.1.2.1 Thuật toán tìm biên

Thuật toán tìm biên là một trong những thuật toán xử lý ảnh để tìm ra các đường biên của các vật thể trong hình ảnh. Các đường biên này được tạo ra bằng cách kết hợp các điểm liền nhau trên cùng một đường biên của vật thể trong ảnh.

Đầu tiên, ảnh được chuyển sang dạng ảnh xám để giảm kích thước dữ liệu cần xử lý. Tiếp theo, một bộ lọc được áp dụng để loại bỏ nhiễu và làm mịn ảnh. Sau đó, ngưỡng hóa ảnh được thực hiện để chuyển ảnh xám thành ảnh nhị phân. Quá trình ngưỡng hóa này sẽ chuyển các điểm ảnh thành các giá trị 0 hoặc 255, tùy thuộc vào giá trị ngưỡng được thiết lập. Các điểm ảnh có giá trị lớn hơn ngưỡng sẽ được gán giá trị 255, các điểm ảnh còn lại sẽ được gán giá trị 0.




Hình 4. Cắt ảnh bằng thuật toán tìm viền.

#### 3.1.2.2 Thuật toán tìm liên thông lớn nhất

Thuật toán tìm liên thông lớn nhất là một trong những thuật toán xử lý ảnh để cắt ảnh. Đầu tiên, sử dụng cắt ngưỡng isodata để phân chia ảnh thành các vùng có độ sáng khác nhau. Phương pháp này sử dụng một ngưỡng ban đầu và tìm ngưỡng tối ưu bằng cách tính trung bình của của giá trị trung bình của hai vùng được tạo ra. Quá trình này được lặp lại cho đến khi ngưỡng không thay đổi đáng kể. Sau khi có được ngưỡng tối ưu, ảnh ban đầu được chuyển đổi sang ảnh nhị phân, trong đó các điểm ảnh có giá trị lớn hơn ngưỡng được gán giá trị 255, các điểm ảnh còn lại gán giá trị 0.  Quá trình này tạo ra một ảnh chỉ còn 2 mức sáng, giúp tách biệt các vùng ảnh quan trọng. Tiếp theo, áp dụng ý tưởng từ thuật toán loang để tìm các thành phần liên thông. Thuật toán bắt đầu từ một điểm xuất phát và kiểm tra các điểm lân cận của nó để xác định xem chúng thuộc cùng một thành phần liên thông hay không. Quá trình này được lặp lại cho tất cả các điểm lân cận thỏa mãn điều kiện liên thông. Thành phần liên thông có diện tích lớn nhất trong ảnh được xác định là phần nhũ ảnh. Tuy nhiên, nó sẽ phụ thuộc rất nhiều vào việc lấy ngưỡng ảnh có phù hợp hay không, gây tốn khá nhiều thời gian để tìm được ngưỡng thích hợp. Và với những ảnh có nhiều nhiều và ngưỡng ảnh không phù hợp sẽ dẫn đến quá trình cắt ảnh sai và độ chính xác của mô hình sẽ thấp. Tuy nhiên, bộ dữ liệu VinDR và CDD-CESM ít nhiễu nên tỉ lệ cắt đúng phần nhũ lên đến 90%.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 5. Cắt ảnh bằng thuật toán tìm liên thông lớn nhất.

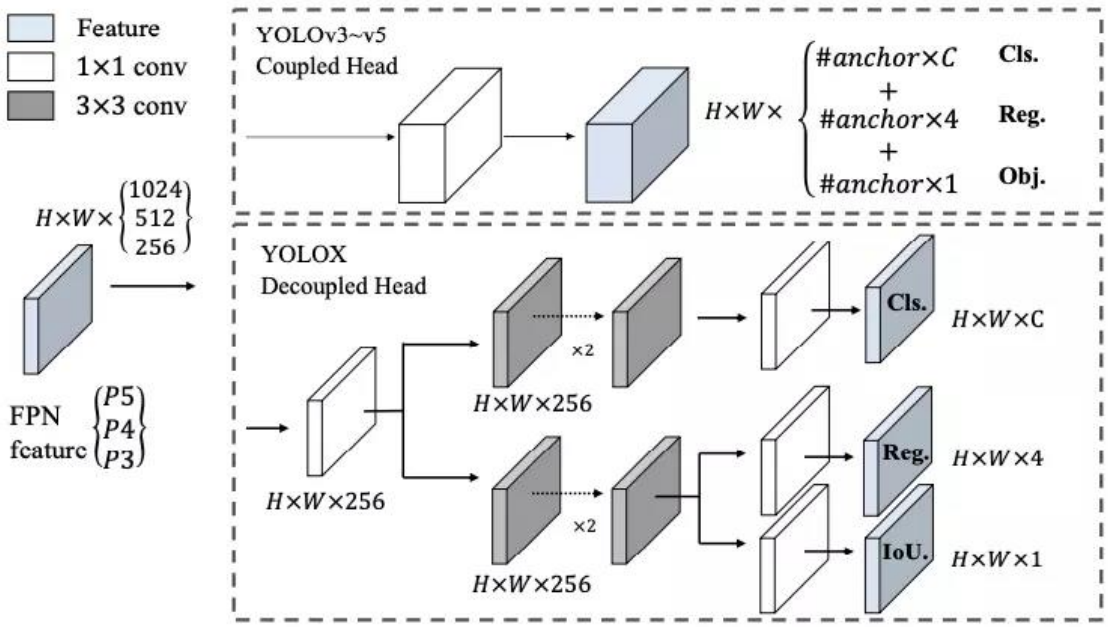
### 3.1.2.3 Mô hình YOLOX




Hình 6. Cắt ảnh bằng YOLOX.

YOLO (You Only Look Once) là một mô hình mạng CNN cho việc phát hiện, nhận dạng và phân loại đối tượng, được tạo ra từ việc kết hợp giữa các convolutional layers và connected layers. Trong đóp các convolutional layers sẽ trích xuất ra các feature của ảnh, còn full-connected layers sẽ dự đoán ra xác suất đó và tọa độ của đối tượng. YOLOX là một mô hình để phát hiện đối tượng trong ảnh nó được cải tiến rất nhiều về tốc độ và độ chính xác. Song với đó, nó sử dụng backbone là Darknet-53 với heads là Decoupled Head đây là điểm độc đáo của YOLOX so với các mô hình khác.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 7. Kiến trúc YOLOX.

Ở sơ đồ này chúng ta có thể thấy rằng đầu vào của head YOLOX là đầu ra từ backbone FPN (darknet) ở 3 tỉ lệ khác nhau – 1024, 512, 256 kênh. Cùng với độ ta cũng thấy rằng đầu ra của YOLOX thực sự là 3 tensors chứa các thông tin khác nhau thay vì 1 tensors lớn chứa tất cả thông tin với các thông tin như sau:

- Cls: lớp của mỗi bounding box.
- Reg: 4 phần của bounding box (x, y, w, h).
- IoU: Chỉ ra mức độ tin cậy của đối tượng trong bounding box.

Một trong những thay đổi của YOLOX là không sử dụng anchor trong phát hiện đối tượng. Thay vì sử dụng các hộp, nó sử dụng centers hoặc key points. Điều này làm cho mô hình trở nên đơn giản, linh hoạt hơn. Trong mô hình phát hiện đối tượng YOLOX sử phương pháp tiếp cận centers để phát hiện đối tượng. Từ đó giá trị dự đoán chỉ là một vector 4D mã hóa vị trí của hộp ở mỗi pixel nền trước thành T = {left, top, right, bottom}. Trong đó left, top, right, bottom là khóa các từ vị của của 4 cạnh của bounding box. Song với đó YOLOX còn sử dụng phương pháp simOTA. Việc sử dụng OTA giúp cải thiện độ chính xác của mô hình được huấn luyện. Nhưng đồng thời cũng tăng 25 % thời gian huấn luyện.

## 3.2 Thuật toán CutMix

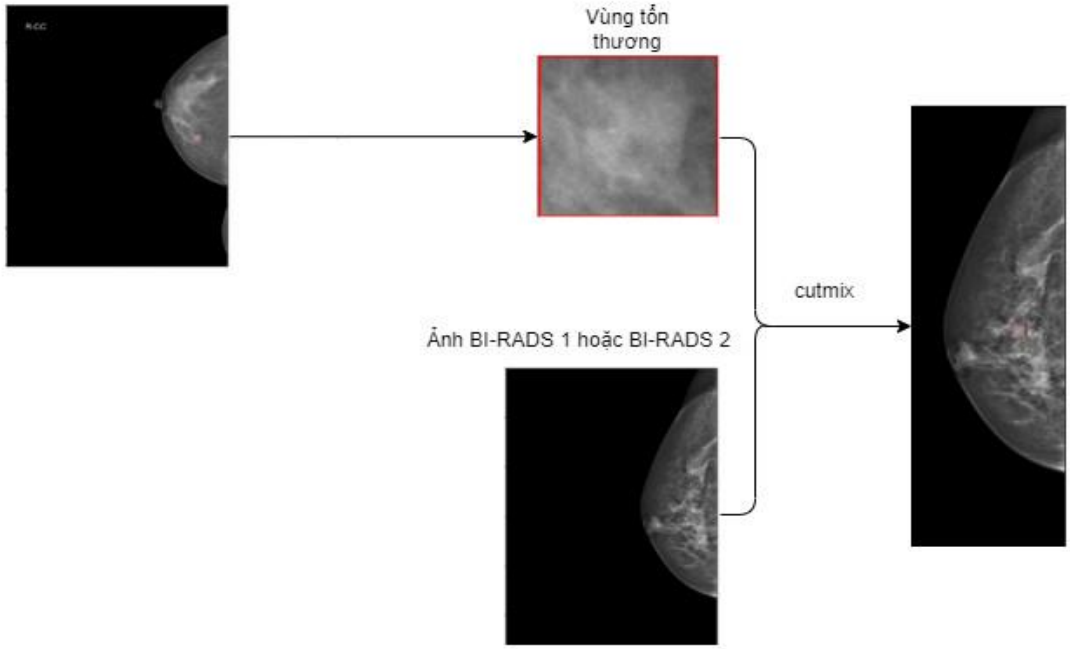
CutMix là một phương pháp tăng cường dữ liệu trong quá trình huấn luyện mô hình học sâu. Trong nghiên cứu này nó được áp dụng để tăng số lượng cho các mẫu bất thường (BI-RADS 3, BI-RADS 4 và BI-RADS 5). Ý tưởng chính của thuật toán này là tạo một mẫu mới bằng cách hợp nhất phép nội suy của cả 2 ảnh và 2 nhãn. Chúng em chuyển vùng tổn thương từ các birads bất thường sang các ảnh bình thường. Đầu tiên, nhóm trích xuất vùng tổn thương từ hình ảnh $A(x_A, y_A)$ với các giá trị vùng tổn thương $Box = (x_{min}, x_{max}, y_{min}, y_{max})$ và nhãn thực tế $y_A \in \{"BIRADS\ 3", "BIRADS\ 4", "BIRADS\ 5"\}$. Một mặt nạ $M \in \{0; 1\}^{W \times H}$ được tạo ra bằng cách giảm các giá trị pixel nền thành 0 và giữ các pixel của vùng tổn thương là 1.

$$M_{ij} = \begin{cases} 1: \ if\ (x_{min} \le i \le x_{max})\ and\ (y_{min} \le j \le y_{max}) \\ 0: \ if\ (x_{min} > i \mid i > x_{max} \mid y_{min} > j \mid j > y_{max}) \end{cases} \quad (2)$$

Trong đó, $x \in R^{W \times H \times C}, y$ tương ứng với một mẫu chụp nhũ ảnh và nhãn của nó, $W, H, C$ lần lượt là chiều rộng, chiều cao và kênh của mẫu tương ứng. Sau đó, mẫu mới $(x', y')$ sẽ được tạo ra như công thức sau

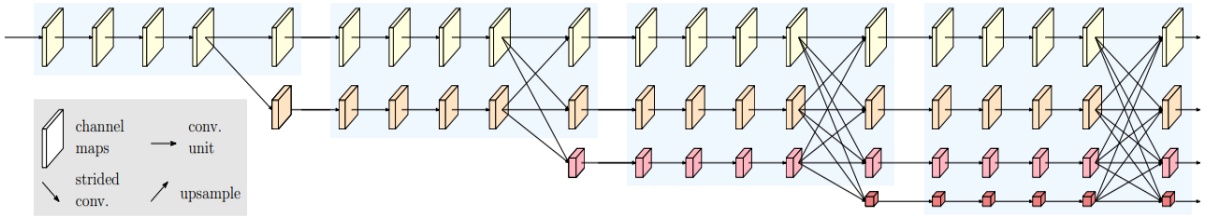
$$x' = M \odot x_A + (1 - M) \odot x_B, y' = y_A \quad (3)$$

Nhãn của ảnh mới sẽ được gán theo nhãn của vùng tổn thương.




Hình 8. Minh họa thuật toán CutMix.

## 3.3 Mô hình HRNet



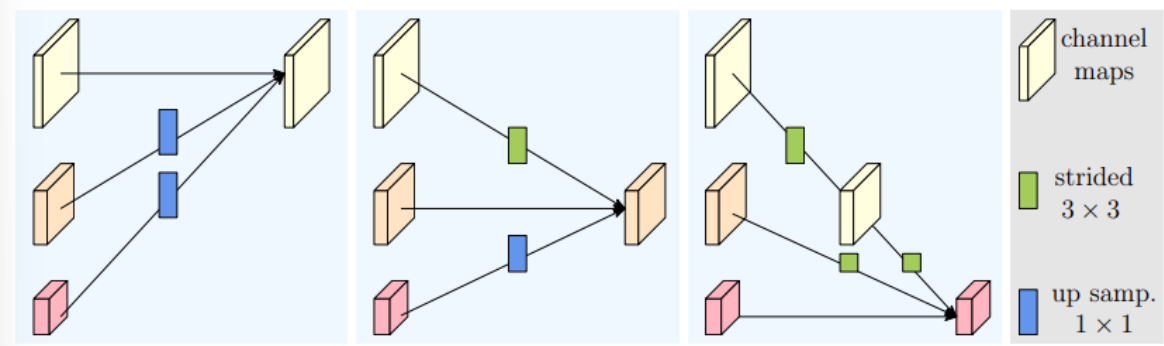

Hình 9. Kiến trúc mô hình HRNet

Ý tưởng chính của mô hình mạng High-Resolution Net là duy trì độ phân giải cao của các đặc trưng xuyên suốt quá trình ảnh được truyền vào mạng. Kiến trúc sẽ chia thành nhiều tầng, tầng phía sau sẽ sinh ra đặc trưng có độ phân giải thấp và sâu hơn, tuy nhiên vẫn duy trì được đặc trưng có độ phân giải ở tầng trước. Hình 9 minh họa kiến trúc HRNet bao gồm 4 giai đoạn, ở giai đoạn số 1, qua nhiều tầng, đặc trưng độ phân giải cao (ký hiệu màu vàng) sẽ được duy trì. Bắt đầu từ tầng số 2, đặc trưng có độ phân giải thấp và sâu hơn (màu cam) sẽ được sinh ra, đặc trưng có độ phân giải cao màu vàng này sẽ được bổ sung ngữ nghĩa bởi các đặc trưng màu cam và ngược lại. Cứ như vậy qua nhiều tầng, các đặc trưng có các độ phân giải khác nhau sẽ được bổ sung ngữ nghĩa cho nhau, dẫn đến cuối cùng ta sẽ thu được đặc trưng độ phân giải cao, tuy nhiên chất lượng vẫn tốt. Luồng xử lý chính được mô tả một cách logic bằng công thức sau:


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

$$\begin{aligned} \mathcal{N}_{11} \rightarrow \mathcal{N}_{21} \rightarrow \mathcal{N}_{31} \rightarrow \mathcal{N}_{41} \\ \searrow \mathcal{N}_{22} \rightarrow \mathcal{N}_{32} \rightarrow \mathcal{N}_{42} \\ \searrow \mathcal{N}_{33} \rightarrow \mathcal{N}_{43} \\ \searrow \mathcal{N}_{44} \end{aligned} \tag{4}$$

Trong đó, $N_{sr}$ là một nhánh đặc trưng tại giai đoạn thứ $s$ và $r$ là chỉ mục độ phân giải. Độ phân giải ở giai đoạn đầu tiên sẽ là $r = 1$, độ phân giải thứ $s$ sẽ bằng $\frac{1}{2^{r-1}}$ độ phân giải ở giai đoạn đầu tiên.




Hình 10. Minh họa cách kết hợp thông tin ngữ nghĩa giữa các đặc trưng độ phân giải cao, trung bình và thấp.

Quan sát hình trên, có 3 đặc trưng có độ phân giải khác nhau. Ta gọi đặc trưng đầu vào là $R^i{}_r$, đặc trưng sau khi kết hợp là $R^o{}_r, r \in 1,2,3$. Một đầu ra tại độ phân giải thứ $r$ ($R^o{}_r$) được tính như sau:

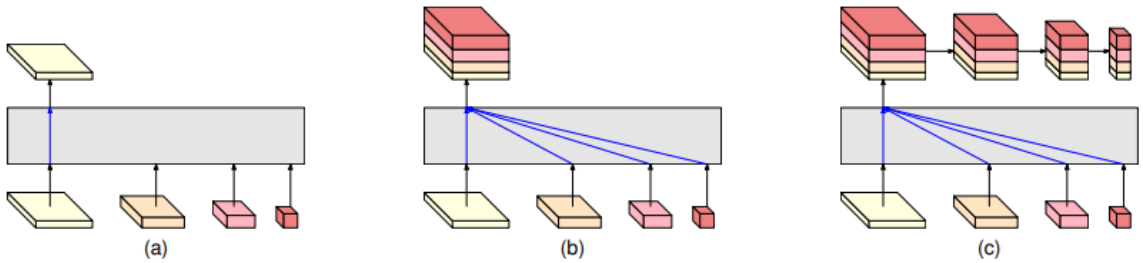
$$R^o{}_r = f_{1r}(R^i{}_1) + f_{2r}(R^i{}_2) + f_{3r}(R^i{}_3) \tag{5}$$

Tại điểm giao giữa giai đoạn 3 và 4 sẽ có thêm một đặc trưng $R^o{}_4$:

$$R^o{}_4 = f_{14}(R^i{}_1) + f_{24}(R^i{}_2) + f_{34}(R^i{}_3) \tag{6}$$

Trong đó, $f_{xr}(\cdot)$ là phép biến đổi đặc trưng có độ phân giải đầu vào là $x$ và độ phân giải đầu ra là $r$. Việc lựa chọn phép biến đổi tùy thuộc vào $x$ và $r$.

- Trường hợp $x = r, f_{xr}(R) = R$.
- Trường hợp $x < r, f_{xr}(R)$ là một phép tích chập sử dụng $(r - s)$ lớp tích chập $3 \times 3$ có $stride = 2$.
- Trường hợp $x > r, f_{xr}(R)$ là một phép upsampling song tuyến, sau đó sẽ đi qua một lớp tích chập $1 \times 1$ để kiểm soát số chiều của đặc trưng.




Hình 11. (a) HRNet-V1 (b) HRNet-V2 (c) HRNet-V2p.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

HRNet có 3 dạng biểu diễn được gọi là HRNet-V1, HRNet-V2 và HRNet-V2p.

- HRNetV1: đầu ra là dạng biểu diễn chỉ lấy từ nhánh đặc trưng có độ phân giải cao (high-resolution). Ba đặc trưng của ba độ phân giải.
- HRNetV2: các tác giả rescale các đặc trưng độ phân giải thấp lên độ phân giải cao bằng một phép upsampling song tuyến mà không thay đổi chiều sâu, nối lại cả bốn đặc trưng. Sau đó, đi qua một lớp tích chập $1 \times 1$ để trộn cả bốn đặc trưng này lại.
- HRNetV2p: các tác giả xây dựng một biểu diễn đa tỷ lệ (multi-level features) bằng cách sử dụng đầu ra từ HRNetV2.

## 4 KẾT QUẢ THỰC NGHIỆM

### 4.1 Dữ liệu

BIRADS đánh giá kết quả của bộ xét nghiệm dựa trên các phân loại từ 1 đến 6, với mỗi cấp độ có ý nghĩa khác nhau. Bao gồm:

- BIRADS 1: Kết quả bình thường, không cần thêm các bộ xét nghiệm bổ sung.
- BIRADS 2: Kết quả bình thường, nhưng cần theo dõi thêm.
- BIRADS 3: Kết quả có khả năng lành tính, nhưng cần theo dõi thêm.
- BIRADS 4: Kết quả có khả năng lành tính/ác tính, cần xét nghiệm bổ sung hoặc phẫu thuật tế bào.
- BIRADS 5: Kết quả có khả năng ác tính, cần xét nghiệm bổ sung hoặc phẫu thuật tế bào.

Ngoài ra thì còn có BIRADS 6: Đã xác định rõ ràng là ung thư vú qua xét nghiệm tế bào.
Độ dày của vú mô tả hình dạng của vú trên ảnh chụp X-quang tuyến vú. Độ dày của vú không phải là thước đo để đánh giá cảm giác của vú. Bác sĩ X-quang có thể ghi chú độ dày của vú trên báo cáo chụp X-quang tuyến vú. Độ dày của vú cao nghĩa là có nhiều mô hơn so với mỡ. Ngược lại, có nhiều mỡ hơn so với mô. Mô vú dày đặc có thể có màu trắng hoặc xám nhạt trên hình ảnh chụp X-quang tuyến vú. Ung thư cũng có màu trắng hoặc xám nhạt nên khó có thể phân biệt được.
Bộ dữ liệu VinDR-Mammography là bộ chụp nhũ ảnh kỹ thuật số quy mô lớn gồm 5000 mẫu với 1 mẫu có 4 góc nhìn tương đương với 20.000 ảnh, được đọc bởi các nhà chụp nhũ ảnh có kinh nghiệm để cung cấp đánh giá ung thư và mật độ vú theo Breast Imaging Report and Data System (BI-RADS). Các vùng bất thường cần kiểm tra cũng được đánh dấu bằng các hình chữ nhật. Hiện, đây là bộ dữ liệu công khai lớn nhất về chụp nhũ ảnh kỹ thuật số có chứa đánh giá BI-RADS và các chú thích bất thường. Về siêu dữ liệu của ảnh DICOM, các thông số kỹ thuật được sử dụng để xử lý, tuổi của bệnh nhân và mô hình của thiết bị ảnh được giữ lại, các thông tin về bệnh nhân đã bị xóa để bảo vệ quyền riêng tư. Đối với thông tin độ tuổi, không có bệnh nhân nào trên 89 tuổi xuất hiện trong tập dữ liệu. Về mặt đánh giá vú tổng thể, các loại đánh giá BI-RADS (từ 1 đến 5) và mật độ vú (A, B, C hoặc D) được cung cấp. Những phát hiện của BI-RADS 2 tức là lành tính, không được đánh dấu vùng tổn thương. Chỉ những phát hiện của BI-RADS 3, 4 hoặc 5, cần kiểm tra theo dõi, được chú thích bởi các hình chữ nhật đánh dấu vùng tổn thương.

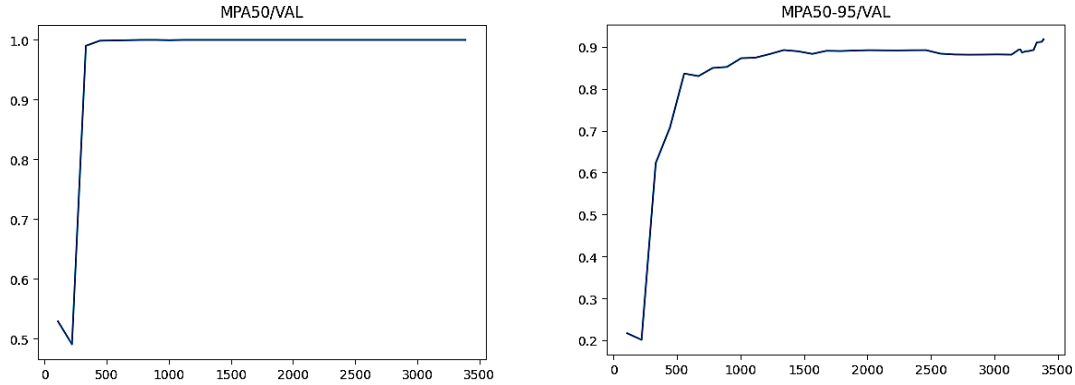
### 4.2 Kết quả cắt ảnh bằng mô hình YOLOX

Bảng 1. Tham số của mô hình YOLOX.

| **Tham số** | **Giá trị** |
|---|---|
| Train - set | 1800 |
| Validation - set | 200 |
| Tổng tham số của mô hình | 9 triệu |

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| | |
|---|---|
| Kích thước ảnh đầu vào | 640x640 |
| loss | IOUloss, Final Loss, BCEloss |




Hình 12. Kết quả mô hình YOLOX với mAP ngưỡng từ 50-95.

mAP của mô hình là 0.92 rất cao với tổng 9 triệu tham số với tốc độ cắt trung bình 0.05s một tấm ảnh. Với thuật toán này vừa tiết kiệm được thời gian và kết quả cắt ảnh từ mô hình này cũng đạt hiệu suất cao hơn so với các thuật toán mà tôi đã thực nghiệm.

### 4.3 Kết quả phân loại BI-RADS

Để đánh giá tổng quan bài toán phân loại 5 lớp BI-RADS từ 1 đến 5. Chúng tôi sử dụng các thang đo: ma trận nhầm lẫn, độ chính xác, f1-score, precision, recall và kappa. Ở đây, điểm Cohen Kappa là một thước đo độ tin cậy giữa hai quan sát viên, hoặc giữa một quan sát viên và một mô hình phân loại. Điểm Kappa của Cohen đo lường mức độ đồng thuận giữa hai quan sát viên bằng cách so sánh số lượng đồng ý thực sự giữa hai người với số lượng đồng ý dự đoán bằng mô hình.

$$Kappa = \frac{Po - Pe}{1 - Pe} \tag{7}$$

Trong đó:

$Po = \frac{TP+TN}{TP+FP+FN+TN}$: Xác suất quan sát được.

$Pe = \frac{(TP+FP)\times(TP+FN) + (FP+TN)\times(FN+TN)}{(TP+TN+FP+FN)^2}$: Xác suất kỳ vọng.

$TP\ (True\ Positive), FP\ (False\ Positive), TN\ (True\ Negative), FN\ (False\ Negative$ đều được định nghĩa dựa trên $confusion\ matrix$ chưa chuẩn hóa như sau:

| | Predicted as Positive | Predicted as Negative |
|---|---|---|
| Actual: Positive | TP | FN |
| Actual: Negative | FP | TN |

Giá trị của điểm Cohen Kappa dao động từ -1 đến 1, trong đó giá trị 1 đại diện cho sự đồng thuận hoàn toàn giữa các quan sát viên hoặc giữa quan sát viên và mô hình phân loại. Giá trị 0 đại diện cho sự đồng thuận ngẫu nhiên và giá trị âm cho sự không đồng thuận hơn so với sự đồng thuận ngẫu nhiên. Các giá trị âm của điểm Cohen Kappa thường được hiểu là mức độ không tốt và cần được cải thiện. Thông thường, một mô hình được coi là tốt nếu điểm Kappa từ 0.6 trở lên.

Sau nhiều lần thực nghiệm với các kết quả đạt được chưa khả quan. Nhóm tiếp tục thử nghiệm với mô hình HRNet qua các bước sau:

**Bước 1:** Tiền xử lý nhũ ảnh.

- Chuyển ảnh Dicom sang png.
- Apply windowing làm cải thiện độ tương phản giữa các vùng mô mềm và mô đặc biệt trong nhũ ảnh.
- Sử dụng thuật toán Cut - mix để tăng cường dữ liệu cho các nhãn ít dữ liệu để giảm mất cân bằng dữ liệu.
- Cắt ảnh bằng mô hình YOLOX.

**Bước 2:** Trích một tập dữ liệu gồm 1130 ảnh và 31430 ảnh.

**Bước 3:** Huấn luyện mô hình với HRNet.

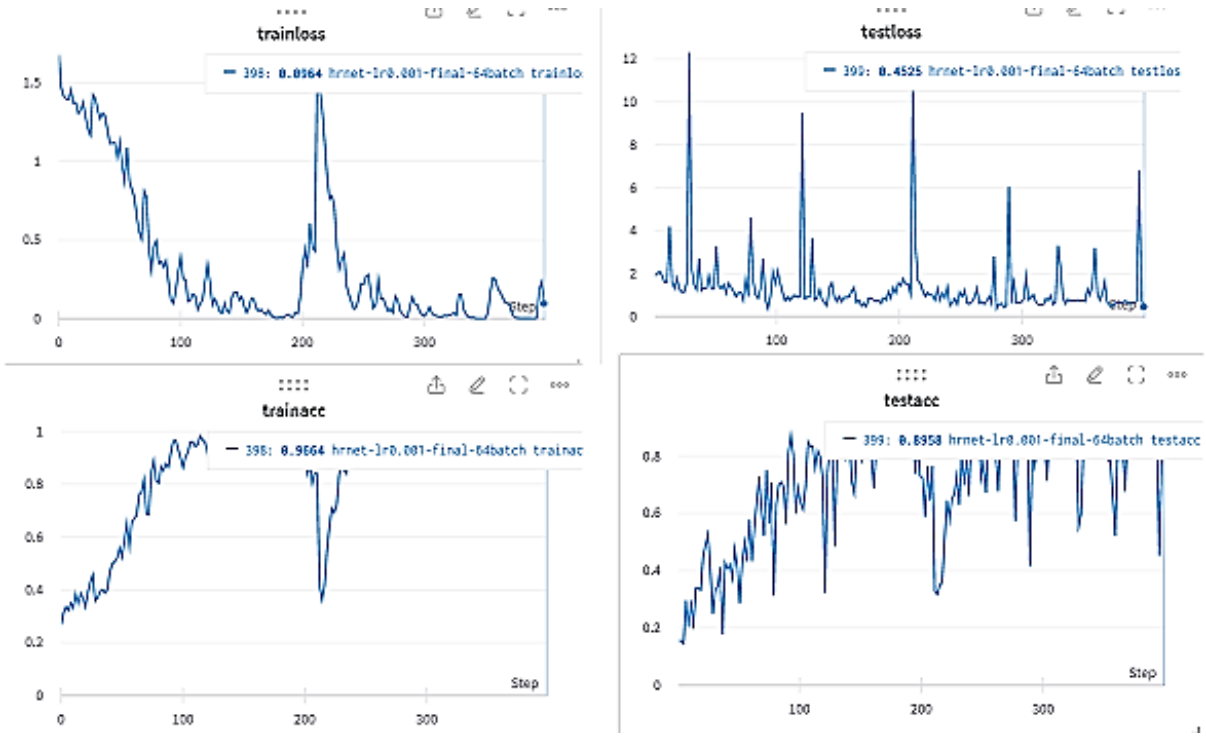
Bảng 2. Bảng các siêu tham số sử dụng trong mô hình HRNET.

| **Tham số** | **Giá trị** |
|---|---|
| epoch | 100-200 |
| learning rate | 0.001 |
| batch size | 8-64 |
| optimize | adam |
| Loss | CrossEntropyLoss |

Bảng 3. Kết quả các lần thực nghiệm với mô hình HRNet.

| **Mô hình HRNet** | **Độ đo** | | | |
|---|---|---|---|---|
| | **Train - acc** | **Train - loss** | **Test - acc** | **Test - loss** |
| Tập 1130 ảnh – 100 epoch | 97.76% | 0.1012 | 38.54% | 2.085 |
| Tập 31430 ảnh – 100 epoch | 96.87% | 0.09102 | 51.15% | 3.000 |
| Tập 31430 ảnh – train 200 epoch, Batch-size=8 | 92.45% | 0.2068 | 83.33% | 0.6925 |
| Tập 31430 ảnh – train 200 epoch, batch_size=64 | 96.63% | 0.0964 | 89.85% | 0.4524 |


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 13. Kết quả mô hình HRNet khi train 200 epoch, batch-size=64.

### 4.3.1 Đánh giá kết quả các mô hình

Bảng 4. So sánh kết quả của các mô hình qua các lần thực nghiệm

| Mô hình | Độ đo | | | | | |
|---|---|---|---|---|---|---|
| | F1-score | Precision | Recall | Kappa | Accuracy | Loss |
| VGG-19 – 16000 ảnh. | 0.195 | - | - | - | 20% | 1.6096 |
| ResNet-50 – cắt ảnh bằng tìm liên thông lớn nhất. | 0.287 | - | - | - | 29.71% | 1.438 |
| Wide ResNet-50 – cắt ảnh bằng tìm liên thông lớn nhất. | 0.68 | - | - | - | 67.03% | 0.9243 |
| RN-BCNN – cắt ảnh bằng YOLOX. | 0.997 | 0.998 | 0.997 | 0.994 | 99.79% | 0.913 |
| DenseNet-121 – upsamling dữ liệu | 0.71 | - | - | - | 70% | 0.8265 |
| DenseNet-201 – upsamling dữ liệu | 0.85 | - | - | - | 86% | 0.3265 |
| HRNet – 1130 ảnh | 0.9776 | 0.975 | 0.977 | 0.95 | 97.76% | 0.1012 |
| HRNet – 31430 ảnh | 0.9245 | 0.9251 | 0.9247 | 0.93 | 96.63% | 0.0964 |

### 4.3.2 So sánh kết quả với các bài báo liên quan

Bảng 5.So sánh kết quả với các bài báo liên quan

| Mô hình | Tập dữ liệu | Độ đo | | | | | |
|---|---|---|---|---|---|---|---|
| | | Accuracy | Loss | F1-score | Kappa | Precision | Recall |
| Nghiên cứu này (HRNet) | VinDR + CDD-CESM | 89.58% | 0.4524 | 0.8516 | 0.867 | 0.9105 | 0.8581 |
| Paper [4] (EfficientNet-B2) | VinDR | - | - | 0.676 | - | - | - |
| Paper [1] (CM-Ensemble) | Dữ liệu riêng tư | 85% | - | - | 0.71 | - | - |
| Paper [3] (EfficientNet-B2) | Dữ liệu riêng tư | 67.02% | - | - | - | - | - |
| Paper [6] (RN-BCNN) | Dữ liệu riêng tư | 85.9% | - | - | - | - | - |

Sau khi hiện thực toàn bộ quá trình huấn luyện và so sánh kết quả với các bài báo nghiên cứu, tôi đã rút ra được những nhận xét cho quá trình hiện thực và cải tiến mô hình huấn luyện như sau:
Về hiệu suất của mô hình qua các lần thực nghiệm. HRNet đạt hiệu quả cao nhất khi ứng dụng phương pháp cutmix tăng cường dữ liệu lên 31430 ảnh đạt độ chính xác ở tập huấn luyện là 96.63% và 89.58% ở tập kiểm thử. Tuy nhiên, hiện dữ liệu quá lớn cộng với đó là batch-size nhỏ mô hình phải huấn luyện quá lâu và gặp khó khăn khi bị ngắt do giới hạn giờ GPU của Kaggle. Điều này có thể ảnh hưởng đến hiệu suất tổng quát của mô hình vì mô hình chưa hội tụ. Tuy nhiên, cũng không thể phủ nhận rằng việc kết hợp phương pháp tiền xử lý dữ liệu mới và tiên tiến với việc sử dụng windowing sau đó cắt ảnh bằng mô hình YOLOX và cân bằng dữ liệu bằng phương pháp upsample và thuật toán cutmix. Mô hình sau nhiều lần thực nghiệm đã có hiệu suất tốt hơn khi đánh giá trên cùng tập dữ liệu VinDr với bài báo "A Novel Transparency Strategy-based Data Augmentation Approach for BI-RADS Classification of Mammograms". Bên cạnh đó, nhóm cũng đã đánh giá ở nhiều độ đo khác nhau để có cái nhìn khách quan tổng thể về bài toán. Qua đây, ta có thể kết luận khi áp dụng các kỹ thuật xử lý ảnh và các mô hình học sâu vào trong bài toán phân loại BI-RADS đã mang lại nhiều kết quả tốt và khả quan khi có thể hỗ trợ phát hiện tình trạng bệnh, phát hiện kịp thời, điều trị nhanh chóng, tăng độ chính xác, chi phí thấp, hỗ trợ các y bác sĩ trong việc đưa ra các chẩn đoán bệnh và phương pháp điều trị.

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Từ những nghiên cứu thực nghiệm đạt được, có thể thấy việc sử dụng mô hình HRNet cho kết quả phân loại rất ấn tượng. Việc áp dụng các phương pháp xử lý ảnh và phương pháp áp dụng thuật toán cutmix giải quyết vấn đề mất cân bằng dữ liệu mang lại hiệu quả đáng kể cho thấy rằng nó có thể cải thiện phân loại BI-RADS nhiệm vụ cực kì quan trọng trong việc chụp quang tuyến vú. Với cách tiếp cận này có thể được áp dụng cho các bài toán trong lĩnh vực y tế với nhiệm vụ phát hiện bất thường và phân loại tổn thương. Tuy nhiên, ở nghiên cứu này tôi chỉ mới áp dụng trên 2 bộ dữ liệu vì thế trong tương lai cần thu thập và mở rộng thêm dữ liệu để mô hình học tốt hơn, phân loại chính xác hơn cũng như khách quan hơn.

## 6 TÀI LIỆU THAM KHẢO

[1] H. Lopez-Almazan, "A deep learning framework to classify breast density with noisy labels regularization," *Computer Methods and Programs in Biomedicine,* May 2022.

[2] Z. Y. Leee, "Classification of mammographic breast density and its correlation wwith BIRADS in elder women using machine learning approach," *Journal of Medical Imaging and Rdiation Sciences,* 2022.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

[3] Huyen Nguyen, "A novel multi-view deep learning approach for BI-RADS and density," *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC),* July 2022.

[4] H. T. N. Sam B. Tran, "A Novel Transparency Strategy-based Data Augmentation Approach for BI-RADS Classification of Mammograms".

[5] K.-J. Tsai, "A High-Performance Deep Neural Network Model for BI-RADS Classification of Screening Mammography," *Sensors,* vol. 22, no. 3, p. 1160, Feb 2022.

[6] S. Siddeeq, "Deep Learning RN-BCNN Model for Breast Cancer BI-RADS Classification".

[7] S. Yun, "CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features," August 2019.

[8] Wikipedia, "Upsampling," [Online]. Available: https://en.wikipedia.org/wiki/Upsampling?fbclid=IwAR3UxUZXsNiluBGwd5ksoAMt4m5KfM8xhKdPQTELtai_wRh706KDzlK8hH8. [Accessed 05 04 2023].

[9] J. Wang, "Deep High-Resolution Representation Learning for Visual Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 43, no. 10, pp. 3349-3364, October 2021.

[10] T. Vu, "Softmax Regression," February 2017.

[11] T. Vu, "Deep Learning," [Online]. Available: https://machinelearningcoban.com/2018/06/22/deeplearning/. [Accessed 10 05 2023].

[12] S.-H. Tsang, "Review: DenseNet — Dense Convolutional Network (Image Classification) - Medium," [Online]. Available: https://towardsdatascience.com/review-densenet-image-classification-b6631a8ef803. [Accessed 05 05 2023].

[13] T. T. Trực, "Optimizer- Hiểu sâu về các thuật toán tối ưu (GD,SGD,Adam,..)," [Online]. Available: https://viblo.asia/p/optimizer-hieu-sau-ve-cac-thuat-toan-toi-uu-gdsgdadam-Qbq5QQ9E5D8?fbclid=IwAR2P6GBv9Nx5wd_Otp6nJAjf-zaVOIxu5I-Hz660fEaMrgc9AIKBolFJWHQ. [Accessed 04 05 2023].

[14] T. D. Thang, "Giới thiệu mạng ResNet," [Online]. Available: https://viblo.asia/p/gioi-thieu-mang-resnet-vyDZOa7R5wj. [Accessed 10 05 2023].

[15] N. K. Sergey (S.) Zagoruyko, "Wide Residual Networks," January 2016.

[16] H. N. Sam B.Tran, "A Novel Transparency Strategy-based Data Augmentation Approach for BI-RADS Classification of Mammograms," 2022.

[17] H. H. Pham, "VinDr-Mammo: A large-scale benchmark dataset for computer-aided detection and diagnosis in full-field digital mammography v1.0.0," March 2022.

[18] H. H. N. N.Đ Thùy Giang, "Ung Thư Vú," Bộ Môn Ung Thư ĐHYD TPHCM.

[19] M. L. McHugh, "Interrater reliability: the kappa statistic," *PubMed Central (PMC),* October 2012.

[20] P. D. Khanh, "YOLO You Only Look Once," March 2020.

[21] A. Z. Karen (K.) Simonyan, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *Computer Vision and Pattern Recognition,* September 2014.

[22] G. Huang, "Densely Connected Convolutional Networks," July 2017.

[23] Z. Ge, "YOLOX: Exceeding YOLO Series in 2021".

[24] N. (N.), "Bài 6: Convolutional neural network," *Deep Learning cơ bản,* April 2019.

**YSC5.F031**

# GIẢI Ô SỐ SUDOKU KÍCH THƯỚC LỚN VỚI GIẢI THUẬT DI TRUYỀN ĐA DẠNG HÓA QUẦN THỂ KẾT HỢP BACKTRACKING

TRẦN MẠNH THÌN [1], TRẦN THANH LƯỢNG [1], ĐỖ HÀ PHƯƠNG[1*]

*. [1]Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành Phố Hồ Chí Minh,*
*[*] dohaphuong@iuh.edu vn*

**Tóm tắt.** Trong bài báo này, chúng tôi thực hiện áp dụng giải thuật di truyền đa dạng hóa quần thể kết hợp với kỹ thuật backtracking để giải ô số sudoku, một trong những game trong thi đấu siêu trí tuệ luôn là mối quan tâm giữa trí tuệ nhân tạo và trí tuệ con người. Hướng nghiên cứu của chúng tôi nhằm cố gắng vượt qua những hạn chế với giải thuật backtracking đối với ô số sudoku kích thước lớn cũng như vấn đề bẫy tối ưu cục bộ với họ giải thuật Depth First Search. Chúng tôi thực nghiệm đối chứng cải tiến giải thuật với giải thuật backtracking trên bộ số sudoku 9x9, 16x16, 25x25 các cấp độ (dễ, trung bình, khó) và cho kết quả khá khả quan.

**Từ khóa.** AI, GA, giải thuật di truyền, Sudoku với GA.

## LARGE SUDOKU SOLUTION WITH POPULATION DIVERSIFICATION GENETIC ALGORITHM COMBINED WITH BACKTRACKING

**Abstract.** In this paper, we apply a diverse population genetic algorithm combined with backtracking technique to solve Sudoku puzzles, one of the games in the realm of superintelligence that has always been of interest to both artificial intelligence and human intelligence. Our research direction aims to overcome the limitations of the backtracking algorithm for large-sized Sudoku puzzles as well as the local optimum trap issue with Depth First Search algorithms. We experimentally compare the improved algorithm with the backtracking algorithm on 9x9, 16x16, 25x25 Sudoku puzzles at various difficulty levels (easy, medium, hard) and obtain quite promising results.

**Keywords.** AI, GA, genetic algorithm, Sudoku with GA.

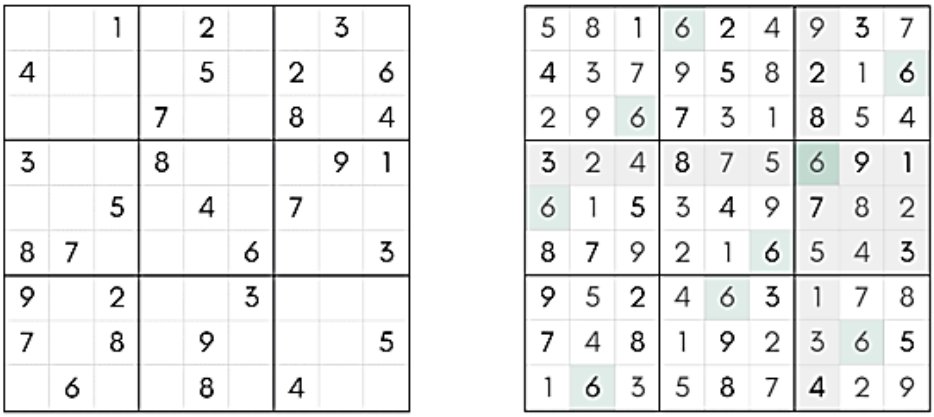
## 1 GIỚI THIỆU

### 1.1 Sudoku

Các trò chơi gần đây đã trở thành một chủ đề nổi bật trong nghiên cứu và phát triển lĩnh vực trí tuệ nhân tạo (AI). Các trò chơi được đề cập điển hình như cờ ca-rô, oẳn tù tì, N-Queens… [1, 2, 3] được sử dụng để chứng minh độ hiệu quả cũng như khả năng giải quyết vấn đề của thuật toán AI. Sudoku cũng là trò chơi giải đố mang tính logic cao, được biết rộng rãi và được yêu thích trên toàn thế giới, Sudoku có hẳn một giải đấu quốc tế gọi là "The World Sudoku Championship" (WSC), được tổ chức lần đầu tiên vào năm 2006 tại Lucca, Ý [4]. Tại WSC, người chơi đối mặt với nhiều thách thức đáng kể trong việc giải các câu đố Sudoku với độ khó và phức tạp cao, đồng thời cạnh tranh và thể hiện sự sáng tạo của mình. Một trong những thách thức lớn là việc giải các câu đố Sudoku kích thước lớn và có độ khó đa dạng. Các bảng Sudoku có kích thước tăng lên đòi hỏi tư duy và suy luận sắc sảo hơn để tìm ra các giải pháp chính xác. Ngoài ra, việc tạo ra các câu đố Sudoku độc đáo và hấp dẫn cũng là một thách thức đối với các nhà thiết kế. Họ phải tạo cấu trúc đối xứng và hài hòa, đồng thời đảm bảo rằng mỗi câu đố chỉ có duy nhất một lời giải [4]. Điều này yêu cầu sự sáng tạo và cẩn thận trong từng chi tiết của câu đố, từ việc xác định vị trí các số đã điền sẵn cho đến cách phân loại độ khó của câu đố. Gần đây Sudoku còn được lựa chọn để tham gia vào các game đấu siêu trí tuệ ở Việt Nam cũng như ở các nước luôn hấp dẫn và được quan tâm ở mọi lứa tuổi.

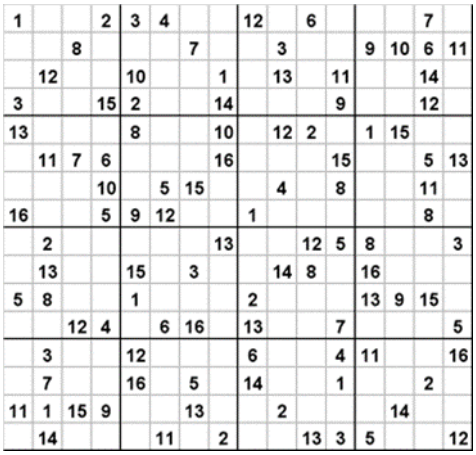
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Sudoku là một trò chơi xuất phát điểm ban đầu ở Nhật Bản, trên một bảng hình vuông bao gồm 81 ô vuông nhỏ (Hình 1). Trên bảng này, mỗi ô vuông được điền số từ 1 đến 9, mục tiêu của trò chơi là các con số bị thiếu vào các ô vuông trống sao cho mỗi hàng, cột, và các vùng khối vuông 3x3 chỉ chứa số từ 1 đến 9 mà không số nào được lặp lại (Hình 1).

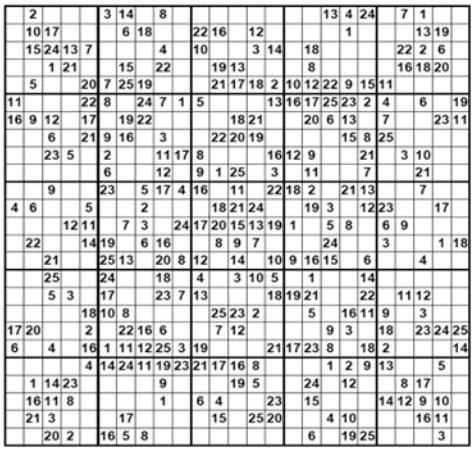


Hình 1: Câu đố tiêu chuẩn và giải pháp đúng của Sudoku

Một số biến thể lớn khác của Sudoku ngoài dạng tiêu chuẩn 9x9, chẳng hạn 16x16 (Hình 2), 25x25 (Hình 3),… Các biến thể này tạo nên tổng thể một đề bài phức tạp hơn rất nhiều lần so với dạng đề tiêu chuẩn và mất rất nhiều thời gian để người chơi có thể giải được.



Hình 2: Biến thể Sudoku 16x16



Hình 3: Biến thể Sudoku 25x25

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Nhìn chung, luật chơi của Sudoku khá đơn giản. Các cấp độ dễ, trung bình và khó của Sudoku được xác định dựa trên số lượng số đã điền sẵn trong bảng Sudoku và độ khó trong việc giải câu đố.

- Cấp độ dễ: Các câu đố Sudoku cấp độ dễ thường có nhiều số điền sẵn, tạo điều kiện thuận lợi cho người chơi bắt đầu và suy luận. Điều này giúp người chơi dễ dàng điền các số còn thiếu và hoàn thành câu đố một cách nhanh chóng. Cấp độ dễ thích hợp cho người mới bắt đầu và là nơi tập trung vào các quy tắc Sudoku cơ bản.
- Cấp độ trung bình: Các câu đố Sudoku cấp độ trung bình có số lượng số điền sẵn trung bình, đòi hỏi người chơi phải áp dụng các quy tắc Sudoku cơ bản và mở rộng phạm vi suy luận. Trong câu đố này, có một số xung đột nhỏ và các số còn thiếu có thể gây ra khó khăn trong việc giải. Cấp độ trung bình thách thức hơn và là nơi người chơi có thể rèn luyện kỹ năng suy luận và giải quyết vấn đề.
- Cấp độ khó: Các câu đố Sudoku cấp độ khó có số lượng số điền sẵn ít, yêu cầu người chơi có kiên nhẫn và kỹ năng giải quyết vấn đề cao hơn để đối phó với sự phức tạp này. Trong câu đố khó, có nhiều xung đột và các số còn thiếu rất ít thông tin dẫn dắt. Giải các câu đố khó đòi hỏi sử dụng các kỹ thuật tiên tiến, suy luận sáng tạo và thậm chí thử nghiệm từng bước một để tìm ra lời giải chính xác. Cấp độ khó là nơi người chơi có thể thách thức và nâng cao trình độ Sudoku của mình.

Những cấp độ này cung cấp những trải nghiệm khác nhau và thích hợp cho mọi người từ người mới chơi đến những người chơi có kinh nghiệm và yêu thích những thách thức Sudoku đa dạng.
Trong AI, giải thuật phổ biến nhất được sử dụng để giải quyết các câu đố này là giải thuật tìm kiếm quay lui (Backtracking Algorithm).

## 1.2 Giải thuật quay lui

Giải thuật quay lui (Backtracking) là một phương pháp giải quyết các vấn đề tìm kiếm và thử nghiệm trong lĩnh vực trí tuệ nhân tạo và thuật toán. Backtracking là một biến thể khác của tìm kiếm theo chiều sâu (DFS) [5]; nó được sử dụng để tìm ra tất cả các giải pháp hoặc tìm ra một giải pháp duy nhất cho các vấn đề mà việc thử tất cả các khả năng bằng vét cạn là quá tốn kém.
Phương pháp backtracking hoạt động theo nguyên tắc "thử và lùi" (Trial and Error). Nó tìm kiếm qua tất cả các khả năng trong không gian tìm kiếm bằng cách thử từng giá trị cho các biến và tiếp tục đi sâu vào mỗi khả năng cho đến khi tìm ra giải pháp hoặc gặp ràng buộc không thỏa mãn. Khi ràng buộc không thỏa mãn, thuật toán quay lại (lùi) và thử các khả năng khác.

Bảng 1: Ưu, nhược điểm của giải thuật quay lui [6]

| Ưu điểm | Nhược điểm |
|---|---|
| • Quay lui có bản chất đánh giá tất cả, thử tất cả các tổ hợp để tìm ra một lời giải.<br>• Các vấn đề của backtracking rất trực quan khi viết mã.<br>• Cách trình bày từng bước của quay lui rất dễ hiểu.<br>• Có thể dễ dàng tìm thấy và khắc phục lỗi mã quay lui<br>• Mã quay lui sử dụng ít dòng mã hơn. | • Khi so sánh với cách loại khác, nó khá là chậm.<br>• Tùy thuộc vào dữ liệu của bạn, có thể sử dụng quay lui để thực hiện tìm kiếm kỹ lưỡng nhưng cuối cùng không mang lại kết quả đáp ứng tiêu chí tìm kiếm.<br>• Quay lui là một thuật toán đệ quy với chi phí tính toán cao, sử dụng nhiều bộ nhớ và CPU.<br>• Do việc sử dụng đệ quy và lưu trữ ngăn xếp cho thông tin chức năng nên có độ phức tạp về không gian cao. |

## 1.3 Giải thuật di truyền

Giải thuật di truyền (GAs) thuộc họ của thuật toán tiến hóa (Evolutionary Algorithm). Họ thuật toán tiến hóa bao gồm nhiều phương pháp tối ưu hóa dựa trên các quy trình tương tự tiến hóa trong tự nhiên, bao gồm giải thuật di truyền, chiến lược tiến hóa, lập trình tiến hóa, và nhiều biến thể khác. [7]
GAs sử dụng ngôn ngữ máy tính để mô phỏng quá trình tiến hoá của một tập hợp những đại diện trừu tượng (gọi là những nhiễm sắc thể), của các giải pháp có thể (gọi là những cá thể) cho bài toán tối ưu hóa vấn đề.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Tập hợp này sẽ tiến triển theo hướng chọn lọc những giải pháp tốt hơn. Các giải pháp được phát triển theo thời gian bằng cách sử dụng các toán tử lai ghép, đột biến; quá trình này tạo ra một tập hợp khác của bộ gen được cải tiến. Thuật toán sẽ dừng khi hoặc là tìm thấy giải pháp hoặc là điều kiện dừng kết khúc. [8]

Bảng 2: Ưu, nhược điểm của giải thuật di truyền [9]

| Ưu điểm | Nhược điểm |
|---|---|
| • GAs là một quy trình tối ưu hóa linh hoạt, có thể áp dụng rộng rãi.<br>• Có thể tối ưu đồng thời nhiều biện pháp song song (đa mục tiêu).<br>• Toán tử có thể được tùy chỉnh để tận dụng các quy tắc hoặc ràng buộc trong một miền cụ thể để cải thiện tốc độ hoặc chất lượng độ hội tụ. | • Khó dự đoán thời gian hội tụ sẽ mất bao lâu – tính ngẫu nhiên trong quá trình có nghĩa là điều này có thể rất khác nhau.<br>• Bởi vì nó yêu cầu biểu diễn và xử lý trên một quần thể kiểu gen khá lớn, nó có thể tốn kém về bộ nhớ và tính toán - những vấn đề rất phức tạp có thể không khả thi. |

## 2 GIẢI QUYẾT VẤN ĐỀ

Có một số cuộc nghiên cứu đã thực hiện cho việc giải Sudoku sử dụng giải thuật tiến hóa. Timo Mantere và Jane Koljionen đến từ Đại học Vassa đã tiếp cận GAs để giải quyết câu đố Sudoku[10]; họ sử dụng GAs với toán tử lai ghép và đột biến đơn giản, song thuật toán thiếu đi sự đa dạng để đạt được kết quả tối ưu. Tiến sĩ JohnM. Weiss cũng đã sử dụng GAs với biểu diễn n mảng 1-D cho câu đố n*n. Ông ấy sử dụng Generational GAs để  sản sinh quần thể giải pháp và dùng chiến lược "khởi động lại" trong trường hợp tìm thấy tối ưu cục bộ. [11]

GAs có khả năng giải quyết vấn đề Sudoku ở bộ số 9x9, và trở nên khó khăn trong cấp độ khó trở lên, độ phức tạp sẽ tăng dần theo số ô trống của trò chơi. Do đó, hàm mục tiêu là yếu tố để có thể giữ lại cá thể tốt nhất. Song, vấn đề Sudoku sẽ phụ thuộc đến giải thuật quay lui.

Trong bài nghiên cứu, GAs được thiết kế đặc biệt cho câu đố 9x9 do hiệu quả cao mà Backtracking có thể mang lại; mặc dù vậy, cách tiếp cận có thể áp dụng cho dạng Sudoku lớn hơn như 16x16. Mục tiêu chính của nó là có thể giải quyết hiệu quả các cấp độ khác nhau cũng như dạng đề lớn của câu đố Sudoku.

GAs được sử dụng ở đây là Steady State GA (SSGA). SSGA thực hiện sản sinh ra một cá thể và thay thế nó bằng một cá thể khác trong quần thể hiện tại. Điểm nổi bậc trong GA này là sử dụng Backtracking để tạo mẫu thông qua đó có thể tối ưu được cá thể tìm kiếm mới. Ngoài ra, một nỗ lực khác giúp đẩy nhanh quá trình tìm kiếm bằng thực hiện song song thông qua sử dụng mô hình quần thể đảo; mô tả chi tiết đã cho bên dưới:

### 2.1 Đại diện trừu tượng

Trong GA, Sudoku được đại diện trừu tượng bằng một cá thể (individual); mỗi cá thể được biểu diễn dưới dạng một nhiễm sắc thể (NST) là một chuỗi gen bao gồm m số lượng gen (m=n^2, n là kích thước của Sudoku). Đối với bài toán nxn, một NST tượng tương cho một giải pháp Sudoku tương ứng; bao gồm n mảng kích thước là n. Đại diện cho câu đố Sudoku được đưa ra như sau:

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Bảng 3: Đại diện cho cá thể Sudoku 9x9

| **Mảng** | **Phần tử** | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 8 | 7 | 6 | 5 | 4 | 3 | 1 | 9 | 2 |
| 2 | 5 | 4 | 3 | 2 | 1 | 9 | 7 | 6 | 8 |
| 3 | 2 | 1 | 9 | 8 | 7 | 6 | 4 | 3 | 5 |
| 4 | 1 | 9 | 8 | 7 | 6 | 5 | 3 | 2 | 4 |
| 5 | 4 | 3 | 2 | 1 | 9 | 8 | 6 | 5 | 7 |
| 6 | 7 | 6 | 5 | 4 | 3 | 2 | 9 | 8 | 1 |
| 7 | 3 | 2 | 1 | 9 | 8 | 7 | 5 | 4 | 6 |
| 8 | 6 | 5 | 4 | 3 | 2 | 1 | 8 | 7 | 9 |
| 9 | 9 | 8 | 7 | 6 | 5 | 4 | 2 | 1 | 3 |

Bảng 4: Đại diện cho cá thể Sudoku 16x16

| **Mảng** | **Phần tử** | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 6 | 3 | 2 | 5 | 10 | 9 | 8 | 11 | 7 | 12 | 15 | 14 | 13 | 4 | 16 |
| 2 | 15 | 11 | 4 | 1 | 10 | 16 | 14 | 13 | 9 | 2 | 12 | 3 | 5 | 6 | 7 | 8 |
| 3 | 13 | 8 | 9 | 12 | 1 | 2 | 4 | 5 | 3 | 16 | 14 | 6 | 15 | 11 | 10 | 7 |
| 4 | 2 | 14 | 16 | 12 | 11 | 13 | 15 | 10 | 1 | 9 | 3 | 5 | 8 | 4 | 7 | 6 |
| 5 | 1 | 15 | 11 | 10 | 4 | 5 | 16 | 2 | 13 | 8 | 9 | 12 | 6 | 7 | 14 | 3 |
| 6 | 12 | 13 | 15 | 7 | 4 | 1 | 5 | 6 | 2 | 3 | 8 | 9 | 14 | 16 | 11 | 10 |
| 7 | 5 | 2 | 6 | 1 | 12 | 4 | 9 | 8 | 15 | 11 | 10 | 7 | 16 | 14 | 13 | 3 |
| 8 | 1 | 2 | 3 | 4 | 7 | 10 | 8 | 11 | 6 | 12 | 5 | 14 | 15 | 13 | 16 | 9 |
| 9 | 10 | 7 | 15 | 11 | 16 | 1 | 2 | 3 | 12 | 13 | 5 | 4 | 8 | 9 | 14 | 6 |
| 10 | 9 | 8 | 3 | 4 | 7 | 6 | 1 | 5 | 15 | 2 | 12 | 16 | 10 | 13 | 14 | 11 |
| 11 | 1 | 2 | 4 | 6 | 9 | 13 | 5 | 8 | 7 | 10 | 11 | 12 | 3 | 16 | 14 | 15 |
| 12 | 16 | 14 | 2 | 3 | 7 | 5 | 10 | 15 | 4 | 6 | 1 | 9 | 11 | 12 | 13 | 8 |
| 13 | 11 | 10 | 2 | 15 | 3 | 6 | 7 | 16 | 9 | 12 | 13 | 8 | 14 | 1 | 5 | 4 |
| 14 | 2 | 3 | 12 | 5 | 1 | 4 | 6 | 7 | 16 | 8 | 9 | 13 | 11 | 10 | 15 | 14 |
| 15 | 1 | 9 | 5 | 3 | 8 | 12 | 13 | 4 | 10 | 6 | 7 | 11 | 2 | 16 | 15 | 14 |
| 16 | 3 | 16 | 1 | 2 | 10 | 4 | 5 | 7 | 8 | 15 | 6 | 9 | 11 | 13 | 12 | 14 |

## 2.2 Hàm thích nghi

Khi tạo ra một cá thể, cần xác định độ thích nghi cho từng cá thể; độ thích nghi ở đây được hiểu như độ sai sót, phạm lỗi của giải pháp đó đối với bài toán Sudoku. Như đã được xác định từ trước, một giải pháp đúng khi và chỉ khi đáp ứng được 3 tiêu chí:

- Mỗi cột chứa số từ 1 – n và không trùng lặp.
- Một hàng chứa số từ 1 – n và không trùng lặp.
- Mỗi khối √n x √n chứa số từ 1 – n và không trùng lặp.

Đối với riêng giải thuật GA, vì cá thể khi tạo ra với ràng buộc phải chứa số từ 1– n ở từng hàng, nên độ thích nghi cá thể được tính bằng cách duyệt qua 2 điều kiện còn lại:

- Thêm số điểm phạm lỗi tương ứng với số lần lặp hơn 1 trong mỗi cột và cũng như thêm 1 điểm đối với số chưa từng xuất hiện trước đó trong cột.
- Thêm số điểm phạm lỗi tương ứng với số lần lặp hơn 1 trong mỗi khối và cũng như thêm 1 điểm đối với số chưa từng xuất hiện trước đó trong khối √n x √n.

Khi sử dụng Backtracking; giải thuật Backtracking tối ưu cao trong vấn đề xử lý Sudoku. Backtracking ở đây được dùng để nhận câu đố Sudoku và tạo ra một mẫu lý tưởng để từ đó có thể tính điểm vi phạm cho các cá thể được sinh ra – đẩy mạnh quá trình tìm kiếm cũng như giữ lại cá thể trội. Độ thích nghi được tính như sau:

- Backtracking thực hiện nhận câu đố và tiến hành giải mã cho ra mẫu đáp án hoàn chỉnh.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

- Tương ứng với mỗi giá trị ô trống trong cá thể, tăng lên 1 điểm đối với giá trị trong ô tương ứng không khớp với mẫu đã cho.

Một khi GA tìm thấy cá thể có độ thích nghi 0 – không có điểm vi phạm thì nhiệm vụ tìm kiếm giải pháp của giải thuật là kết thúc.

## 2.3 Khởi tạo

Kích thước của quần thể cho giải thuật GA này là 1000 cá thể tại mỗi đảo. Mỗi cá thể được khởi tạo bằng cách ngẫu nhiên sao cho tại mỗi đoạn gen đều chứa số từ 1 – n. Ngoài ra, các số cố định trong bài toán (đề bài) bắt buộc giữ cố định và không thể di chuyển ra khỏi vị trí ban đầu; khởi tạo quần thể cũng quan tâm đến ràng buộc này vì đây là điều hiển nhiên.

## 2.4 Lai ghép

Toán tử lai ghép sử dụng 2 hoặc nhiều hơn số cha mẹ để thực hiện tạo ra một hay hai cá thể con, song trong nghiên cứu này, chúng tôi chỉ chọn ra 2 cá thể cha mẹ để tiến hành tạo ra 2 cá thể mới sau mỗi lần lai ghép. Bên cạnh đó, sẽ giữ lại một cá thể con duy nhất để tham gia vào quá trình tiến hóa tiếp theo; cá thể con được chọn là cá thể có độ thích nghi cao hơn cá thể con còn lại sau quá trình lai ghép, bất kể cá thể đó có đột biến hay không.

Cá thể mới được tạo ra thông qua toán tử lai ghép, mỗi dòng được áp dụng một cách tách biệt. Cá thể chỉ tổ hợp lại với cùng một hàng theo thứ tự ban đầu. Toán tử lai ghép được sử dụng trong GA này là lai ghép PMC (Partial Mapped Crossover) [12]. Quá trình này được thực hiện qua các bước sau:

1) Chọn ngẫu nhiên hai điểm cắt trên cá thể cha mẹ.
2) Sao chép nội dung giữa hai điểm cắt từ cá thể cha sang cá thể con 1, và từ cá thể mẹ sang cá thể con 2.
3) Đối với các vị trí còn lại, kiểm tra sự trùng lặp giữa các phần được sao chép. Nếu có thực hiện quá trình tráo đổi để đảm bảo rằng không có sự trùng lặp xảy ra.
4) Đưa ra hai cá thể con mới sau khi hoàn thành lai ghép

## 2.5 Đột biến

Toán tử đột biến thay đổi hay hoán vị vị trí giá trị trong NST để trở thành một cá thể mới khác. Trong GA này tổng cộng có 2 phương pháp đột biến thông dụng trong thuật toán di truyền. Các phương pháp đột biến được đề cập bên dưới:

- Đột biến hoán vị (Swap Mutation [13]): một số gen được chọn ngẫu nhiên trong cá thể và sau đó hoán đổi vị trí với nhau. Điều này có nghĩa là các giá trị gen sẽ được đổi chỗ với nhau một cách ngẫu nhiên, ngoại trừ các giá trị cố định. Đột biến này được áp dụng cho một cá thể, tất cả các hàng của cá thể có tỉ lệ áp dụng với loại đột biến này một cách riêng biệt.
- Đột biến xáo trộn (Scramble Mutation [14]): một đoạn gen liên tiếp trong cá thể được chọn ngẫu nhiên và sau đó xáo trộn vị trí của các giá trị gen một cách ngẫu nhiên, ngoại trừ các giá trị cố định. Đột biến này được áp dụng cho một cá thể, tất cả các hàng của cá thể có tỉ lệ áp dụng với loại đột biến này một cách riêng biệt.

## 2.6 Độ đa dạng của quần thể

Vấn đề đa dạng – duy trì sự đa dạng quần thể khỏe mạnh, có liên quan chặt chẽ đến việc đạt được trạng thái cân bằng hợp lý giữa thăm dò và khai thác [15]. Quần thể càng đa dạng giúp tăng không gian tìm kiếm giải pháp và giảm thiểu tình trạng rơi vào tối ưu hóa cục bộ. Trong nghiên cứu này, độ đa dạng của quần thể được áp dụng cho quần thể riêng biệt.

Độ đa dạng của quần thể được xác định thông qua độ thích các cá thể trong quần thể. Ứng với trên 1/3 số lượng cá thể có cùng một điểm thích nghi, quần thể sẽ được gọi là kém đa dạng và cần tiến hành các toán tử đặc biệt cải thiện độ đa dạng của quần thể.

## 2.7 Chọn lọc cha mẹ

Chọn lọc cha mẹ - chọn hai cá thể trong quần thể để tham gia quá trình lai ghép và đột biến để tạo ra cá thể con thay thế cho một cá thể trong quần thể hiện tại. Chọn lọc cha mẹ được thực hiện theo phương pháp dựa trên độ đa dạng của quần thể tại thời điểm đó; quá trình chọn lọc được đề cập bên dưới:

- Chọn lọc xếp hạng định rõ (Tournament Selection – TS [16]): được sử dụng để chọn ra các cá thể tốt nhất trong một nhóm con nhỏ từ quần thể ban đầu. Trong nghiên cứu này, TS được thực hiện khi quần thể đa dạng; trong quá trình thực hiện sẽ chọn ra 1 nhóm gồm 4 cá thể ngẫu nhiên từ quần thể hiện tại, từ 4 cá thể này tiếp tục chọn ra 2 cá thể có độ thích nghi cao hơn để làm cá thể cha mẹ.
- Chọn lọc ngẫu nhiên (Random Selection – RS [16]): là toán tử chọn lọc đơn giản nhất trong các phương pháp chọn lọc, tại đây mỗi cá thể đều có cùng một xác suất được lựa chọn là 1/n. Trong nghiên cứu này, RS sẽ thực hiện chọn lọc ngẫu nhiên 2 cá thể từ quần thể ban đầu để làm cá thể cha mẹ, việc thực hiện RS dựa trên độ đa dạng của quần thể là kém đa dạng.

## 2.8 Chọn lọc sinh tồn

Lựa chọn sinh tồn thực hiện 2 phương pháp để thay thế cá thể trong quần thể cũ bằng cá thể mới được sinh ra. Sự thay thế này phản ánh đến cách tiếp cận SSGA, cách tiếp cận bên dưới:

- Thay thế cá thể xấu hơn mức trung bình: cá thể được lựa chọn để thay thế với cá thể con bằng cách lựa chọn ngẫu nhiên 1 cá thể trong quần thể cũ với điều kiện độ thích nghi của cá thể kém hơn mức trung bình độ thích nghi của quần thể. Sau khi chọn được cá thể, thực hiện thay thế này bằng cá thể con được sinh ra. Điều này giúp quần thể dần đẩy nhanh quá trình tối ưu giải pháp. Phương pháp này thực hiện khi quần thể đa dạng cao.
- Thay thế cá thể ngẫu nhiên: khi độ đa dạng của quần thể kém, một phương pháp thay thế được thực thi; chọn ngẫu nhiên ra một cá thể trong quần thể [9], sau đó thay thế cá thể này bằng cá thể con được sinh ra.

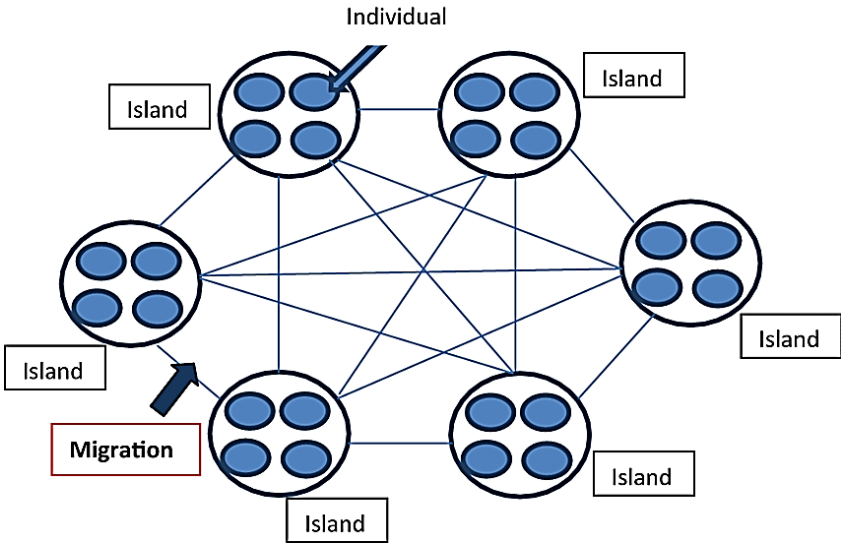
## 2.9 Xác suất lai ghép và đột biến

Xác suất để thực hiện quá trình lai ghép và đột biến như sau:
P(crossover) – Pc = 0.9; P(mutation) – Pm = 0.1, Pc 0.9 tức là trong quá trình lai ghép, 2 cá thể cha mẹ có tỉ lệ tạo ra 2 cá thể mới là 90% cơ hội; nếu rơi vào 10% còn lại, cá thể cha hoặc mẹ có độ thích nghi cao hơn sẽ được chọn làm cá thể con. Trong khi đó Pm mặc định là 0.1, ứng với cá thể mới sau quá trình lai ghép có cơ hội 10% đột biến; ngoài ra, nếu độ đa dạng của quần thể đảo hiện tại kém đa dạng Pm sẽ được đẩy lên 90% tạo ra không gian tìm kiếm mới. Tỉ lệ lựa chọn phương pháp đột biến trong 2 phương pháp là 50%.

## 2.10 Mô hình đảo

Quần thể lớn ban đầu được chia thành một số quần thể nhỏ nhất định, được gọi là "đảo", được gọi một cách tổng thể là "mô hình đảo" trong GA. [17] Mô hình đảo chuyển đổi GA thành một không gian có thể phân tách tuyến tính, làm giảm đáng kể thời gian tính toán. Các tham số quan trọng nhất của các mô hình đảo là khoảng thời gian di chuyển và quy mô di cư. Tại một khoảng thời gian đều đặn trong quá trình thực thi, các đảo trao đổi cá thể của chúng với nhau, đó là khoảng thời gian di chuyển; trong khi số lượng cá thể trao đổi được gọi là kích thước di chuyển.
Trong nghiên cứu này, khoảng cách di cư là 1/6 kích thước quần thể, tương ứng với 6 đảo trong mô hình. Trên mỗi 2 lần chu kỳ nhỏ của đảo, các cá thể tại mỗi đảo sẽ trao đổi với nhau theo hình thức: lựa chọn cá thể kém nhất ở đảo A và thay thế nó bằng cá thể ngẫu nhiên của đảo B.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 4: Mô hình đảo, GA song song với mô hình đảo [18]

Tại mỗi đảo, quá trình tiến hóa được thực hiện một cách độc lập với nhau. Các đảo chỉ trao đổi cá thể qua lại với nhau ở chu kỳ nhất định.

### 2.11 Điều kiện dừng

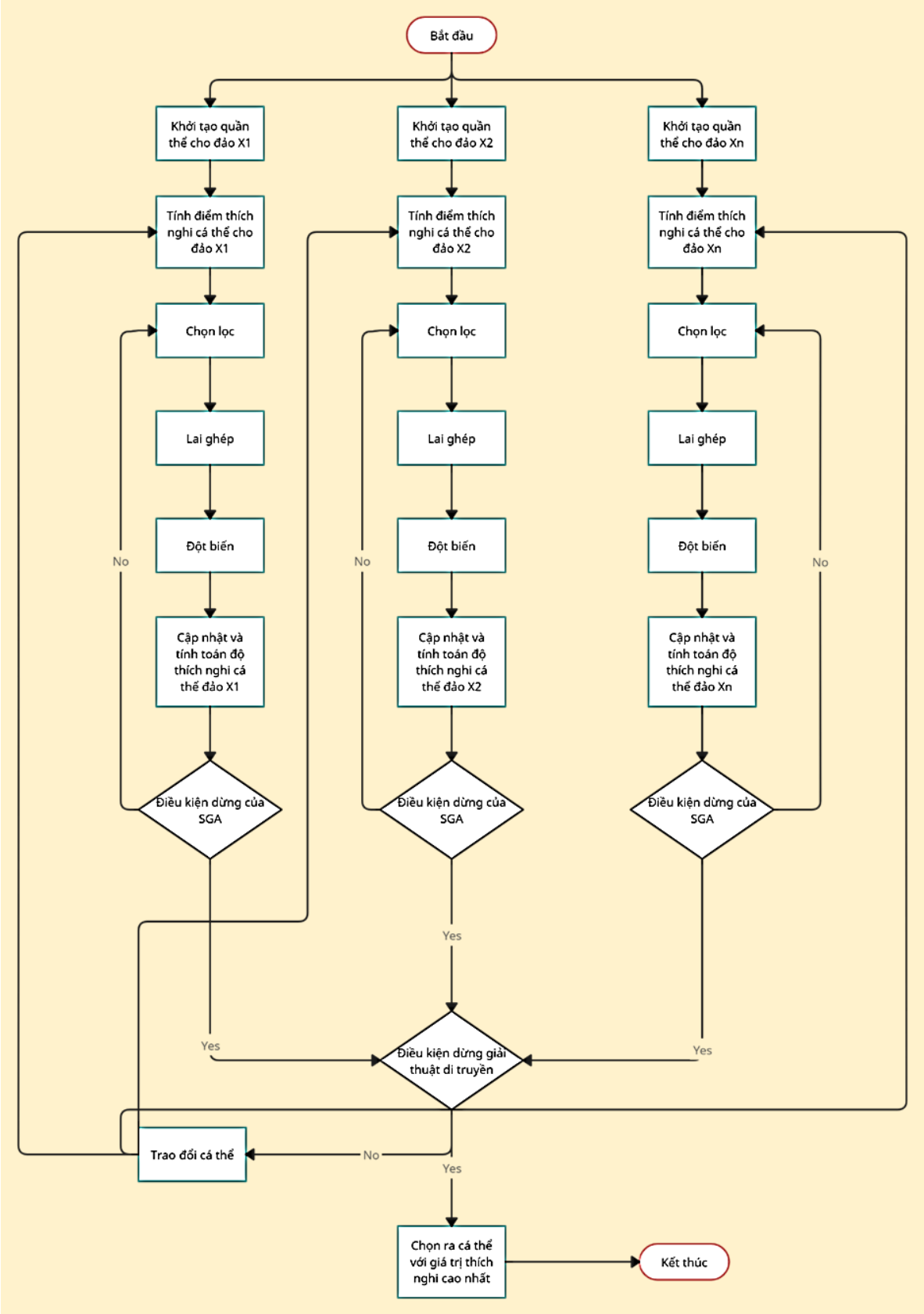
Trong thực nghiệm GA này, điều kiện để dựng quá trình tiến hóa khi và chỉ khi tìm ra giải pháp đúng – tức cá thể có 0 điểm vi phạm hay độ thích nghi bằng 0 (điểm vi phạm càng nhỏ, độ thích nghi càng nhỏ, khả năng cho ra giải pháp đúng càng cao), hoặc kết thúc vòng lặp tiến hóa, vòng lặp mặc định là 50.000 thế hệ đối với cấp độ dễ và trung bình, 300000 thế hệ đối với cấp độ khó.

## 3 THỰC NGHIỆM

### 3.1 Thực nghiệm với giải thuật GAs đa dạng hoá quần thể

Trong nghiên cứu này, chúng tôi đã sử dụng Giải thuật GAs đa dạng hóa quần thể và tiến hành 20 lần thực nghiệm trên các bộ đề Sudoku 9x9 với ba cấp độ khó khác nhau: dễ, trung bình và khó. Kết quả cho thấy giải thuật GAs khá hiệu quả với các bộ đề cấp độ dễ, nhưng có hạn chế đối với các bộ đề cấp độ trung bình và đặc biệt là cấp độ khó.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

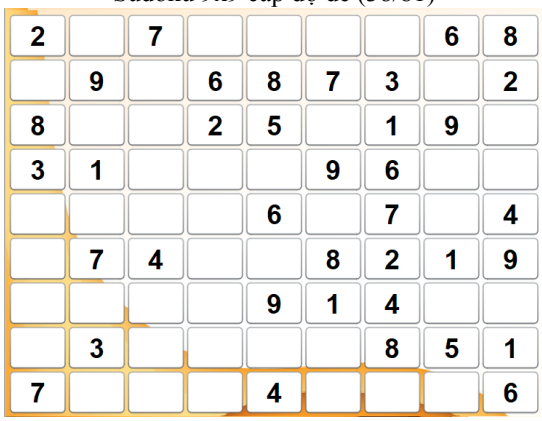



Hình 5: Sơ đồ triển khai giải thuật di truyền đa dạng hóa quần thể


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

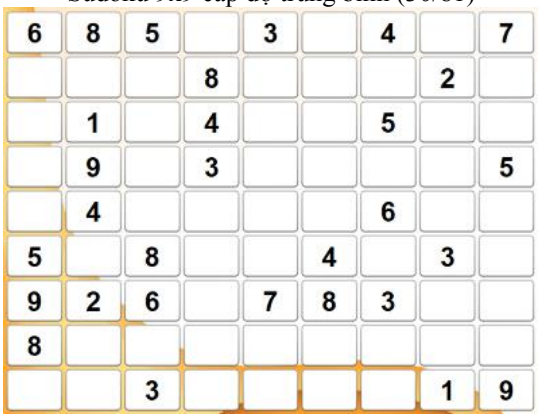
Bảng 5: Minh họa cho 1 số ví dụ chạy ô số Sudoku 9x9 với giải thuật Gas đa dạng hóa quần thể

Sudoku 9x9 cấp độ dễ (38/81)



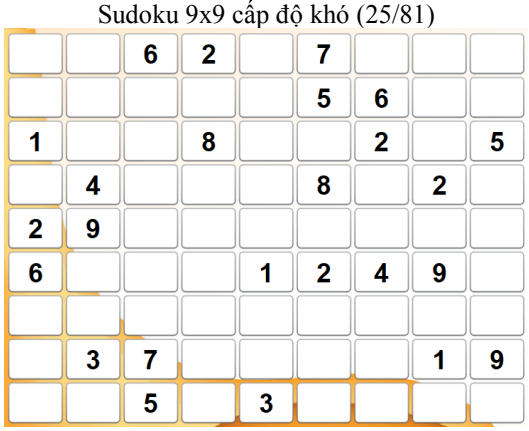

| Lần Test | Thế hệ | Giá trị thích nghi | | Đảo xuất hiện | Vị trí cá thể tốt nhất | Thời gian thực hiện (giây) |
|---|---|---|---|---|---|---|
| | | Cá thể xấu nhất | Cá thể tốt nhất | | | |
| 1 | 4772 | 60 | 0 | 3 | 677/1000 | 5.95 |
| 2 | 5005 | 59 | 0 | 4 | 840/1000 | 3.83 |
| 3 | 5223 | 59 | 0 | 2 | 315/1000 | 3.61 |
| 4 | 4944 | 62 | 0 | 4 | 431/1000 | 3.483 |
| 5 | 5224 | 84 | 0 | 2 | 260/1000 | 3.112 |
| 6 | 4944 | 76 | 0 | 4 | 920/1000 | 2.925 |
| 7 | 5779 | 50 | 0 | 4 | 184/1000 | 3.701 |
| 8 | 5084 | 62 | 0 | 6 | 658/1000 | 3.556 |
| 9 | 5321 | 59 | 0 | 5 | 108/1000 | 3.313 |
| 10 | 5460 | 50 | 0 | 1 | 876/1000 | 3.836 |
| 11 | 5157 | 57 | 0 | 5 | 752/1000 | 3.23 |
| 12 | 4829 | 60 | 0 | 5 | 770/1000 | 3.476 |
| 13 | 5097 | 56 | 0 | 3 | 424/1000 | 3.67 |
| 14 | 5335 | 53 | 0 | 4 | 90/1000 | 3.757 |
| 15 | 4955 | 63 | 0 | 2 | 463/1000 | 3.127 |
| 16 | 4511 | 73 | 0 | 5 | 900/1000 | 2.811 |
| 17 | 5800 | 48 | 0 | 6 | 854/1000 | 3.628 |
| 18 | 5178 | 54 | 0 | 5 | 592/1000 | 3.86 |
| 19 | 5331 | 63 | 0 | 4 | 293/1000 | 3.626 |
| 20 | 5257 | 56 | 0 | 3 | 495/1000 | 3.711 |

Sudoku 9x9 cấp độ trung bình (30/81)




| Lần Test | Thế hệ | Giá trị thích nghi | | Đảo xuất hiện | Vị trí cá thể tốt nhất | Thời gian thực hiện (giây) |
|---|---|---|---|---|---|---|
| | | Cá thể xấu nhất | Cá thể tốt nhất | | | |
| 1 | 10930 | 44 | 0 | 6 | 350/1000 | 2.613 |
| 2 | 13647 | 80 | 0 | 2 | 322/1000 | 3.352 |
| 3 | 9835 | 56 | 0 | 3 | 794/1000 | 2.61 |
| 4 | 9946 | 68 | 0 | 5 | 768/1000 | 2.215 |
| 5 | **50000** | 105 | 6 | - | - | Cực đại |
| 6 | 11579 | 42 | 0 | 1 | 660/1000 | 2.505 |
| 7 | 9847 | 70 | 0 | 3 | 914/1000 | 2.220 |
| 8 | **50000** | 105 | 12 | - | - | Cực đại |
| 9 | 9351 | 81 | 0 | 5 | 18/1000 | 3.31 |
| 10 | 9460 | 50 | 0 | 1 | 14/1000 | 3.86 |
| 11 | 10151 | 57 | 0 | 5 | 75/1000 | 2.234 |
| 12 | 9827 | 61 | 0 | 5 | 700/1000 | 3.476 |
| 13 | 5097 | 56 | 0 | 3 | 201/1000 | 3.67 |
| 14 | **50000** | 86 | 6 | - | - | Cực đại |
| 15 | 9905 | 63 | 0 | 2 | 36/1000 | 3.12 |
| 16 | 9511 | 73 | 0 | 5 | 840/1000 | 2.81 |
| 17 | 10800 | 48 | 0 | 6 | 800/1000 | 3.62 |
| 18 | 9150 | 54 | 0 | 5 | 457/1000 | 3.86 |
| 19 | 9987 | 63 | 0 | 4 | 154/1000 | 3.626 |
| 20 | 9150 | 50 | 0 | 3 | 710/1000 | 3.71 |

Sudoku 9x9 cấp độ khó (25/81)

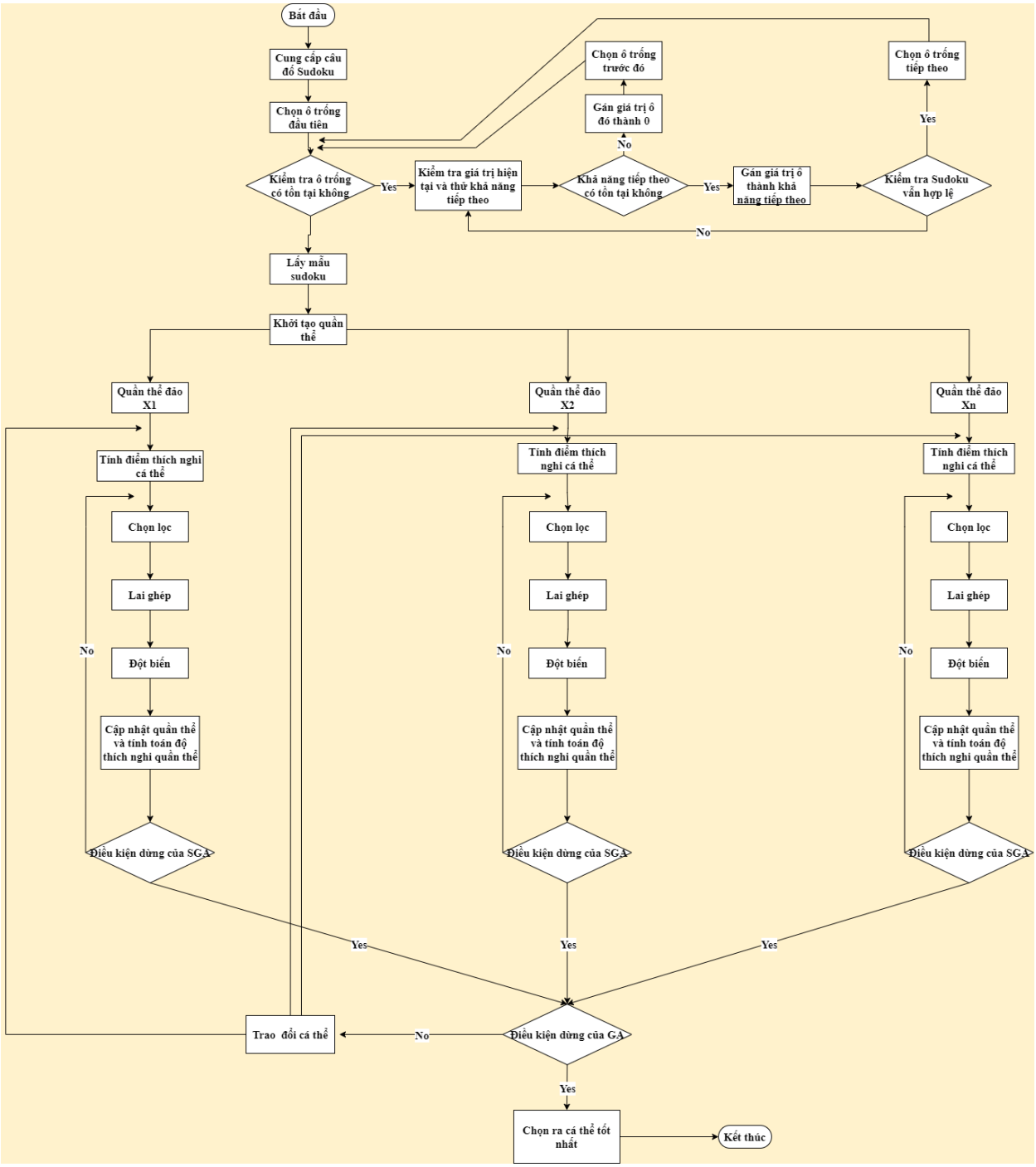



| Lần Test | Thế hệ | Giá trị thích nghi | | Đảo xuất hiện | Vị trí cá thể tốt nhất | Thời gian thực hiện (giây) |
|---|---|---|---|---|---|---|
| | | Cá thể xấu nhất | Cá thể tốt nhất | | | |
| 1 | **300000** | 111 | 6 | - | - | Cực đại |
| 2 | **300000** | 111 | 6 | - | - | Cực đại |
| 3 | **300000** | 105 | 6 | - | - | Cực đại |
| 4 | **300000** | 105 | 6 | - | - | Cực đại |
| 5 | **279681/300000** | 105 | 0 | 1 | 541/1000 | 178.942 |
| 6 | **300000** | 111 | 6 | - | - | Cực đại |
| 7 | **300000** | 105 | 6 | - | - | Cực đại |
| 8 | **300000** | 105 | 6 | - | - | Cực đại |
| 9 | **300000** | 111 | 6 | - | - | Cực đại |
| 10 | **300000** | 105 | 6 | - | - | Cực đại |
| 11 | **300000** | 111 | 6 | - | - | Cực đại |
| 12 | **255011/300000** | 99 | 0 | 1 | 420/1000 | 173.451 |
| 13 | **300000** | 111 | 6 | - | - | Cực đại |
| 14 | **300000** | 111 | 6 | - | - | Cực đại |
| 15 | **300000** | 105 | 6 | - | - | Cực đại |
| 16 | **194051/300000** | 99 | 0 | 1 | 851/1000 | 163.441 |
| 17 | **300000** | 105 | 6 | - | - | Cực đại |
| 18 | **300000** | 111 | 6 | - | - | Cực đại |
| 19 | **300000** | 105 | 6 | - | - | Cực đại |
| 20 | **212012/300000** | 99 | 0 | 1 | 987/1000 | 170.410 |

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

### 3.2 Thực nghiệm GAs đa dạng hoá quần thể kết hợp Backtracking

Nghiên cứu tiến hành việc kết hợp giải thuật di truyền đa dạng hoá quần thể kết hợp với Backtracking. Chúng tôi thực nghiệm trên các bộ số Sudoku 9x9, 16x16 với các cấp độ: dễ, trung bình và khó, thực nghiệm cho thấy việc kết hợp hai phương pháp cho kết quả khả quan.



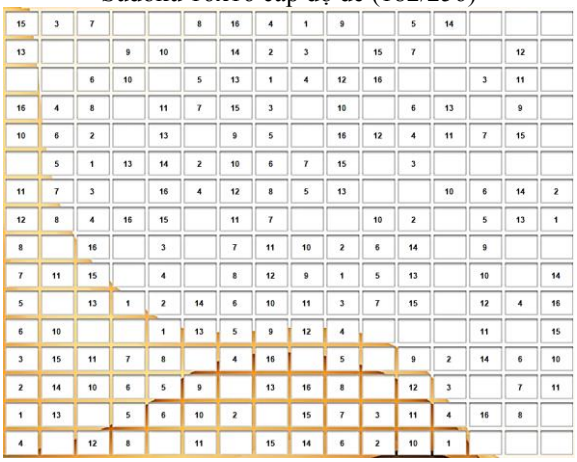

Hình 6: Sơ đồ triển khai giải thuật di truyền đa dạng hóa quần thể kết hợp backtracking


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Dưới đây là kết quả so sánh của bộ đề Sudoku 16x16 với các cấp độ qua các lần thực nghiệm:

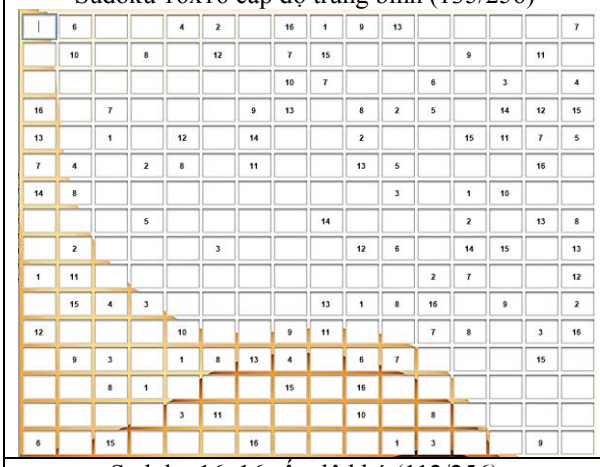
Bảng 6: So sánh các lần kiểm thử trên bộ đề sudoku 16x16

Sudoku 16x16 cấp độ dễ (182/256)



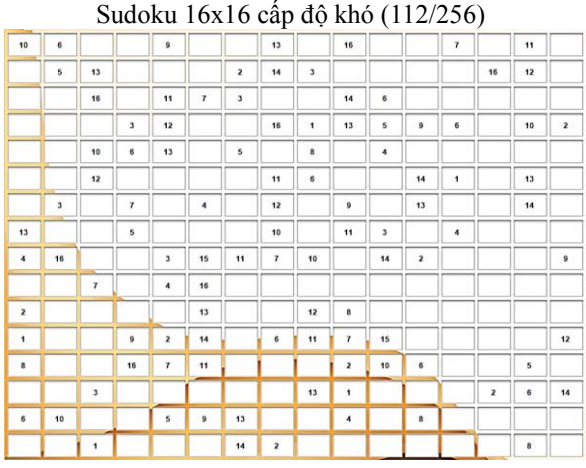
| Lần thử | GA | | | GA with BA | | |
|---|---|---|---|---|---|---|
| | Gen tốt nhất | Gen xấu nhất | Chu kỳ thực hiện | Gen tốt nhất | Gen xấu nhất | Chu kỳ thực hiện |
| 1 | 0 | 95 | 6428 | 0 | 27 | 5296 |
| 2 | 0 | 95 | 6440 | 0 | 29 | 4897 |
| 3 | 0 | 110 | 6039 | 0 | 28 | 4654 |
| 4 | 0 | 95 | 6541 | 0 | 31 | 5320 |
| 5 | 0 | 105 | 6245 | 0 | 26 | 4986 |
| 6 | 0 | 110 | 6024 | 0 | 25 | 4814 |
| 7 | 0 | 89 | 6841 | 0 | 20 | 5352 |
| 8 | 0 | 99 | 6571 | 0 | 26 | 5287 |
| 9 | 0 | 105 | 6345 | 0 | 25 | 5305 |
| 10 | 0 | 95 | 6214 | 0 | 25 | 5254 |

Sudoku 16x16 cấp độ trung bình (135/256)



| Lần thử | GA | | | GA with BA | | |
|---|---|---|---|---|---|---|
| | Gen tốt nhất | Gen xấu nhất | Chu kỳ thực hiện | Gen tốt nhất | Gen xấu nhất | Chu kỳ thực hiện |
| 1 | 54 | 295 | 50000 | 0 | 27 | 15209 |
| 2 | 67 | 316 | 50000 | 0 | 29 | 14897 |
| 3 | 49 | 275 | 50000 | 0 | 28 | 14657 |
| 4 | 31 | 209 | 50000 | 0 | 31 | 15324 |
| 5 | 54 | 286 | 50000 | 0 | 26 | 14985 |
| 6 | 61 | 301 | 50000 | 0 | 35 | 14872 |
| 7 | 19 | 184 | 50000 | 0 | 28 | 13540 |
| 8 | 45 | 266 | 50000 | 0 | 29 | 14781 |
| 9 | 54 | 245 | 50000 | 0 | 31 | 14501 |
| 10 | 85 | 313 | 50000 | 0 | 30 | 15640 |

Sudoku 16x16 cấp độ khó (112/256)



| Lần thử | GA | | | GA with BA | | |
|---|---|---|---|---|---|---|
| | Gen tốt nhất | Gen xấu nhất | Chu kỳ thực hiện | Gen tốt nhất | Gen xấu nhất | Chu kỳ thực hiện |
| 1 | 77 | 316 | 50000 | 0 | 44 | 15295 |
| 2 | 61 | 324 | 50000 | 0 | 41 | 16887 |
| 3 | 75 | 298 | 50000 | 0 | 39 | 16977 |
| 4 | 65 | 331 | 50000 | 0 | 55 | 15644 |
| 5 | 61 | 316 | 50000 | 0 | 51 | 15985 |
| 6 | 77 | 310 | 50000 | 0 | 44 | 15872 |
| 7 | 77 | 328 | 50000 | 0 | 44 | 15541 |
| 8 | 64 | 384 | 50000 | 0 | 41 | 16010 |
| 9 | 69 | 340 | 50000 | 0 | 43 | 17501 |
| 10 | 70 | 343 | 50000 | 0 | 45 | 16410 |

Kết quả của các lần thực nghiệm đã cung cấp một cái nhìn rõ ràng và đáng tin cậy về khả năng giải quyết bài toán Sudoku bằng phương pháp GA. Chúng tôi đã thực hiện phân tích và đánh giá kết quả một cách chi tiết, đưa ra số liệu và thống kê để chứng minh tính khả quan và hạn chế của giải thuật.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Đặc biệt, chúng tôi đã quan sát được rằng số lượng gene xấu trong giải thuật có tỉ lệ thuận với cấp độ khó của bài toán Sudoku, cho dù là GA đơn thuần hay GA kết hợp Backtracking (GA+BA). Tuy nhiên, phương pháp GA+BA đã cho thấy khả năng tối ưu cao hơn và đạt được kết quả tuyệt vời hơn so với GA đơn thuần. Với các bộ đề có cấp độ dễ, phương pháp GA+BA chỉ có khoảng 20% số lượng gene xấu so với GA. Với bộ đề cấp độ trung bình, tỷ lệ này là xấp xỉ 10%, và với bộ đề cấp độ khó, tỷ lệ này cũng xấp xỉ 13.5%.
Những kết quả này thể hiện rõ ràng sự hiệu quả và tính khả thi của việc kết hợp GA và Backtracking trong việc giải quyết bài toán Sudoku, đồng thời cung cấp thông tin quan trọng về việc điều chỉnh các yếu tố gene xấu trong thuật toán để tối ưu hóa hiệu suất giải thuật
Chương trình của dự án được thực thi trên nền tảng Java, với công nghệ giao diện Java Swing, kèm với các thư viện bên ngoài hỗ trợ quá trình đọc dữ liệu và hiển thị giao diện. Các thư viện sử dụng: jfreechart, jcommon, json-simple được bao gồm trong mã nguồn. Source code thực thi có sẵn trên URL: https://github.com/TranManhThin/Solving_Sudoku_with_GA_and_BA
Dự án cũng cung cấp một phiên bản khác của chương trình, xử lý bài toán hoàn toàn bằng GA mà không có sự tham gia của Backtracking. Hỗ trợ quá trình mở rộng giải thuật dễ dàng trong tương lai.

## 3.3 Kết quả

Dự án của chúng tôi đã tiến hành thử nghiệm Genetic Algorithm (GA) để giải bài toán Sudoku 9x9, 16x16, 25x25. Tuy nhiên, chúng tôi gặp khó khăn khi giải các bài toán Sudoku khó từ bộ ô số 9x9 vì GA thường tốn nhiều thời gian và hội tụ sớm, đặc biệt không có kết quả với việc giải các bộ ô số 16x16 cấp độ khó, 25x25 cấp độ dễ, trung bình và khó. Để đạt được kết quả tối ưu, chúng tôi đã phải tăng cường số lần lặp hoặc thực hiện nhiều lần "bắt đầu lại" để tìm kiếm giải pháp tốt nhất.
Tuy nhiên, chúng tôi đã phát hiện rằng việc kết hợp GA và Backtracking đem lại hiệu quả cao hơn. Thuật toán Backtracking giúp khắc phục hạn chế của GA khi giải bài toán Sudoku, trong khi GA vẫn hỗ trợ quá trình tối ưu hóa. Sự kết hợp giữa hai phương pháp này đã mang lại giải pháp tốt cho bài toán Sudoku trong thời gian ngắn và đạt được độ tối ưu cao.
Mô hình đảo của chúng tôi cho phép song song hóa việc tối ưu hóa, giúp tăng tốc quá trình giải quyết. Các cá thể đúng được chọn lọc dễ dàng và tham gia vào quá trình tiến hóa, từ đó chúng tôi đã đạt được kết quả ấn tượng trên các bộ đề Sudoku 9x9, như được thể hiện qua Bảng 7 và Bảng 8.

Bảng 7: Xử lý với GA

| Loại đề | Chu kỳ tiến hóa trung bình |
|---|---|
| Dễ | 3452.6 |
| Trung bình | 9109.5 |
| Khó | - |

Bảng 8: Xử lý với GA kết hợp BA

| Loại đề | Chu kỳ tiến hóa trung bình |
|---|---|
| Dễ | 3059.7 |
| Trung bình | 3826.5 |
| Khó | 4378.4 |

Tóm lại, kết hợp GA và Backtracking đã cải thiện hiệu suất giải bài toán Sudoku, giúp đạt được kết quả tối ưu trong thời gian ngắn hơn và vượt qua những khó khăn mà hai phương pháp này gặp riêng lẻ. Từ kết quả thực nghiệm cho thấy việc kết hợp cho kết quả khả quan với bộ ô số 16x16 các cấp độ dễ, trung bình và

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

khó. Hiện tại kết quả nghiên cứu vẫn chưa có dấu hiệu khả quan với bộ ô số 25x25 và trong tương lai chúng tôi sẽ tiếp tục nghiên cứu các hướng tiếp cận cải tiến khác để đi sâu vào giải quyết vấn đề này.

### 3.4 Kết luận

GAs là một giải thuật tối ưu phổ biến và bền vững được ứng dụng rộng rãi trong nhiều bài toán tối ưu thực tế. Mặc dù không đảm bảo tối ưu tuyệt đối, GAs đã chứng minh khả năng tìm kiếm các giải pháp chấp nhận được trong nhiều tình huống.

Trong bài toán Sudoku, GAs không đảm bảo tìm ra lời giải hoàn hảo, nhưng kết quả của việc kết hợp GAs với các thuật giải khác và tối ưu hóa đa mục tiêu đem lại triển vọng hứa hẹn. Kết hợp này giúp cải thiện khả năng giải quyết Sudoku một cách hiệu quả hơn. Ngoài ra, một điểm mạnh của GAs là khả năng linh hoạt kết hợp với các thuật giải chuyên biệt khác hoặc hỗ trợ qua lại với các giải thuật khác. Sự kết hợp này nhằm giảm thiểu khuyết điểm cơ bản của mỗi giải thuật tham gia, tạo nên một phương pháp toàn diện và mạnh mẽ hơn trong việc giải quyết các bài toán phức tạp.

Khi kết hợp GAs với phương pháp Backtracking, chúng ta có cơ hội tận dụng những ưu điểm của cả hai phương pháp. GAs có thể tìm kiếm các giải pháp gần tối ưu, trong khi Backtracking có thể xác nhận tính chính xác của các lời giải. Sự kết hợp này giúp giải quyết các bài toán phức tạp một cách hiệu quả, cân nhắc giữa tính đa dạng và tính chính xác trong quá trình tìm kiếm lời giải. Tóm lại, kết hợp giữa GAs và Backtracking là một phương pháp mạnh mẽ và đa tiện ích để giải quyết các bài toán phức tạp, tận dụng lợi thế của từng phương pháp và hạn chế nhược điểm của chúng.

## 4 TÀI LIỆU THAM KHẢO


1. Rémi Bédard-Couture, Nawwaf N. Kharma, "Playing Iterated Rock-Paper-Scissors with an Evolutionary Algorithm," in 11th International Conference on Evolutionary Computation Theory and Applications, 2019.
2. Kumar Chellapilla, David Fogel, "Evolving an Expert Checkers Playing Program without Using Human Expertise," IEEE Transactions on Evolutionary Computation, vol. 5, no. 4, pp. 422-428, 2001.
3. Habiboghli, Ali; Jalali, Tayebeh, "A Solution to the N-Queens Problem Using Biogeography-Based Optimization," 2017.
4. Wikipedia, "Sudoku," [Online]. Available: http://en.wikipedia.org/wiki/Sudoku.
5. Stuart Russell and Peter Norvig, Artificial Intelligence: A model approach, 2009.
6. N. Khandelwal, "Introduction to Backtracking," [Online]. Available: https://www.shiksha.com/online-courses/articles/introduction-to-backtracking/.
7. Eiben and Smith. Springer-Verlag, Introduction to Evolutionary Computing, 2007.
8. Trần Kim Hương, Nguyễn Thị Ngọc Chi, "GIẢI THUẬT DI TRUYỀN (GAs) VÀ CÁC ỨNG DỤNG," in Hội Nghị Nckh Khoa Sp Toán-Tin, 2015.
9. A. Laver, "Topic 10 Evolutionary Programming, Genetic Algorithms," 2015.
10. Timo Mantere and Janne Koljonen, "Solving and Rating Sudoku Puzzles with Genetic Algorithms," Evolutionary Computation, 2007.
11. Weiss, John M., "Genetic Algorithms and Sudoku," 2009.
12. Ahmed Lahjouji El Idrissi, Chakir Tajani, Mohamed Sabbane, "New crossover operator for genetic algorithm to resolve the fixed charge transportation problem," 2017.
13. Shih-Hsin Chen, Mu-Chung Chen, "Operators of the Two-Part Encoding Genetic Algorithm in Solving the Multiple Traveling Salesmen Problem," 2011.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

4. S. M. Anwer, "Solving the problem of balancing single model assembly lines using Meta heuristic Algorithms," 2020.
5. Brian Mc Ginley, John Maher, Colm O' Riordan, F. Morgan, "Maintaining Healthy Population Diversity Using Adaptive Crossover, Mutation, and Selection," 2011.
6. Alaa Khamis, PhD, SMIEEE, "Genetic Algorithms," in Computational Intelligence/Intelligent Systems Design, Canada, 2016.
7. Darrell Whitley , Soraya Rana , Robert B. Heckendorn, "The Island Model Genetic Algorithm: On Separability, Population Size and Convergence," Journal of Computing and Information Technology, pp. 33-47, 1998.
8. Saroja Subbaraj, Revathi T., "Hybrid dual-objective parallel genetic algorithm for heterogeneous multiprocessor scheduling," 2020.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F032**

# ỨNG DỤNG XỬ LÝ SONG SONG NHÂN MA TRẬN LỚN ĐỂ MÃ HÓA CÔNG KHAI TRUYỀN THÔNG ĐIỆP VỚI HỆ MÃ HÓA MCELIECE

HUỲNH DUY QUANG [1], TRẦN XUÂN THẮNG [1], ĐỖ HÀ PHƯƠNG[1*]

[1]*Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành Phố Hồ Chí Minh,*
[*] *dohaphuong@iuh.edu.vn*

**Tóm tắt.** Trong bài báo này, dựa trên nghiên cứu về tính toán song song và bảo mật trong truyền dữ liệu, chúng tôi sử dụng thư viện Joblib để thực hiện phép nhân ma trận lớn với tính song song cho truyền dữ liệu với hệ mã hóa công khai McEliece. Kết quả đã được thực hiện mã hóa dữ liệu cho nguồn gửi và phục hồi dữ liệu tại nguồn nhận.

**Từ khóa:** McEliece, Joblib, xử lý song song, mã hóa công khai, ma trận song song

# THE APPLICATION OF PARALLEL PROCESSING WITH LARGE MATRIX MULTIPCATION FOR PUBLIC KEY MESSAGE ENCRYPTION USING MCELIECE CRYPTOSYSTEM

**Abstract.** In this paper, based on the research on parallel computation and security in data transmission, we use Joblib library to implement large matrix multiplication with application parallelism for data transmission with the system McEliece public encryption. The result has been done to encrypt data for the sending source and recover the data at the receiving source.

**Keywords**: Mceliece, Joblib, parallel processing, public encryption, parallel matrix.

## 1 GIỚI THIỆU VỀ HỆ MÃ HÓA MCELIECE

### 1.1 Tổng quan về hệ mã hóa McEliece:

Hệ mật mã công khai McEliece là một hệ mật mã khóa công khai đầu tiên dựa trên lý thuyết mã hóa đại số, được giới thiệu năm 1978 [1]. An ninh của hệ mật mã này dựa trên độ khó của bài toán giải mã theo Syndrome và đã được chứng minh là bài toán NP-complete (Nondeterministic Polymonial time) đầy đủ [2]. Trên thực tế, khác với các hệ thống khai thác hệ số nguyên hoặc logarit rời rạc, nó dựa vào độ khó của việc giải mã mã khối tuyến tính mà không theo bất kỳ cấu trúc cụ thể nào [3].

Hệ thống mật mã McEliece ban đầu sử dụng ma trận trình tạo của mã Goppa nhị phân làm khóa riêng, khai thác ma trận biến đổi dày đặc và ma trận hoán vị để ngụy trang khóa bí mật thành khóa công khai. Nó đã chống lại việc phân tích mật mã trong hơn ba mươi năm, cho đến nay vẫn không có cuộc tấn công đa thức nào vào hệ thống này. Tuy nhiên, sức mạnh tính toán ngày càng tăng lên và sự sẵn sàng của các quy trình tấn công được tối ưu hóa khiến hệ mã cần cập nhật các tham số ban đầu của nó [4].

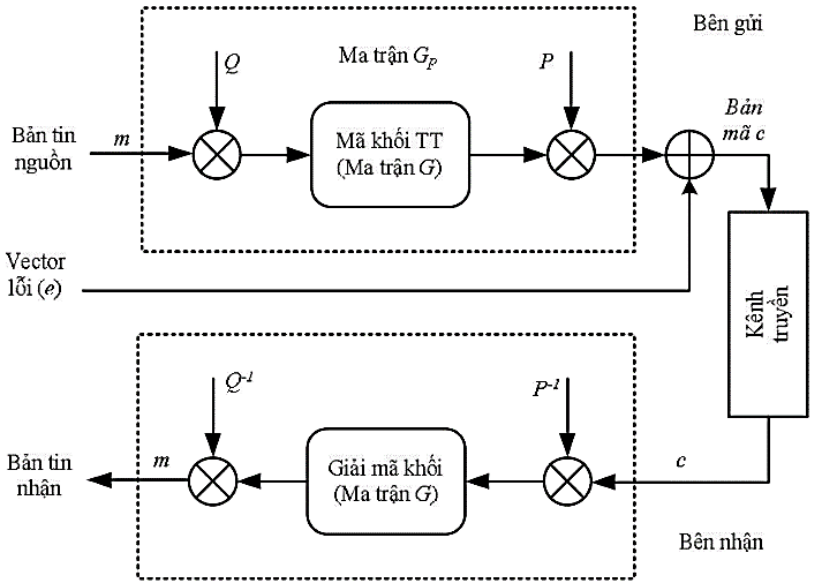
Vì vậy, hệ mật này được xếp vào nhóm mật mã sau lượng tử và những năm gần đây đã được cộng đồng các nhà mật mã học nghiên cứu rộng rãi. Ưu điểm nổi bật của hệ mật mã McEliece là tính bảo mật cao, thời gian thực hiện mã hoá và giải mã nhanh, yêu cầu thiết bị thực hiện đơn giản [5].

Tuy nhiên, hệ mật mã này chưa được đưa vào ứng dụng trong thực tế xuất phát từ nhược điểm cơ bản của nó là tỷ lệ mã hóa thấp (~1/2), kích thước khóa lớn (1024× 524 bit đối với hệ mật đề xuất ban đầu) do đó đòi hỏi dung lượng bộ nhớ lớn.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

## 1.2 Quá trình thực hiện mã hóa của hệ mã hóa McEliece

Dưới đây là sơ đồ hệ mật mã đầu tiên sử dụng tính ngẫu nhiên trong mã hóa. Thuật toán dựa trên độ khó của giải mã khối tuyến tính. Thuật toán ban đầu sử dụng mã nhị phân Goppa, dễ dàng trong việc giải mã nhờ thuật toán của Patterson [6]. Khóa công khai thu được từ khóa mật bằng cách che giấu từ mã đã chọn giống như một từ mã tuyến tính. Để thực hiện, ma trận sinh G của mã nhị phân được xáo trộn với ma trận khả nghịch Q và ma trận hoán vị P.




Hình 1. Sơ đồ khối thuật toán McEliece

Hệ mật mã McEliece có 3 thuật toán: thuật toán tạo khóa, nhằm tạo khóa công khai và khóa bí mật; thuật toán mã hóa, nhằm mã hóa thông tin đầu ra cho bên nhận và cuối cùng là thuật toán giải mã, nhằm giải mã thông tin đầu vào từ người gửi. Trong đó, bản tin nguồn được biểu diễn ở dạng một chuỗi số nhị phân được chia thành các khối con ký hiệu là m có độ dài là k bit. Các thuật toán của hệ mật được thực hiện như sau [1]:

**Quá trình tạo khóa:** chọn một mã tuyến tính nhị phân C có khả năng sửa được t lỗi. Mã Goppa được đặc trưng bởi ma trận sinh G kích thước k×n và có khả năng sửa được một vector lỗi ngẫu nhiên dài n bit có trọng số nhỏ hơn hoặc bằng t:

Bước 1: tạo một ma trận khả nghịch Q kích thước k×k, có ma trận nghịch đảo là $Q^{-1}$.

Bước 2: tạo một ma trận hoán vị P kích thước n×n, có ma trận nghịch đảo là $P^{-1}$

Bước 3: tính toán ma trận Gp = Q*G*P kích thước k×n (1)

Bước 4: lưu trữ khóa công khai (Gp, t) và khóa bí mật (G, Q, P).

**Quá trình mã hóa:** quá trình mã hoá một bản tin trên hệ mật mã McEliece được thực hiện như sau: bên nhận muốn nhận được bản tin được mã hoá bằng thuật toán McEliece, bên gửi sẽ sử dụng khoá công khai Gp lên bản tin m để tạo thành bản mã và gửi đến bên nhận thông qua kênh truyền, các bước mã hóa được thực hiện như sau:

Bước 1: biểu diễn bản tin m ở dạng một chuỗi nhị phân có độ dài k bit.

Bước 2: tạo một vector e ngẫu nhiên có độ dài n và có trọng số nhỏ hơn hoặc bằng t

Bước 3: tính toán bản mã c sau đó gửi cho bên nhận: c = m*Gp + e (2)

**Quá trình giải mã:** sau khi bên nhận đã nhận được bản mã từ bên gửi qua kênh truyền, bên nhận sẽ thực hiện giải mã như sau:

Bước 1: tính phép toán $c*P^{-1}$:

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

$$c*P^{-1} = (m*Gp + e)*P^{-1}$$
$$= m(G*Q*P)*P\text{-}1 + e*P^{-1}$$
$$= m*Q*G + e*P^{-1} \quad (3)$$

Bước 2: sử dụng thuật toán giải mã sửa lỗi đối với c*P-1 để tìm được m*Q:

$$m' = m*Q \quad (4)$$

Bước 3: xác định bản tin m:

$$m = m'*Q^{-1} = (m*Q)Q^{-1} \quad (5)$$

Ta có $c*P^{-1} = m*Q*G + e*P^{-1}$ và P là ma trận hoán vị nên $e*P^{-1}$ có trọng số lớn nhất là t. Mã Goppa Gp có thể sửa được t lỗi và từ mã m*Q*G có thể sửa được t lỗi nhờ thuật toán Patterson hoặc sử dụng các thuật toán khác. Do đó ta sẽ tính được từ mã m' = m*Q. Để lấy bản tin gốc ta nhân m' với ma trận nghịch đảo của Q ta có $m'*Q^{-1} = m$, đây chính là bản tin gốc ban đầu.

Thông thường để đảm bảo độ mật cao, hệ mật mã McEliece yêu cầu kích thước khóa lên tới 1024 bit (tương đương với 210), hơn nữa, để khắc phục phương án tấn công theo kiểu vét cạn (tính tất cả các trường hợp có thể có của vector tín hiệu đầu vào), hệ mật mã McEliece yêu cầu kích thước bản tin đầu vào khá lớn ($k \geq 524$).

Những vấn đề này dẫn đến việc đòi hỏi thiết bị mã hoá và giải mã phải có dung lượng bộ nhớ khá lớn, do đó làm chậm thời gian của quá trình xử lý tín hiệu.

Bảng 1: Ưu điểm và nhược điểm của hệ mã hóa McEliece

| **Ưu điểm** | **Nhược điểm** |
|---|---|
| • *Thứ nhất, khả năng chống lại các kỹ thuật tấn công thông thường*: hệ mật mã McEliece sử dụng các tính toán phức tạp trong lý thuyết thông tin và đại số tuyến tính để mã hóa và giải mã thông tin. Điều này làm cho nó khó bị tấn công bởi các kỹ thuật tấn công thông thường.<br>• *Thứ hai*, *không dùng đến phép tính chia*: hệ mật mã McEliece không sử dụng phép tính chia trong quá trình giải mã, điều này giúp cho quá trình giải mã nhanh hơn so với các hệ mật mã khác.<br>• *Thứ ba*, *khóa bí mật có thể được cập nhật*: khóa bí mật của hệ mật mã McEliece có thể được cập nhật một cách đơn giản bằng cách sử dụng các phép biến đổi ma trận. Điều này giúp cho hệ thống có khả năng chống lại các cuộc tấn công trên khóa.<br>• *Thứ tư*, *khóa công khai nhỏ hơn so với các hệ mật mã khác*: kích thước khóa công khai của hệ mật mã McEliece nhỏ hơn so với các hệ mật mã công khai khác như RSA hay ECC. Điều này giúp cho quá trình truyền dữ liệu trên mạng internet được thực hiện một cách nhanh hơn.<br>• *Thứ năm*, *độ bảo mật cao*: hệ mật mã McEliece được xem là một trong những hệ mật mã công khai có độ bảo mật cao nhất hiện | • *Thứ nhất*, *kích thước khóa*: hệ mật mã McEliece yêu cầu kích thước khóa rất lớn. Điều này làm cho việc truyền tải và lưu trữ khóa trở nên khó khăn.<br>• *Thứ hai, kích thước dữ liệu*: kích thước dữ liệu được mã hóa trong hệ mã hóa McEliece cũng rất lớn, vì vậy hệ thống sử dụng hệ mật mã McEliece sẽ tốn nhiều bộ nhớ hơn.<br>• *Thứ ba*, *tấn công tập trung*: hệ mật mã McEliece dễ bị tấn công tập trung, trong đó kẻ tấn công có thể tạo ra nhiều mã hóa sai lệch và phân tích chúng để tìm ra khóa mật. Tuy nhiên, đây là tấn công được coi là khó thực hiện trong thực tế. |

| | |
|---|---|
| nay. Tuy nhiên, điều này còn phụ thuộc vào kích thước khóa bí mật và độ phức tạp của phép tính tìm khóa bí mật. | |

### 1.4 Ứng dụng hệ mã hóa công khai McEliece trong thực tiễn:

Hệ mã hóa công khai McEliece là một hệ mã hóa dựa trên việc sử dụng ma trận ngẫu nhiên và các phép toán ma trận để mã hóa thông điệp. Dưới đây là một số ví dụ về ứng dụng thực tế của hệ mã hóa công khai McEliece:

*Mã hóa dữ liệu trong mạng lưới cảm biến không dây: Hệ mật mã McEliece có thể được sử dụng để bảo vệ dữ liệu truyền qua mạng lưới cảm biến không dây, trong đó các nút mạng lưới có tài nguyên hạn chế. Với tốc độ mã hóa và giải mã nhanh, hệ mật mã McEliece là sự lựa chọn tốt cho các ứng dụng đòi hỏi bảo mật và hiệu suất cao.*

*Mã hóa và chữ ký số:* Hệ mật mã McEliece có thể được sử dụng để bảo vệ các thông tin nhạy cảm như chữ ký số và mã hóa Email, đặc biệt là trong các ứng dụng đòi hỏi bảo mật cao như ngân hàng trực tuyến và giao dịch tài chính.

*Bảo vệ dữ liệu trong các hệ thống đám mây:* Hệ mật mã McEliece có thể được sử dụng để bảo vệ dữ liệu trong các hệ thống đám mây, nơi dữ liệu được lưu trữ trên các máy chủ từ xa. Với tính năng mã hóa và giải mã nhanh, hệ mật mã McEliece là sự lựa chọn tốt để bảo vệ dữ liệu truyền qua mạng.

*Bảo vệ thông tin trong các ứng dụng IoT:* Hệ mật mã McEliece có thể được sử dụng để bảo vệ thông tin trong các ứng dụng IoT (Internet of Things), nơi các thiết bị định vị và cảm biến thông tin được kết nối với mạng.

Với tính năng mã hóa và giải mã nhanh, hệ mật mã McEliece là sự lựa chọn tốt cho các ứng dụng đòi hỏi bảo mật và hiệu suất cao như trong quản lý năng lượng, quản lý nước và các ứng dụng y tế thông minh.

## 2 GIỚI THIỆU VỀ THƯ VIỆN JOBLIB

### 2.1 Thư viện Joblib trong tính toán song song

Thư viện Joblib là một thư viện Python được sử dụng để thực hiện các tác vụ tính toán song song trên nhiều CPU (Central Processing Unit) hoặc máy tính khác nhau. Nhằm sử dụng để đơn giản hóa và tăng tốc quá trình lưu trữ và tải dữ liệu và các đối tượng trong quá trình phân tán tính toán. Nó cung cấp các công cụ linh hoạt để thực hiện xử lý dữ liệu song song, bao gồm sử dụng nhiều luồng (Threads) hoặc các tiến trình (Processes) khác nhau để thực hiện các tác vụ đồng thời.
Joblib cung cấp các công cụ mạnh mẽ để lưu trữ và tải lại các đối tượng Python như mảng numpy, đối tượng Sklearn và các đối tượng phức tạp khác một cách hiệu quả. Nó hỗ trợ các công nghệ nén dữ liệu như Zlib và Blosc để giảm kích thước lưu trữ và tốc độ tải dữ liệu. Thư viện này cũng cho phép tính toán song song và phân tán thông qua việc sử dụng Multiprocessing, giúp tận dụng tối đa các lõi CPU và tăng tốc độ tính toán.
Một trong những tính năng quan trọng của Joblib là khả năng tự động chọn số lượng tiến trình tối ưu để sử dụng, giúp tăng tốc độ tính toán và tránh tình trạng cạnh tranh tài nguyên. Joblib là một thư viện rất mạnh mẽ và hữu ích cho các ứng dụng tính toán khoa học và máy học. Nó giúp tăng tốc độ tính toán và cải thiện hiệu suất của các ứng dụng, đặc biệt là khi xử lý các tập dữ liệu lớn. Joblib thường được sử dụng trong các tác vụ liên quan đến học máy, xử lý dữ liệu lớn và tính toán khoa học trong Python, khi bạn cần lưu trữ và tải lại các đối tượng phức tạp một cách nhanh chóng và hiệu quả.

### 2.1 Tính năng của thư viện Joblib:

*Khả năng xử lý dữ liệu lớn:* "Joblib" là một thư viện mạnh mẽ cho phép xử lý dữ liệu lớn, đặc biệt là khi sử dụng nhiều CPU hoặc máy tính khác nhau để thực hiện các tính toán song song.

*Tự động chọn số lượng tiến trình tối ưu:* "Joblib" có khả năng tự động chọn số lượng tiến trình tối ưu để sử dụng, giúp tăng tốc độ tính toán và tránh tình trạng cạnh tranh tài nguyên.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

*Hỗ trợ nhiều loại tính toán song song:* "Joblib" hỗ trợ nhiều loại tính toán song song, bao gồm sử dụng nhiều luồng hoặc các tiến trình khác nhau để thực hiện các tác vụ tính toán.

*Tương thích với nhiều thư viện Python:* "Joblib" tương thích với nhiều thư viện Python phổ biến khác nhau, bao gồm NumPy, SciPy, Pandas và Scikit-learn.

*Được sử dụng rộng rãi trong các ứng dụng thực tế:* "Joblib" được sử dụng rộng rãi trong các ứng dụng thực tế, bao gồm trong lĩnh vực khoa học dữ liệu, machine learning và deep learning. Sau khi Process đã hoàn thành thực thi, quy trình mới kết thúc và đối tượng Process bị hủy

*Hỗ trợ các loại đầu vào khác nhau:* "Joblib" có thể xử lý các loại đầu vào khác nhau, bao gồm cả các đối tượng Python và các mảng NumPy.

*Tính năng Memory Mapping:* "Joblib" hỗ trợ tính năng Memory Mapping, cho phép lưu trữ các mảng lớn trên đĩa mà không cần tải chúng vào bộ nhớ RAM.

*Gỡ lỗi và lập hồ sơ:* có thể khó xác định phần mã nào mất nhiều thời gian nhất để thực thi khi tạo một ứng dụng lớn có nhiều chức năng. Joblib cung cấp các công cụ đơn giản để định hình hiệu suất mã của bạn, cho phép bạn xác định vị trí và tăng tốc các khu vực chậm nhất trong ứng dụng của mình. [7]

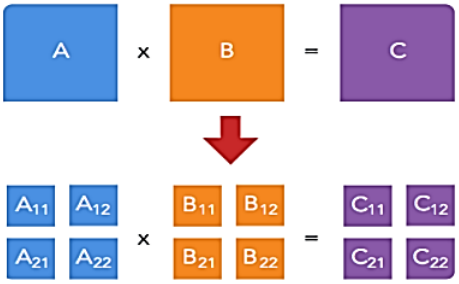
*Khả năng lặp lại:* thực hiện các phép tính giống nhau nhiều lần có thể tốn thời gian khi làm việc với bộ dữ liệu khổng lồ. Để sử dụng lại kết quả tính toán tốn thời gian mà không phải chạy lại mã, Joblib cung cấp một phương tiện để lưu trữ kết quả. Bằng cách này, bạn có thể tiết kiệm thời gian và đảm bảo khả năng lặp lại các kết quả của mình. [7]

*Kiểm tra:* viết các bài kiểm tra là rất quan trọng khi tạo một chương trình phức tạp vì chúng đảm bảo rằng mã thực hiện như dự định. Joblib cung cấp phương tiện để chạy thử nghiệm đồng thời để bạn có thể tìm hiểu nhanh hơn về trạng thái mã của mình. Điều này có thể tăng tốc quá trình phát triển của bạn và cho phép bạn viết và thực hiện nhiều bài kiểm tra hơn trong thời gian ngắn hơn. [7]

*Thử nghiệm:* chạy đồng thời nhiều lần lặp mã có thể hữu ích khi tạo thuật toán mới hoặc thử nghiệm các chiến lược khác nhau. Joblib cung cấp một phương pháp đơn giản để chạy đồng thời nhiều lần lặp lại mã của bạn để bạn có thể nhanh chóng so sánh kết quả của chúng và xác định chiến lược nào hoạt động tốt nhất [7].

## 3 GIỚI THIỆU VỀ THUẬT TOÁN STRASSEN:

Trong đại số tuyến tính, thuật toán Strassen, được đặt theo tên của Volker Strassen, là một thuật toán được sử dụng để nhân ma trận. Nó nhanh hơn thuật toán nhân ma trận tiêu chuẩn cho các ma trận lớn, với độ phức tạp lý thuyết tốt hơn, mặc dù thuật toán đơn giản thường hiệu quả hơn đối với các ma trận nhỏ. Tuy thuật toán Strassen chậm hơn so với các thuật toán nhanh nhất đã được biết đến cho các ma trận vô cùng lớn, nhưng những thuật toán đó không có ích trong thực tế vì chúng chậm hơn rất nhiều đối với các ma trận có kích thước thực tế. Đối với các ma trận nhỏ, cũng tồn tại các thuật toán nhanh hơn. [8]
Thuật toán Strassen hoạt động trên mọi nhóm vòm, chẳng hạn như phép cộng và nhân, tuy nhiên nó không hoạt động trên tất cả các nhóm vòm. Ví dụ như min-plus hoặc đại số Boolean. Trong những trường hợp như vậy, thuật toán thông thường vẫn có thể được sử dụng và được gọi là phép nhân ma trận tổ hợp [8].
Ý tưởng thuật toán của Strassen là áp dụng chia để trị để giải quyết bài toán theo hướng của giải thuật cơ bản trên. Cụ thể là: với mỗi ma trận vuông A, B, C có kích thước n×n, ta chia chúng thành 4 ma trận con, và biểu diễn tích A × B = C theo các ma trận con đó:




$$C_{1,1} = A_{1,1}B_{1,1} + A_{1,2}B_{2,1}$$
$$C_{1,2} = A_{1,1}B_{1,2} + A_{1,2}B_{2,2}$$
$$C_{2,1} = A_{2,1}B_{1,1} + A_{2,2}B_{2,1}$$
$$C_{2,2} = A_{2,1}B_{1,2} + A_{2,2}B_{2,2}$$

**Hình 2.** Ví dụ minh họa phép nhân hai ma trận

Tuy nhiên với cách phân tích này thì chúng ta vẫn cần 8 phép nhân để tính ra ma trận C. Đây là phần quan trọng nhất của vấn đề.
Chúng ta định nghĩa ra các ma trận M mới và biểu diễn chúng dưới dạng các phần tử của C theo M. Bằng cách này, chúng ta chỉ cần 7 phép nhân thay vì 8 như phương pháp cũ. Thực hiện đệ quy quá trình trên cho đến khi ma trận có cấp hai. Độ phức tạp của thuật toán Strassen là O(nlog7) ≈ O(n2.807)

$$M_1 = (A_{1,1} + A_{2,2})(B_{1,1} + B_{2,2})$$
$$M_2 = (A_{2,1} + A_{2,2})B_{1,1}$$
$$M_3 = A_{1,1}(B_{1,2} - B_{2,2})$$
$$M_4 = A_{2,2}(B_{2,1} - B_{1,1})$$
$$M_5 = (A_{1,1} + A_{1,2})B_{2,2}$$
$$M_6 = (A_{1,1} - A_{2,1})(B_{1,1} + B_{1,2})$$
$$M_7 = (A_{1,2} - A_{2,2})(B_{2,1} + B_{2,2})$$

$$C_{1,1} = M_1 + M_4 - M_5 + M_7$$
$$C_{1,2} = M_3 + M_5$$
$$C_{2,1} = M_2 + M_4$$
$$C_{2,2} = M_3 + M_1 - M_2 - M_6$$

**Hình 3.** Định nghĩa các ma trận M mới và biểu diễn ma trận C mới

## 4 SO SÁNH THUẬT TOÁN SONG SONG VÀ THUẬT TOÁN TUẦN TỰ

### 4.1 Thuật toán tuần tự và thuật toán song song

*Thuật toán tuần tự (Sequential Algorithm):* là một loại thuật toán thực hiện các công việc tuần tự trên một tài nguyên xử lý duy nhất và kết quả của bước trước đó được sử dụng cho bước tiếp theo. Thuật toán tuần tự thường dễ hiểu, tuy nhiên nó có thể trở nên chậm chạp nếu bài toán cần phải xử lý một lượng lớn dữ liệu hoặc các bước xử lý phức tạp.

*Thuật toán song song (Parallel Algorithms):* là một lĩnh vực trong khoa học máy tính tập trung vào việc phát triển các thuật toán có khả năng thực hiện nhiều công việc cùng một lúc trên nhiều tài nguyên xử lý song song. Mục tiêu là tận dụng sức mạnh tính toán của các hệ thống song song để cải thiện hiệu suất và tốc độ xử lý của các ứng dụng. Dùng phương pháp thực hiện nhiều tác vụ cùng một lúc trên nhiều CPU hoặc bộ xử lý đồng thời. Thông thường, việc thực hiện các tác vụ đó sẽ được phân tán trên nhiều CPU hoặc bộ xử lý, cho phép mỗi CPU hoặc bộ xử lý đóng góp vào việc thực hiện các tác vụ của thuật toán song song một cách độc lập, tạo ra sự đồng bộ và tăng tốc độ xử lý.

### 4.2 Ưu điểm và nhược điểm của thuật toán song song và tuần tự:

**Bảng 2**. Ưu điểm và nhược điểm của thuật toán song song và thuật toán tuần tự

| | **Thuật toán song song** | **Thuật toán tuần tự** |
|---|---|---|
| **Ưu điểm** | • *Tăng hiệu suất:* Thuật toán song song có thể tận dụng được nhiều tài nguyên tính toán song song, giúp rút ngắn thời gian xử lý và tăng hiệu suất tổng thể. Các công việc có thể | • *Đơn giản:* Thuật toán tuần tự dễ hiểu và triển khai. Không yêu cầu nhiều kiến thức về đồng bộ hoặc quản lý tài nguyên phức tạp. |


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| | | |
|---|---|---|
| | được thực hiện đồng thời, giảm thời gian chờ đợi và tận dụng sức mạnh tính toán của nhiều lõi hoặc máy tính.<br>• *Mở rộng được:* Thuật toán song song cho phép mở rộng dễ dàng trong việc xử lý dữ liệu lớn hoặc các bài toán có kích thước lớn. Bằng cách sử dụng nhiều tài nguyên, nó có khả năng mở rộng tốt và xử lý hiệu quả các tác vụ phức tạp.<br>• *Giải quyết các bài toán phân tán*: Thuật toán song song phù hợp cho việc giải quyết các bài toán phân tán, trong đó các phần khác nhau của công việc có thể được thực hiện trên các nút xử lý riêng biệt. Điều này giúp tăng tính khả dụng và tốc độ xử lý. | • *Dễ debug*: Vì các công việc được thực hiện tuần tự, việc debug và xác định lỗi trong thuật toán tuần tự thường dễ dàng hơn so với các thuật toán song song hoặc phân tán.<br>• *Độ tin cậy cao:* Với thuật toán tuần tự, không có xung đột hoặc đối chọi giữa các công việc cần xử lý đồng thời, điều này giúp tăng độ tin cậy và giảm khả năng xảy ra lỗi. |
| **Nhược điểm** | • *Khó triển khai và debug*: Thuật toán song song đòi hỏi kiến thức đồng bộ hóa, quản lý tài nguyên và phân phối công việc một cách chính xác. Triển khai và debug thuật toán song song có thể phức tạp hơn so với thuật toán tuần tự.<br>• *Xung đột tài nguyên*: Sử dụng nhiều tài nguyên đồng thời có thể dẫn đến xung đột và tranh chấp tài nguyên. Điều này yêu cầu sự đồng bộ hóa tốt giữa các tiến trình và quản lý tài nguyên một cách cẩn thận để tránh xung đột và lỗi.<br>• *Overhead phân tán*: Việc phân phối công việc, đồng bộ hóa và truyền thông giữa các nút trong một hệ thống song song có thể gây ra overhead | • *Hiệu suất kém*: Thuật toán tuần tự thường không tận dụng được tài nguyên máy tính và khả năng xử lý đồng thời của các lõi hoặc tiểu vi xử lý. Điều này dẫn đến hiệu suất thấp và thời gian xử lý kéo dài, đặc biệt khi xử lý các công việc lớn hoặc phức tạp.<br>• *Thiếu sự tận dụng tài nguyên*: Khi một công việc đang được thực hiện, các tài nguyên khác không được sử dụng. Điều này dẫn đến sự lãng phí tài nguyên và giới hạn khả năng tận dụng đồng thời của hệ thống.<br>• *Không thích ứng với các vấn đề phân tán*: Trong một môi trường phân tán, nhiều công việc có thể được thực hiện song song và tận dụng được sức mạnh tính toán của nhiều nút. Trái lại, thuật toán tuần tự không thể tận dụng được lợi ích này và không phù hợp với các bài toán phân tán hoặc song song. |

### 4.3 Thời gian xử lý ma trận của thuật toán song song và thuật toán tuần tự

Trong nghiên cứu này, chúng tôi sẽ thực hiện các thực nghiệm cho xử lý song song và xử lý tuần tự phép nhân các ma trận có kích thước 200x200, 400x400, 600x600, 800x800 và kích thước 1000x1000. Các giá trị phần tử của ma trận được phát sinh ngẫu nhiên.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

| Number of Core | Size of Matrix A | Size of Matrix B | Number of parallel executions [second] | | | |
|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 |
| 2 | 200x200 | 200x200 | 31.019 | 28.787 | 28.846 | 30.09 |
| | 400x400 | 400x400 | 195.266 | 191.897 | 197.755 | 194.474 |
| | 600x600 | 600x600 | 1349.329 | 1367.385 | 1414.69 | 1377.943 |
| | 800x800 | 800x800 | 1415.655 | 1408.976 | 1410.736 | 1408.425 |
| | 1000x1000 | 1000x1000 | 1432.5 | 1427.931 | 1412.616 | 1425.989 |
| 5 | 200x200 | 200x200 | 11.803 | 10.761 | 10.539 | 10.631 |
| | 400x400 | 400x400 | 71.371 | 70.567 | 71.08 | 71.069 |
| | 600x600 | 600x600 | 491.415 | 485.507 | 486.006 | 484.567 |
| | 800x800 | 800x800 | 492.401 | 506.254 | 512.489 | 501.739 |
| | 1000x1000 | 1000x1000 | 516.426 | 521.549 | 535.563 | 528.463 |
| 8 | 200x200 | 200x200 | 6.674 | 6.084 | 6.204 | 6.923 |
| | 400x400 | 400x400 | 45.334 | 45.296 | 45.254 | 45.416 |
| | 600x600 | 600x600 | 316.648 | 317.04 | 316.644 | 316.316 |
| | 800x800 | 800x800 | 327.135 | 329.649 | 329.513 | 327.707 |
| | 1000x1000 | 1000x1000 | 319.981 | 321.582 | 322.925 | 324.352 |
| 12 | 200x200 | 200x200 | 6.596 | 6.174 | 6.184 | 6.353 |
| | 400x400 | 400x400 | 44.644 | 45.199 | 46.091 | 46.374 |
| | 600x600 | 600x600 | 321.682 | 322.358 | 325.857 | 315.682 |
| | 800x800 | 800x800 | 319.731 | 324.205 | 323.848 | 324.11 |
| | 1000x1000 | 1000x1000 | 318.851 | 324.374 | 321.351 | 321.394 |

Hình 4. Thời gian thực hiện nhân hai ma trận của thuật toán song song

Theo thống kê dữ liệu trong bảng 1, ta thấy rằng với số lượng Core cấp phát cho chương trình khác nhau thì thời gian thực hiện nhân hai ma trận cũng sẽ khác nhau: về cơ bản thì với mỗi kích thước ma trận là lũy thừa của 2 (ví dụ kích thước $2^{10}$ = 1024 so với kích thước $2^9$=512) thì chương trình thực hiện nhân ma trận sẽ chậm gấp 7 – 7.5 lần so với mốc kích thước lũy thừa 2 của ma trận trước đó (1432s/195s=7.2 lần).
Đối với thuật toán Strassen thì kích thước của các ma trận đầu vào phải là lũy thừa của 2 ($2^1$, $2^2$, $2^3$, ….): điều này sẽ giúp cho thuật toán Strassen hoạt động hiệu quả hơn, vì nó sẽ giảm số lần phải chia ma trận để tính toán ma trận con và giảm độ phức tạp tính toán. Do đó khi thực hiện nhân ma trận song song, ta sẽ tăng kích thước của ma trận lên đến kích thước của lũy thừa của 2 (1, 2, 4, 8, 16, 32, ….. 1024, 2048, ….) và sau khi thực hiện nhân ma trận xong sẽ trả lại kích thước ban đầu. Tuy rằng thời gian thực hiện ứng với mỗi mốc lũy thừa của 2 sẽ gấp 7 - 7.5 lần thời gian thực hiện so với mốc trước nhưng nhìn chung thì vẫn tối ưu hơn thuật toán tuần tự.

| Number of Core | Size of Matrix A | Size of Matrix B | Sequential executions [seconds] | | | |
|---|---|---|---|---|---|---|
| | | | 1 | 2 | 3 | 4 |
| 2 | 200x200 | 200x200 | 3.625 | 3.568 | 3.826 | 3.765 |
| | 400x400 | 400x400 | 31.256 | 31.259 | 30.264 | 31.875 |
| | 600x600 | 600x600 | 105.265 | 105.843 | 104.233 | 104.996 |
| | 800x800 | 800x800 | 245.162 | 258.167 | 260.112 | 255.111 |
| | 1000x1000 | 1000x1000 | 482.663 | 484.5 | 482.103 | 485.221 |
| 5 | 200x200 | 200x200 | 3.827 | 3.849 | 3.819 | 3.82 |
| | 400x400 | 400x400 | 30.812 | 30.805 | 30.853 | 31.313 |
| | 600x600 | 600x600 | 106.572 | 105.449 | 105.592 | 106.002 |
| | 800x800 | 800x800 | 246.976 | 256.229 | 262.652 | 262.5 |
| | 1000x1000 | 1000x1000 | 483.576 | 484.258 | 482.154 | 484.126 |
| 8 | 200x200 | 200x200 | 3.916 | 3.933 | 3.798 | 3.78 |
| | 400x400 | 400x400 | 30.622 | 30.639 | 30.556 | 30.672 |
| | 600x600 | 600x600 | 104.153 | 105.59 | 110.988 | 104.349 |
| | 800x800 | 800x800 | 247.688 | 254.35 | 254.415 | 252.976 |
| | 1000x1000 | 1000x1000 | 485.664 | 484.119 | 484.67 | 484.536 |
| 12 | 200x200 | 200x200 | 3.73 | 3.727 | 3.703 | 3.708 |
| | 400x400 | 400x400 | 30.508 | 30.432 | 30.389 | 30.347 |
| | 600x600 | 600x600 | 103.783 | 112.253 | 111.772 | 107.112 |
| | 800x800 | 800x800 | 254.129 | 248.744 | 257.431 | 255.49 |
| | 1000x1000 | 1000x1000 | 484.2 | 481.447 | 481.709 | 481.518 |

Hình 5. Thời gian thực hiện nhân ma trận của thuật toán tuần tự


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Theo thống kê dữ liệu trong bảng 2 về thời gian thực hiện nhân ma trận bằng thuật toán tuần tự, ta nhận thấy rằng dù cấp phát bao nhiêu CPU cho chương trình thì thời gian thực hiện khá là tương đối nhau với mỗi mốc kích thước ma trận (mốc kích thước lũy thừa của 2 như trên)
Tuy nhiên thuật toán tuần tự kém hiệu quả so với thuật toán song song ở trên vì mỗi lần tăng kích thước ma trận thì thời gian thực hiện cũng tăng theo tương ứng.

## 5 HIỆN THỰC MÃ HÓA CÔNG KHAI TRUYỀN THÔNG ĐIỆP VỚI HỆ MÃ MCELIECE:

Thông qua việc sử dụng hệ mật mã công khai McEliece trong mã hóa thông điệp, chúng tôi đã mã hóa và giải mã thành công dữ liệu ở dạng text '.txt' và dạng bảng '.xlsx':

```
Original Data:
Trong ví dụ này, chúng ta sử dụng thư viện Joblib để
thực hiện xử lý nhân ma trận song song.
Sau đó tiến hành tạo khóa, mã hóa, giải mã và lưu data vào file txt.

Trong ví dụ này, chúng ta sử dụng thư viện Joblib để
thực hiện xử lý nhân ma trận song song.
Sau đó tiến hành tạo khóa, mã hóa, giải mã và lưu data vào file txt.
```

Hình 6. Dữ liệu ban đầu của dữ liệu cần mã hóa

Từ dữ liệu gốc ban đầu (Hình 6 - Original Data), sau khi chuyển đổi dữ liệu sang ma trận số (Hình 7 - Data Matrix) và trải qua quá trình tạo khóa, quá trình mã hóa của hệ mật mã McEliece, ta thu được bản mã (hay dữ liệu sau khi mã hóa) tại hình 8 (Cipher Matrix):

```
Data Matrix:
[[ 2 32  5 18  3 22 21  6 23 29  0 35 -1 -1 -1 -1]
 [24  9 33 13 17 15 26 19 20 -1 -1 -1 -1 -1 -1 -1]
 [ 1 34 25 10 28 12 16 11  8 16 30 14  4 31  7 27]
 [-1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1]
 [ 2 32  5 18  3 22 21  6 23 29  0 35 -1 -1 -1 -1]
 [24  9 33 13 17 15 26 19 20 -1 -1 -1 -1 -1 -1 -1]
 [ 1 34 25 10 28 12 16 11  8 16 30 14  4 31  7 27]]
```

Hình 7. Ma trận số của dữ liệu ban đầu

```
Cipher Matrix:
[[ 8.090e+02  1.000e+00  7.950e+02  1.010e+03  8.740e+02  9.680e+02
   6.630e+02  6.020e+02  1.224e+03  1.046e+03  5.330e+02  9.930e+02
   1.024e+03  0.000e+00  7.040e+02  1.027e+03  1.000e+00  9.380e+02
   1.000e+00  1.000e+00  1.141e+03]
 [ 5.160e+02  1.000e+00  5.410e+02  7.440e+02  7.110e+02  6.340e+02
   5.700e+02  4.280e+02  9.140e+02  7.200e+02  3.780e+02  8.150e+02
   7.520e+02  0.000e+00  6.080e+02  7.220e+02  1.000e+00  6.220e+02
   0.000e+00  0.000e+00  9.250e+02]
 [ 1.084e+03  0.000e+00  1.115e+03  1.540e+03  1.165e+03  1.362e+03
   1.036e+03  7.890e+02  1.614e+03  1.387e+03  7.550e+02  1.425e+03
   1.441e+03  0.000e+00  1.169e+03  1.348e+03  1.000e+00  1.225e+03
   1.000e+00  0.000e+00  1.515e+03]
 [-5.700e+01  0.000e+00 -6.200e+01 -8.400e+01 -6.600e+01 -7.500e+01
  -5.800e+01 -4.300e+01 -9.300e+01 -7.900e+01 -4.500e+01 -8.200e+01
  -8.200e+01  0.000e+00 -6.400e+01 -7.600e+01  1.000e+00 -6.700e+01
   1.000e+00  1.000e+00 -8.900e+01]
 [ 8.090e+02  0.000e+00  7.960e+02  1.011e+03  8.730e+02  9.680e+02
   6.630e+02  6.020e+02  1.224e+03  1.046e+03  5.340e+02  9.940e+02
   1.023e+03  0.000e+00  7.050e+02  1.027e+03  1.000e+00  9.370e+02
   1.000e+00  1.000e+00  1.142e+03]
 [ 5.150e+02  0.000e+00  5.410e+02  7.430e+02  7.110e+02  6.340e+02
   5.710e+02  4.290e+02  9.140e+02  7.200e+02  3.780e+02  8.150e+02
   7.520e+02  1.000e+00  6.080e+02  7.220e+02  1.000e+00  6.210e+02
   1.000e+00  0.000e+00  9.250e+02]
 [ 1.084e+03  0.000e+00  1.115e+03  1.540e+03  1.166e+03  1.362e+03
   1.035e+03  7.890e+02  1.615e+03  1.388e+03  7.550e+02  1.424e+03
   1.440e+03  0.000e+00  1.168e+03  1.348e+03  0.000e+00  1.224e+03
   1.000e+00  0.000e+00  1.515e+03]]
```

Hình 8. Ma trận dữ liệu khi được mã hóa

Từ bản mã (dữ liệu sau khi mã hóa – Cipher Matrix), trải qua quá trình giải mã (Hình 9 – Plain Matrix), ta thu được bản rõ (dữ liệu sau khi giải mã hay dữ liệu ban đầu) ở hình 10 (Plain Data):

```
Plain Matrix:
[[ 2. 32.  5. 18.  3. 22. 21.  6. 23. 29.  0. 35. -1. -1. -1. -1.]
 [24.  9. 33. 13. 17. 15. 26. 19. 20. -1. -1. -1. -1. -1. -1. -1.]
 [ 1. 34. 25. 10. 28. 12. 16. 11.  8. 16. 30. 14.  4. 31.  7. 27.]
 [-1. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1. -1.]
 [ 2. 32.  5. 18.  3. 22. 21.  6. 23. 29.  0. 35. -1. -1. -1. -1.]
 [24.  9. 33. 13. 17. 15. 26. 19. 20. -1. -1. -1. -1. -1. -1. -1.]
 [ 1. 34. 25. 10. 28. 12. 16. 11.  8. 16. 30. 14.  4. 31.  7. 27.]]
```

Hình 9. Ma trận dữ liệu sau khi được giải mã

```
Plain Data:
Trong ví dụ này, chúng ta sử dụng thư viện Joblib để
thực hiện xử lý nhân ma trận song song.
Sau đó tiến hành tạo khóa, mã hóa, giải mã và lưu data vào file txt.

Trong ví dụ này, chúng ta sử dụng thư viện Joblib để
thực hiện xử lý nhân ma trận song song.
Sau đó tiến hành tạo khóa, mã hóa, giải mã và lưu data vào file txt.
```

Hình 10. Dữ liệu sau khi được khôi phục


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 6 KẾT LUẬN

Xử lý song song cho phép chia nhỏ các nhiệm vụ mã hóa thành các phần nhỏ hơn và thực hiện chúng đồng thời trên nhiều luồng xử lý độc lập. Điều này có thể đem lại hiệu quả tính toán cao hơn và giảm thời gian thực hiện của quá trình mã hóa

Hệ mật mã McEliece sử dụng thuật toán giải mã đơn giản nhiều hơn so với các hệ mã hóa khác, điều này có nghĩa là nó có thể được triển khai nhanh chóng và hiệu quả trên các thiết bị có tài nguyên hạn chế.

Hơn nữa hệ mật mã McEliece cũng được biết đến như một trong những hệ mã hóa chịu được tấn công bởi các máy tính lượng tử. Điều này có ý nghĩa rằng nó có thể được sử dụng để bảo vệ các thông tin quan trọng khỏi những cuộc tấn công của máy tính lượng tử trong tương lai.

Một trong những ưu điểm của hệ mật mã McEliece là tính toán phức tạp của nó rất cao, do đó nó khó bị tấn công bởi các kỹ thuật tấn công thông thường. Tuy nhiên, hệ thống này cũng có một số hạn chế, bao gồm kích thước của khóa công khai và khóa bí mật rất lớn, và khả năng xử lý của nó cũng khá chậm

## 7 HƯỚNG PHÁT TRIỂN

*Tối ưu hóa nhân ma trận:* Việc tối ưu hóa ma trận nhân là một yếu tố quan trọng trong hệ mật mã McEliece. Nhóm cần nghiên cứu và tìm kiếm các kỹ thuật tối ưu hóa nhân ma trận để cải thiện tính hiệu quả của quá trình mã hóa và giải mã.

*Tối ưu hóa hiệu suất:* Một trong những khía cạnh quan trọng cần cải thiện là tốc độ mã hóa và giải mã. Nhóm sẽ nghiên cứu các phương pháp tối ưu hóa hiệu suất cho hệ mật mã McEliece, bao gồm cải thiện tốc độ tạo khóa và tăng tốc độ xử lý phần cứng thông qua các phương pháp tối ưu hóa.

*Giảm kích thước khóa:* Giảm kích thước khóa để làm cho hệ mật mã McEliece trở nên hữu ích hơn trong các ứng dụng thực tế. Nhóm sẽ tìm hiểu về các phương pháp như mã hóa đơn giản hóa, kỹ thuật nén dữ liệu hoặc tạo ra các hệ thống mã hóa McEliece với kích thước khóa nhỏ hơn.

*Nghiên cứu và phát triển các biến thể của hệ mật mã McEliece:* Để cải thiện tính bảo mật và hiệu quả, nhóm sẽ nghiên cứu và phát triển các biến thể mới của hệ mã này, bao gồm các phiên bản có kích thước khóa nhỏ hơn và tính bảo mật cao hơn.

*Kết hợp với các công nghệ khác:* Một phương pháp khác để phát triển hệ mã hóa McEliece là kết hợp nó với các công nghệ khác để tạo ra hệ mã hóa kết hợp.

*Phát triển và nâng cấp* cho website và hệ mật mã McEliece nhằm mã hóa được tất cả mọi loại tập tin.

## 8 TÀI LIỆU THAM KHẢO

[1]. McEliece R. J. (1978). A Public-Key Cryptosystem Based on Algebraic Coding Theory, The Deep Space Network Progress Report, pp: 114-116.

[2]. Berlekamp E., McEliece R., and Tilborg H. v. (1978), "On the Inherent Intractability of Certain Coding Problems", IEEE Transactions on Information Theory, 24(3), pp: 384-386.

[3]. E. Berlekamp, R. McEliece, H. van Tilborg, On the inherent intractability of certain coding problems. IEEE Trans. Inf. Theory 24

[4]. D.J. Bernstein, T. Lange, C. Peters, Attacking and defending the McEliece cryptosystem, in Post-Quantum Cryptography. LNCS, vol. 5299 (Springer, Berlin/Heidelberg, 2008), pp. 31–46Return

[5]. Bernstein D. J., Buchmann J., and Dahmen E. (2009), Post-quantum cryptography, Springer-Verlag Berlin Heidelberg, pp: 95-145.

[6]. Patterson N. J. (1975), "The Algebraic Decoding of Goppa Codes", IEEE Transactions on Information Theory, IT-21(2), pp: 203-207

[7]. Pallav Sharma (2023) “How to Save and Load Machine Learning Models in Python Using Joblib Library?”

[8]. Wikipedia (edited on 29 May 2023) “Strassen algorithm”

**YSC5.F033**

# IMPROVING THE PROPOSAL AND APPROVAL SYSTEM FOR EMULATION AND REWARD CONTESTS AT IUH

QUANG-MANH PHAM, TAN-LOI PHAM, PHUOC-LOC NGUYEN [1], HUU-QUANG NGUYEN [2*]

[1] *Information Systems students - Faculty of Information Technology – Industrial University of Ho Chi Minh City,*

[2] *Department of Information Systems – Faculty of Information Technology – Industrial University of Ho Chi Minh City;*

** nguyenhuuquang@iuh.edu.vn*

**Abstract.** The Industrial University of Ho Chi Minh City holds annual contests to recognize exceptional achievements by students, lecturers, and staff. Currently, the school's executive board employs a manual approach to approving candidates, where they fill out information on the system, upload supporting documents, and undergo manual evaluation. However, manual evaluation is a challenging process given the significant number of students and teaching staff. To address this challenge, the article suggests improving the existing system to alleviate obstacles in the school's emulation approval procedure. The proposed system is developed on a website platform using the MVC architecture model based on the PHP programming language, the Laravel framework, and MongoDB. Before implementing the system, it is evaluated using a database of students, professors, and school employees to determine its effectiveness.

**Keywords.** Emulation and reward, review system, improving system.

## CẢI TIẾN HỆ THỐNG ĐĂNG KÝ XÉT DUYỆT THI ĐUA KHEN THƯỞNG TẠI TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH

**Tóm tắt.** Hằng năm, trường Đại học Công nghiệp Thành phố Hồ Chí Minh tổ chức các đợt thi đua khen thưởng nhằm ghi nhận thành tích xuất sắc của sinh viên, giảng viên và nhân viên. Hiện nay, ban chấp hành Đoàn trường đang sử dụng phương thức xét duyệt thủ công bằng cách cho các ứng cử viên điền thông tin vào hệ thống, tải lên các minh chứng liên quan và tiến hành xét duyệt từng tiêu chí, tiêu chuẩn và kết luận ứng cử viên có đạt được danh hiệu đã đăng ký hay không. Tuy nhiên, với số lượng sinh viên và đội ngũ giảng viên, cán bộ công nhân viên tương đối nhiều thì xét duyệt thủ công là việc không dễ dàng. Để giải quyết vấn đề trên, bài báo đề xuất việc cải tiến hệ thống hiện có nhằm giảm bớt những khó khăn trong việc xét duyệt thi đua của trường. Hệ thống đề xuất được phát triển trên nền tảng website sử dụng mô hình kiến trúc MVC dựa trên ngôn ngữ lập trình PHP, Laravel framework và MongoDB. Hệ thống được thực nghiệm với cơ sở dữ liệu sinh viên, giảng viên và cán bộ công nhân viên của nhà trường để đánh giá hiệu quả trước khi đưa vào triển khai sử dụng thực tế.

**Từ khóa.** Hệ thống thi đua khen thưởng, đánh giá hệ thống, cải tiến hệ thống

## 1 INTRODUCTION

In contemporary organizational settings, including companies, enterprises, and state agencies, the practice of commendation serves as a regular means of acknowledging, appreciating, honoring, and incentivizing the efforts and contributions made by individuals and collectives. Annually, the Ho Chi Minh Communist Youth Union at Industrial University of Ho Chi Minh City orchestrates registration and commendation campaigns to pay tribute to exceptional achievements accomplished by students, lecturers, and staff members. Beginning in 2020, the process of evaluating commendations has been standardized through the issuance of relevant documents, such as guidelines for appraising titles like "Advanced Youth Following

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Uncle Ho's Teachings" [1] [2], "Outstanding Young Lecturer" [3], and "Young Specialist - Excellent - Friendly" [4]. Previously, the university's youth executive committee utilized a manual approach to registration and evaluation, wherein candidates were required to draft commendation proposals and furnish physical copies of supporting evidence. Subsequently, the managing unit compiled the proposals and evidence and submitted them, along with a word-processed document encompassing a list of individuals and collectives recommended for commendation, to the department responsible for evaluating these campaigns. Regrettably, the potential of these records remains largely untapped for subsequent tasks and procedures, necessitating the generation of new documents, lists, and commendation certificates.

To address the aforementioned challenges, the student Dinh Tuan Anh and collaborators [5] advocate for the development of a registration and evaluation system tailored to commendation campaigns. This system digitalizes specific aspects of the commendation evaluation process, encompassing the creation of commendation titles and registration campaigns by administrators, as well as allowing candidates to register accounts and actively participate in commendation initiatives. Nonetheless, the aforementioned system fails to adequately tackle the management of supporting evidence and the evaluation process required to determine whether candidates fulfill the criteria for the registered titles.

Considering the present circumstances, requirements, and prevailing issues outlined above, this paper proposes an enhanced information system that supports the registration and evaluation process of commendation campaigns. The system optimizes the evaluation function of commendation campaigns while facilitating the supervision of candidate numbers and the administration of accompanying evidence. Furthermore, the system enables the systematic and transparent arrangement of commendation campaigns and their corresponding timelines. Of notable significance, the system incorporates the capacity to lock or unlock the registration period for commendation titles, granting users the ability to ascertain which commendation campaigns are ongoing, upcoming, or completed.

The subsequent sections of this paper are structured as follows: Section 2 discusses pertinent theories relevant to the research content, encompassing the commendation evaluation process at Industrial University of Ho Chi Minh City, the utilization of the Model-View-Controller (MVC) architectural model utilizing the PHP programming language and Laravel framework, as well as the interaction with the MongoDB database. Section 3 provides a detailed analysis of the system design. Section 4 delineates the application development process based on the previously examined system design analysis. Section 5 outlines the application testing process, while Section 6 elucidates the outcomes of the application development and offers directions for future enhancements and developments.

## 2 THEORETICAL RESEARCH FOUNDATION

### 2.1 Process of Evaluating Commendation Campaigns

At present, Industrial University of Ho Chi Minh City implements a system whereby a variety of titles are available for candidates, encompassing students, lecturers, and specialists affiliated with the institution, allowing them to partake in the registration process for commendation. Each group of candidates adheres to distinct criteria and standards for assessment, which are explicitly outlined within the corresponding instructional materials [1] [2] [3] [4]. Nevertheless, the registration and evaluation procedures pertaining to commendation campaigns solely diverge in terms of the criteria, standards, and supporting evidence associated with each candidate group. Overall, the commendation registration and evaluation process generally follows a uniform framework, with the specific steps for registration and evaluation expounded upon in Figure 1.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

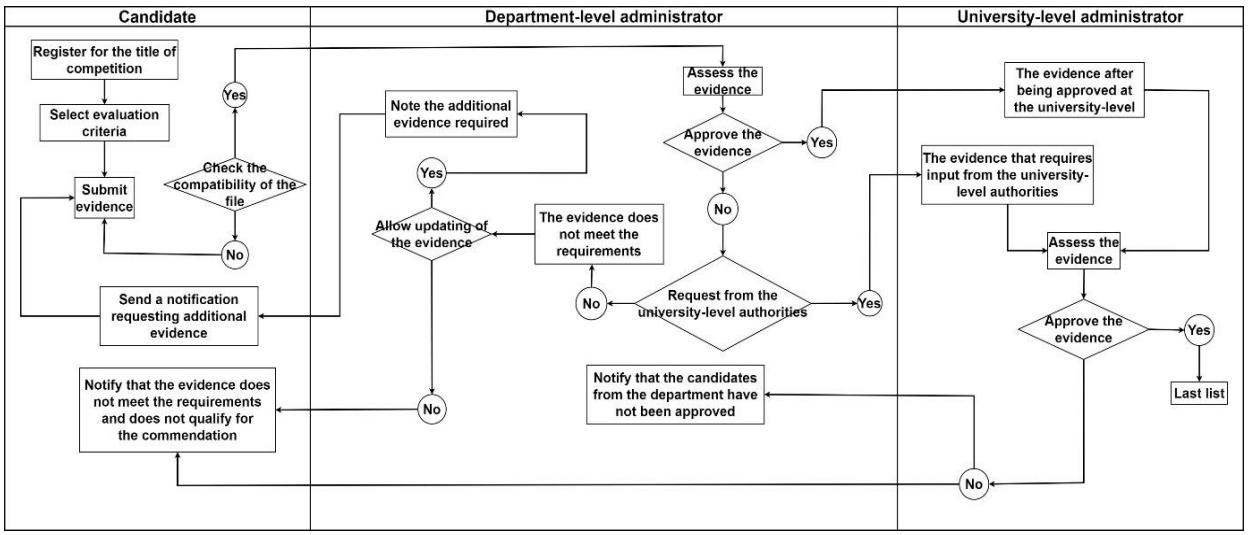



Figure 1. The process of registration for competition and approval of rewards.

The commendation evaluation system follows a step-by-step procedure as outlined below: Candidates first register for titles through their individual candidate pages and submit the required supporting evidence corresponding to each title. Upon successful title registration, the system automatically transfers the candidate's supporting evidence to the page of the department administrator responsible for evaluation.

Once the department administrator receives the candidate's supporting evidence, they commence their own evaluation process, which entails three possible scenarios: If the candidate's supporting evidence satisfies all the established criteria, the department administrator marks the "Approved" option for each piece of evidence and proceeds to click the "Approved" button. In cases where the candidate's supporting evidence fails to meet the criteria, the department administrator refrains from selecting the "Approved" option and instead clicks the "Rejected" button. If the department administrator has reservations or uncertainties regarding the candidate's supporting evidence, they refrain from selecting the "Approved" option and instead click the "Review" button. Regardless of the scenario, the supporting evidence is then forwarded to the university along with the evaluation status at the department level.

During the university-level evaluation process, the university administrator assesses the supporting evidence received. For evidence in the "Approved" status, the university administrator selects the "Achieved" button. In the case of evidence marked as "Review," the university administrator re-evaluates the candidate's evidence and chooses either the "Achieved" or "Not Achieved" option, depending on the assessment outcome, before clicking the corresponding button. For candidates whose supporting evidence is in the "Rejected" status, the university administrator has the ability to review and re-evaluate their evidence. However, they are required to provide remarks in the adjacent box indicating that the evidence was not approved and communicate this information to the respective department administrator.

## 2.2 Laravel Framework

In the context of construction projects, the term "framework" [6] denotes a structural support system, while in the field of information technology, it pertains to a compilation of libraries and tools that facilitate the streamlined development of software by programmers. Laravel [7], a highly acclaimed website framework, is distinguished by its lucid syntax and adherence to the Model-View-Controller (MVC) architectural pattern. This framework encompasses various functionalities such as bundles, composer for seamless installation, Restful API support, Artisan command-line tools, database migration capabilities, authentication mechanisms, view management, unit testing, and automatic pagination. Laravel offers numerous benefits [8], including access to the latest features of PHP, extensive documentation, compatibility with caching systems, efficient database administration, and reverse routing. Nonetheless, it should be noted that Laravel may encounter compatibility challenges [8] when transitioning between different versions and exhibit increased resource consumption when employed in mobile applications.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 2.3 MVC pattern

The MVC (Model-View-Controller) [9] paradigm in Laravel encompasses a cohesive architectural pattern that comprises three distinct and autonomous components: the Model, View, and Controller. The Model [10] represents the underlying business logic, encompassing methods for data processing, database retrieval, and entity descriptions. The View [10] assumes the responsibility of rendering information and facilitating user interaction, housing graphical user interface (GUI) elements such as textboxes and images. In the context of web applications, this often entails generating HTML content for display within the user's browser, although the Views can take various forms. Acting as an intermediary, the Controller [10] assumes the role of a coordinator between the Model and View. It receives input from the user via the View and subsequently interacts with the Model to execute specific actions, ultimately returning the results to the View for presentation. This modular separation of concerns within Laravel's MVC architecture promotes a structured and systematic approach to software development.

## 2.4 MongoDB

MongoDB [11] represents an open-source NoSQL database management system that prioritizes a document-oriented approach, facilitating the storage of data in adaptable JSON-like documents. By eliminating rigid tabular structures, MongoDB enables agile and dynamic data modeling, while its query language allows for efficient and intricate document-level queries. Noteworthy features of MongoDB include its support for horizontal scalability, data replication, file storage capabilities, and robust data processing functionalities, positioning it as a well-suited choice for high-performance applications dealing with extensive datasets. The system facilitates ad-hoc querying, leverages indexing for enhanced performance, and employs replica sets and sharding techniques to ensure scalability. Furthermore, MongoDB seamlessly integrates with GridFS for the storage of large files and provides comprehensive support for complex calculations and aggregations through the use of aggregation pipelines and JavaScript execution. Additional features encompass collection size limits, transaction support, and diverse querying capabilities. However, it is important to note that MongoDB lacks join operations, necessitates substantial memory resources, imposes a 16MB document size limit, and does not fully adhere to the ACID properties typically associated with traditional relational databases. As a result, careful consideration is required when deploying MongoDB for applications that demand high transactional integrity and stringent data consistency requirements. It is worth highlighting that MongoDB benefits from an active and vibrant user community, supported by an abundance of resources offering assistance and guidance.

## 2.5 Laragon

Laragon [12] represents a robust and versatile development environment accommodating diverse programming languages such as PHP, Node.js, Python, Java, Go, and Ruby. Noteworthy for its remarkable velocity, extensibility, and minimalistic attributes, Laragon furnishes a dependable and resourceful milieu for constructing and administering contemporary web applications. Prominent features of Laragon encompass intuitive URLs, effortless portability of the Laragon folder, isolation from the underlying operating system, and automated configuration. Moreover, Laragon extends support to both Apache and Nginx web servers, thereby affording a cutting-edge architectural framework conducive to modern web development practices. Thus, Laragon serves as an intuitive and formidable instrument for developers seeking a streamlined workflow and optimal performance outcomes.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

# 3 SYSTEM DESIGN ANALYSIS AND APPLICATION DEVELOPMENT

## 3.1 System functional diagram

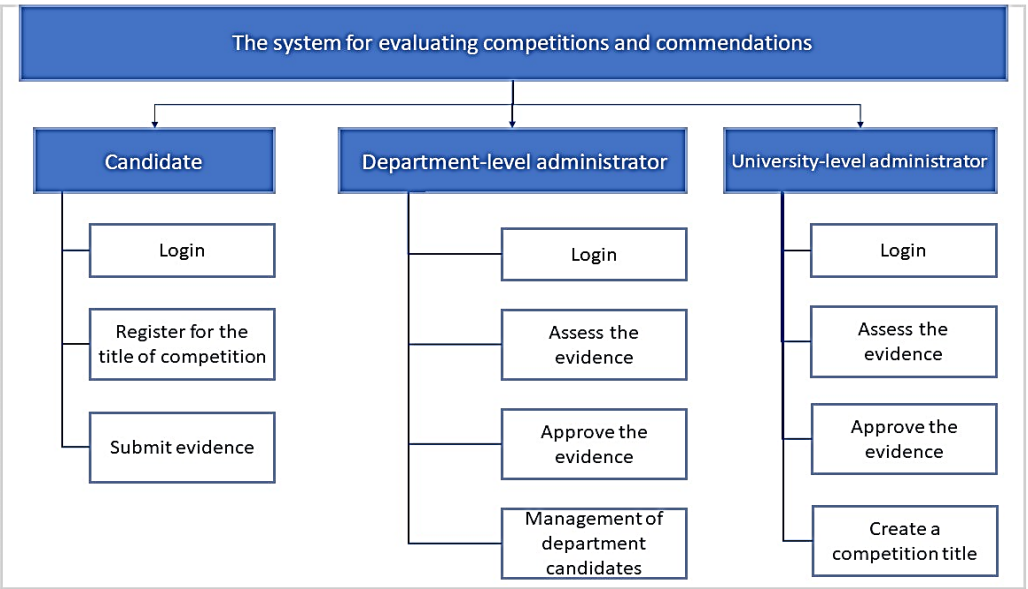



Figure 2. Diagram of system functionality

The system manages the registration of candidates for commendation campaigns and approves their titles. As illustrated in Figure 2, the system is designed for three types of users: candidates, including students, specialists, and faculty members who are studying and working at Industrial University of Ho Chi Minh City, department-level administrators, and university-level administrators. The system has two main functions: candidate registration for commendation campaigns and approval of registered profiles at the department and university levels. Additionally, the system performs several other important basic functions such as candidate management, title creation, creation of campaign registration rounds, user account management…

## 3.2 The process of approving candidate's title registration

The process of approving a candidate's title registration is illustrated in Figure 3. When candidates intend to enroll for a new title, they will access the system by logging in. The homepage will exhibit a comprehensive roster of existing titles. Upon selecting the "Register" button, candidates will be prompted to submit the necessary evidence in accordance with the specified requirements of the title. Subsequently, candidates are required to diligently furnish all the requisite information within the designated "Answer" section and furnish their substantiating documentation. Following the successful completion of the answer submission and supporting document upload, the registration details will be forwarded to the Administrator for meticulous evaluation and subsequent endorsement.

## 3.3 The process for evaluating candidates' commendation titles at the department level

When the Department Administrator at the faculty commences the approval process, they will access the administrative interface where the list of students will be displayed. Subsequently, the Administrator will thoroughly examine the candidates' supporting documentation and responses, taking into consideration the specific criteria and requirements associated with the title. The Administrator will diligently evaluate each candidate to ensure their compliance with all the established criteria and requisites. The outcome of the evaluation process can result in one of three statuses: Rejected, Under Review, Approved. Ultimately, all evaluated candidates will be forwarded to the homepage of the University Administrator for further processing and consideration. Figure 4 provides a visual representation of the procedural framework concerning the evaluation of commendation titles conferred upon departmental candidates.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

### 3.4 The process for evaluating candidates' commendation titles at the university-level level

Upon commencement of the accreditation process at the university level, the administrator thoroughly examines and reevaluates the evidence provided by all candidates. In instances where discrepancies arise between the evaluations conducted by the university-level administrator and the department-level administrator, the former takes meticulous note of such variations and forwards them to the latter for further deliberation. Ultimately, the university-level administrator appraises the candidates, classifying them into two distinct categories: "Not Achieved" and "Achieved." Those who are deemed to have met the requisite criteria receive an official certificate of accreditation conferred by the university-level administrator. Figure 5 illustrates the steps involved in the evaluation process of candidates' titles by the university-level administrator.

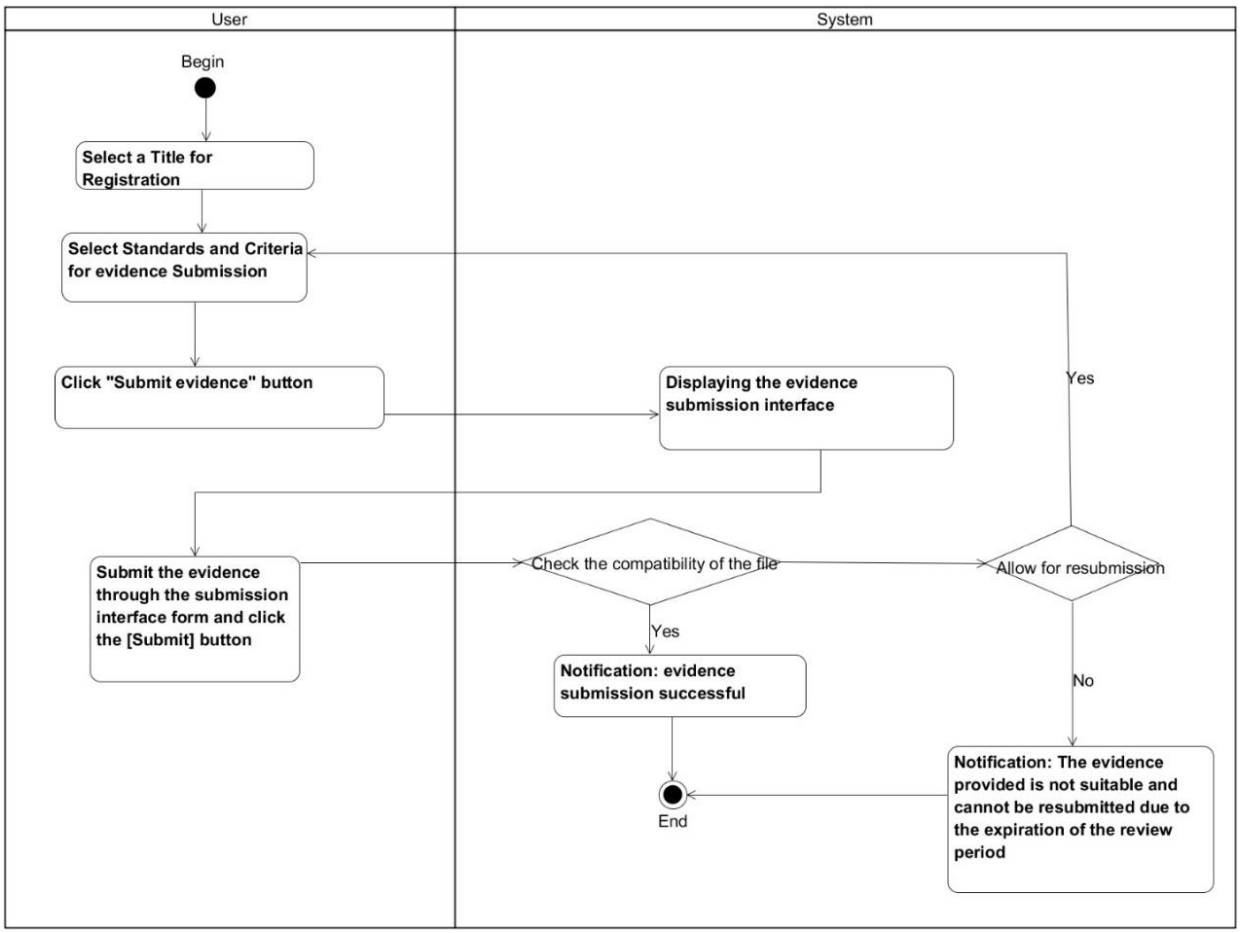



Figure 3. The process of approving candidate's title registration

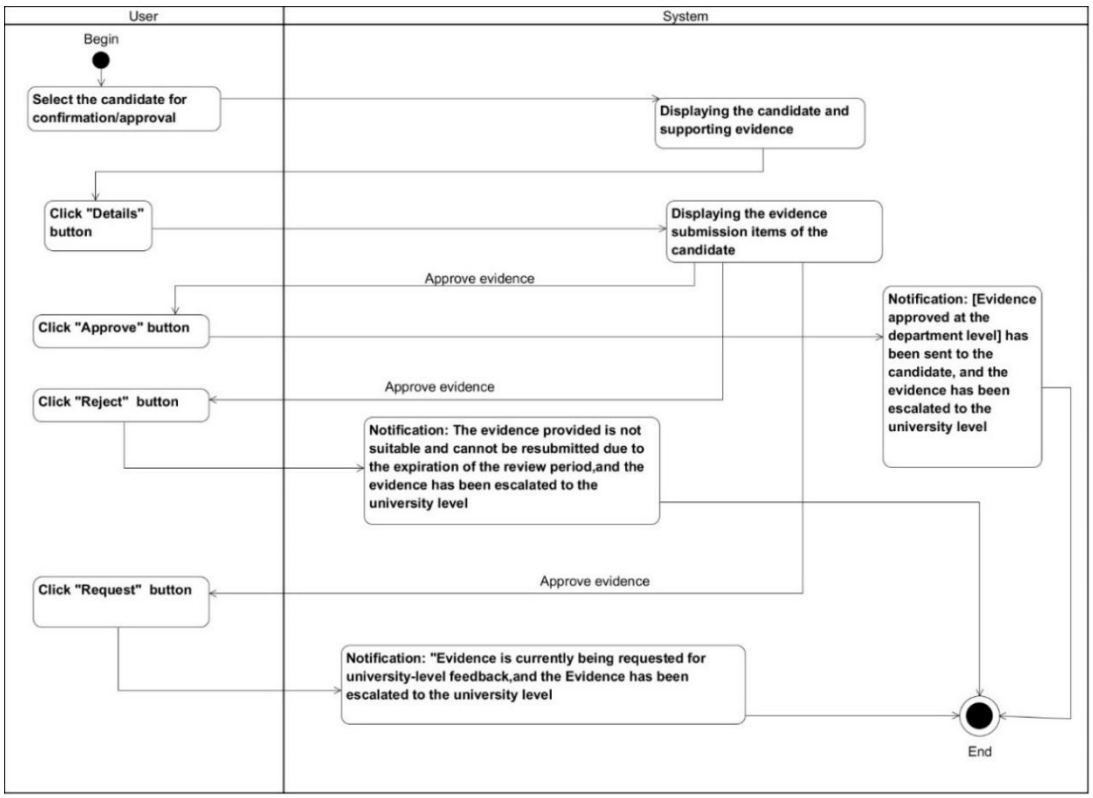



Figure 4. The process for evaluating candidates' commendation titles at the department level

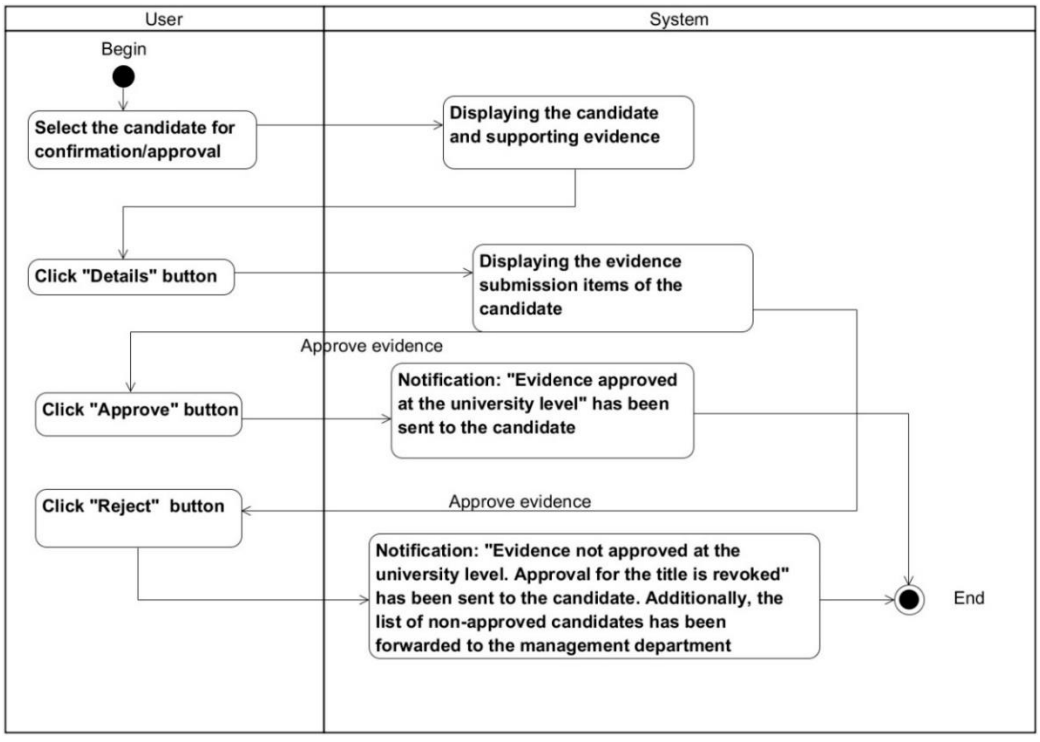



Figure 5. The process for evaluating candidates' commendation titles at the university level

## 3.5 Building applications

The application has been deployed on a computer equipped with a Core i7 processor, 16 GB of RAM, and the Windows 10 operating system. The system architecture follows a three-tier model, utilizing the PHP

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

programming language along with the Laravel framework, facilitating seamless interaction with the underlying database. Data management within the system is facilitated by the employment of the MongoDB database management system. The system encompasses essential functionalities encompassing the registration process and the evaluation of commendation campaigns. The following table provides a description of some important features of the system.

Table 1. Description of some important features of the system

| Function | Description |
|---|---|
| Title Creation | Enables the creation of new commendation titles to accommodate diverse recognition categories. |
| Campaign Management | Facilitates the creation and management of commendation campaigns, including defining criteria and timelines. |
| Candidate Registration | Enables candidates to register their participation in the commendation process by providing their information. |
| Evidence Submission | Provides a mechanism for candidates to submit supporting evidence required for the commendation process. |
| Commendation Approval | Allows administrators to review and approve commendation applications submitted by candidates. |
| User Account Management | Allows administrators to manage user accounts, including registration, authentication, and access control. |
| Reporting and Analysis | The system provides comprehensive reporting and analysis to track commendation progress and generate insights, including statistics on candidate participation and achievement of titles at the department and school levels. |

Figures 6-8 depict the interfaces of the system's core functionalities, including the candidate's title registration process and the evaluation of titles at the department and university levels. Figure 9 represents the interface displaying the certification of attained titles for candidates after the completion of the evaluation processes.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

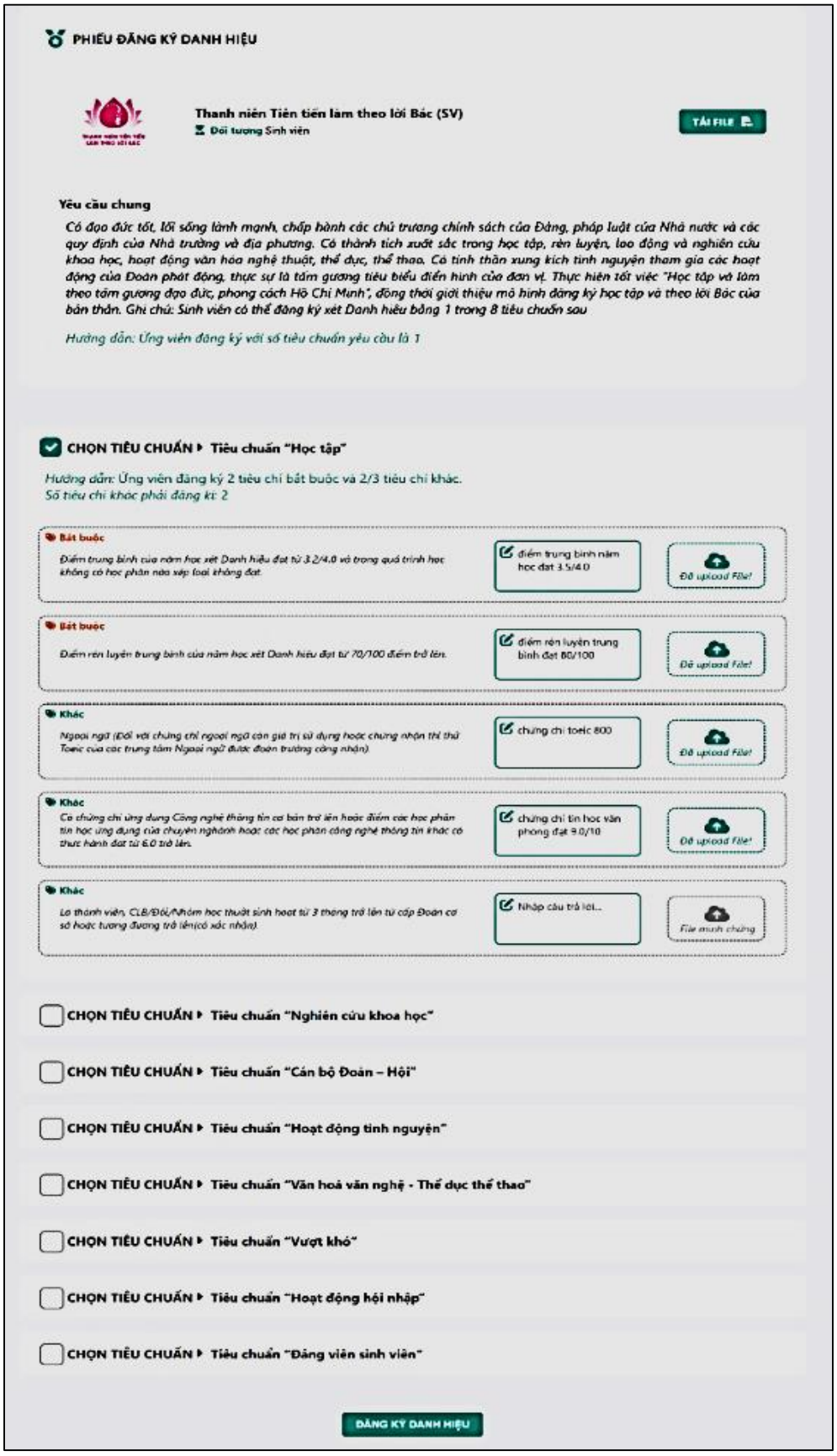



Figure 6. Interface for Candidate's Commendation Title Approval Registration

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

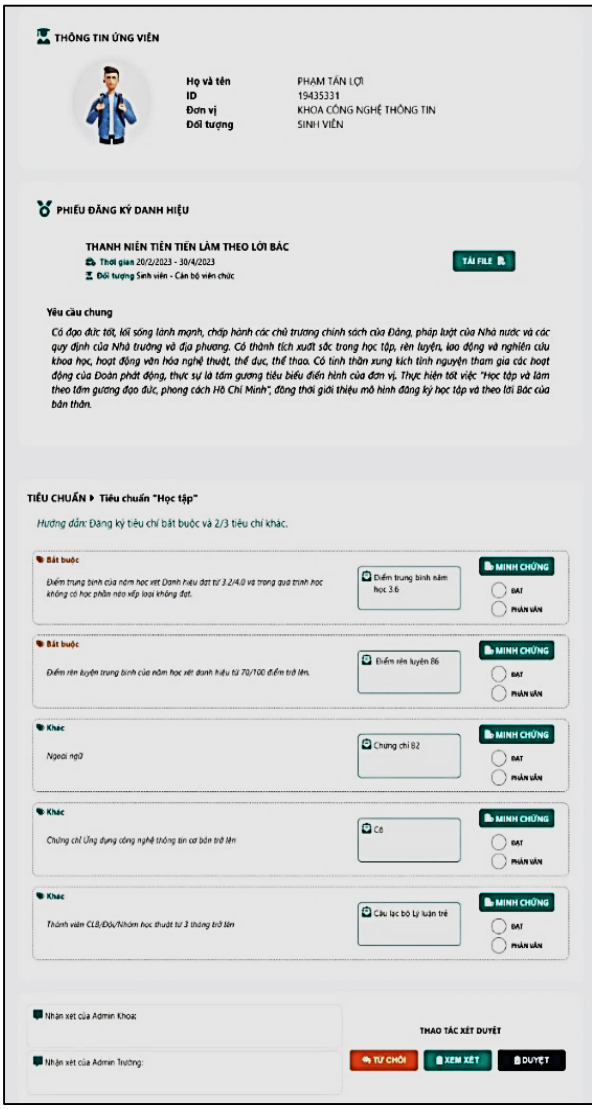



Figure 7. Interface for evaluating candidates' commendation titles at the department level

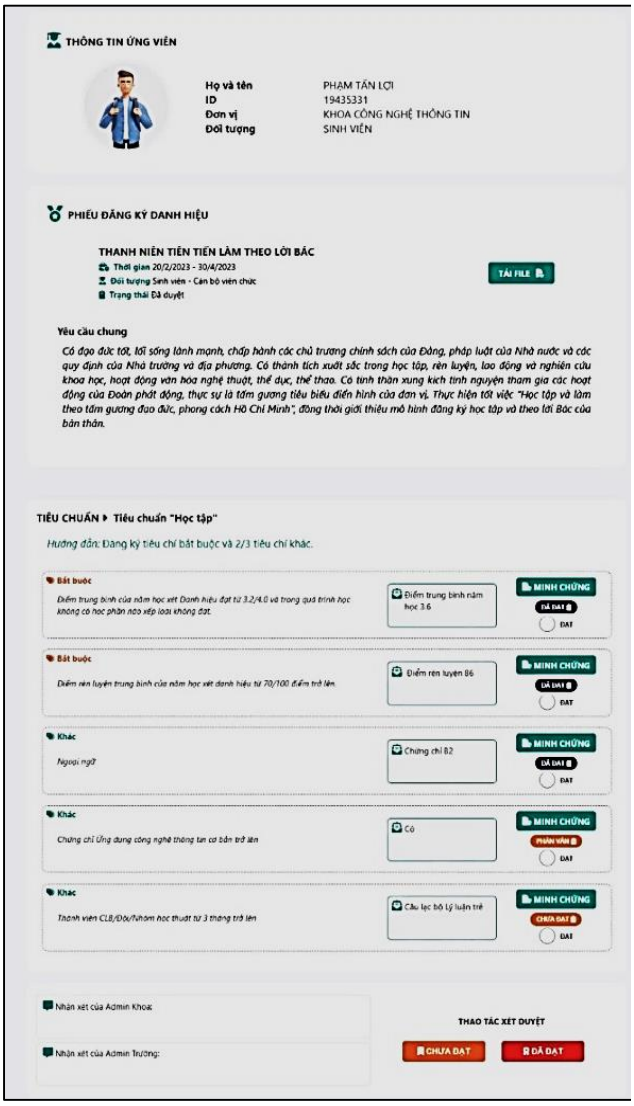



Figure 8. Interface for evaluating candidates' commendation titles at the university-level level

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

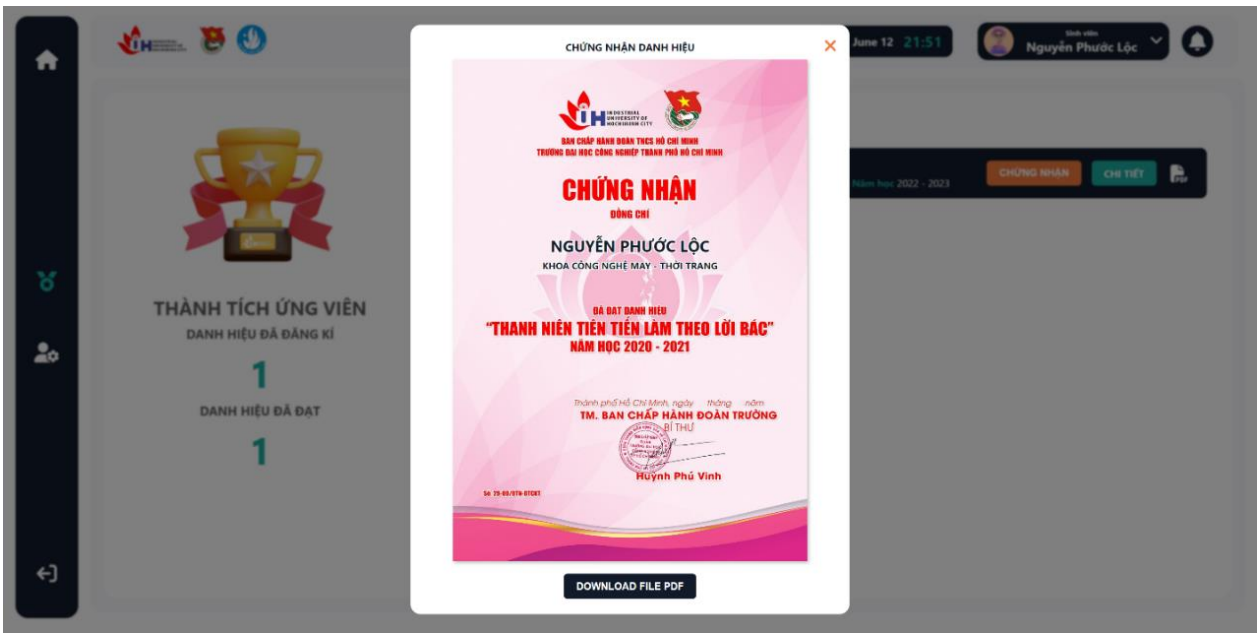



Figure 9. the interface displaying the certification of attained titles for candidate

## 4 TEST CASE

Following the completion of analysis and system development, the research team employed a dataset comprising individuals who actively participated in the evaluation of the "Advanced Youth Following President Ho Chi Minh's Ideals" accolade during the year 2022 for the purpose of conducting preliminary function testing before system deployment. The test dataset encompassed a total of 150 candidates who had registered for the evaluation, encompassing students, specialists, and lecturers affiliated with various departments within the esteemed Industrial University of Ho Chi Minh City. Presented in Table 2 is a depiction of the testing procedures carried out pertaining to several essential system functionalities. The utilization of this test dataset served as a means for the research team to duly assess the efficacy and efficiency of the system's fundamental functionalities. By scrutinizing these pivotal operations, the team succeeded in ascertaining the system's preparedness for deployment, alongside its competence in effectively managing the commendation process. The dataset encompassed a diverse range of candidates, thereby mirroring real-world scenarios and furnishing invaluable insights that can be leveraged for the future enhancement and optimization of the system.

Table 2. Test case

| Function | Scenario | Step by step | Data test | Result |
|---|---|---|---|---|
| Register a title | Candidate applying for a title and submit evidence | Step 1: Candidate logging into system<br>Step 2: The candidate selects the title and presses the 'Register' button<br>Step 3: Select standards and criteria for submit evidence<br>Step 4: Enter the answer in the text box | Title: “Progressive young individuals obeying Uncle's guidance”<br>Standard: Standard “Learning”<br>• Mandatory criteria:<br>Mandatory criteria 1:<br>Text box: “Answer 1.1”<br>Upload supporting evidence field: “evidence1.png”<br>Mandatory criteria 2:<br>Text box: “Answer 1.2” | Notification:<br>You have successfully registered your title |

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| | | | | |
|---|---|---|---|---|
| | | Step 5: Upload file evidence<br>Step 6: Presses “Register a title” button | Upload supporting evidence field: “evidence2.pdf”<br>• Additional criteria:<br>Additional criteria 1:<br>Text box: “Answer 2.1”<br>Upload supporting evidence field: “evidence1.pdf”<br>Additional criteria 2:<br>Text box: “Answer 2.2”<br>Upload supporting evidence field: “evidence2.pdf” | |
| Department-level approval process | After the candidates have successfully registered their title, the department administrator will review their supporting documents for approval | Step 1: The department administrator logs into the system<br>Step 2: The department administrator selects the "Approval" function<br>Step 3: The department administrator selects a candidate with the status "Pending approval"<br>Step 4: The department administrator checks the box "Approve" for all the supporting evidences<br>Step 5: Presses “Approve” button | • Candidate’s evidences:<br>Evidence 1: “Approve”<br>Evidence 2: “Approve”<br>Evidence 3: “Approve”<br>Evidence 4: “Approve” | The approved supporting documents of the candidate will be forwarded to the university-level data with the status "Approved" |
| The university-level approval process | The university administrator reviews the candidate's supporting documents after the department-level evaluation process | Step 1: The university administrator logs into the system<br>Step 2: The university administrator selects the "Approval" function<br>Step 3: The university administrator selects a candidate with the status "Rejected" | • Candidate’s evidences:<br>Evidence 1: “Approve”<br>Evidence 2: “Approve”<br>Evidence 3: “Approve”<br>Evidence 4: “Approve” | The system notifies the candidate that they have achieved the title and provides them with a certificate of title confirmation |

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

| | | | | |
|---|---|---|---|---|
| | | Bước 4: The university administrator checks the box "Approved" for all the supporting evidences<br>Step 5: Presses “Approve” button | | |

## 5 CONCLUSION

The system has been constructed and tested using data from students, faculty members, and staff at the Industrial University of Ho Chi Minh City. Employing a three-tier MVC architecture and the NoSQL MongoDB database, the research team has successfully developed a commendation and reward approval system at the aforementioned institution. This system effectively fulfills the core functionalities of the commendation and reward approval process, enabling candidates to register and submit evidence in order to obtain titles, while also facilitating automated evaluations conducted by department-level and university-level administrators. Moreover, the system offers a range of valuable features, including the creation of commendation campaigns, the establishment of new titles, candidate management, and the generation of comprehensive statistics and reports. Once implemented, the system will provide crucial assistance to the university by automating the commendation and reward approval workflow. The research team intends to further refine the system to meet the specific requirements of the institution and explore potential deployment on mobile platforms or alternative operating systems beyond Windows. Currently, efforts are concentrated on enhancing system completeness, maximizing user convenience, and optimizing user interaction through the mobile user interface.

## REFERENCES

[1] Executive Committee of Industrial University of Ho Chi Minh City, "Decision on the promulgation of the Regulation on the title 'Progressive youth following Uncle's words," 2020. [Online]. Available: https://vanphong.tuoitreiuh.vn/vpdtn/detail/Quyet-dinh-so-05-Ban-hanh-Quy-che-danh-hieu-Thanh-nien-tien-tien-lam-theo-loi-Bac-30/.

[2] Executive Committee of Industrial University of Ho Chi Minh City, "Decision on the promulgation of the Regulation on the titles 'Advanced Collective Following Uncle Ho's Words' and 'Advanced Youth Following Uncle Ho's Words' (amended, supplemented)," 2021. [Online]. Available: https://vanphong.tuoitreiuh.vn/vpdtn/detail/Quyet-dinh-so-23-Ban-hanh-Quy-che-danh-hieu-Tap-the-tien-tien-lam-theo-loi-Bac-va-Thanh-nien-tien-tien-lam-theo-loi-Bac-sua-doi-bo-sung-160/.

[3] Executive Committee of Industrial University of Ho Chi Minh City, "Decision on the promulgation of the Regulation on the Outstanding Young Educator Award," 2020. [Online]. Available: https://vanphong.tuoitreiuh.vn/vpdtn/detail/Quyet-dinh-so-06-Ban-hanh-Quy-che-Giai-thuong-Nha-giao-tre-tieu-bieu-31/.

[4] Executive Committee of Industrial University of Ho Chi Minh City, "Decision on the promulgation of the Regulation on the 'Outstanding Young, Competent, and Friendly Civil Servant Award," 2020. [Online]. Available: https://vanphong.tuoitreiuh.vn/vpdtn/detail/Quyet-dinh-so-07-Quy-che-Giai-thuong-Can-bo-cong-chuc-vien-chuc-tre-gioi-than-thien-92/.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

[5] A. T. Dinh , H. N. Nguyen, T. H. Hoang and D. T. Do, "Building a reward and recognition approval system," Department of Information Systems, 2022.

[6] Devmaster Academy via All Laravel, "What is a framework? The difference between a framework and a library.," Devmaster Academy, 2020. [Online]. Available: https://devmaster.edu.vn/framework-la-gi-su-khac-biet-giua-framework-va-library.html.

[7] O. Taylor, "Laravel - The PHP Framework for Artisan," Laravel LLC, [Online]. Available: https://laravel.com/docs.

[8] Admin, "Advantages and disadvantages of Laravel.," https://hoclaravel.net/, 2021. [Online]. Available: https://hoclaravel.net/tong-quan/uu-diem-va-nhuoc-diem-cua-laravel/.

[9] J. Chadwick, T. Snyder và H. Panda, Programming ASP.NET MVC 4, O'Reilly Media, Inc., 2012.

[10] nhiepphong, "What is the MVC model?," hoclaptrinh.vn, 2019. [Online]. Available: https://hoclaptrinh.vn/posts/tim-hieu-mo-hinh-mvc-la-gi.

[11] MongoDB, Inc., "MongoDB Documentation," MongoDB, Inc., 2023. [Online]. Available: https://www.mongodb.com/docs/.

[12] K. Leo, "https://laragon.org/," 2023. [Online]. Available: https://laragon.org/docs/.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

**YSC5.F034**

# ĐÁNH GIÁ KHẢ NĂNG HỎI - ĐÁP CỦA CÁC MÔ HÌNH NGÔN NGỮ CỠ NHỎ TRÊN MIỀN TRI THỨC KHOA HỌC DỮ LIỆU

NGUYỄN VĂN ANH TUẤN (*), NGUYỄN ĐÌNH THANH, CAO NGUYỄN GIA HƯNG, NGUYỄN VĂN HẬU, DƯƠNG TRẦN KIM NGÂN, TRẦN KỲ, NGUYỄN VĂN HOÀI NAM, TRẦN VĂN QUYẾN, NGUYỄN HỮU TÌNH

*Khoa Công nghệ Thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*

** nvatuan3@gmail.com*

**Tóm tắt.** Trong nghiên cứu này, chúng tôi đánh giá khả năng hiểu và đưa ra câu trả lời của các mô hình ngôn ngữ cỡ nhỏ, dưới 1 tỷ tham số. Chúng tôi sử dụng các mô hình BloomZ 560M, GPT2 112.4M, ViT5 large 880M, XGLM 564.5M được tiền huấn luyện trước đó trên tiếng Việt để huấn luyện tiếp trên hơn 102M token dữ liệu miền công nghệ thông tin và khoa học dữ liệu (đo bằng GPT2 của OpenAI). Tiếp đó, chúng tôi tinh chỉnh mô hình trên hơn 26K dữ liệu hỏi đáp, gồm hơn 6K dữ liệu liên quan đến khoa học dữ liệu và 20K dữ liệu tổng quát. Sau cùng, các mô hình được đánh giá trên 450 dữ liệu hỏi đáp của các chủ đề của miền tri thức khoa học dữ liệu bao gồm: Học máy, Học sâu, Khoa học dữ liệu, Phân tích dữ liệu, Kỹ thuật dữ liệu, Xác suất, Thống kê, Trực quan hóa dữ liệu và ngôn ngữ lập trình Python, tổng cộng 450 câu hỏi-đáp được tạo cẩn thận bằng tay. Kết quả cho thấy rằng tuy các mô hình ngôn ngữ này bị hạn chế về mặt tham số, nhưng vẫn có thể hiểu và trả lời ra tốt kết quả.

**Từ khóa.** Khoa học dữ liệu, Hệ thống trả lời câu hỏi, mô hình ngôn ngữ cỡ nhỏ.

## EVALUATE OF QUESTION-ANSWERING CAPABILITIES OF SMALL-SCALE LANGUAGE MODELS IN DATA SCIENCE DOMAIN

**Abstract.** In this study, we evaluate the understanding and answering ability of small language models, under 1 billion parameters. We use the pre-trained BloomZ 560M, GPT2 113.4M, ViT5 large 880M, XGLM 564.5M models on Vietnamese to continue training on over 102M tokens of information technology and data science data (measured by OpenAI's GPT2). Next, we fine-tune the model on over 26K question-answer data, including over 6K data related to data science and 20K general data. Finally, the models are evaluated on 450 question-answer data on topics in the field of data science knowledge including: Machine Learning, Deep Learning, Data Science, Data Analysis, Data Engineering, Probability, Statistics, Data Visualization and Python programming language, a total of 450 questions and answers carefully created by hand. The results show that although these language models are limited in parameters, they can still understand and answer good results.

**Keywords.** Data Science, Question Answering System, small-scaled language models.

## 1 GIỚI THIỆU

Sự thành công của ChatGPT [1] đã khiến cộng đồng nghiên cứu khoa học về xử lý ngôn ngữ tự nhiên nói chung và phát triển mô hình ngôn ngữ nói riêng nhộn nhịp trở lại. Với việc xuất hiện hàng loạt các mô hình ngôn ngữ lớn khác nhau như PaLM [2], Chinchilla [3], GPT3 [4], GPT4 [5], LLaMa [6], Alpaca [7], Vicuna [8], Koala [9], Dolly [10], MPT [11], Falcon [12] đã tạo rất nhiều sản phẩm nhờ sự trợ giúp của cộng đồng nghiên cứu về mô hình ngôn ngữ lớn, như ChatPDF [13], privateGPT [14], HuggingFace Chat [15], …Có thể thấy được rằng việc phát triển các mô hình ngôn ngữ lớn này có thể sẽ không dừng lại sớm trong tương lai gần và có thể phát triển xa hơn nữa [16]. Tuy nhiên, các mô hình ngôn ngữ lớn này dường như không thể được tiếp cận quá rộng rãi đến môi trường nghiên cứu của sinh viên trẻ, nhất là ở những trường đại học,

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

cộng đồng mã nguồn mở, họ đều không có quá nhiều kinh phí dành cho việc duy trì một lượng tính toán khổng lồ. Như việc Meta sử dụng 2048 A100 80GB GPU để phát triển LLaMa [17], GPT3 là 10000 V100 GPU, BloombergGPT là 1.3 triệu giờ GPU 40GB A100, và một số ví dụ khác từ những mô hình khổng lồ PaLM hay Chinchilla, GPT4. Các mô hình ngôn ngữ lớn (từ 1 tỷ đến 13 tỷ tham số) tuy nhỏ hơn, nhưng chi phí để huấn luyện cũng tương đối cao, như Alpaca, Koala, Vicuna, MPT đều sử dụng từ 4 - 8 A100 GPU. Cộng đồng cũng phát triển các cách thức để huấn luyện mô hình ngôn ngữ lớn tiết kiệm hơn, như LoRA, QLoRA, LLaMa Adapter, SqueezeLLM, việc huấn luyện các mô hình dùng kỹ thuật này đối với mô hình 7 tỷ tham số có lẽ chỉ cần 1 GPU với 24GB [18], [19], [20], [21], chi phí lúc này hầu như đã tiết kiệm hơn hẳn, nhưng đổi lại việc tinh chỉnh trên dữ liệu chỉ dẫn có lẽ sẽ không tốt bằng việc tiền huấn luyện truyền thống, vì nguồn văn bản thô lúc nào cũng tốt hơn và đôi khi việc huấn luyện chỉ dẫn chỉ dùng để cho mô hình hiểu cách hỏi và trả lời [22], và đôi khi việc tiền huấn luyện trên văn bản thô lại đem hiệu quả cao trước khi đi huấn luyện chỉ dẫn [23].

Dựa vào các tiêu chí của các trường đại học, cũng như các bạn sinh viên ở Việt Nam, và sự phát triển của ngành công nghiệp AI thì nhu cầu phát triển cũng như ứng dụng AI vào công việc, thỏa mãn đam mê nghiên cứu là rất lớn. Vì thế, nên có một nghiên cứu tập trung vào các mô hình cỡ nhỏ, nhưng cho thấy rằng việc huấn luyện trên mô hình cỡ nhỏ không phải là một sự lựa chọn tồi khi kết quả được cải thiện. LaMini-LM [24], là một loạt các mô hình ngôn ngữ từ lớn đến nhỏ được học dựa trên tri thức của mô hình ngôn ngữ lớn (ChatGPT) để bắt chước tri thức và khả năng của các mô hình lớn hơn, nghiên cứu cho thấy rằng mô hình cỡ nhỏ có thể có kết quả còn tốt hơn mô hình baseline lớn hơn gấp 10 lần (700 triệu tham số so với 7 tỷ tham số), chỉ bằng cách huấn luyện trên một tập dữ liệu chỉ dẫn lớn hơn. Chúng ta có thể học hỏi phương pháp của họ bằng cách thu thập thêm dữ liệu tiền huấn luyện và chỉ dẫn để tinh chỉnh mô hình.

Thêm nữa, ngành khoa học dữ liệu đã và đang rất nổi trong thời điểm hiện tại, có 28 triệu kết quả tìm kiếm trên Bing và 3 tỷ 720 triệu kết quả tìm kiếm trên Google vào ngày 20/06/2023 sử dụng từ khóa "Data Science", nhiều hơn cả "Computer Science" (Bing là 23 triệu 300 nghìn, Google là 2 tỷ 570 triệu) và "Software Engineer" (19 triệu 300 nghìn trên Bing và 742 triệu trên Google). Các con số khổng lồ đó đã cho thấy ngành Khoa học dữ liệu đang phát triển rất tốt ở thời điểm hiện tại, và lượng thông tin cần học cũng như hấp thu rất nhiều. Vì thế việc phát triển một mô hình ngôn ngữ phục vụ mục đích hỏi đáp để tóm lược thông tin nhanh rất cần và tiềm năng của chúng rất được trọng dụng trong việc ứng dụng vào dạy và học, giúp giải quyết các vấn đề đặc thù trong ngành.

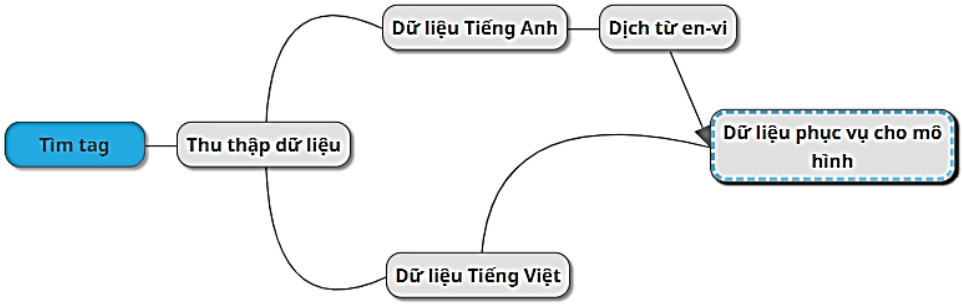
Nghiên cứu này tập trung vào việc đánh giá các kết quả của các mô hình ngôn ngữ cỡ nhỏ, phù hợp với phần cứng được cung cấp miễn phí trên Colab hay Kaggle nhưng vẫn đem lại kết quả tốt nhờ vào việc tiền huấn luyện tiếp tục trên dữ liệu văn bản thô và dữ liệu tinh chỉnh hỏi-đáp trên miền tri thức của Khoa học dữ liệu. Bên cạnh đó, đây cũng là tiền đề cho việc ứng dụng AI vào trong việc dạy và học tại ngành Khoa học dữ liệu của Trường Đại học Công nghiệp Thành phố Hồ Chí Minh.

## 2 TIẾP CẬN

### 2.1 Dữ liệu

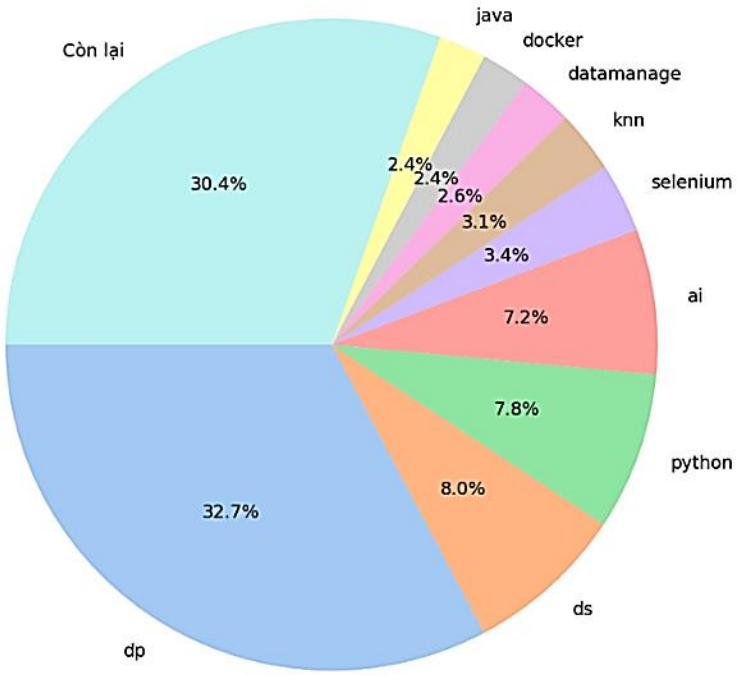
#### 2.1.1 Dữ liệu tiền huấn luyện

**Thu thập:** Nhóm chúng tôi chia việc thu thập các tag cụ thể của ngành khoa học dữ liệu, để tạo thành 175 tag liên quan nhất. Sau đó, chúng tôi tiến hành tìm kiếm các trang web, nguồn tài liệu liên quan trực tiếp cho từng tag, và thu thập dữ liệu từ các nguồn này với thư viện request và BeautifulSoup, kết hợp thêm đa luồng và Scrapy để quá trình thu thập nhanh hơn.



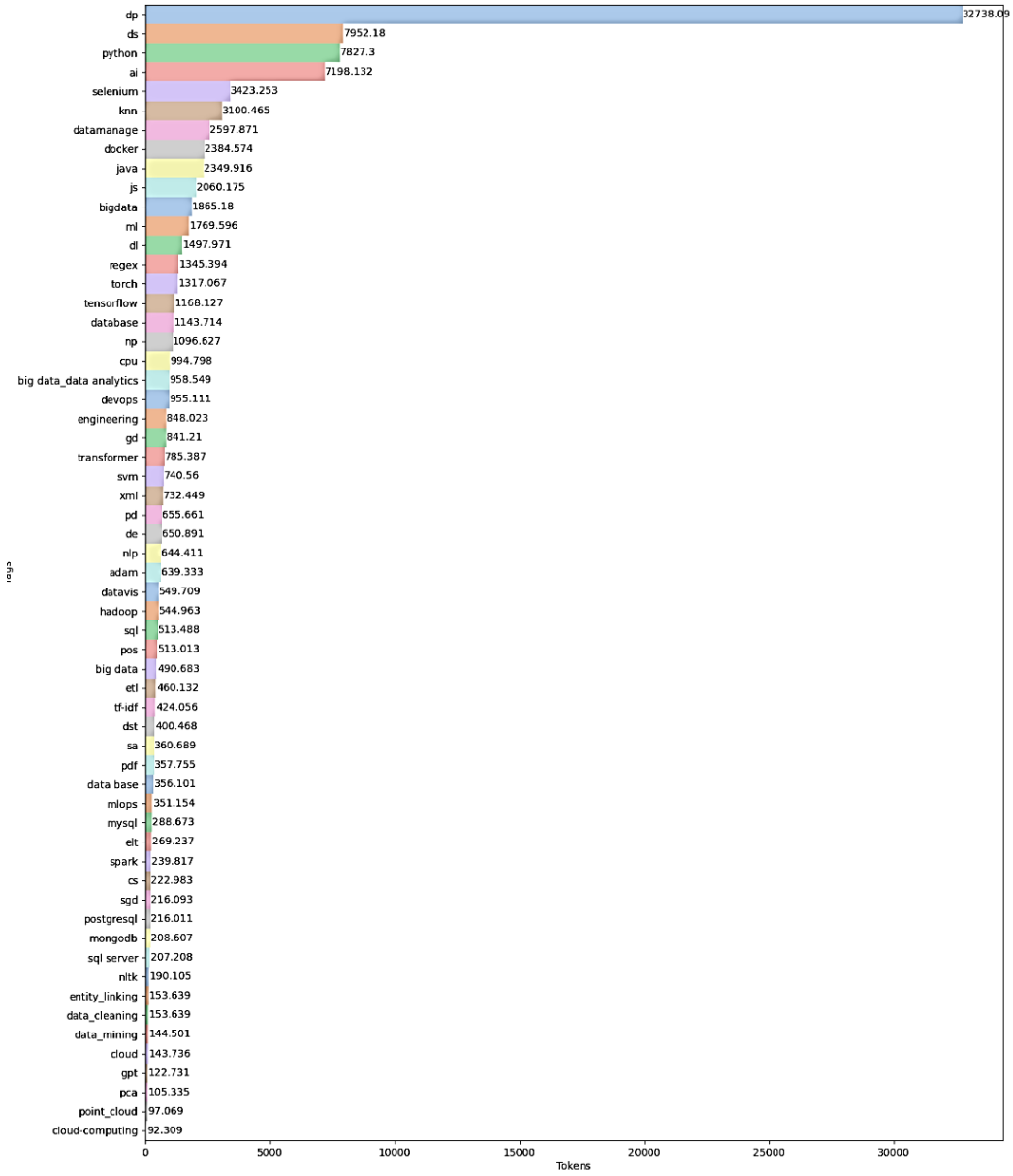

Hình 1: Quy trình tạo ra dữ liệu.

Phân tích:




Hình 2: Biểu đồ tròn 10 tag thể hiện tỉ lệ token (đơn vị nhỏ nhất trong xử lý văn bản, tương ứng với từ, dấu câu hoặc cụm từ giúp máy tính hiểu văn bản) giữa các tag trong tập tiền huấn luyện.

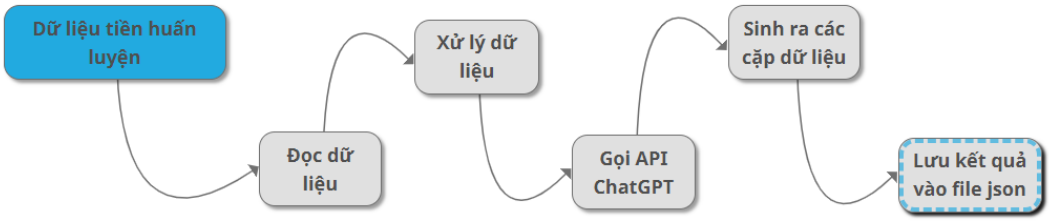

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 3: Biểu đồ số lượng token của mỗi tag.

### 2.1.2 Dữ liệu tinh chỉnh hỏi đáp

**Thu thập:** Từ văn bản thô ở bước thu thập dữ liệu tiền huấn luyện, chúng tôi tiếp tục sử dụng các thư viện python như requests, json và tiktoken,... đồng thời chúng tôi dùng chat.forefront.ai - một trang web cung cấp API ChatGPT kết hợp với sử dụng đa luồng để sinh ra các cặp câu hỏi và trả lời trong thời gian 2 ngày. Việc sử dụng API từ trang web trên giúp tiết kiệm thời gian hơn so với các phương pháp khác như VinAI, Bing, Google Dịch,... Chúng tôi tạo bằng API Chat Completion để đảm bảo chất lượng dữ liệu, phù hợp với ngữ cảnh và bám sát chủ đề khoa học dữ liệu, đồng thời tối ưu hóa quá trình sinh dữ liệu. Kết quả sau hơn 2 ngày làm việc là một tập dữ liệu có kích thước khoảng 83.76MB.




Hình 4. Sơ đồ mô tả quá trình tạo các cặp Câu hỏi và Trả lời.

**Thông tin từ dữ liệu:** Đối với dữ liệu tinh chỉnh hỏi đáp, chúng tôi chỉ lọc ra những tag có mức độ thông tin cao và liên quan chặt chẽ với miền tri thức của khoa học dữ liệu. Cụ thể là các tag như ML, DL, DS, SQL, PD, NP, Cloud Computing,... Chúng tôi lọc ra một số tag chính nhưng vẫn đảm bảo sự đa dạng từ dữ liệu. Từ những tag này, chúng tôi tạo ra 169,481 cặp dữ liệu hỏi đáp bằng ChatGPT, qua quá trình tinh lọc, chúng tôi thu thập được bộ hỏi-đáp còn lại chứa trên 6K dữ liệu, liên quan đến miền tri thức khoa học dữ liệu và các câu trả lời của dữ liệu thu thập đã được kiểm chứng bởi chuyên gia. Dưới đây là một số mẫu dữ liệu có trong tập tinh chỉnh hỏi đáp của chúng tôi:

Bảng 1: Thông tin Tags.

| Tag Code | Tag |
|---|---|
| ML | Machine Learning |
| CLOUD | Cloud Computing |
| SQL | Database |
| PY | Python |
| TF | Tensorflow |
| PT | Pytorch |
| NP | Numpy |
| DL | Deep Learning |
| PD | Pandas |
| DS | Data Science |
| DE | Data Engineer |
| DA | Data Analyst |
| DST | Data Scientist |
| AI | Trí Tuệ Nhân Tạo |
| ETC | Các tag khác |

Dưới đây là mẫu prompt chúng tôi dùng trong ChatGPT API để tạo các cặp dữ liệu hỏi-đáp từ các văn bản đã có, với *{tag}* và *{text}* hai vị trí chúng tôi sử dụng để thay thế tag và đoạn văn bản được cung cấp.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

```
I ONLY ACCEPT RESULTS IN VIETNAMESE Based on the following text about {tag}, please create at least 10 questions and corresponding answers, and each question can be long or short. Questions should be as general as possible Please generate questions that are context-dependent or completely independent of the given context. If the question completely independent of the given context, please fill blank in context.
Exampletrue context:
[
  {
    'question': 'Convert from celsius to fahrenheit. Temperature in Celsius: 15',
    'answer': 'Temperature in Fahrenheit: 59'
  }

    {
    'question': 'Supervised learning là gì?',
    'answer': 'Supervised learning là một phương pháp học có giám sát trong machine learning',
  }
    {
    'question': 'Use the given data to calculate the median.\n1, 2, 3, 4, 5, 6, 7, 8, 9, 10',
    'answer': '5.5'
  }
]

I ONLY ACCEPT RESULTS (question, answer) IN VIETNAMESE LANGUAGE AND RETURNED IN JSON FORMAT LIKE THIS: {[{ 'question': 'question 1' ,'answer': 'answer 1' }, { 'question': 'question 2', 'answer': 'answer 2' }]}

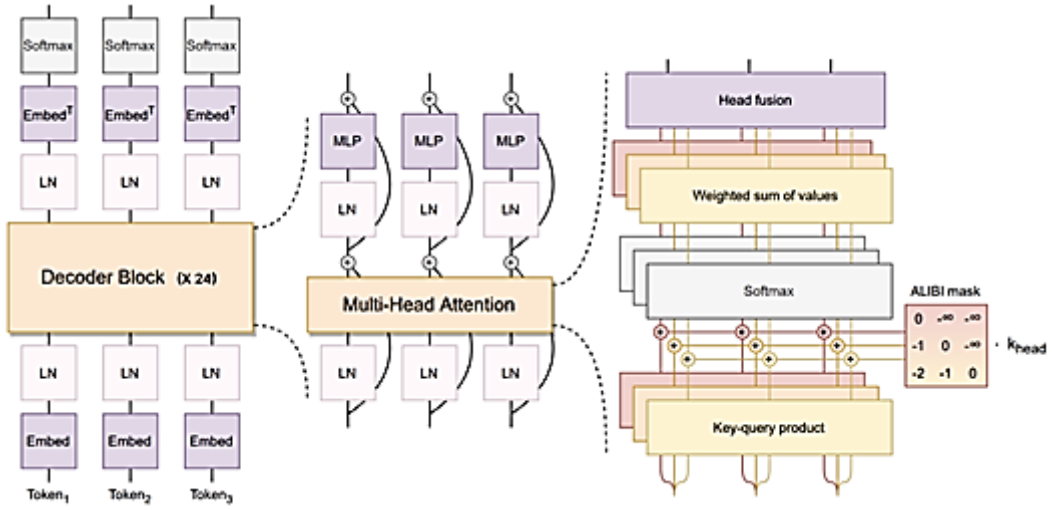
Text: {text}
```

Dưới đây là một số mẫu dữ liệu hỏi đáp đã tạo ra được từ prompt ở trên:
*Q: "Deep Learning khác gì so với Machine Learning?"*
*A: "Deep Learning được xây dựng dựa trên những khái niệm phức tạp hơn Machine Learning."*
*Q: "Deep Learning là gì?"*
*A: "Deep Learning là một phương pháp học máy sâu trong đó mô hình học tập được xây dựng từ nhiều lớp xử lý thông tin."*
*Q: "Deep Learning có những ưu điểm gì?"*
*A: "Deep Learning có khả năng học tập và tự động cải thiện hiệu suất mô hình, có thể xử lý dữ liệu lớn và phức tạp, và có thể tạo ra các mô hình dự đoán chính xác hơn so với các phương pháp truyền thống khác."*

## 2.2 Kiến trúc và huấn luyện mô hình ngôn ngữ

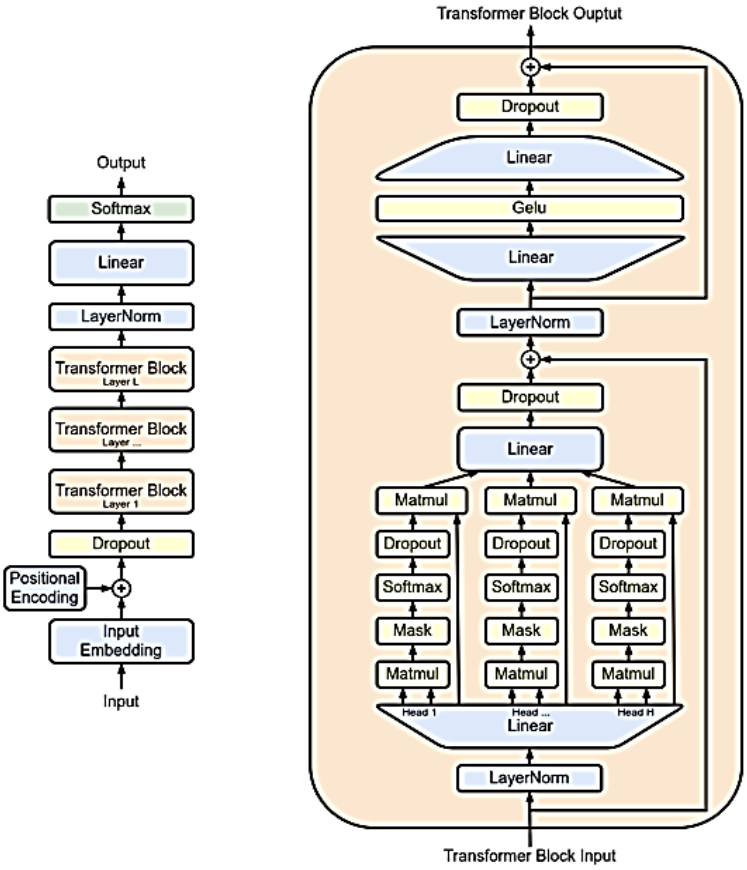
### 2.2.1 Mô hình ngôn ngữ cơ sở

**BLOOMZ:** Bloomz là một dự án được thực hiện bởi đội ngũ các nhà nghiên cứu thuộc BigScience [25]. Các nhà nghiên cứu ở BigScience tinh chỉnh các mô hình ngôn ngữ đa ngôn ngữ được đào tạo trước BLOOM trên tập dữ liệu gồm nhiều ngôn ngữ (xP3), và kết quả cho thấy các mô hình sau quá trình tinh chỉnh có khả năng tổng quát hóa qua các tác vụ và ngôn ngữ chưa từng được tiếp xúc. Từ đó, chúng tôi quyết định chọn Bloomz để thực nghiệm. Có 6 mô hình Bloomz được BigScience giới thiệu tương ứng với 6 mô hình Bloom [26]. Tuy nhiên, do gặp nhiều hạn chế phần cứng, nhóm chúng tôi quyết định lựa chọn mô hình Bloomz-560m với 560 triệu tham số và 24 layer để thực nghiệm trong dự án của chúng tôi.




Hình 5. Kiến trúc mô hình Bloomz (560m).

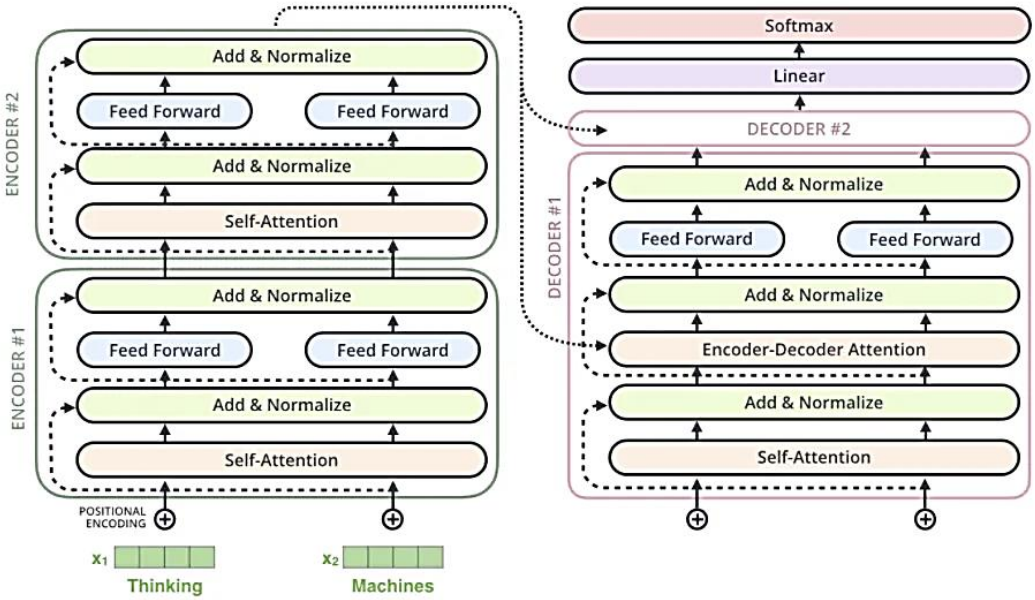
**GPT2:** là một mô hình ngôn ngữ mạnh mẽ phát triển bởi OpenAI. Dựa trên kiến trúc Transformer, GPT-2 có khả năng xử lý ngôn ngữ tự nhiên và tạo ra văn bản có ý nghĩa. Với việc được đào tạo trên dữ liệu lớn từ Internet, GPT-2 có khả năng hiểu và phân tích ngữ cảnh một cách toàn diện. Mô hình này có thể sinh tiếp tục văn bản, thực hiện nhiều tác vụ ngôn ngữ và đã được ứng dụng rộng rãi trong lĩnh vực chatbot.




Hình 6. Kiến trúc mô hình GPT2.

**ViT5**: là một mô hình ngôn ngữ phát triển dựa trên công nghệ học sâu và kiến trúc Transformer. Được phát triển bởi nhóm nghiên cứu ở VietAI, dựa trên mô hình T5 (text to text transformer) của Google.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 7: Kiến trúc mô hình ViT5

Được đào tạo trên bộ dữ liệu gồm 71GB dữ liệu tiếng Việt trong bộ CC 100 dataset [27] cho quá trình pretraining cùng kiến trúc encoder-decoder với 24 layer. Vit5-large có thể tạo ra văn bản có ý nghĩa và tự nhiên, và được ứng dụng rộng rãi trong các ứng dụng chatbot để tương tác với người dùng tiếng Việt. ViT5-lager chứng minh được sự hiệu quả trong nhiều nhiệm vụ xử lý ngôn ngữ tự nhiên cùng với việc được pretraining trên một lượng lớn data tiếng Việt là lý do nhóm chúng tôi chọn mô hình này để thực nghiệm.
**XGLM:** là một mô hình huấn luyện đa ngôn ngữ, được dựa trên kiến trúc GPT-3 của OpenAI [28], và điều chỉnh tùy chỉnh cho tiếng Việt. Mô hình này đã được đánh giá tích cực trong việc áp dụng và thích ứng vào ngôn ngữ cụ thể. Với phiên bản 564 triệu tham số công khai từ Facebook AI, XLGM đã trải qua quá trình tiền huấn luyện đa ngôn ngữ, với tỉ lệ tiếng Việt khoảng 0,6% tương ứng với 11,199 triệu token. Kết quả thực nghiệm cho thấy mô hình XLGM đạt được hiệu suất cạnh tranh trong nhiều nhiệm vụ ngôn ngữ, từ dịch máy đến phân tích cảm xúc. Phân tích chi tiết về tham số và kiến trúc đã cho thấy sự hiệu quả của việc điều chỉnh mô hình cho ngôn ngữ mục tiêu. Mô hình XLGM là một bước tiến quan trọng trong nghiên cứu và ứng dụng xử lý ngôn ngữ tự nhiên, tận dụng tiềm năng của việc thích ứng mô hình huấn luyện đa ngôn ngữ vào ngôn ngữ cụ thể.

### 2.2.2 Phương pháp huấn luyện

**Pretraining:** Chúng tôi tiến hành tiền huấn luyện mô hình ngôn ngữ cơ sở trên các mô hình cho chatbot. Thay vì khởi tạo lại từ đầu, chúng tôi tiếp tục quá trình huấn luyện để tiết kiệm thời gian và tài nguyên. Điều này giúp cải thiện hiệu suất mô hình mà không cần bắt đầu lại từ đầu. Quá trình huấn luyện yêu cầu phần cứng mạnh mẽ và thời gian huấn luyện phụ thuộc vào quy mô dữ liệu và phần cứng sử dụng. Các tham số của mô hình được tinh chỉnh để tối ưu hiệu suất.
**Fine-tuning:** Sau giai đoạn huấn luyện mô hình trên văn bản thuần để mô hình có khả năng hiểu tri thức cơ bản, chúng tôi tiếp tục quá trình huấn luyện mô hình trên dữ liệu tinh chỉnh cho việc hỏi đáp các vấn đề trong lĩnh vực khoa học dữ liệu. Tuy nhiên, do hạn chế về phần cứng và mục đích chính là kiểm tra khả năng nội suy thông tin và tri thức của mô hình, quá trình này được thực hiện với mục tiêu thử nghiệm hơn là huấn luyện chính thức.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Bảng 2. Thông tin huấn luyện mô hình.

| | **Bloomz** | **ViT5** | **GPT2** | **XGLM** |
|---|---|---|---|---|
| Phần cứng huấn luyện | P-100 | | | |
| Thời gian huấn luyện | 12h | 20h | 8h | 12h |
| Epoch | 15 | 15 | 10 | 15 |

# 3 KẾT QUẢ

## 3.1 Kết quả tiền huấn luyện

Trong phần này, chúng tôi đánh giá hiệu suất của mỗi mô hình dựa trên hai chỉ số chính, bao gồm:

- Loss trên tập test trước khi thực hiện quá trình pretrain.
- Loss trên tập test sau khi thực hiện quá trình pretrain.

Chúng tôi tổng hợp kết quả đánh giá thành một bảng có kích thước (số lượng mô hình) x (3 cột), trong đó cột thứ hai là loss trước khi pretrain của tập test, cột thứ ba là loss sau khi pretrain của tập test. Bảng 3 cung cấp cái nhìn tổng quan về hiệu suất của từng mô hình và sự tác động của quá trình pretrain lên kết quả đạt được.

Việc đánh giá kết quả tiền huấn luyện giúp chúng tôi hiểu rõ hơn về sự phát triển và cải thiện của các mô hình trước khi tiến hành giai đoạn fine-tuning.

Bảng 3. Bảng so sánh loss của tập test trước khi pretrain và sau khi pretrain.

| Mô hình | Loss trước khi pretrain của tập test | Loss sau khi pretrain của tập test |
|---|---|---|
| **Bloomz** | 7.764 ± 0.336 | 2.336 ± 1.762 |
| **ViT5** | 5.737 ± 1.621 | 1.305 ± 0.855 |
| **GPT2** | 2.678 ± 0.543 | 0.422 ± 0.032 |
| **XGLM** | 3.484 ± 0.472 | 0.742 ± 0.246 |

Bảng 4. Bảng chi tiết thông số các siêu tham số của các mô hình.

| **Hyperparameter** | **Bloomz** | **ViT5** | **GPT2** | **XGLM** |
|---|---|---|---|---|
| Architecture design | Decoder-only | Encoder-decoder | Decoder-only | Decoder-only |
| **Tham số mô hình** | | | | |
| Parameters | 559M | 880M | 124.4M | 564.5 M |
| Precision | float16 | | | |
| Layers | 24 | 24 | 12 | 24 |
| Hidden dim. | 1024 | 1024 | 768 | 1024 |
| Attention heads | 16 | 16 | 12 | 16 |
| Vocab size | 250,680 | 36,100 | 50,257 | 256,008 |
| Sequence length | 2048 | 2048 | 1024 | 2048 |

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| Activation | GELU | | | |
|---|---|---|---|---|
| **Siêu tham số tiền huấn luyện** | | | | |
| Global Batch Size | 256 | | | |
| Learning | 5.0e-5 | 2.0e-5 | 1.5e-3 | 1.5e-3 |
| Total training tokens | 95,553,672 | | | |
| Warmup tokens | 477,768 | | | |
| Decay style | cosine | | | |
| Min. learning rate | 1e-5 | | | |
| Adam ($\beta_1$, $\beta_2$) | (0.9, 0.999) | | | |
| Weight decay | 0.01 | | | |
| Gradient clipping | 64 | 256 | 8 | 8 |
| **Siêu tham số tinh chỉnh** | | | | |
| Global Batch Size | 256 | 256 | 256 | 128 |
| Gradient accumulation | 64 | 64 | 64 | 64 |
| Learning | 1.0e-5 | 1.0e-5 | 1.5e-3 | 1.5e-5 |
| Decay style | Cosine | Cosine | Cosine | Cosine |
| Weight decay | 0.01 | 0.01 | 0.01 | 0.01 |

## 3.2 Kết quả của tinh chỉnh hỏi đáp

### 3.2.1 Phương pháp đánh giá

**BLEU:** phương pháp đánh giá chất lượng đầu ra của hệ thống chatbot. Nó tính toán sự tương đồng giữa các n-gram trong đầu ra dự đoán và đầu ra tham chiếu, với thang đo từ 0 đến 1 trong đó 1 biểu thị sự tương đồng hoàn toàn giữa đầu ra dự đoán và đầu ra tham chiếu và 0 là biểu thị sự không tương đồng giữa đầu ra dự đoán và đầu ra tham chiếu.

**ROUGE:** phương pháp đánh giá chất lượng đầu ra của hệ thống chatbot. Nó đo lường sự tương đồng giữa các n-gram trong đầu ra dự đoán và đầu ra tham chiếu tạo bởi con người. ROUGE tính toán các giá trị R1, R2 và R3 để đo lường khớp từng từ, cụm từ 2 từ và 3 từ liên tiếp. Giá trị ROUGE nằm trong khoảng từ 0 đến 1, với 1 biểu thị sự khớp hoàn toàn và 0 biểu thị không khớp.

### 3.2.2 Kết quả

Bảng 5. Thống kê trung bình và độ lệch chuẩn của các tag trong đánh giá chatbot.

| Tag | Mean | Std |
|---|---|---|
| DA | 1.5 | 1.5 |
| DE | 2.2 | 2.4 |
| DL | 1.1 | 0.8 |
| DS | 1.4 | 1.6 |
| ML | 1.6 | 1.9 |
| Probability | 1.3 | 1.5 |
| Python | 3.8 | 2.4 |
| Statistics | 3.8 | 2.4 |
| Visualization | 1.2 | 1.3 |

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

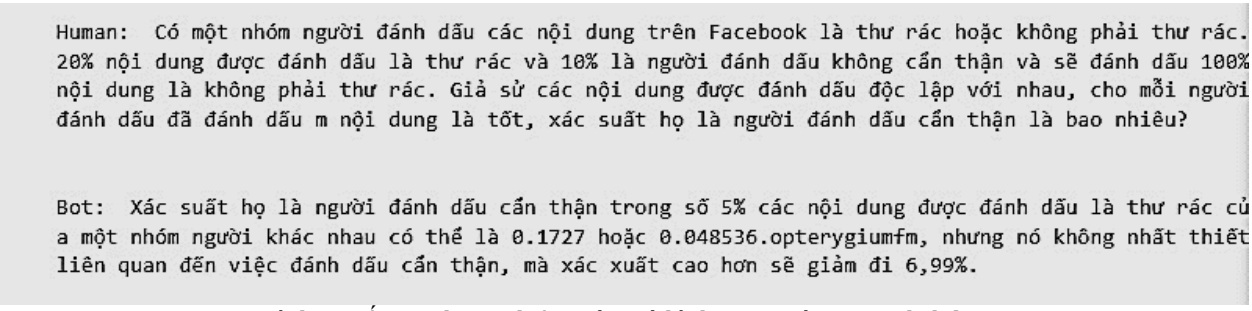
Bảng 6. Bảng kết quả đánh giá BLEU (%) của mô hình.

| TAG | BLEU (%) | | | |
|---|---|---|---|---|
| | **Bloomz** | **ViT5** | **GPT2** | **XGLM** |
| DA | 0.33 | 4.41 | 0.92 | 0.51 |
| DE | 0.56 | 10.56 | 1.01 | 0.82 |
| DL | 0.52 | 3.38 | 0.74 | 0.75 |
| DS | 0.41 | 10.12 | 0.67 | 0.35 |
| ML | 0.00 | 3.17 | 0.84 | 0.00 |
| Probability | 0.00 | 24.87 | 0.00 | 0.00 |
| Python | 0.00 | 8.36 | 2.08 | 1.08 |
| Statistics | 0.00 | 6.68 | 0.00 | 0.00 |
| Visualization | 0.35 | 7.27 | 0.36 | 0.00 |
| Trung bình | 0.24 | 8.76 | 0.74 | 0.39 |

Bảng 7. Bảng kết quả đánh giá ROUGE (%) của mô hình.

| TAG | ROUGE (%) | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | **BLOOMZ** | | | **ViT5** | | | **GPT-2** | | | **XGLM** | | |
| | **R1** | **R2** | **R3** | **R1** | **R2** | **R3** | **R1** | **R2** | **R3** | **R1** | **R2** | **R3** |
| DA | 11.70 | 1.91 | 6.24 | 26.13 | 13.89 | 20.37 | 13.32 | 2.64 | 7.93 | 14.52 | 2.36 | 6.58 |
| DE | 8.02 | 1.31 | 4.59 | 31.21 | 18.82 | 27.11 | 10.13 | 2.61 | 6.92 | 11.27 | 2.10 | 5.92 |
| DL | 11.39 | 1.71 | 5.71 | 24.14 | 9.43 | 17.76 | 11.21 | 1.82 | 7.09 | 14.04 | 2.73 | 6.14 |
| DS | 10.45 | 1.72 | 5.40 | 24.38 | 12.17 | 21.62 | 8.01 | 1.68 | 5.52 | 10.47 | 1.76 | 5.24 |
| ML | 12.94 | 1.64 | 6.09 | 17.97 | 6.61 | 12.85 | 11.28 | 2.03 | 6.07 | 14.85 | 2.41 | 6.41 |
| Probability | 1.53 | 0.09 | 1.40 | 49.73 | 42.10 | 49.56 | 3.30 | 0.47 | 2.84 | 0.99 | 0.12 | 0.85 |
| Python | 9.72 | 1.61 | 5.23 | 29.06 | 17.28 | 24.37 | 13.50 | 3.53 | 8.50 | 10.94 | 2.16 | 6.20 |
| Statistics | 9.41 | 1.25 | 4.45 | 26.21 | 10.87 | 18.71 | 9.77 | 1.44 | 6.20 | 11.90 | 2.09 | 5.51 |
| Visualization | 7.46 | 1.21 | 4.48 | 24.66 | 13.04 | 20.67 | 7.63 | 1.47 | 4.80 | 14.52 | 2.36 | 6.58 |
| Trung bình | 9.18 | 1.38 | 4.84 | 28.16 | 16.02 | 23.66 | 9.79 | 1.97 | 6.21 | 10.82 | 1.89 | 5.29 |

### 3.2.3 Kết quả thực nghiệm




Hình 8. Kết quả thực nghiệm trên mô hình ViT5 trên tag Probability

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

```
Human:  Python là gì?


Bot:  Python là một ngôn ngữ lập trình được tạo ra bởi nhà toán học Guido Van Rossum và được giới t
hiệu chính thức vào năm 1991. Đây là một trong những ngôn ngữ lập trình được sử dụng nhiều nhất thế
giới và đã cách mạng hóa ngành khoa học máy tính. Nó được viết bằng các ngôn ngữ lập trình động, kh
ông có kết cấu và không phụ thuộc vào PATH (Hệ thống tệp). Trình biên dịch và trình đọc mã cũng đượ
c viết bằng Python. Điều này làm cho Python trở thành một trong những ngôn ngữ lập trình dễ đọc và
dễ viết nhất. Python cung cấp cho người dùng các thư viện và framework để làm việc với các lĩnh vực
khác nhau như khoa học dữ liệu, phát triển web, học máy, xử lý ngôn ngữ tự nhiên, v.v. Nó cũng cung
cấp nhiều thư viện và framework cho phát triển web và phân tích dữ liệu. Nó cũng có một loạt các th
ư viện và framework liên quan như Beautiful Soup, Beautiful JSON, JSONP, Tensorflow, Keras, H2O, Mi
nd
```

Hình 9. Kết quả thực nghiệm trên mô hình BLOOMZ trên tag Python

```
Human:  Khoa học dữ liệu là gì?


Bot:  Khoa học dữ liệu là nghiên cứu về cách xây dựng các mô hình dự đoán và phân loại có khả năng
theo dõi các biến đầu ra liên tục hoặc chuỗi. Các mô hình này có thể được sử dụng cho cả phân loại
và hồi quy. Các biến đầu ra có thể là các giá trị thực hoặc các giá trị đầu ra của các mô hình dự đ
oán. Khoa học dữ liệu là một nhánh của học máy, nó tập trung vào việc xây dựng và huấn luyện các mô
hình dự đoán và phân loại mạnh mẽ sử dụng dữ liệu chuỗi thời gian. Các mô hình dự đoán thường được
sử dụng cho các tác vụ phân loại và hồi quy, trong khi các mô hình phân loại và hồi quy thường được
sử dụng cho các tác vụ theo dõi biến đầu ra nhị phân. Khoa học dữ liệu là một lĩnh vực phát triển n
hanh chóng, với nhiều dự án và công cụ nghiên cứu đang được phát triển. Các công cụ học máy phổ biế
n khác bao gồm BigDL, XGBoost và Random Forests
```

Hình 10. Kết quả thực nghiệm trên mô hình BLOOMZ trên tag DS

```
Human:  Khoa học dữ liệu là gì?


Bot:  Có hai dạng phản hồi: Phản hồi từ (từ) và Phản hồi từ (từ). Phản hồi từ bao gồm các câu được
viết dưới dạng câu hỏi có ý nghĩa, câu hỏi mà chúng ta có thể lặp lại nhiều lần trong quá trình trả
lời các câu đúng. Phản hồi từ bao gồm các câu trả lời đã được lặp lại. Phản hồi từ bao gồm các câu
trả lời đã được lặp lại.
- Câu hỏi mà bạn đang nói về có thể được đặt ra để xác định câu trả lời. Hãy nhớ rằng câu hỏi có ý
nghĩa là phản hồi từ được lặp lại. Điều này không bao gồm tất cả câu hỏi có ý nghĩa. Các câu hỏi th
ường được lặp lại bởi một người nào đó với những người khác hoặc từ một tổ chức khác mà chúng ta đa
ng nói đến. Một số câu hỏi được trả lời bởi người này. Các câu hỏi khác thường được trả lời bởi ngư
ời khác. Bạn cần phải nhớ các câu hỏi này, và không nên đặt câu hỏi như vậy
```

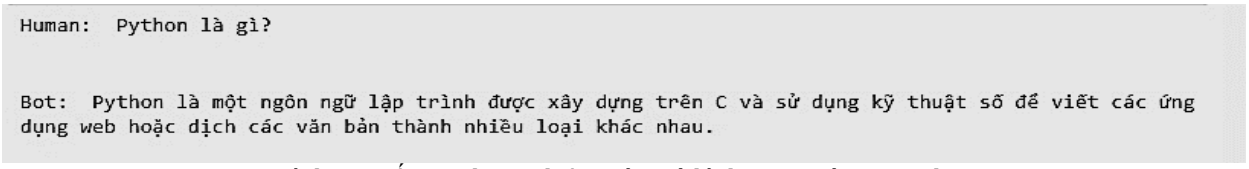
Hình 11. Kết quả thực nghiệm trên mô hình GPT-2 trên tag DS

```
Human:  Khoa học dữ liệu là gì?

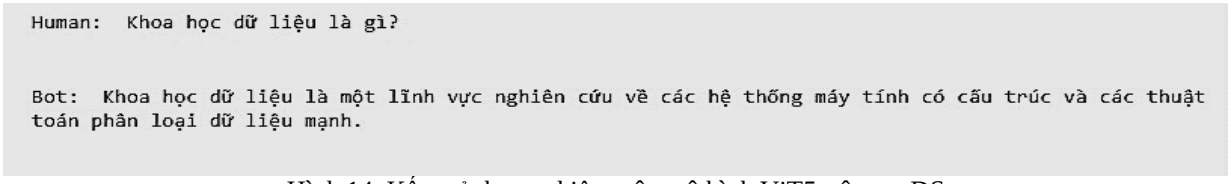

Bot:  Ứng dụng khoa học dữ liệu là một lĩnh vực của khoa học dữ liệu, bao gồm việc tạo ra các mô hì
nh và dự đoán các mối quan hệ giữa các dữ liệu. Nó bao gồm việc tạo ra các mô hình và dự đoán các m
ối quan hệ giữa các dữ liệu bằng cách sử dụng các thuật toán và công cụ khác nhau. Nó bao gồm việc
tạo ra các mô hình và dự đoán các mối quan hệ giữa các dữ liệu bằng cách sử dụng các thuật toán và
công cụ khác nhau. Nó bao gồm việc tạo ra các mô hình và dự đoán các mối quan hệ giữa các dữ liệu b
ằng cách sử dụng các thuật toán và công cụ khác nhau. Nó bao gồm việc tạo ra các mô hình và dự đoán
các mối quan hệ giữa các dữ liệu bằng cách sử dụng các thuật toán và công cụ khác nhau. Nó bao gồm
việc tạo ra các mô hình và dự đoán các mối quan hệ giữa các dữ liệu bằng cách sử dụng các thuật toá
n và công cụ khác nhau. Nó bao gồm
```

Hình 12. Kết quả thực nghiệm trên mô hình XGLM trên tag Python

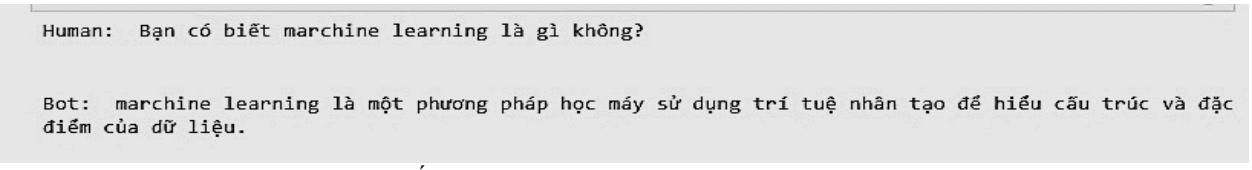
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 13. Kết quả thực nghiệm trên mô hình ViT5 trên tag Python




Hình 14. Kết quả thực nghiệm trên mô hình ViT5 trên tag DS




Hình 15. Kết quả thực nghiệm trên mô hình ViT5 trên tag DS

## 3.3 Thảo luận

**Hiệu suất BLEU:** Kết quả BLEU (%) cho mỗi mô hình chatbot cho thấy mức độ tương đồng giữa đầu ra dự đoán của chatbot và đầu ra tham chiếu. Mô hình ViT5 có hiệu suất BLEU cao nhất với giá trị trung bình là 8.76, cho thấy mô hình này có xu hướng đưa ra các dự đoán tốt hơn so với các mô hình còn lại, bên cạnh đó thì GPT2 với giá trị trung bình là 0.74 cho thấy mô hình đem lại khá tốt, trong khi Bloomz và XGLM có hiệu suất thấp hơn, với giá trị BLEU trung bình lần lượt là 0.24 và 0.39, điều này cho thấy chúng có thể cần được cải thiện để đạt được kết quả tốt hơn. Từ đó cho thấy mô hình ViT5 và GPT2 có khả năng sinh ra các câu trả lời gần giống với câu trả lời chính xác.

**Hiệu suất ROUGE:** Chúng tôi nhận thấy hiệu suất của các TAG có sự khác biệt. ViT5 có hiệu suất ROUGE cao nhất với các giá trị R1, R2 và R3 đều cao hơn so với các mô hình còn lại, cho thấy mô hình này có xu hướng tạo ra các câu dịch gần giống với văn bản tham chiếu. Trong khi đó, GPT-2 có hiệu suất ROUGE khá tốt với giá trị R1 và R3 tương đối cao, tuy nhiên giá trị R2 không cao bằng các mô hình khác. BLOOMZ và XGLM có hiệu suất thấp hơn, với các giá trị ROUGE trung bình thấp hơn nhiều so với ViT5 và GPT-2. Tóm lại, hiệu suất tốt của mô hình ViT5 trong việc tạo ra các câu dịch gần giống văn bản tham chiếu. GPT-2 cũng đạt được một hiệu suất tốt, trong khi BLOOMZ và XGLM cần được cải thiện để tăng cường độ chính xác trong việc tạo ra các câu dịch. Từ mô hình ViT5 cho thấy khả năng khớp từng từ và cụm từ liên tiếp giữa chatbot và đầu ra tham chiếu có thể cải thiện.

## 4 KẾT LUẬN

Trong nghiên cứu này, chúng tôi đã đánh giá hiệu suất của các mô hình chatbot (Bloomz, ViT5, GPT2 và XGLM) trong việc trả lời các câu hỏi liên quan đến các lĩnh vực như DA, DE, DL, DS, ML, Probability, Python, Statistics và Visualization. Đánh giá được thực hiện dựa trên hai chỉ số quan trọng là BLEU (%) và ROUGE (%). Kết quả cho thấy mô hình ViT5 và XGLM có khả năng tạo ra các câu trả lời có độ chính xác và độ tương đồng ngữ nghĩa cao hơn so với mô hình Bloomz và GPT2. Đặc biệt, mô hình ViT5 đã đạt kết quả tốt nhất trong tất cả các thẻ, đồng thời vượt trội với độ chính xác và độ tương đồng ngữ nghĩa cao nhất trong các thẻ DE, DS, và Probability. Tuy nhiên, cần nhận thức về các hạn chế và yếu điểm trong quá trình đánh giá hiệu suất của các mô hình chatbot. Một số thẻ có BLEU và ROUGE thấp có thể do đặc thù của chủ đề hoặc thiếu dữ liệu tương ứng. Hơn nữa, việc đánh giá chỉ số dựa trên hai chỉ số này có thể hạn chế trong việc đánh giá sự tương tác và tương thích với người dùng thực tế. Tổng quát, mô hình ViT5 đã cho thấy tiềm năng và hiệu quả trong việc xây dựng một chatbot có khả năng hiểu và trả lời các câu hỏi từ người dùng. Các kết quả nghiên cứu này cung cấp một cơ sở để phát triển và cải thiện chatbot trong tương lai, đồng thời mở ra những hướng nghiên cứu tiềm năng khác nhau để nâng cao hiệu suất và tương tác của chatbot trong các lĩnh vực ứng dụng khác nhau.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

## 5 TÀI LIỆU THAM KHẢO

[1] S. Altman, "Creating safe AGI that benefits all of humanity," OpenAI, 9 12 2015. [Online]. Available: https://openai.com/. [Accessed 1 10 2022].

[2] A. Chowdhery, "PaLM: Scaling Language Modeling with Pathways," Arxiv, 5 10 2022. [Online]. Available: https://arxiv.org/abs/2204.02311. [Accessed 9 1 2023].

[3] J. Hoffmann, "Training Compute-Optimal Large Language Models," Arxiv, 29 3 2022. [Online]. Available: https://arxiv.org/abs/2203.15556. [Accessed 1 2 2023].

[4] T. B. Brown, "Language Models are Few-Shot Learners," Arxiv, 22 7 2020. [Online]. Available: https://arxiv.org/abs/2005.14165. [Accessed 3 12 2022].

[5] A. Ecoffet, "GPT-4 Technical Report," Arxiv, 27 3 2023. [Online]. Available: https://arxiv.org/abs/2303.08774. [Accessed 12 4 2023].

[6] G. Izacard, "LLaMA: Open and Efficient Foundation Language Models," Arxiv, 27 2 2023. [Online]. Available: https://arxiv.org/abs/2302.13971. [Accessed 3 3 2023].

[7] R. Taori, "Alpaca: A Strong, Replicable Instruction-Following Model," CRFM, 13 3 2023. [Online]. Available: https://crfm.stanford.edu/2023/03/13/alpaca.html. [Accessed 29 3 2023].

[8] T. V. Team, "Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality," LMSYSORG, 30 3 2023. [Online]. Available: https://lmsys.org/blog/2023-03-30-vicuna/. [Accessed 10 4 2023].

[9] X. Geng, "Koala: A Dialogue Model for Academic Research," BAIR, 3 4 2023. [Online]. Available: https://bair.berkeley.edu/blog/2023/04/03/koala/. [Accessed 26 4 2023].

[10] Matthayes, "Databrickslabs," Github, 24 3 2023. [Online]. Available: https://github.com/databrickslabs/dolly. [Accessed 4 4 2023].

[11] T. M. N. Team, "Introducing MPT-7B: A New Standard for Open-Source, Commercially Usable LLMs," MOSAICML, 5 5 2023. [Online]. Available: https://www.mosaicml.com/blog/mpt-7b. [Accessed 14 5 2023].

[12] Falcon, "Call for Proposals: Falcon 40B - World's Top Ranked AI Model Empowers Exceptional Use Cases with Training Compute Power in Call for Proposals," Hugging Face, 25 5 2023. [Online]. Available: https://huggingface.co/tiiuae/falcon-40b. [Accessed 29 5 2023].

[13] pdf.chat, "Upload and Chat PDF Analyze .pdf files with AI," Chat PDF.ai, 4 8 2018. [Online]. Available: https://pdf.chat/. [Accessed 29 11 2022].

[14] Imartinez, "PrivateGPT," Github, 8 5 2023. [Online]. Available: https://github.com/imartinez/privateGPT. [Accessed 28 5 2023].

[15] HF, "Users of HuggingChat are authenticated through their HF user account.," HuggingChat, 25 4 2023. [Online]. Available: https://huggingface.co/chat/privacy. [Accessed 1 6 2023].

[16] C. Dilmegani, "The Future of Large Language Models," AImultiple, 10 4 2023. [Online]. Available: https://research.aimultiple.com/future-of-large-language-models/. [Accessed 5 5 2023].

[17] G. Izacard, "LLaMA: Open and Efficient Foundation Language Models," Arxiv, 27 2 2023. [Online]. Available: https://arxiv.org/abs/2302.13971. [Accessed 4 4 2023].

[18] Juletx, "Add machine-translated Alpaca dataset in 6 languages and finetuned mo…," Github, 13 3 2023. [Online]. Available: https://github.com/tloen/alpaca-lora. [Accessed 7 5 2023].

[19] E. Beeching, "Fine-tuning 20B LLMs with RLHF on a 24GB consumer GPU," Huggingface, 9 3 2023. [Online]. Available: https://huggingface.co/blog/trl-peft. [Accessed 28 4 2023].

[20] Artidoro, "QLoRA: Efficient Finetuning of Quantized LLMs," Github, 24 5 2023. [Online]. Available: https://github.com/artidoro/qlora. [Accessed 20 7 2023].

[21] ChrisLiu6, "LLaMA-Adapter: Efficient Fine-tuning of LLaMA," Github, 12 6 2023. [Online]. Available: https://github.com/OpenGVLab/LLaMA-Adapter. [Accessed 1 7 2023].

[22] O. Levy, "LIMA: Less Is More for Alignment," Arxiv, 18 5 2023. [Online]. Available: https://arxiv.org/abs/2305.11206. [Accessed 6 6 2023].

[23] S. Gururangan, "Don't Stop Pretraining: Adapt Language Models to Domains and Tasks," 23 4 2020. [Online]. Available: https://arxiv.org/abs/2004.10964. [Accessed 14 2 2023].

[24] M. Wu, "LaMini-LM: A Diverse Herd of Distilled Models from Large-Scale Instructions," Arxiv, 27 4 2023. [Online]. Available: https://arxiv.org/abs/2304.14402. [Accessed 8 5 2023].

[25] HF, "A one-year long research workshop," Bigscience, 22 4 2021. [Online]. Available: https://bigscience.huggingface.co/. [Accessed 27 1 2023].

[26] Bloom, "The BLOOM model has been proposed with its various versions through the BigScience Workshop.," Huggingface, 17 6 2022. [Online]. Available: https://huggingface.co/docs/transformers/model_doc/bloom. [Accessed 5 2 2023].

[27] GH, "Introduced by Conneau et al. in Unsupervised Cross-lingual Representation Learning at Scale," CC100, 20 5 2022. [Online]. Available: https://paperswithcode.com/dataset/cc100. [Accessed 22 2 2023].

[28] X. V. Lin, "Few-shot Learning with Multilingual Generative Language Models," Arxiv, 11 1 2022. [Online]. Available: https://arxiv.org/pdf/2112.10668.pdf. [Accessed 3 3 2023].

[29] H. Lopez-Almazan, "A deep learning framework to classify breast density with noisy labels regularization," *Computer Methods and Programs in Biomedicine,* May 2022.

[30] Z.Y.Leee, "Classification of mammographic breast density and its correlation wwith BIRADS in elder women using machine learning approach," *Journal Of Medical Imaging and Rdiation Sciences,* 2022.

[31] Huyen Nguyen, "A novel multi-view deep learning approach for BI-RADS and density," *2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC),* July 2022.

[32] H. N. Sam B.Tran, "A Novel Transparency Strategy-based Data Augmentation Approach for BI-RADS Classification of Mammograms," 2022.

[33] K. Jang, "A High-Performance Deep Neural Network Model for BI-RADS Classification of Screening Mammography," 2022.

[34] S. Siddeeq, "Deep Learning RN-BCNN Model for Breast Cancer BI-RADS Classification".

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F035**

# HỆ THỐNG IOT HỖ TRỢ QUẢN LÝ, CHĂM SÓC VÀ THEO DÕI SỨC KHỎE THÚ CƯNG TỰ ĐỘNG

TRẦN THỊ MINH KHOA[1,*], MÃ TIỂU LINH[1], TRẦN ANH VŨ[1], ĐẶNG THỊ PHÚC[1]

[1] *Khoa Công nghệ thông tin, Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*
** ttmkhoa@iuh.edu.vn*

**Tóm tắt.** Trong thời điểm hiện tại, một trong những lĩnh vực khoa học công nghệ được nhiều người yêu động vật quan tâm là hệ thống chăm sóc thú cưng tự động. Mục đích của dự án là hỗ trợ chủ vật nuôi cho thú cưng ăn đúng giờ, ngay cả khi họ không ở nhà. Được xây dựng dựa trên nền tảng công nghệ Internet of Things, toàn bộ hệ thống được thiết kế một cách tối ưu nhất, giảm thiểu số lượng linh kiện và giá thành nhưng vẫn giữ chất lượng và độ bền cao. Điểm đặc biệt của dự án là truyền thông hai chiều giữa chủ nhân và thú cưng, hệ thống cho phép họ theo dõi tình trạng thú cưng và đưa ra quyết định phù hợp. Với vai trò là vi xử lý chính, Arduino Mega 2560 cùng Arduino Uno điều khiển các thiết bị thông tin, nhận dữ liệu từ cảm biến và truyền qua Raspberry Pi 3B+ là bộ xử lý trung gian để giao tiếp với Firebase. Sau khi nhận dữ liệu giữa Firebase và Arduino Mega 2560 cùng Arduino Uno, Raspberry Pi 3B+ sẽ phân tích so sánh 2 dữ liệu này. Kết quả là dữ liệu của Firebase được cập nhật, hiển thị trang web cho người dùng, đồng thời gửi dữ liệu về Arduino để nó xử lý và đưa ra các quyết định điều khiển các thiết bị điện tử trên hệ thống.

**Từ khóa.** Mạng lưới vạn vật kết nối, hệ thống chăm sóc thú cưng tự động, Arduino, Raspberry

## AN IOT SUPPORT SYSTEM FOR PET MANAGEMENT, CARE, AND HEALTH MONITORING

**Abstract.** In the current time, one of the areas of science and technology that many animal lovers are interested in is the automatic pet care system. The aim of the project is to assist pet owners to feed their pets on time, even when they are not at home. Built on the foundation of Internet of Things technology, the entire system is designed in the most optimal way, minimizing the number of components and costs but still maintaining high durability. A special feature of the project is two-way communication between owners and pets, the system allows them to monitor the pet's condition and make appropriate decisions. As the main processor, the Arduino Mega 2560 and Arduino Uno control the communication devices, receive data from the sensor, and transmit it through the Raspberry Pi 3B+, an intermediate processor to communicate with Firebase. After receiving data between Firebase and Arduino, Raspberry Pi 3B+ will analyze and compare these two data. As a result, data on Firebase is updated and displayed on the trang web to users. At the same time, it also sends data to Arduino for it to process decisions to control the electronic devices on the system.

**Keywords.** Internet of Things, Automatic Pet Care System, Arduino, Raspberry

## 1 GIỚI THIỆU

Trong cả nền kinh tế toàn cầu và cuộc sống hàng ngày, tự động hóa ngày càng trở nên cần thiết. Các hệ thống thủ công đang dần được thay thế bằng các hệ thống tự động. Trong khi cơ khí hóa cung cấp nhiều loại máy móc được vận hành hỗ trợ con người đáp ứng các yêu cầu về cơ bắp trong công việc thì tự động hóa ngoài tăng tính tiện lợi còn giúp làm giảm đáng kể nhu cầu về mặt cảm xúc và tinh thần của con người. Trên thực tế, Internet of Things (IoT) không chỉ đơn giản là một công nghệ mà là sự kết hợp giữa nhiều công nghệ để tạo ra một dịch vụ thông tin được thu thập từ nhiều cảm biến và thiết bị khác nhau [1, 2].

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

Ngày nay, bởi phong cách sống thay đổi nên chúng tôi có thể dễ dàng thấy được số hộ gia đình độc thân đang tăng lên và họ lựa chọn sống cùng với thú cưng của mình. Một trong những vấn đề khó khăn đối với những hộ này là làm sao để chăm sóc và cho thú cưng ăn uống hợp lý, đúng giờ khi họ vắng nhà. Dù các dịch vụ chăm sóc thú cưng cá nhân xuất hiện ngày càng nhiều xung quanh nhưng lại đòi hỏi chi phí cao và cần một thời gian dài để tạo dựng sự tin cậy giữa đôi bên.

Theo nhiều kết quả nghiên cứu, việc kiểm soát ăn uống tự động sẽ hỗ trợ nhiều vấn đề mà vật nuôi gặp phải do ăn uống không đều đặn. Máy cho ăn và uống tự động là một trong những phát minh tự động hóa nổi bật. Hệ thống giúp chủ vật nuôi chăm sóc người bạn của họ một cách tốt nhất. Mục đích của dự án là hỗ trợ chủ vật nuôi cho thú cưng ăn đúng giờ, ngay cả khi họ không ở nhà. Ngoài ra, nó cũng có thể giúp chủ sở hữu hiểu chế độ ăn uống của thú cưng, điều cần thiết để đảm bảo an toàn cho thú cưng. Việc cho thú cưng ăn uống điều độ không phải là một vấn đề nan giải nữa khi mọi thứ được vận hành một cách tự động và có thể điều khiển, theo dõi thông tin từ xa.

Vì những lý do trên, đã tạo cơ hội thúc đẩy việc xây dựng hệ thống chăm sóc thú cưng tự động sử dụng công nghệ IoT một cách mạnh mẽ. Trong bài viết này, chúng tôi đề xuất một hệ thống chăm sóc thú cưng tự động hỗ trợ cung cấp thức ăn và nước uống khi chủ vắng nhà, đồng thời có thể theo dõi cân nặng, tình trạng của thú cưng và đưa ra một số đề xuất cho người dùng. Hệ thống sử dụng Raspberry Pi 3B+ kết nối với cơ sở dữ liệu thời gian thực Firebase và bo mạch Arduino Mega 2560, Arduino Uno qua giao thức UART. Tất cả các thông tin được thu thập từ các cảm biến và thiết bị sẽ được xử lý, hiển thị trên trang web để người dùng có thể dễ dàng quản lý, theo dõi và điều khiển từ xa mà không bị hạn chế về mặt không gian lẫn thời gian.

## 2 CÁC NGHIÊN CỨU LIÊN QUAN

Thông thường, bất kỳ loài thú cưng nào cũng cần phải có sự chăm sóc của người chủ sở hữu. Một số loài vật nuôi không thể tự kiểm soát thói quen ăn uống, chúng sẽ ăn bất kỳ khi nào miễn là có sẵn thức ăn. Vấn đề nảy sinh và có thể ngày càng nghiêm trọng khi người chủ không có nhà để tự chăm sóc thú cưng trong một khoảng thời gian dài. Để khắc phục được vấn đề này, chúng tôi cần một hệ thống chăm sóc thú cưng tự động để đảm bảo rằng thú cưng sẽ được khỏe mạnh dù chủ vắng nhà.

Máy cung cấp thức ăn tự động cho thú cưng ứng dụng nguyên lý trọng lực bao gồm một hộp chứa lớn phía trên để thức ăn tự động rơi ra khi thú cưng ăn hết thức ăn có trong khay mà không cần sử dụng nguồn điện và có thể sử dụng ở mọi nơi [3,4]. Tương tự máy cho ăn, máy cho uống cũng hoạt động với nguyên lý hoạt động này. Loại máy cho ăn này đảm bảo thú cưng có thể tiếp cận thức ăn, nước uống cả ngày thậm chí một khoản thời gian dài cho đến khi thức ăn trong hộp chứa không còn nữa nhưng lại không thể kiểm soát được khối lượng thức ăn mà thú cưng sẽ ăn vào có thể ảnh hưởng đến sức khỏe của chúng.

Trong một nghiên cứu khác, mẫu máy phân phối thức ăn tự động được thiết kế sử dụng Raspberry Pi để cải thiện việc cho thú cưng ăn ở nhà. Nghiên cứu chỉ ra rằng Raspberry Pi là một công cụ đa năng hỗ trợ giao tiếp với trình điều khiển thiết bị, cảm biến và động cơ bước để thức ăn rơi ra khay khi vật nuôi đến gần. Theo kết quả nghiên cứu, thiết kế bộ phận phân phối tốt sẽ đem lại ảnh hưởng tốt với cả thú cưng và cả gia đình bởi mọi việc trở nên thuận tiện và dễ dàng hơn rất nhiều.

Năm 2019, một hệ thống cho thú cưng ăn tự động được điều khiển bởi điện thoại đã được phát triển. Thiết bị này phân phối lượng thức ăn chính xác vào thời điểm thích hợp tùy theo từng loại vật nuôi khác nhau. Dự án sử dụng Arduino Mega để cung cấp cho người dùng một phương pháp thuận tiện trong việc cho thú cưng ăn. Bằng cách sử dụng điện thoại, chủ sở hữu chỉ cần cài đặt chính xác lượng thức ăn cung thời gian và nó sẽ được tự động phân phối một cách chính xác. Dữ liệu thu thập được ngay sau đó sẽ được truyền qua Ethernet đến PCB và gửi tín hiệu đến các cổng phân phối thức ăn.

Máy giám sát vị trí và cho mèo ăn tự động cũng là một phát minh vượt bậc trong lĩnh vực công nghệ chăm sóc thú cưng cao cấp [5]. Thiết bị được tạo ra bởi hai module độc lập là UNO và Node MCU ESP8266 kết hợp với nhiều loại cảm biến phù hợp như cảm biến siêu âm, động cơ servo hay module GPS để theo dõi vị trí. Sản phẩm được tạo ra và phát triển có khả năng cho thú cưng ăn khi chủ vắng nhà thông qua việc sử dụng ứng dụng di động có thể nhấn nút điều khiển để hệ thống làm đầy bát thức ăn. khi mức thức ăn và nước uống giảm đến mức cảnh báo, hệ thống còn gửi thông báo cho chủ nhân. Ngoài ra, hệ thống sẽ giúp người nuôi thú cưng nhận biết về vị trí của người bạn nhỏ nhờ theo dõi vị trí của chúng. Kết quả của dự án là tạo thành công một mẫu máy hoạt động có thể tương tác từ xa qua ứng dụng di động.

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Trong một bài báo năm 2014, từng có một đề xuất một hệ thống "Pet Feeder" được lập trình trên vi điều khiển [6]. Dự án phát triển lập trình một máy cho thú cưng ăn, phụ thuộc vào tốc độ quay của động cơ bước và giúp người dùng phân phối thức ăn bởi động cơ DC. Thời gian cho từng giai đoạn phân phối thức ăn có thể được cài đặt bởi người nuôi thú cưng.

Một hệ thống chăm sóc thú cưng tự động đem lại nhiều lợi ích trong việc nuôi dưỡng thú cưng bởi đây là công việc yêu cầu cam kết liên tục và tốn nhiều thời gian. Hệ thống mang lại tiện lợi cho người dùng bằng cách giúp họ cho thú cưng ăn, uống khi họ vắng nhà. Công việc bận rộn khiến việc theo dõi chế độ ăn uống của thú cưng đúng giờ và đúng phương pháp trở nên khó khăn khiến tình trạng sức khỏe của thú cưng ngày càng tệ đi hoặc gặp một số bệnh không mong muốn. Từ tất cả những lý do trên ta có thể thấy rằng việc phát triển một hệ thống phân phát đồ ăn và thức uống cho thú cưng là vô cùng cần thiết.

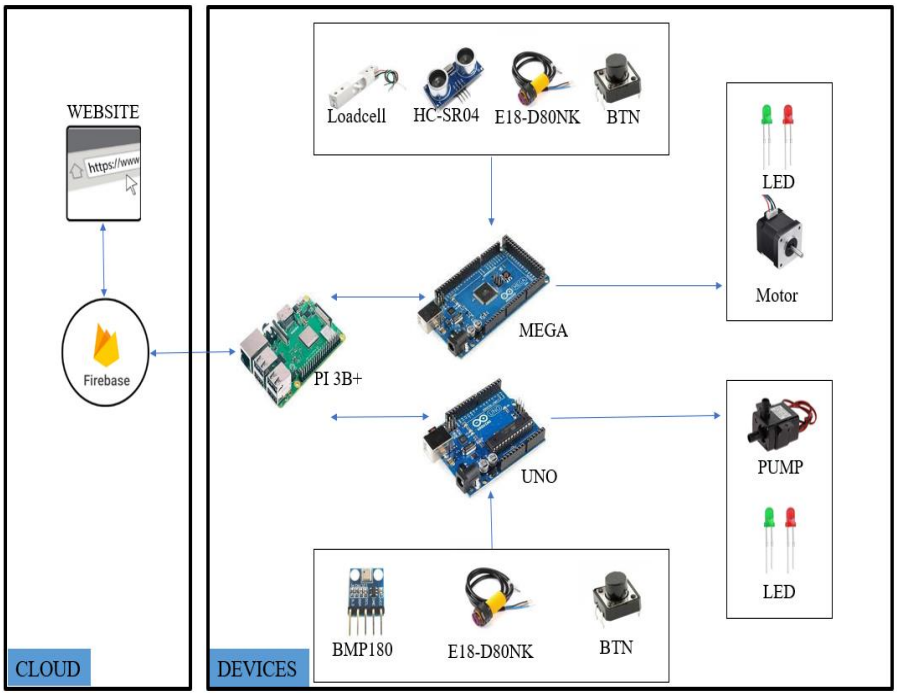
## 3 XÂY DỰNG HỆ THỐNG IOT HỖ TRỢ QUẢN LÝ, CHĂM SÓC VÀ THEO DÕI SỨC KHỎE THÚ CƯNG TỰ ĐỘNG

### 3.1 Mô tả hệ thống

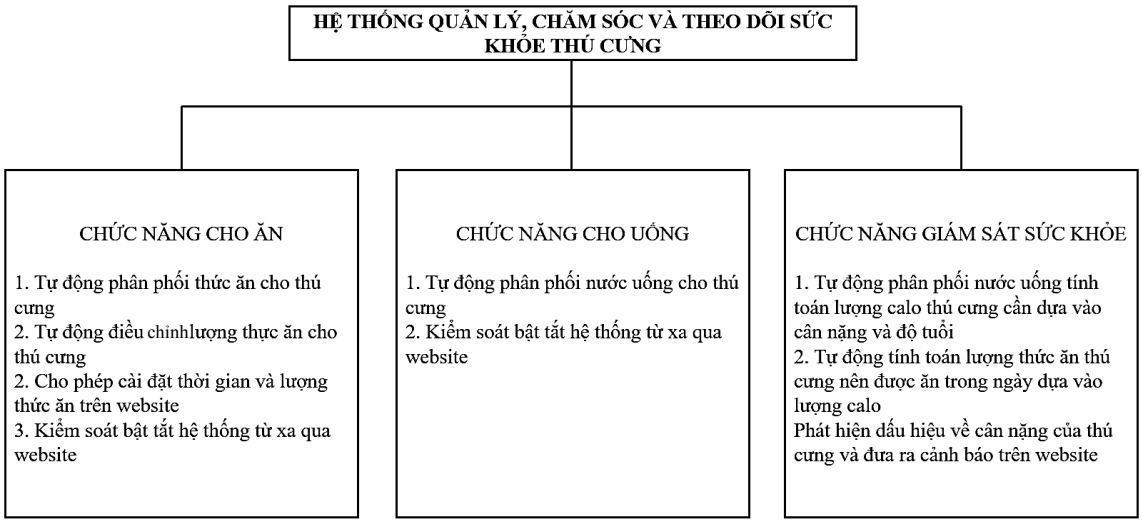
Trong dự án này, chúng tôi sử dụng ba kit xử lý chính để thực hiện nhiệm vụ. Hai kit đầu tiên là Arduino Mega 2560 và Arduino Uno, sẽ là trung tâm điều khiển các thiết bị điện tử và thu thập dữ liệu từ các cảm biến. Sau đó, dữ liệu này sẽ được gửi đến kit thứ ba là Raspberry Pi 3B+. Raspberry Pi 3B+ sẽ đóng vai trò là trung tâm trung gian để liên lạc giữa Arduino và Firebase đến trang web. Nó sẽ nhận dữ liệu từ Firebase và Arduino, phân tích và so sánh hai loại dữ liệu này để cập nhật lại thông tin trên Firebase và điều khiển các thiết bị điện tử trên hệ thống.

Về phương thức truyền dữ liệu, trên Raspberry Pi 3B+ sẽ được lập trình để đợi nhận dữ liệu từ Arduino Mega 2560 cùng Arduino Uno và Firebase. Khi phát hiện sự khác biệt về dữ liệu giữa Firebase và dữ liệu trên Raspberry Pi 3B+ (dữ liệu này cũng được lưu trữ trên Arduino Mega 2560 - Máy cung cấp đồ ăn và Arduino Uno - Máy cung cấp nước uống), Raspberry Pi sẽ cập nhật dữ liệu mới nhất và lưu trữ nó để tiếp tục so sánh với dữ liệu tiếp theo. Đồng thời, nó sẽ cập nhật lại dữ liệu trên Firebase và hai kit Arduino.

Hình 1 thể hiện sơ đồ tổng quan của hệ thống. Hình 2 thể hiện sơ đồ chức năng của hệ thống quản lý, chăm sóc và theo dõi sức khỏe thú cưng.




Hình 1. Sơ đồ tổng quan hệ thống




Hình 2. Sơ đồ chức năng hệ thống

## 3.2 Thiết bị phần cứng

Dựa vào yêu cầu thiết kế hệ thống chăm sóc thú cưng tự động chúng tôi cần kết hợp giữa vi xử lý cùng nhiều loại cảm biến, thiết bị khác nhau.

Bảng 1. Thông số kỹ thuật của các loại trang thiết bị trong hệ thống

| STT | Thiết bị | Thông số kỹ thuật |
|---|---|---|
| 1 | Raspberry pi 3B+ | CPU Broadcom BCM2837B0 quad-core A53 (ARMv8) 64-bit @ 1.4GHz<br>GPU Broadcom Videocore-IV<br>RAM 1GB LPDDR2 SDRAM<br>Networking: Gigabit Ethernet (via USB channel), 2.4GHz and 5GHz 802.11b/g/n/ac Wi-Fi<br>Bluetooth 4.2, Bluetooth Low Energy (BLE)<br>Storage: Micro-SD<br>GPIO: 40-pin GPIO header, populated.<br>Ports: HDMI, 3.5mm analog audio-video jack, 4x USB 2.0, Ethernet, Camera Serial Interface (CSI), Display Serial Interface (DSI)<br>Dimensions: 82mm x 56mm x 19.5mm, 50g |
| 2 | Arduino mega 2560 | Vi điều khiển: AVR ATmega 2560 (8bit)<br>Nguồn cung cấp: 7-12V (Bộ điều chỉnh sẵn có cho bộ điều khiển)<br>Số chân I/O số: 54<br>Số chân I/O tương tự: 16<br>Xung clock: 16 MHz (nhà sản xuất cài đặt là 1MHz)<br>Bộ nhớ flash: 128 KB<br>SRAM: 8 KB<br>Giao tiếp: USB (Lập trình với ATmega8), ICSP (lập trình), SPI, I2C và USART<br>Bộ Timer: 2 (8bit) + 4 (16bit) = 6 Timer<br>PWM: 12 (2-16 bit)<br>ADC: 16 (10 bit) |

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| | | |
|---|---|---|
| | | USART: 4<br>Ngắt thay đổi chân: 24 |
| | Arduino Uno | Chip điều khiển chính: ATMEGA328P loại IC dán<br>Chip nạp và giao tiếp UART: CH340<br>Nguồn nuôi mạch: 5VDC từ cổng USB hoặc nguồn ngoài cắm từ giắc tròn DC (nếu sử dụng nguồn ngoài từ giắc tròn DC Hshop.vn khuyên bạn nên cấp nguồn từ 6~9VDC để đảm bảo mạch hoạt động tốt, nếu bạn cắm 12VDC thì IC ổn áp rất nóng, dễ cháy và gây hư hỏng mạch).<br>Số chân Digital I/O: 14 (trong đó 6 chân có khả năng xuất xung PWM).<br>Số chân PWM Digital I/O: 6<br>Số chân Analog Input: 6<br>Dòng điện DC Current trên mỗi chân I/O: 20 mA<br>Dòng điện DC Current chân 3.3V: 50 mA<br>Flash Memory: 32 KB (ATmega328P), 0.5 KB dùng cho bootloader.<br>SRAM: 2 KB (ATmega328P)<br>EEPROM: 1 KB (ATmega328P)<br>Clock Speed: 16 MHz<br>LED_BUILTIN: 13<br>Kích thước: 68.6 x 53.4 mm |
| 3 | Cảm biến siêu âm HC-SR04 | Điện áp hoạt động: 5VDC<br>Dòng tiêu thụ: 10~40mA<br>Tín hiệu giao tiếp: TTL<br>Chân tín hiệu: Echo, Trigger.<br>Góc quét:<15 độ<br>Tần số phát sóng: 40Khz<br>Khoảng cách đo được: 2~450cm (khoảng cách xa nhất đạt được ở điều kiện lý tưởng với không gian trống và bề mặt vật thể bằng phẳng, trong điều kiện bình thường cảm biến cho kết quả chính xác nhất ở khoảng cách <100cm).<br>Sai số: 0.3cm (khoảng cách càng gần, bề mặt vật thể càng phẳng sai số càng nhỏ).<br>Kích thước: 43mm x 20mm x 17mm |
| 4 | Cảm Biến Vật Cản Hồng Ngoại E18-D80NK | Model: OMDHON E18-D80NK<br>Số dây tín hiệu: 3 dây (2 dây cấp nguồn và 1 dây tín hiệu).<br>Chân tín hiệu ngõ ra: cấu trúc cực thu hở Transistor NPN - Open Collector nên sẽ cần phải có trở kéo (khoảng 1~10K) lên chân nguồn dương VCC để tạo thành tín hiệu mức cao (High).<br>Nguồn điện cung cấp: 5VDC<br>Khoảng điều chỉnh phát hiện vật cản của cảm biến: 3~80cm (điều chỉnh bằng biến trở trên cảm biến).<br>Góc khuếch tán (góc chiếu): 3~5 độ<br>Dòng kích ngõ ra: < 300mA.<br>Có led hiển thị ngõ ra màu đỏ.<br>Chất liệu sản phẩm: vỏ ngoài nhựa ABS, phía trong đổ keo chống nước, chống va đập.<br>Kích thước: 18 x 70mm |

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

| | | |
|---|---|---|
| 5 | Cảm biến áp suất không khí BMP180 | Điện áp cung cấp: 1.8 ~ 3.6V<br>Công suất tiêu thụ thập: 0.5uA tại 1Hz<br>Giao tiếp chuẩn I2C<br>Tốc độ I2C max: 3.5MHz<br>Độ nhiễu rất thấp: lên đến 0.02hPa (17cm)<br>Có sẵn bộ hiệu chỉnh bên trong.<br>Dải đo áp suất: 300 hPa ~ 1100 hPa (+9000m đến -500m)<br>Trọng lượng: 1.18g<br>Kích thước: 21mm x 18mm |
| 6 | Cảm biến trọng lượng loadcell 5kg, loadcell 20kg | Model YZC - 133<br>Tải trọng: 5Kg/20Kg<br>Rated Output (mV/V): 1.0 +- 0.15<br>Độ lệch tuyến tính (%): 0.05<br>Creep (5min) %: 0.1<br>Ảnh hưởng nhiệt độ tới độ nhạy %RO/độ C: 0.003<br>Ảnh hưởng nhiệt độ tới điểm không %RO/độ C: 0.02<br>Độ cân bằng điểm không %RO: +-0.1<br>Trở kháng đầu vào (Ω): 1066 +- 20<br>Trở kháng ngõ ra (Ω): 1000 +- 20<br>Trở kháng cách li (MΩ) 50V: 2000<br>Điện áp hoạt động: 5V<br>Nhiệt độ hoạt động: -20 ~ 65 độ C<br>Safe Overload %RO: 120<br>Ultimate overload %RO: 150<br>Chất liệu cảm biến: Nhôm<br>Độ dài dây: 180mm |
| 7 | Động cơ Step motor NEMA 17 (Động cơ bước) | Kích thước chuẩn: 42x42mm và chiều dài 48mm<br>Đường kính trục: 5mm<br>Số bước trên một vòng: 200 (1.8 độ mỗi bước)<br>Momen xoắn tối đa: từ 0.4 đến 1.8 Nm<br>Điện áp định mức: từ 2 đến 12VDC<br>Dòng định mức: từ 0.4 đến 1.2A<br>Độ chính xác: +-5%<br>Nhiệt độ hoạt động: từ -20 đến 50 độ C<br>Độ ồn: từ 25 đến 40 dB |
| 8 | Động cơ bơm nước không chổi than 12V AD20P-1230A | Điện áp định mức: 12V DC<br>Dòng điện định mức: 500mA<br>Công suất tiêu thụ: 4,8W<br>Dòng định mức tối đa: 400mA<br>Tốc độ dòng chảy tối đa: 240L/H<br>Đầu tối đa (chiều cao nâng): 3M<br>Độ ồn: <40dB (hầu hết 35dB).<br>Lớp chống nước: IP68 (có thể được lắp đặt chìm)<br>Tuổi thọ: Hơn 20000 giờ.<br>Nguồn điện: Bảng điều khiển năng lượng mặt trời, nguồn điện DC, pin.<br>Kích thước: 5,5cm x 3,4cm x 4,1cm |

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

| | | |
|---|---|---|
| | | Đường kính ống vào/ra: 8mm<br>Nhiệt độ hoạt động: 0 độ C đến 60 độ C |
| 9 | Nguồn xung 12V 10A (nguồn tổ ong) | Điện áp đầu vào: 110V/220V – 50/60Hz.<br>Điện áp đầu ra: 12V – 10A.<br>Tuân thủ an toàn: CCC/FCC/CE.<br>Nhiệt độ làm việc: Từ 0 độ C đến 40 độ C.<br>Nhiệt độ lưu trữ: Từ -20 độ C đến 60 độ C.<br>Độ ẩm môi trường thích nghi: Từ 0 đến 95% sẽ không ngưng tụ.<br>Không có chế độ quạt làm mát, giải nhiệt.<br>Kích thước: 200 x 100 x 44 mm.<br>Trọng lượng: 420 gams |
| 10 | Đèn LED xanh, LED đỏ | LED 5mm<br>Ánh sáng thông thường<br>Mức độ phát sáng: 100-200 mcd<br>Điện áp chuyển tiếp: 1.8-2.2VDC<br>Màu sắc: Đỏ/Xanh |

### 3.3 Thiết kế hệ thống cung cấp nước tự động

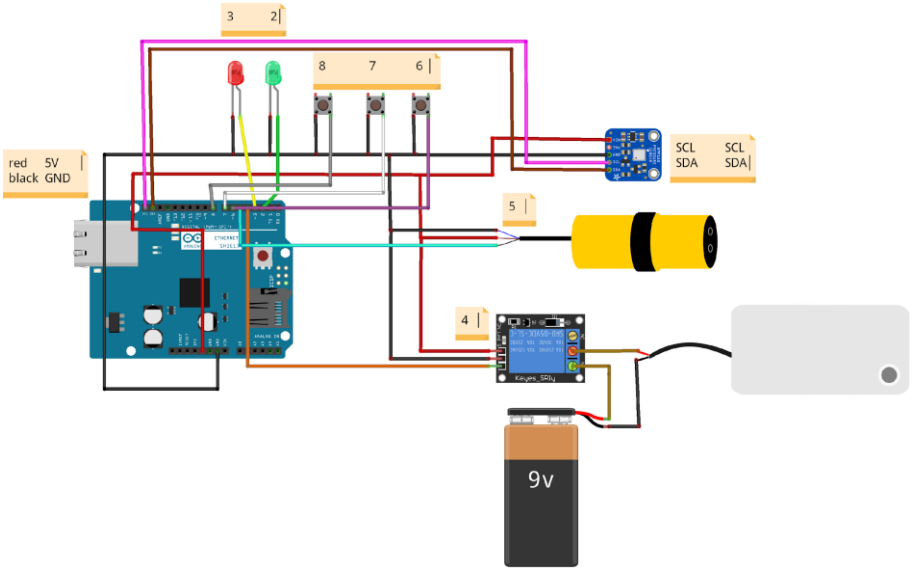
Hệ thống cung cấp nước tự động cho thú cưng bao gồm các thiết bị sau:

- Arduino Uno: Vi xử lý
- Cảm biến vật cản hồng ngoại E18-D80NK: tác dụng nhận biết thú cưng có ở gần hệ thống hay không
- Cảm biến áp suất không khí BMP180: tác dụng đo lường lượng nước còn lại trong máy
- Động cơ bơm nước không chổi than 12V AD20P-1230A: tác dụng bơm nước cho mèo uống
- Nguồn xung 12V10A (nguồn tổ ong): cung cấp điện cho máy bơm và động cơ step
- Đèn Led xanh, Led đỏ: tác dụng thông báo cho người dùng về tình trạng lượng nước còn trong hệ thống

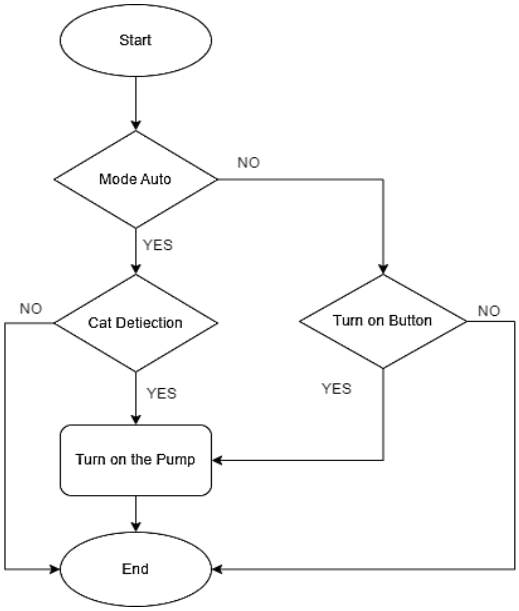
Có thể thấy, khi chức năng tự động được kích hoạt, hệ thống sẽ sử dụng cảm biến chướng ngại vật hồng ngoại để phát hiện sự có mặt của thú cưng, bật bơm nước khi phát hiện mèo và sẽ tắt bơm khi không phát hiện để tiết kiệm nước. Ngược lại, khi chức năng tự động không được kích hoạt, hệ thống sẽ kiểm tra xem bơm nước hiện đang hoạt động hay không. Nếu bơm đang hoạt động, hệ thống sẽ bật bơm. Nếu không, hệ thống sẽ tắt bơm. Cuối cùng, hệ thống sẽ kiểm tra giá trị của các biến, bao gồm mức nước trong bể nước, và cập nhật trạng thái của các chân đầu ra của hệ thống tương ứng. Bất kỳ thay đổi nào của các biến này sẽ kích hoạt cập nhật trạng thái của hệ thống và dữ liệu sẽ được truyền đến Raspberry Pi 3B+ để tiến hành phân tích và xử lý thêm.

Hình 3 và Hình 4 thể hiện sơ đồ nối mạch hệ thống và sơ đồ luồng dữ liệu của hệ thống cung cấp nước tự động cho thú cưng.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 3. Sơ đồ mạch hệ thống cung cấp nước tự động




Hình 4. Sơ đồ luồng dữ liệu hệ thống cung cấp nước tự động

Cuối cùng, có một điểm cần lưu ý khi sử dụng máy cung cấp nước tự động cho thú cưng là hệ thống sẽ không tự động bơm nước vào bình chứa khi đã hết mà người dùng phải tự bổ sung thêm tương tự như máy cho ăn tự động.

## 3.4 Thiết kế hệ thống cung cấp thức ăn tự động

Hệ thống cung cấp nước tự động cho thú cưng bao gồm các thiết bị sau:

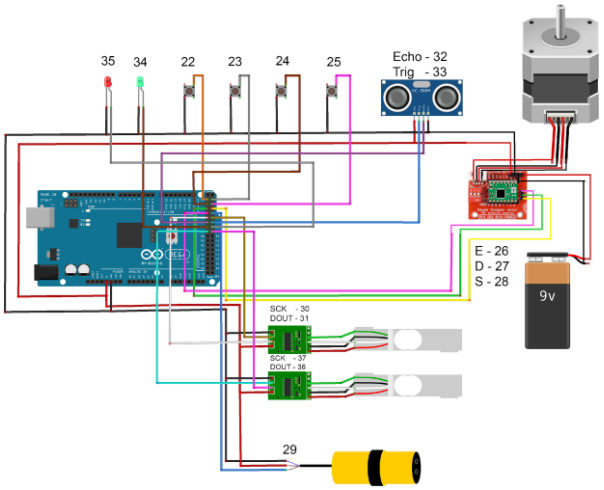
- Arduino Mega 2560: Vi xử lý
- Cảm biến vật cản hồng ngoại E18-D80NK: tác dụng nhận biết thú cưng có ở gần hệ thống hay không
- Cảm biến siêu âm HC-SR04: tác dụng đo lượng đồ ăn đang có trong máy đồ ăn
- Cảm biến trọng lượng Loadcell 5kg, loadcell 20kg: tác dụng đo trọng lượng đồ án và trọng lượng của thú cưng


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

- Động cơ Step motor NEMA 17 (Động cơ bước): tác dụng quay khay đồ ăn cho mèo ăn
- Đèn Led xanh, Led đỏ: tác dụng thông báo cho người dùng về tình trạng lượng lượng thức ăn còn trong hệ thống

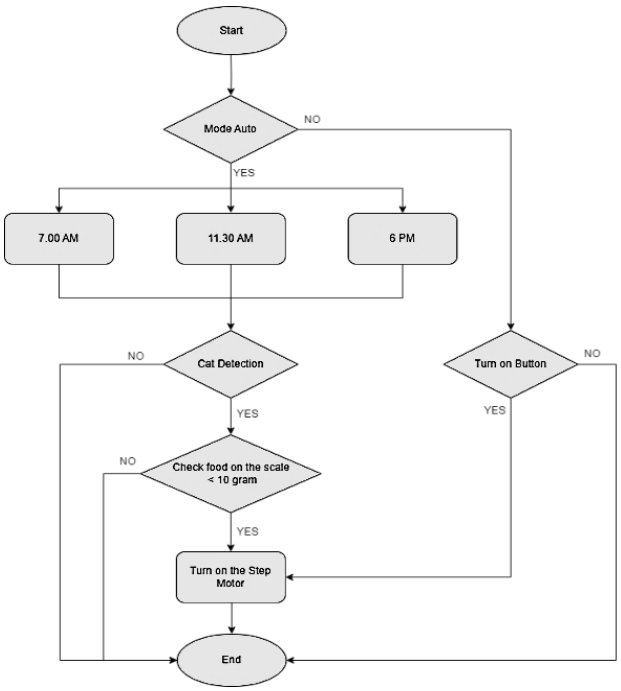
Trong quá trình hoạt động, chương trình sẽ thực hiện các công việc sau: tính toán trọng lượng thức ăn đã ăn và trọng lượng của thú cưng. Nếu cảm biến phát hiện thú cưng đang ở trước hệ thống, trọng lượng thức ăn ban đầu sẽ được lưu trữ. Khi cảm biến không phát hiện thú cưng nữa, chương trình sẽ tính toán lượng thức ăn chúng đã ăn và lưu trữ. Khi thú cưng đứng lên cân, chương trình sẽ tiến hành cân, lưu trữ trọng lượng của thú cưng và truyền dữ liệu. Ngoài ra, hệ thống cũng sẽ kiểm tra và hiển thị lượng thức ăn còn lại, bật đèn LED màu đỏ nếu tỷ lệ thức ăn còn lại lớn hơn 20% và bật đèn LED màu xanh nếu không. Tùy trường hợp mà người dùng có thể chọn xử lý việc cho thú cưng ăn bằng cách cho phép chế độ tự động hoặc điều khiển thủ công. Nếu chế độ tự động được bật, hệ thống sẽ tự động cho thú cưng ăn, nếu có đủ thức ăn và có thú cưng trên cân. Nếu chế độ điều khiển thủ công được bật, khi người dùng nhấn nút trên hệ thống, chương trình sẽ cho ăn thú cưng nếu đầy đủ thức ăn. Nếu không đủ thức ăn, chương trình sẽ hiển thị cảnh báo và hỏi người dùng có muốn cho ăn hay không. Nếu nhấn nút cơ khí, chương trình sẽ cho thú cưng ăn ngay lập tức.

Hình 5 và Hình 6 thể hiện sơ đồ nối mạch hệ thống và sơ đồ luồng dữ liệu của hệ thống cung cấp thức ăn tự động cho thú cưng.




Hình 5. Sơ đồ mạch hệ thống cung cấp thức ăn tự động

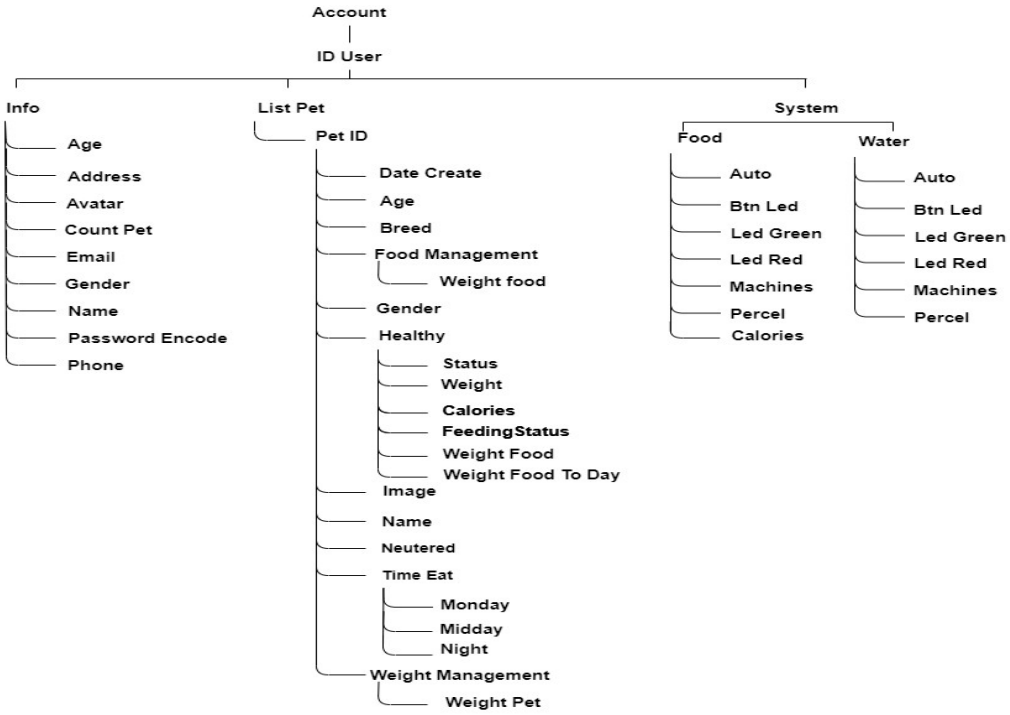
© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 6. Sơ đồ luồng dữ liệu hệ thống cung cấp thức ăn tự động

## 3.5 Thiết kế trang web quản lý và theo dõi

Raspberry Pi 3B+ đóng vai trò là trung tâm điều khiển trong hệ thống liên kết giữa Arduino Mega 2560, Arduino Uno, Firebase và trang web. Raspberry Pi 3B+ có khả năng nhận và xử lý dữ liệu từ Firebase và Arduino, đồng thời cập nhật dữ liệu lên Firebase để hiển thị trên trang web. Ngoài ra, Raspberry Pi 3B+ còn gửi dữ liệu đến hai Arduino để điều khiển các thiết bị điện tử trong hệ thống. Hình 7 mô tả cây dữ liệu hệ thống khi thiết kế và triển khai trang web quản lý.




Hình 7. Cây dữ liệu của hệ thống

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

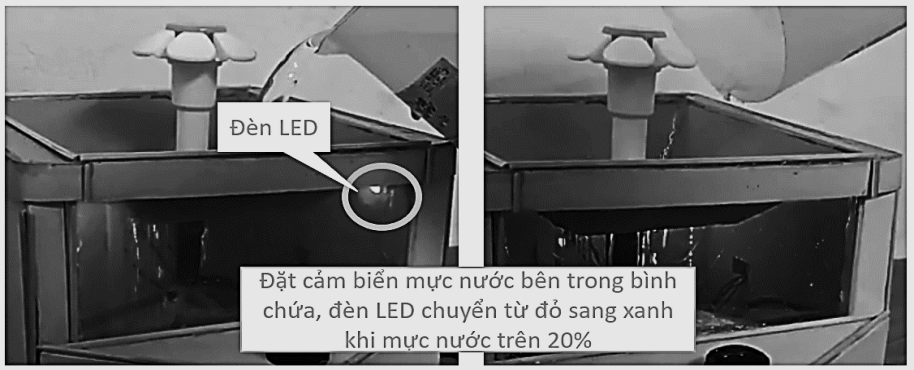
Trang web được thiết kế cho phép người dùng điều khiển trực tiếp hệ thống cho ăn và uống cho thú cưng. Chỉ cần các thao tác đơn giản trên trang web, hệ thống sẽ lập tức cho thú cưng ăn uống mà không cần kiểm tra nhưng yêu cầu về thời gian hay có thú cưng đang ở trên cân hay không. Đặc biệt hơn nữa, hệ thống khi thấy lượng đồ người dùng cho thú cưng ăn trong ngày đã đạt mức yêu cầu về calo thì sẽ cảnh báo cho người dùng là lượng đồ ăn người dùng đã cho ăn vượt mức yêu cầu về calo sẽ gây béo phì cho thú cưng. Trong trường hợp này, trang web sẽ hiển thị message để cảnh báo và xem quyết định của phía người dùng là tiếp tục cho ăn hay dừng lại. Trang web được xây dựng dựa trên nền tảng Python kết hợp với C và Javascript, bao gồm nhiều tính năng hỗ trợ người dùng sử dụng một cách thuận tiện và hiệu quả hơn.

# 4 KẾT QUẢ HIỆN THỰC HỆ THỐNG

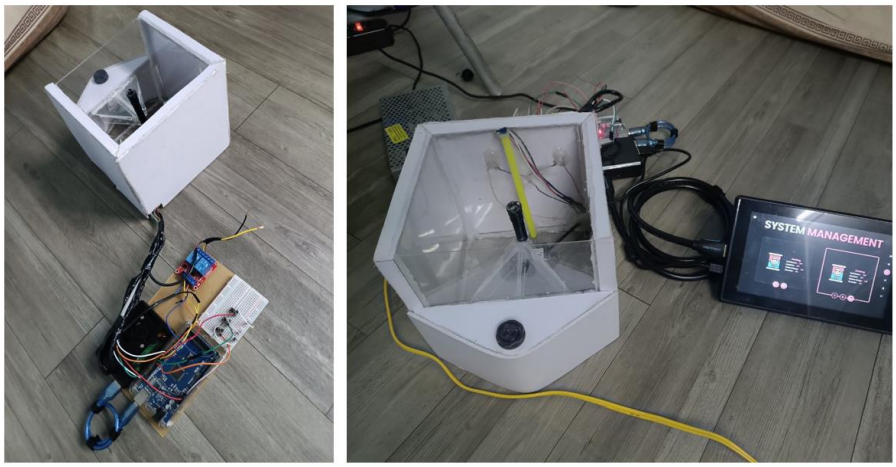
## 4.1 Chức năng cung cấp nước tự động

Chức năng tự động cung cấp nước uống cho thú cưng gồm các tính năng sau:

- Cảm biến mức nước: Đặt một cảm biến trong bình chứa nước để theo dõi mức nước hiện tại. Cảm biến này sẽ liên tục cập nhật thông tin về lượng nước còn lại trong hệ thống, khi mực nước thay đổi đèn LED sẽ chuyển màu (Hình 8).




Hình 8. Đèn LED chuyển màu khi mực nước thay đổi




Hình 9. Module cung cấp nước tự động sau khi triển khai

- Cảm biến phát hiện mèo: Khi cảm biến phát hiện mèo tới gần, bơm sẽ hoạt động và tự động cung cấp nước cho mèo
- Hệ thống cung cấp nước: Sử dụng một bơm nước hoặc van điều khiển để tự động điều chỉnh lượng nước được phân phối từ bình chứa tới khay uống của thú cưng. Bơm nước hoặc van điều khiển này có thể được kích hoạt từ xa thông qua một giao diện điều khiển trực tuyến, ví dụ như một trang web.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

- Hình 9 là sản phẩm sau khi hiện thực chức năng cho hệ thống cung cấp nước tự động.

## 4.2 Chức năng cung cấp thức ăn tự động

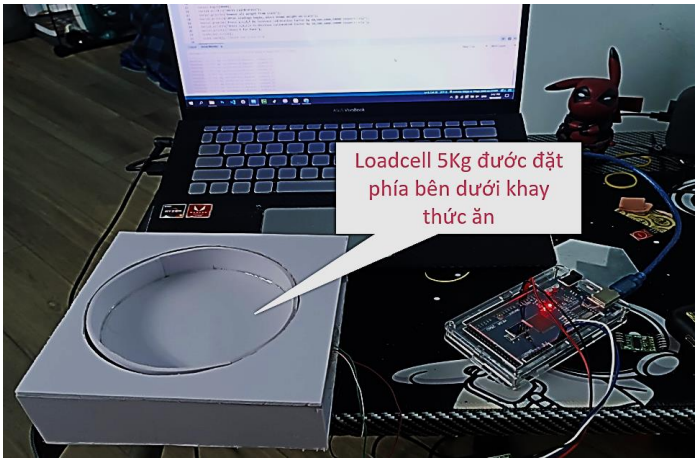
Chức năng tự động cung cấp thức ăn cho thú cưng bao gồm các tính năng sau:

- Bình chứa thức ăn có: Sử dụng một bình chứa thức ăn có thể lưu trữ một lượng lớn thức ăn cho thú cưng. Bình chứa này sẽ có một cơ chế tự động để phân phối lượng thức ăn theo thời gian và lượng calo được cài đặt.
- Cảm biến mức thức ăn: Đặt một cảm biến trong bình chứa thức ăn để theo dõi mức thức ăn hiện tại. Cảm biến này sẽ liên tục cập nhật thông tin về lượng thức còn lại trong hệ thống, đèn LED xanh sẽ chuyển đỏ lượng thức ăn dưới 20% (Hình 10).




Hình 10. Bình chứa thức ăn kết hợp cảm biến và đèn LED

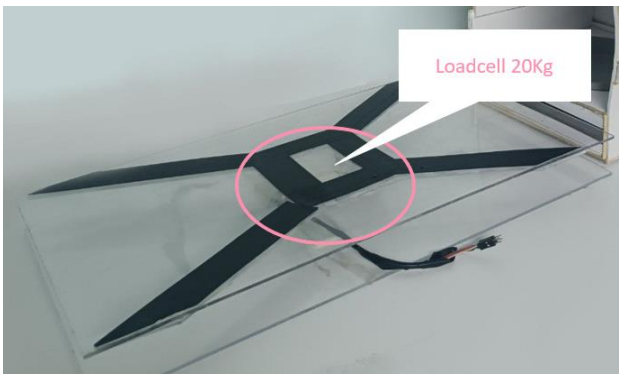
- Cảm biến cân nặng: Đặt một cảm biến cân nặng để theo dõi lượng thức ăn còn lại trong bình chứa. Cảm biến này sẽ tự động cập nhật thông tin về lượng thức ăn còn lại trong hệ thống (Hình 11).




Hình 11. Cân thức ăn - loadcell 5kg

- Cảm biến cân nặng thú cưng: Sử dụng một cảm biến cân nặng để đo và cập nhật cân nặng của thú cưng. Thông tin này sẽ giúp xác định lượng thức ăn đã được tiêu thụ và theo dõi sự phát triển của thú cưng (Hình 12).


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 12. Cân trọng lượng - loadcell 20kg

- Động cơ bước: Cảm biến phát hiện mèo hoạt động sẽ bật động cơ bước, mỗi lần xoay, một lượng thức ăn sẽ được gạt đến khay thức ăn cho thú cưng.

Hình 13 là sản phẩm hiện thực của hệ thống cung cấp thức ăn tự động dành cho thú cưng sau khi tích hợp các loại cảm biến, thiết bị và vi xử lý chính.



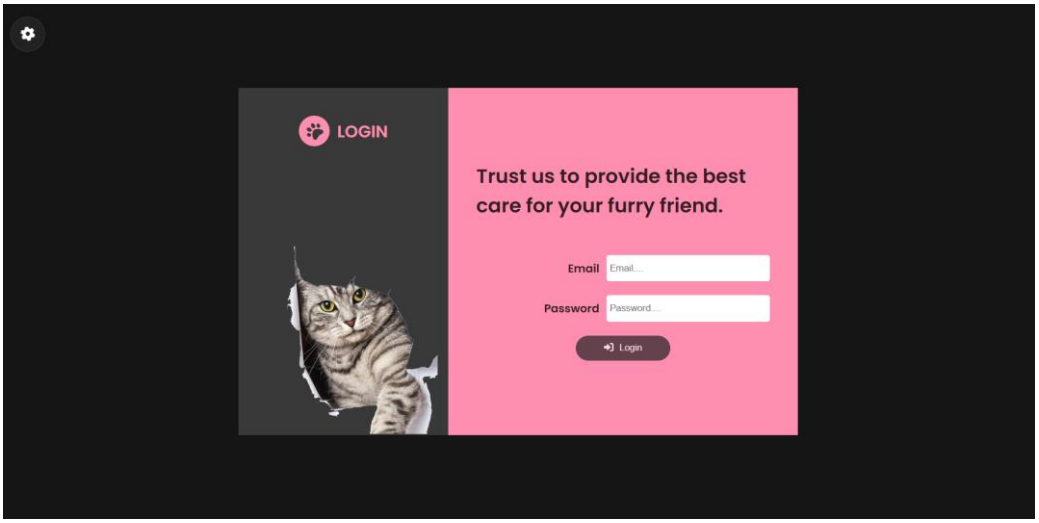
Hình 13. Module cung cấp thức ăn tự động sau khi triển khai

## 4.3 Trang web quản lý, theo dõi thú cưng

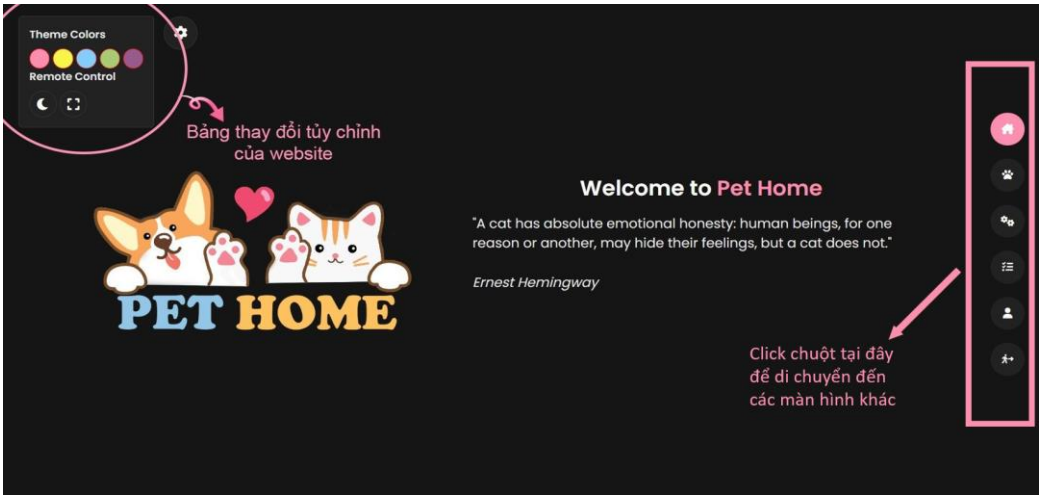
Trang web quản lý, theo dõi thú cưng bao gồm các trang chính như sau:

- Đăng nhập (Hình 14): Sau khi nhập Email và Password, dựa theo phân quyền tài khoản mà sẽ đưa người dùng đến trang Quản lý thú cưng (Pet Management) với tài khoản Quản trị viên (Admin) hay trang Quản lý người dùng (User Management) với các tài khoản người dùng khác.
- Trang chủ (Hình 15): Người dùng sẽ xem giới thiệu sơ lược về thông tin hệ thống tại đây.
- Danh sách thú cưng (Hình 16): Hiển thị danh sách thú cưng mà người dùng nuôi dưỡng và thông tin tổng quan của thú cưng: Hình ảnh, Breed (Chủng loại), Age (Tuổi), Gender (Giới tính), Neutered (Triệt sản hay chưa).
- Thông tin chi tiết thú cưng (Hình 17): Khi nhấn chọn xem chi tiết, người dùng có thể xem những thông tin như Thông tin chung, Thống kê và thông báo, Biểu đồ cân nặng và thức ăn,…
- Quản lý hệ thống (Hình 18): Thể hiện thông tin trạng thái hoạt động của máy cung cấp thức ăn và nước uống tự động, tại đây người dùng có thể điều khiển bật/tắt chức năng hệ thống từ xa.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 14. Trang đăng nhập

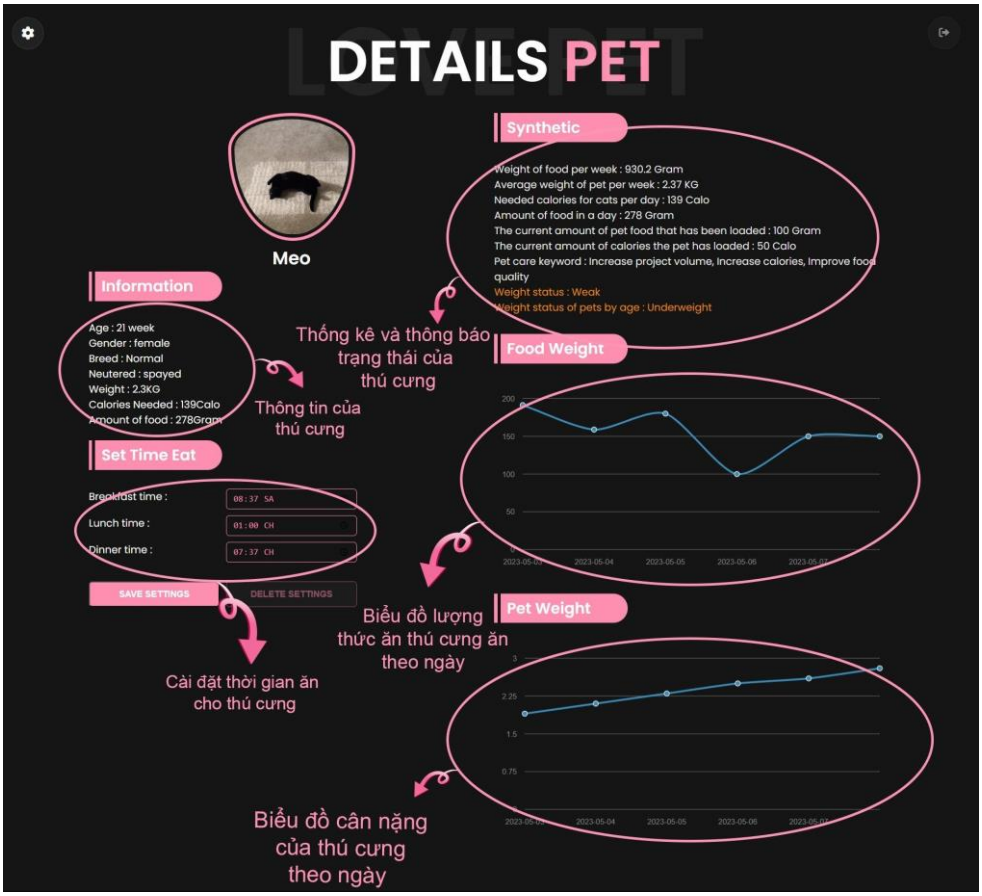



Hình 15. Trang chủ

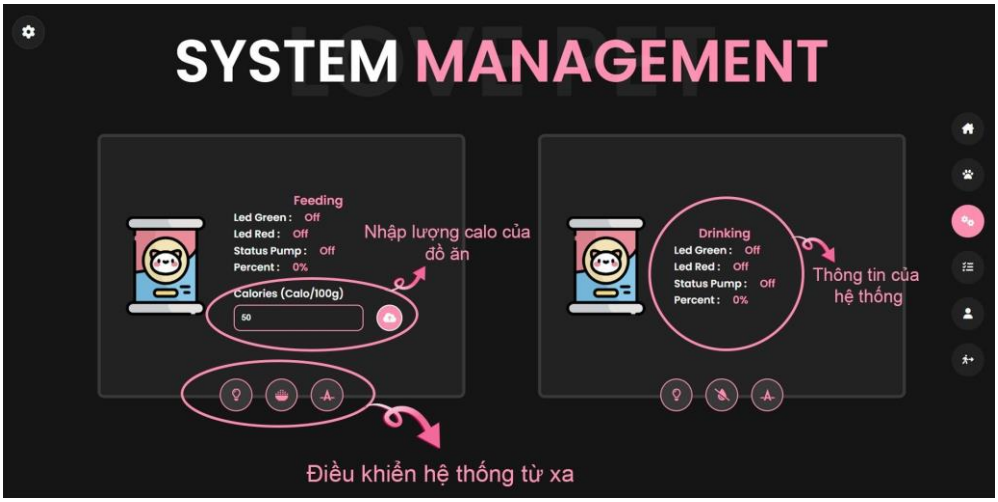



Hình 16. Danh sách thú cưng


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 17. Thông tin chi tiết thú cưng




Hình 18. Quản lý hệ thống

## 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Hệ thống quản lý, chăm sóc và theo dõi sức khỏe thú cưng có những ưu điểm vượt trội như tự động hóa việc cung cấp nước và thức ăn cho thú cưng, giúp tiết kiệm thời gian và công sức của chủ nhân. Ngoài ra, hệ thống còn có chức năng tự động tính toán lượng thức ăn, kiểm tra cân nặng của thú cưng, giúp chủ nhân nắm bắt được tình trạng sức khỏe của thú cưng và đưa ra những đề nghị về giải pháp phù hợp mà các dòng

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

máy hiện tại trên thị trường vẫn chưa có. Đặc biệt, máy cho ăn và cho uống tự động có chi phí hoàn thành không quá cao khi so với lợi ích mà nó đem lại cũng là một điểm đáng chú ý.

Hệ thống quản lý, chăm sóc và theo dõi sức khỏe thú cưng vẫn còn nhiều hạn chế khi chỉ mới được chạy thử nghiệm với mèo. Tuy đã giúp tiết kiệm thời gian và công sức cho chủ nhân nhưng không thể hoàn toàn thay thế việc chăm sóc thú cưng mà vẫn cần sự can thiệp của họ để giám sát và điều chỉnh lại hệ thống khi cần thiết. Thêm vào đó, hệ thống còn giới hạn trong việc kiểm soát và điều chỉnh các thông số cấu hình của nó, và để hoàn thiện hệ thống, yêu cầu sự đầu tư về kỹ thuật và thiết bị cần thiết.

Để cải thiện tính hiệu quả của hệ thống quản lý, chăm sóc và theo dõi sức khỏe thú cưng, đội ngũ phát triển đã đề xuất một số hướng phát triển tiếp theo. Đầu tiên, mở rộng hệ thống để hỗ trợ nhiều loại thú cưng hơn để người dùng có thể sử dụng hệ thống cho nhiều loài thú cưng khác nhau. Thứ hai, cải thiện tính năng của hệ thống bằng cách thêm các tính năng như kiểm tra sức khỏe chi tiết hơn, giám sát chất lượng không khí xung quanh vị trí thú cưng đang sinh sống. Thứ ba, phát triển thêm tính năng giám sát hành vi của thú cưng, giúp chủ nhân nắm bắt thông tin về hoạt động, sức khỏe, tâm lý và nhu cầu của thú cưng. Thứ tư, nâng cao tính tương tác với thú cưng bằng cách tích hợp giọng nói hoặc kết nối với các thiết bị như camera để giúp chủ nhân giao tiếp và quan sát thú cưng một cách trực quan hơn. Thứ năm, tích hợp các công nghệ mới để nâng cao tính năng của hệ thống, ví dụ như trí tuệ nhân tạo để phát hiện các dấu hiệu bệnh tật của thú cưng hoặc phân biệt giữa các loại thú cưng khác nhau. Thứ sáu, tối ưu hóa hệ thống để tiết kiệm năng lượng và giảm chi phí thấp nhất có thể cho người dùng.

## 6 TÀI LIỆU THAM KHẢO

[1] F. Mattern and C. Floerkemeier, “From the Internet of Computers to the Internet of Things”, Informatik-Spektrum, vol. 33, no. 2, (2010), pp. 107-121.

[2] O. Vermesan and P. Friess, “Internet of Things-From Research and Innovation to Market Deployment”, River Publishers, (2014).

[3] B. Ravi Babu, P. Pavan Kumar, Dr. P. G. Kuppusam. (2019). Arduino Mega-based PET Feeding Automation. IOSR Journal of Electronics and Communication Engineering (IOSR-JECE), 14(4), 13-16

[4] Delgado Villanueva, Alexi, and Hilary Nicole Vargas Alcantara. (2020). Pet food dispenser design using Raspberry Pi. International Journal of Emerging Trends in Engineering Research, 8(4).

[5] Nur Izzatul Nadia Binti M Razif, Nadilah Binti Mohd Ralim. (2020). Automatic Cat Feeder And Location Tracker. Journal of Computing Technologies and Creative Content, 5(1), 27-32.

[6] Gelila Berhan, T., Ahemed, W. T., & Birhan, T. Z. (2014). Programmable Pet Feeder. International Journal of Scientific Engineering and Research (IJSER), 3(11), 99–104

 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**YSC5.F036**

# XÂY DỰNG ỨNG DỤNG TRÍCH XUẤT THÔNG TIN TỪ DANH THIẾP SỬ DỤNG KỸ THUẬT HỌC SÂU

DANG THI PHUC [1](*), LE DINH HIEU[1], LE MINH PHU[1], TRAN THI MINH KHOA[1], LE THI ANH TUYET[1]

[1] *Faculty of Information Technology, Industrial University of Ho Chi Minh City*

*phucdt@iuh.edu.vn, 18075581. dinhhieu210899@gmail.com, minhphu18499@gmail.com ,ttmkhoa@iuh.edu.vn, lethianhtuyet@iuh.edu.vn*

**Tóm tắt.** Hiện nay, hầu hết người làm kinh doanh cá nhân hay các công ty, doanh nghiệp có nhu cầu để lại thông tin liên lạc đều sở hữu cho mình một mẫu thiết kế danh thiếp để tiện trong việc kinh doanh và liên lạc với khách hàng khi cần thiết và giới thiệu công ty, doanh nghiệp với khách hàng, đối tác hay những người chưa biết về doanh nghiệp mình. Vấn đề đặt ra khi lưu trữ trực tiếp các danh thiếp sẽ có khả năng mất thông tin. Trong bài báo, chúng tôi xây dựng ứng dụng tự động trích xuất các thông tin quan trọng từ danh thiếp, đảm bảo việc lưu trữ thông tin an toàn chính xác và dễ tìm kiếm. Chúng tôi đề xuất mô hình học sâu để trích xuất các đặc trưng này. Dữ liệu được thu thập từ nhiều loại danh thiếp khác nhau, trên mỗi danh thiếp được khoanh vùng và gán nhãn thành 4 lớp "tên", "email", "số điện thoại" và "địa chỉ". Dữ liệu được tiền xử lý bằng các thuật xử lý ảnh và được huấn luyện trên mô hình YOLOv8n, YOLOv8l, YOLOv8x. Kết quả huấn luyện trên mô hình cho thấy độ chính xác mAP là 98%. Mô hình được triển khai lên hệ thống và đem lại hiệu quả khi hoạt động.

**Từ khoá.** Học sâu, YOLOv8n, YOLOv8l, YOLOv8x, ứng dụng truy xuất thông tin danh thiếp

## BUILDING AN APPLICATION TO EXTRACT INFORMATION FROM BUSINESS CARDS USING DEEP LEARNING

**Abstract.** Currently, most individuals who are in business or companies and enterprises have a need to provide contact information and therefore possess their own business card design for convenience in conducting business and communicating with customers when necessary. It is also used to introduce their company or business to customers, partners, or those who are not familiar with their business. The problem arises when directly storing business cards, as there is a possibility of information loss. In the paper, we have developed an automated application to extract important information from business cards, ensuring safe and accurate storage of information that is easy to retrieve. We propose a deep learning model to extract these features. Data is collected from various types of business cards, where each card is segmented and labeled into four classes: "name," "email," "phone number," and "address." The data is preprocessed using image processing techniques and trained on YOLOv8n, YOLOv8l, and YOLOv8x models. The training results on the model show an accuracy of 98% mean Average Precision (mAP). The model is deployed in the system and proves to be effective in operation.

**Keyword.** Deep learning, YOLOv8n, YOLOv8l, YOLOv8x, business card information retrieval application.

## 1 GIỚI THIỆU

Trong kỷ nguyên công nghệ 4.0 như hiện nay thế giới đang chạy đua để phát triển tự động hóa, chuyển đổi số, giải quyết các quy trình sản xuất thủ công thay thế bằng những quy trình tự động, phân tích dữ liệu lớn để đưa ra các hoạch định về chiến lược kinh doanh. Bên cạnh đó cũng có một lượng lớn dữ liệu ở dạng vật lý là tài liệu giấy đang hiện hữu thì danh thiếp được cá nhân, doanh nghiệp, … sử dụng rất phổ biến. Tất cả

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

những loại giấy tờ này muốn truy xuất lưu trữ thông tin sẽ rất tốn kém về mặt con người, thời gian thực hiện, không gian lưu trữ so với các phương pháp công nghệ tiên tiến hiện nay. Để giải quyết các vấn đề trên, đã có các nghiên cứu về công nghệ phục vụ vấn đề số hóa tài liệu để có thể áp dụng các công nghệ tiên tiến trong việc phân tích dữ liệu đối với các tài liệu mà mình đang có [1]. Đó là việc thay thế nhập liệu thông tin danh thiếp thủ công vào trong hệ thống bằng việc trích xuất tự động. Giảm thời gian thực hiện, giảm mức độ sai sót trong lúc thực hiện thủ công, giảm chi phí. Một số nghiên cứu gần đây cũng đã đạt được những thành công trong việc lưu trữ thông tin tự động từ danh thiếp [2]. Các ứng dụng này hầu hết lưu tất cả thông tin từ danh thiếp dẫn đến thông tin khó truy xuất, việc tìm kiếm khó khăn và không cần thiết [3]. Bài toán nhận diện đối tượng đã được nghiên cứu từ những năm 1980 và đạt được nhiều thành tựu trong những năm gần đây. Nổi bật đáng kể đến là xu hướng ứng dụng các mô hình máy học vào bài toán nhận diện. Các thuật toán máy học mang lại thành công đáng kể có thể kể đến như K Nearest Neighbor (KNN), support vector machine (SVM), ANN [3,4,5], trong đó thuật toán học sâu, cụ thể là mô hình CNN là một bước tiến lớn giúp cải tiến độ chính xác cho bài toán nhận diện đối tượng một cách đáng kể, một số mô hình học sâu đạt hiệu quả cao như MTCNN [6], VGG16[7] ResNet [8]. Các mô hình này đã đạt được nhiều thành tựu đáng kể trong các lĩnh vực y khoa, thị giác máy tính… Những năm gần đây, xu hướng nhận diện đối tượng real time trên các thiết bị nhỏ gọn đang phát triển, các mô hình học sâu cũng ngày càng cải tiến hơn để phục vụ cho bài toán này, một số mô hình được đề xuất để giải quyết bài toán này như R_CNN[9], SSD[10], Fast R-CNN[11], Fast R-CNN[12], Mask_RCNN[13], RetinaNet[14], YOLO[15,16] trong đó mô hình YOLO được lựa chọn nhiều cho bài toán nhận dạng đối tượng bởi khả năng đáp ứng về độ chính xác và tốc độ của YOLO đảm bảo được cho rất nhiều bài toán thực tế hiện nay.

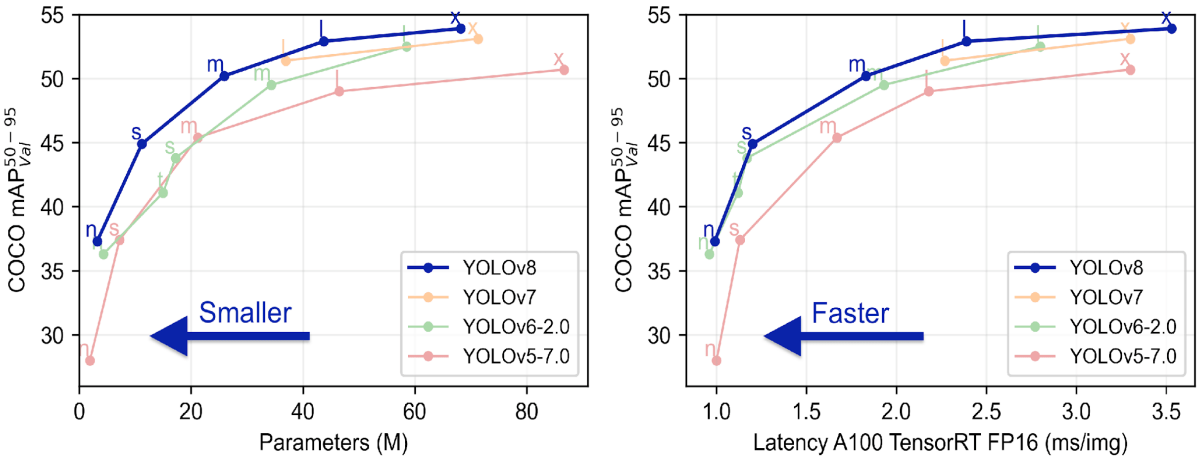
Trong bài báo, chúng tôi ứng dụng kỹ thuật deep learning để trích xuất các thông tin quan trọng từ danh thiếp. Dữ liệu được thu thập từ nhiều loại danh thiếp khác nhau bao gồm 1013 ảnh, trên mỗi danh thiếp được gán nhãn thành 4 lớp. Dữ liệu được huấn luyện thông qua pre-train mô hình YOLOv8 [17] để nhận dạng vị trí và lớp của các đối tượng trong ảnh. Từ các lớp và vị trí được xác định ở bước huấn luyện, chúng tôi tiến hành cắt thành các ảnh nhỏ theo vị trí và lớp tương ứng. Các ảnh được cắt sau đó được tiền xử lý bằng kỹ thuật xử lý ảnh để đảm bảo phần văn bản rõ ràng, nổi bật làm tăng độ chính xác cho bước sau. Những hình ảnh này, sẽ đưa vào Tesseract để nhận diện ký tự trên ảnh và chuyển thành dạng văn bản.

## 2 CÁC THUẬT TOÁN LIÊN QUAN

### 2.1 You Only Look Once (YOLO)

YOLO [4] là viết tắt của You Only Look Once là một thuật toán phát hiện đối tượng phổ biến trong lĩnh vực thị giác máy tính. YOLO có kiến trúc đơn giản và yêu cầu dữ liệu đào tạo tối thiểu, giúp dễ dàng thực hiện và thích ứng với các tác vụ mới. Thuật toán YOLO lấy một hình ảnh làm đầu vào và sau đó sử dụng một mạng nơ-ron tích chập sâu đơn giản để phát hiện các đối tượng trong hình ảnh. YOLO có nhiều phiên bản khác nhau, trong đó YOLOv8 [15,16,17,18] là dòng mô hình Phát hiện đối tượng dựa trên YOLO mới nhất từ Ultralytics cung cấp hiệu suất tốt. YOLOv8 tận dụng các phiên bản YOLO trước đó nhưng được cải tiến nhanh hơn và chính xác hơn đồng thời cung cấp một khung thống nhất cho các mô hình đào tạo để thực hiện phát hiện đối tượng, phân đoạn phiên bản và phân loại hình ảnh. So sánh hiệu quả của YOLOv8 được mô tả ở hình 1.

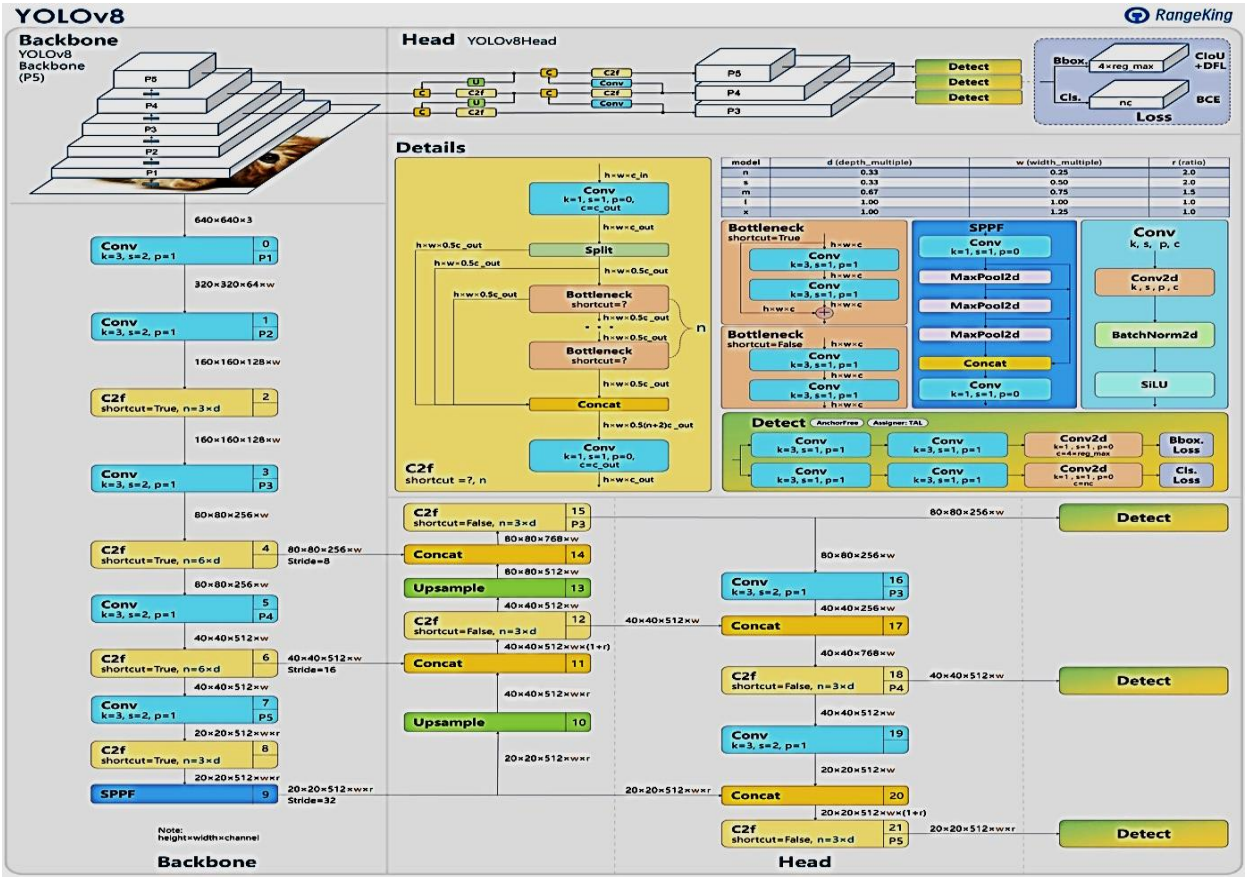
 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh




Hình 1: So sánh các mô hình YOLO

Ưu điểm của YOLOv8 là có tỷ lệ chính xác cao được đo bằng COCO và Roboflow 100, YOLOv8 đi kèm với rất nhiều tính năng thuận tiện cho nhà phát triển, từ CLI dễ sử dụng đến gói Python có cấu trúc tốt. Có một cộng đồng lớn xung quanh YOLO và một cộng đồng đang phát triển xung quanh mô hình YOLOv8, nghĩa là có nhiều người trong giới thị giác máy tính có thể hỗ trợ khi cần hướng dẫn. So với đánh giá YOLOv5, mô hình YOLOv8 tạo ra kết quả tương tự trên mỗi tập dữ liệu hoặc cải thiện kết quả đáng kể.
Kiến trúc mạng của YOLOv8 được xây dựng dựa trên các phiên bản trước của thuật toán YOLO. YOLOv8 sử dụng mạng thần kinh tích chập có thể được chia thành hai phần chính: xương sống và phần đầu. Phiên bản sửa đổi của kiến trúc CSPDarknet53 tạo thành xương sống của YOLOv8. Kiến trúc này bao gồm 53 lớp tích chập và sử dụng các kết nối từng phần giữa các giai đoạn để cải thiện luồng thông tin giữa các lớp khác nhau. Phần đầu của YOLOv8 bao gồm nhiều lớp tích chập, theo sau là một loạt các lớp được kết nối đầy đủ. Các lớp này chịu trách nhiệm dự đoán các hộp giới hạn, điểm đối tượng và xác suất lớp cho các đối tượng được phát hiện trong một hình ảnh. Một trong những tính năng chính của YOLOv8 là sử dụng cơ chế tự chú ý trong phần đầu của mạng. Cơ chế này cho phép mô hình tập trung vào các phần khác nhau của hình ảnh và điều chỉnh tầm quan trọng của các tính năng khác nhau dựa trên mức độ phù hợp của chúng với nhiệm vụ. Một tính năng quan trọng khác của YOLOv8 là khả năng thực hiện phát hiện đối tượng đa tỷ lệ. Mô hình sử dụng mạng kim tự tháp đặc trưng để phát hiện các đối tượng có kích thước và tỷ lệ khác nhau trong một hình ảnh. Mạng kim tự tháp tính năng này bao gồm nhiều lớp phát hiện các đối tượng ở các tỷ lệ khác nhau, cho phép mô hình phát hiện các đối tượng lớn và nhỏ trong một hình ảnh. Dựa trên hệ số tỷ lệ, YOLOv8 có các mô hình kích thước khác nhau: tỷ lệ nano (n), nhỏ (s), trung bình (m), lớn (l) và cực lớn (x) để đáp ứng nhu cầu của các tình huống khác nhau. Không giống như các phiên bản YOLO khác, cấu trúc C2f với luồng gradient phong phú hơn sẽ thay thế cấu trúc C3 của YOLOv5 trong mạng đường trục và phần cổ, đồng thời các số kênh khác nhau được sửa đổi cho các mô hình tỷ lệ khác nhau. Ngoài ra, hai lớp kết nối tích chập của mô-đun Neck đã bị loại bỏ. Cuối cùng, số khối C2f của đường trục nằm trong khoảng từ 3-6-9-3 đến 3-6-6-3. Các nhánh phân loại và hồi quy tách rời thay thế nhánh đối tượng trước đó. Kiến trúc mạng của YOLOv8 được mô tả ở hình 2.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 2: Kiến trúc mạng YOLOv8

Hàm mất mát của của mô hình YOLO sử dụng 3 đầu ra từ PAN Neck, để phát hiện objects tại 3 scale khác nhau. Tuy nhiên, Glenn Jocher nhận thấy rằng sự ảnh hưởng của các object tại mỗi scale đến Objectness Loss là khác nhau, do đó, công thức của Objectness Loss được thay đổi như sau [16]:

$$L_{obj} = 4.0 \cdot L_{obj}^{small} + 1.0 \cdot L_{obj}^{medium} + 0.4 \cdot L_{obj}^{large} \quad (1)$$

Thay đổi này giúp YOLO tập trung vào các đối tượng nhỏ hơn và cân bằng giữa việc dự đoán bounding box và phân loại đối tượng .

## 2.2 Tesseract

Tesseract [19,20]- là một thư viện nhận dạng ký tự quang học mã nguồn mở, không phải là một phần mềm ở dạng thương mại. Tesseract được đánh giá là một trong số ít những thư viện nhận dạng ký tự quang học mã nguồn mở tốt nhất hiện nay. Kiến trúc tổng quan của Tesseract là giả định các đầu vào là hình ảnh nhị phân với các vùng văn bản đa giác tùy chọn được xác định. Phân tích thành phần được kết nối là bước đầu tiên trong đó các phác thảo của các thành phần được lưu trữ. Các phác thảo được tập hợp lại với nhau, hoàn toàn bằng cách lồng vào nhau, thành các Blobs. Phác thảo ở đây là khái niệm trong Tesseract chỉ đến những vùng văn bản được phát hiện dựa trên các đốm màu trong cùng khu vực và được bọc lại bằng các outlines. Đây là kết quả của quá trình phân tích bố cục trang đưa ra. Blobs (Binary large object) là một kiểu dữ liệu lưu trữ một tập hợp dữ liệu nhị phân được lưu trữ dưới dạng một thực thể duy nhất. Dữ liệu được lưu trữ thường là các đốm màu biểu diễn một hình ảnh, âm thanh hoặc các đối tượng đa phương tiện. Tesseract được ứng dụng rất nhiều trong bài toán nhận dạng ký tự quang học và đạt được nhiều kết quả đáng kể [21].

## 2.3 Các thông số đánh giá mô hình

**Accuracy (Độ chính xác)**: Đây là tỷ lệ giữa số lượng dự đoán chính xác và tổng số lượng dữ liệu được dùng để đánh giá mô hình. Accuracy là một thông số quan trọng và thường được sử dụng để đánh giá mô hình, nhưng nó không phải là đánh giá tốt nhất cho bài toán nhận dạng ảnh, đặc biệt là trong trường hợp các lớp dữ liệu không cân bằng về số lượng.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

**Precision (Độ chắc chắn):** Precision là tỷ lệ giữa số lượng dự đoán đúng và tổng số lượng dự đoán được thực hiện cho một lớp cụ thể. Thông số này cho biết mức độ chắc chắn của mô hình trong việc dự đoán một lớp nào đó. Nếu mô hình có độ chính xác cao nhưng độ chắc chắn thấp, nghĩa là nó có thể dự đoán sai một số trường hợp.

**Recall (Độ phủ):** Recall là tỷ lệ giữa số lượng dự đoán đúng và tổng số lượng thực tế cho một lớp cụ thể. Thông số này cho biết mức độ hoàn thành của mô hình trong việc dự đoán tất cả các trường hợp của một lớp nào đó. Nếu mô hình có độ chính xác thấp nhưng độ phủ cao, nghĩa là nó có thể phát hiện được tất cả các trường hợp nhưng có thể dự đoán sai một số trường hợp.

**F1-score:** F1-score là trung bình điều hòa của precision và recall. Nó được tính bằng công thức:

$$\mathrm{F1-score} = \frac{2*(\mathrm{precision} * \mathrm{recall})}{\mathrm{precision} + \mathrm{recall}} \quad (2)$$

F1-score cho biết mức độ chính xác của mô hình trên toàn bộ các lớp.

**mAP:** là trung bình cộng của tất cả các giá trị AP (Average Precision) của các lớp đối tượng trong tập dữ liệu. Trong đó, AP được tính bằng cách tính diện tích dưới đường Precision-Recall (PR) curve cho từng lớp đối tượng. Precision là tỉ lệ số lượng đối tượng được nhận dạng đúng trên tổng số đối tượng được dự đoán là positive. Recall là tỉ lệ số lượng đối tượng được nhận dạng đúng trên tổng số đối tượng thực tế.

## 3 KẾT QUẢ THỰC NGHIỆM

### 3.1 Bộ dữ liệu

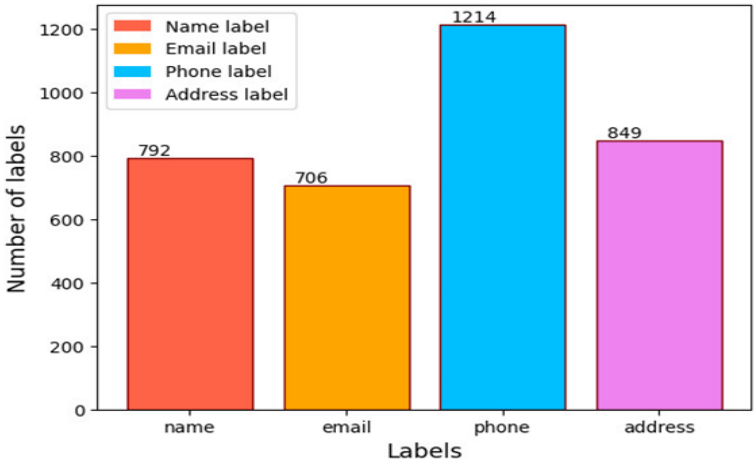
Bộ dữ liệu chúng tôi sử dụng là bộ dữ liệu hình ảnh được thu thập một cách thủ công từ các Hình ảnh danh thiếp của các công ty, trang web in các mẫu danh thiếp, từ các cửa hàng tiện lợi, siêu thị, quán ăn, từ bạn bè, người thân, tờ rơi danh thiếp quảng cáo. Tập dữ liệu cuối cùng sau quá trình tiền xử lý đạt 1013 ảnh. Mẫu danh thiếp được mô tả ở hình 4.




Hình 4: Một số mẫu danh thiếp

Dữ liệu sau đó được gán nhãn thành các vùng sau: *name* (tên) là họ và tên của chủ thể trên danh thiếp, lớp tên thường được hiển thị với cỡ chữ lớn, rõ ràng và nổi bật. *Email* (email) là địa chỉ thư điện tử (email) của chủ thể danh thiếp, lớp email có cỡ chữ nhỏ hơn lớp tên và có chứa ký tự "@" đặc trưng cho tên miền email. *phone* (số điện thoại) là số điện thoại liên lạc của chủ danh thiếp, lớp số điện thoại thường có cỡ chữ nhỏ, bao gồm số, các ký tự ngăn cách như ".", "-", ký tự bắt đầu là số "0" hoặc mã vùng ví dụ như "+84" là mã vùng điện thoại của Việt Nam. *address* (địa chỉ) là thông tin địa chỉ liên hệ, nơi đặt văn phòng, trụ sở làm việc của chủ danh thiếp hoặc công ty, lớp địa chỉ thường có cỡ chữ nhỏ. Địa chỉ gồm có 2 phần cơ bản. Phần đầu tiên bao gồm số nhà, tên đường hoặc thôn, ấp và phần còn lại sẽ theo địa giới hành chính tương ứng với 3 cấp quản lý hành chính là tên cấp xã: xã, phường, thị trấn; cấp huyện: huyện, quận, thị xã, thành phố trực thuộc tỉnh; cấp tỉnh: tỉnh, thành phố trực thuộc trung ương. Từ biểu đồ cho thấy số lượng của mỗi loại có sự khác nhau. Dữ liệu số điện thoại (phone) có số lượng cao nhất với 1214 nhãn. Các lớp còn lại có số lượng tương đồng nhau với email có số lượt thấp nhất là 706 nhãn, tiếp theo là tên (name) với 792 nhãn và cuối cùng là địa chỉ với 849 nhãn. Biểu đồ phân bố dữ liệu được mô tả ở hình 5.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 




Hình 5: Biểu đồ thể hiện sự phân bố số lượng các lớp của dữ liệu

**Tiền xử lý dữ liệu:** Thu thập được dữ liệu cần phải phân loại các danh thiếp đúng yêu cầu. Có đầy đủ thông tin như tên, địa chỉ, email, số điện thoại. Dữ liệu phải rõ, không bị mờ, bể ảnh, phải được chụp chính diện. Dữ liệu ban đầu có thể không đúng trọng tâm, có thể bị lệch trái, lệch phải. Cần phải có bước tiền xử lý xoay ảnh sao cho ảnh không bị lệch, cắt hình ảnh chỉ giữ lại phần trọng tâm là danh thiếp, gia giảm độ sáng tối để dữ liệu được rõ ràng. Dữ liệu có thông tin mờ, ảnh không đúng góc chụp chính diện, ảnh bị lệch, ảnh tối thiếu độ sáng là các lỗi thường gặp của dữ liệu. Một số lỗi thường gặp khi xử lý được mô tả ở hình 6.




Hình 6: Một số lỗi của ảnh danh thiếp

## 3.2 Thiết kế mô hình

Mô hình được huấn luyện với 200 epochs, kích thước ảnh đầu vào được model YOLOv8 chuẩn hóa về 640. Dữ liệu huấn luyện được chia thành 4 lớp bao gồm name ứng với nhãn 0, email ứng với nhãn 1, phone ứng với nhãn 2 và address ứng với nhãn 3. Các model được huấn luyện với GPU của Google Colab.

## 3.3 Kết quả so sánh

Kết quả huấn luyện các mô hình được mô tả ở bảng 1.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

Bảng 1: Bảng so sánh độ chính xác của các mô hình trên từng lớp

| Model | Parameters | Classes | Precision (%) | Recall (%) | mAP50 (%) | mAP50-95 (%) |
|---|---|---|---|---|---|---|
| YOLOv8n | 3,006,428 | Name (0) | 96.2 | 94 | 99 | 78.4 |
| | | Email (1) | 97.5 | 92.6 | 98.7 | 74.9 |
| | | Phone (2) | 95.1 | 99.5 | 98.5 | 71.6 |
| | | Address (3) | 97.1 | 94.9 | 99 | 81.2 |
| YOLOv8l | 43,609,692 | Name (0) | 96.1 | 90.3 | 97.3 | 82.1 |
| | | Email (1) | 100 | 99.5 | 99.5 | 76.9 |
| | | Phone (2) | 96.8 | 100 | 99.4 | 76.3 |
| | | Address (3) | 96.6 | 100 | 99.5 | 84.5 |
| YOLOv8x | 68,127,420 | Name (0) | 93.5 | 96.3 | 97.4 | 82.2 |
| | | Email (1) | 96.2 | 100 | 99.3 | 79.4 |
| | | Phone (2) | 97.4 | 97.7 | 99.4 | 77.1 |
| | | Address (3) | 100 | 98.7 | 99.5 | 83 |

Kết quả từ các lần huấn luyện được tổng hợp với các thông tin về độ chính xác, thời gian xử lý, thời gian huấn luyện và kích thước của mô hình được mô tả ở bảng 2.

Bảng 2: So sánh độ hiệu quả của các mô hình

| Model | Precision (%) | Recal (%) | mAP50 (%) | mAP50-95 (%) | Speed (ms) | Time complete | Size weight |
|---|---|---|---|---|---|---|---|
| YOLOv8n | 96.5 | 95.3 | 98.8 | 76.5 | **3.3** | **2.345h** | **6.2MB** |
| YOLOv8l | **97.4** | 97.4 | 98.9 | 80 | 12 | 3.121h | 87.6MB |
| YOLOv8x | 96.8 | **98.2** | **98.9** | **80.4** | 18.5 | 4.284h | 136.7MB |

Theo thông tin từ Bảng 1 số lượng tham số có sự khác biệt lớn giữa các mô hình. Bắt đầu với mô hình YOLOv8n, đây là một mô hình nhẹ ưu tiên tốc độ nên lượng tham số chỉ 3 triệu. Với mô hình YOLOv8l và YOLOv8x, đây là 2 mô hình nặng ưu tiên độ chính xác nên lượng tham số tương đối lớn. YOLOv8l có 43.6 triệu tham số hơn lượng tham số của YOLOv8n gần 15 lần và YOLOv8x có đến 68.1 triệu tham số hơn mô hình YOLOv8n tới 23 lần.

Với lượng tham số càng lớn thì độ chính xác càng tốt nhưng thời gian huấn luyện, thời gian nhận dạng dữ liệu và kích thước của file weight cũng lớn theo. Từ bảng 2 có thể thấy được các chỉ số về độ chính xác và hiệu suất nhận diện của mô hình YOLOv8x có phần tốt hơn các mô hình còn lại. Với mAP50 đạt 80.4% bằng với mô hình YOLOv8l và cao hơn mô hình YOLOv8n 3.9%. Với mAP50-95 thì mô hình YOLOv8x thể hiện sự vượt trội với 80.4% cao nhất trong các mô hình.

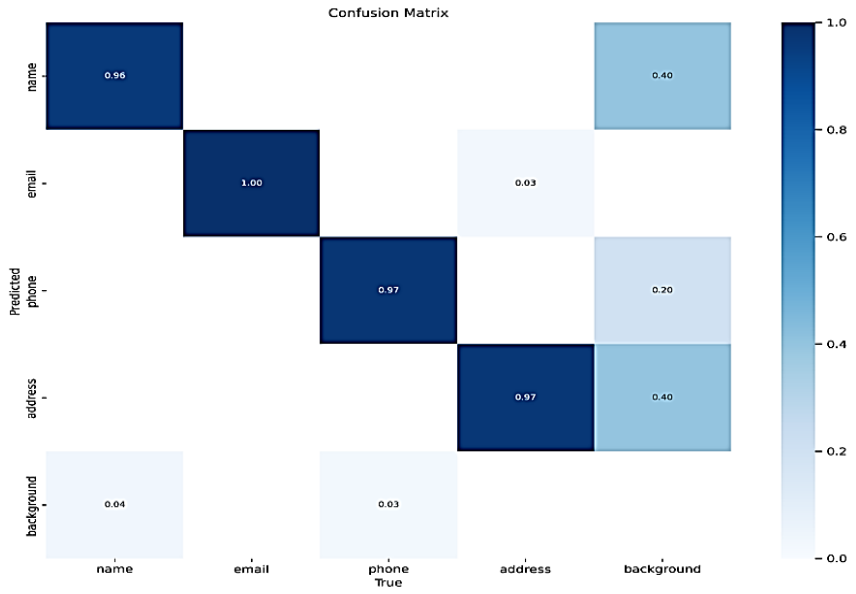
Với tốc độ nhận diện (Speed) thì mô hình YOLOv8x mất nhiều thời gian để nhận diện hơn các mô hình còn lại. Mô hình YOLOv8x cần 18.5ms cao hơn mô hình YOLOv8l 6.5ms và hơn mô hình YOLOv8n tới 15.2ms.

Do lượng tham số lớn nên thời gian huấn luyện cũng có sự chênh lệch lớn. Mô hình YOLOv8n chỉ cần 2.345 giờ, mô hình YOLOv8l cần 3.121 giờ và cuối cùng mô hình YOLOv8x cần đến 4.284 giờ để có thể hoàn thành quá trình huấn luyện cao gấp đôi thời gian so với mô hình YOLOv8n.

Từ các so sánh trên, có thể thấy được điểm mạnh yếu của các mô hình. Mô hình YOLOv8n là một mô hình nhẹ có thời gian xử lý nhanh và thích hợp cho các hệ thống tích hợp cần sự nhanh chóng và kích thước lưu trữ nhỏ. Với model YOLOv8l và YOLOv8x thích hợp cho hệ thống cần sự chính xác cao, không bó buộc trong thời gian xử lý và không gian lưu trữ file weight.

Trong ứng dụng này cần một mô hình có độ chính xác cao để có thể nhận diện được các dữ liệu cần cho bước xử lý tiếp theo chuyển ảnh thành văn bản, đồ án xây dựng một hệ thống xử lý backend riêng không tích hợp xử lý vào chương trình giao diện ở frontend nên kích thước của file weight không cần sự nhỏ gọn mà phải ưu tiên tối đa cho độ chính xác. Trong đó, mô hình YOLOv8x và YOLOv8l đáp ứng được các yêu

cầu về độ chính xác của ứng dụng, về hiệu quả YOLOv8l nhẹ và tốc độ nhận dạng nhanh đáng kể so với YOLOv8x. Do đó mô hình YOLOv8l được đề xuất triển khai lên hệ thống. Ma trận nhầm lẫn của mô hình YOLOv8l được mô tả ở hình 6.
Một vài so sánh kết quả nhận diện của các model YOLOv8l được mô tả ở hình 7.




Hình 6. Ma trận nhầm lẫn của mô hình YOLOv8l




Hình 7. Kết quả dự đoán của mô hình YOLOv8l

## 3.4 Triển khai mô hình lên ứng dụng

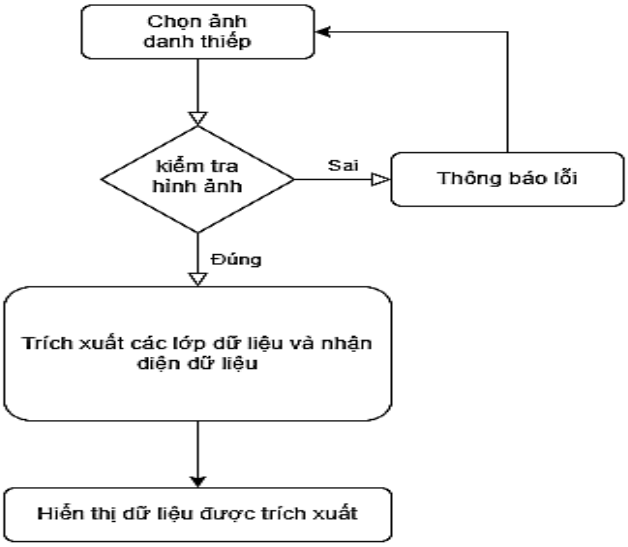
Sau khi huấn luyện các mô hình và đưa ra những kết quả so sánh, từ đó chúng tôi chọn ra mô hình có độ chính xác cao. Model YOLOv8l được triển khai Backend với ngôn ngữ Python và sử dụng Flask là một Python micro-framework cho phép xây dựng Web API. Sau khi người dùng chọn "Gửi" hệ thống sẽ gửi request cho server và dữ liệu được xử lý nhận diện ở Backend. Kết quả response của server sẽ được hiển thị ở khung "Kết quả".
Trong trường hợp dữ liệu đầu vào là ảnh bị lệch, nghiêng. Hệ thống sẽ trả về thông báo lỗi để người dùng điều chỉnh lại hình ảnh đầu vào đúng với yêu cầu của hệ thống. Thời gian xử lý được mô tả ở bảng 3. Ở bảng 3 cho thấy tốc độ xử lý khá nhanh đối với khi triển khai mô hình lên hệ thống và thoả điều kiện của hệ thống. Hệ thống được mô tả ở hình 6 và giao diện được mô tả ở hình 10.


© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

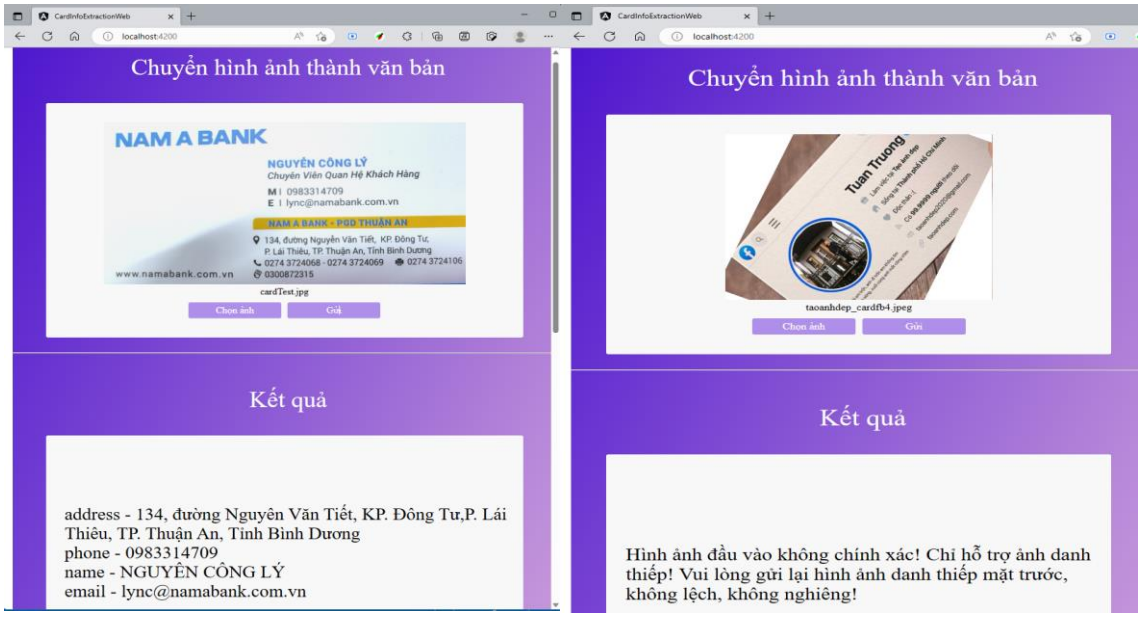
Bảng 3: Thời gian xử lý thông tin của hệ thống

| Trường hợp | Dung lượng ảnh | Thời gian xử lý |
|---|---|---|
| Dữ liệu đầu vào đúng | 140 KB | 3.06s |
| Dữ liệu đầu vào sai | 76.0 KB | 99ms |

Các bước triển khai được mô tả ở hình 9.




Hình 9: Sơ đồ triển khai lên ứng dụng




Hình 10. Giao diện của hệ thống

## 4 KẾT LUẬN

Trong bài báo, chúng tôi xây dựng một ứng dụng trích xuất thông tin quan trọng từ danh thiếp bằng kỹ thuật Deep Learning. Mô hình được huấn luyện trên tập dữ liệu được thu thập thủ công bao gồm 1013 ảnh

từ các mẫu danh thiếp khác nhau, trong đó chúng tôi trích xuất các thông tin bao gồm số điện thoại gồm 1214 nhãn, email 706 nhãn, tiếp theo là tên với 792 nhãn, địa chỉ với 849 nhãn. Chúng tôi tiến hành huấn luyện trên các mô hình học sâu YOLOv8n, YOLOv8l, YOLOv8x. Các mô hình được kiểm chứng trên tập test và được so sánh đánh giá độ chính xác cũng như hiệu quả của mô hình. Kết quả cho thấy 2 mô hình YOLOv8l và YOLOv8x có độ chính xác khá cao và chênh lệch không đáng kể. Tuy nhiên kích thước mô hình, thời gian huấn luyện và tốc độ nhận dạng đối tượng của YOLOv8l tốt hơn mô hình YOLOv8x. Dựa trên kết quả của các mô hình chúng tôi lựa chọn mô hình YOLOv8l để triển khai lên hệ thống để đem lại hiệu quả tốt nhất cho người dùng. Từ các lớp và vị trí được xác định ở bước huấn luyện, chúng tôi tiến hành cắt thành các ảnh nhỏ theo vị trí và lớp tương ứng và tiền xử lý bằng kỹ thuật xử lý ảnh để đảm bảo phần văn bản rõ ràng, nổi bật làm tăng độ chính xác, những hình ảnh này, sẽ đưa vào Tesseract để nhận diện ký tự trên ảnh và chuyển thành dạng văn bản. Trong bài báo còn một số hạn chế là dữ liệu còn khá ít không đáp ứng được hết nhu cầu của người sử dụng và mô hình vẫn chưa đạt được độ chính xác cao nhất. Trong tương lai chúng tôi sẽ xây dựng bộ dữ liệu đa dạng hơn, nhiều địa điểm hơn, xây dựng thêm các chức năng mới cho ứng dụng và tăng cường phần cứng để huấn luyện mô hình tốt hơn.

## TÀI LIỆU THAM KHẢO


1. X. Zhao, Q. Gao, S. Shen and Z. Wang, "Business Card Recognition and E-mail Delivery Based on NAO Robot," *2018 Chinese Control And Decision Conference (CCDC)*, Shenyang, China, 2018, pp. 5595-5599, doi: 10.1109/CCDC.2018.8408107.
2. Z. Zhang and S. Huang, "Credit Card Fraud Detection via Deep Learning Method Using Data Balance Tools," *2020 International Conference on Computer Science and Management Technology (ICCSMT)*, Shanghai, China, 2020, pp. 133-137, doi: 10.1109/ICCSMT51754.2020.00033.
3. M. L. Gambo, A. Zainal and M. N. Kassim, "A Convolutional Neural Network Model for Credit Card Fraud Detection," *2022 International Conference on Data Science and Its Applications (ICoDSA)*, Bandung, Indonesia, 2022, pp. 198-202, doi: 10.1109/ICoDSA55874.2022.9862930.
4. F. M. A. Mazen, R. A. A. Seoud and Y. O. Shaker, "Deep Learning for Automatic Defect Detection in PV Modules Using Electroluminescence Images," in *IEEE Access*, vol. 11, pp. 57783-57795, 2023, doi: 10.1109/ACCESS.2023.3284043
5. J. Xiang and G. Zhu, "Joint Face Detection and Facial Expression Recognition with MTCNN," *2017 4th International Conference on Information Science and Control Engineering (ICISCE)*, Changsha, China, 2017, pp. 424-427, doi: 10.1109/ICISCE.2017.95.

A. Krizhevsky, I. Sutskever, G.E. Hinton, "Imagenet classication with deep convolutional neural networks," *Proceedings of the 25th Interna- tional Conference on Neural Information Processing Systems*-Volume 1. NIPS'12, pp. 1097–1105. Curran Associates Inc., Red Hook, NY, USA, 2012.

6. K. He, X. Zhang, S. Ren, J. Sun, "Deep residual learning for image recognition, " *CoRR abs/1512.03385* ,2015.
7. R. Girshick, J. Donahue, T. Darrell and J. Malik, "Region-Based Convolutional Networks for Accurate Object Detection and Segmentation," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 1, pp. 142-158, 1 Jan. 2016, doi: 10.1109/TPAMI.2015.2437384.
8. W. Liu *et al*, " SSD: Single Shot MultiBox Detector, " *Computer Vision – ECCV 2016. ECCV 2016. Lecture Notes in Computer Science()*, vol 9905. Springer, Cham, 2016. https://doi.org/10.1007/978-3-319-46448-0_2.
9. R. Girshick, "Fast R-CNN," *2015 IEEE International Conference on Computer Vision (ICCV)*, Santiago, Chile, 2015, pp. 1440-1448, doi: 10.1109/ICCV.2015.169.


 © 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh

10. S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 1 June 2017, doi: 10.1109/TPAMI.2016.2577031.
11. S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137-1149, 1 June 2017, doi: 10.1109/TPAMI.2016.2577031.
12. K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 2980-2988, doi: 10.1109/ICCV.2017.322.
13. T. -Y. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal Loss for Dense Object Detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 318-327, 1 Feb. 2020, doi: 10.1109/TPAMI.2018.2858826.
14. J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 779-788, doi: 10.1109/CVPR.2016.91.
15. J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 6517-6525, doi: 10.1109/CVPR.2017.690.
16. Y. Zhou, W. Zhu, Y. He and Y. Li, "YOLOv8-based Spatial Target Part Recognition," *2023 IEEE 3rd International Conference on Information Technology, Big Data and Artificial Intelligence (ICIBA)*, Chongqing, China, 2023, pp. 1684-1687, doi: 10.1109/ICIBA56860.2023.10165260.
17. N. B. A. Karna, M. A. P. Putra, S. M. Rachmawati, M. Abisado and G. A. Sampedro, "Toward Accurate Fused Deposition Modeling 3D Printer Fault Detection Using Improved YOLOv8 With Hyperparameter Optimization," in *IEEE Access*, vol. 11, pp. 74251-74262, 2023, doi: 10.1109/ACCESS.2023.3293056.
18. [Ray Smith, "An overview of the Tesseract OCR engine", *Ninth international conference on document analysis and recognition (ICDAR 2007)*, vol. 2, 2007.
19. S. Thakare, A. Kamble, V. Thengne and U. R. Kamble, "Document Segmentation and Language Translation Using Tesseract-OCR," *2018 IEEE 13th International Conference on Industrial and Information Systems (ICIIS)*, Rupnagar, India, 2018, pp. 148-151, doi: 10.1109/ICIINFS.2018.8721372.
20. C. Sonavane, P. Kulkarni, O. Podey and P. Rewane, "Smart Surveillance and Tracking System using Resnet and Tesseract-OCR," *2021 IEEE Pune Section International Conference (PuneCon)*, Pune, India, 2021, pp. 1-6, doi: 10.1109/PuneCon52575.2021.9686493.

© 2023 Trường Đại học Công nghiệp Thành phố Hồ Chí Minh 

# KỶ YẾU HỘI NGHỊ KHOA HỌC TRẺ LẦN 5 NĂM 2023
# YSC2023 - TRI THỨC TRẺ TRONG KỶ NGUYÊN SỐ
# LĨNH VỰC CÔNG NGHỆ THÔNG TIN

TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP THÀNH PHỐ HỒ CHÍ MINH

**NHÀ XUẤT BẢN**
**ĐẠI HỌC CÔNG NGHIỆP TP HỒ CHÍ MINH**
12 Nguyễn Văn Bảo – P. 4 – Q. Gò Vấp – TPHCM
**ĐT**: (028) 3894 0390 – 816
**Email**: nhaxuatban@iuh.edu.vn

*Chịu trách nhiệm xuất bản:* PHẠM TRUNG KIÊN

*Biên tập:* LÊ THỊ TIỂU NHI
*Sửa bản in:* ĐOÀN THANH ĐIỀN
*Trình bày bìa:* VĂN SANG

*Đối tác liên kết: Trường Đại học Công nghiệp Thành phố Hồ Chí Minh*

**ISBN: 978-604-920-195-0**

In 55 cuốn khổ 20 × 28 cm theo Quyết định xuất bản số: 25/QĐ-NXBĐHCN ngày 24/10/2023 với xác nhận đăng kí xuất bản số 2882-2023/CXBIPH/1–14/ĐHCNTPHCM ngày 23/08/2023. In tại Xưởng in NXB Đại học Công nghiệp TPHCM, nộp lưu chiểu tháng 12/2023.