|  |  |
| --- | --- |
| **bỘ GIÁO DỤC VÀ ĐÀO TẠO** | **BỘ NÔNG NGHIỆP VÀ PTNT** |

TRƯỜNG ĐẠI HỌC THỦY LỢI



nguyễn cao thiêm

XÂY DỰNG ứng dụng điểm danh sinh viên qua qr code và công nghệ gps trên nền tảng ios

ĐỒ ÁN TỐT NGHIỆP

HÀ NỘI, NĂM 2022

|  |  |
| --- | --- |
| **bỘ GIÁO DỤC VÀ ĐÀO TẠO** | **BỘ NÔNG NGHIỆP VÀ PTNT** |

TRƯỜNG ĐẠI HỌC THỦY LỢI

nguyễn cao thiêm

XÂY DỰNG ứng dụng điểm danh sinh viên qua qrcode và công nghệ gps trên nền tảng ios

|  |  |
| --- | --- |
| Ngành : | Công nghệ thông tin |
| Mã số: |  |

|  |  |
| --- | --- |
| NGƯỜI HƯỚNG DẪN | **TS. Nguyễn Quỳnh Diệp** |

HÀ NỘI, NĂM 2022

|  |  |
| --- | --- |
| Logo-WRU | **CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc lập - Tự do - Hạnh phúc**  ----------★----------  **NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP** |

Họ tên sinh viên: Nguyễn Cao Thiêm Hệ đào tạo: Đại học chính quy

Lớp: 60TH3 Ngành: Công nghệ thông tin

Khoa: Công nghệ thông tin

1- TÊN ĐỀ TÀI: XÂY DỰNG ỨNG DỤNG ĐIỂM DANH SINH VEIEN QUA QRCODE VÀ CÔNG NGHỆ GPS TRÊN NỀN TẢNG IOS

2- CÁC TÀI LIỆU CƠ BẢN:

<https://www.w3schools.com/>

<https://gambaru.io/en/blog/sql-la-gi-cach-su-dung-sql>

<https://maclife.vn/staruml-ve-thiet-ke-so-do-mo-ta-he-thong.html>

3 - NỘI DUNG CÁC PHẦN THUYẾT MINH VÀ TÍNH TOÁN:

|  |  |
| --- | --- |
| Nội dung các phần | Tỷ lệ (%) |
| Chương 1: Tổng quan | 10% |
| Chương 2: Cơ sở lý thuyết và công nghệ | 20% |
| Chương 3: Đặc tả yêu cầu hệ thống | 10% |
| Chương 4: Phân tích và thiết kế | 35% |
| Chương 5: Xây dựng hệ thống | 25% |

4. GIÁO VIÊN HƯỚNG DẪN TỪNG PHẦN

|  |  |
| --- | --- |
| Nội dung các phần | Giáo viên hướng dẫn |
| Chương 1: Tổng quan | TS. Nguyễn Quỳnh Diệp |
| Chương 2: Cơ sở lý thuyết và công nghệ | TS. Nguyễn Quỳnh Diệp |
| Chương 3: Đặc tả yêu cầu hệ thống | TS. Nguyễn Quỳnh Diệp |
| Chương 4: Phân tích và thiết kế | TS. Nguyễn Quỳnh Diệp |
| Chương 5: Xây dựng hệ thống | TS. Nguyễn Quỳnh Diệp |

5. NGÀY GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

Ngày ............ tháng ......... năm 2022

|  |  |
| --- | --- |
| **Trưởng bộ môn**  *(Ký và ghi rõ Họ tên)* | **Giáo viên hướng dẫn chính**  *(Ký và ghi rõ Họ tên)* |

TS. Nguyễn Quỳnh Diệp

Nhiệm vụ Đồ án tốt nghiệp đã được hội đồng thi tốt nghiệp của khoa thông qua

Ngày . . . tháng . . . năm 2022 **Chủ tịch hội đồng**

*(Ký và ghi rõ Họ tên)*

Sinh viên đã hoàn thành và nộp bản Đồ án tốt nghiệp cho hội đồng thi ngày 25 tháng 12 năm 2022

**Sinh viên làm đồ án tốt nghiệp**

*(Ký và ghi rõ Họ tên)*

Nguyễn Cao Thiêm

**TIẾN ĐỘ THỰC HIỆN**

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Thời gian** | **Nội dung công việc** | **Kết quả dự kiến đạt được** |
| 1 | 26/9 –05/10 | * Tìm hiểu cơ sở lý thuyết | * Viết báo cáo phần CSLT |
| 2 | 5/10 – 20/10 | * Phân tích thiết kế hệ thống | * Hồ sơ phân tích thiết kế |
| 3 | 20/10 – 01/12 | * Xây dựng và cài đặt hệ thống | * Một phiên bản của hệ thống * Viết báo cáo phần cài đặt, đánh giá kết quả chạy thử |
| 4 | 01/12 – 25/12 | * Các chỉnh sửa hệ thống nếu có * Hoàn thiện báo cáo | * Phiên bản hệ thống hoàn chỉnh * Bản luận văn hoàn chỉnh |

|  |  |
| --- | --- |
| logo | TRƯỜNG ĐẠI HỌC THUỶ LỢI  **KHOA CÔNG NGHỆ THÔNG TIN**  BẢN TÓM TẮT ĐỀ CƯƠNG ĐỒ ÁN TỐT NGHIỆP |

TÊN ĐỀ TÀI: Xây dựng ứng dụng điểm danh sinh viên qua QRCode và công nghệ GPS trên nền tảng iOS.

*Sinh viên thực hiện*: Nguyễn Cao Thiêm

*Lớp*: 60TH3

*Giáo viên hướng dẫn*: TS. Nguyễn Quỳnh Diệp

**TÓM TẮT ĐỀ TÀI**

Dựa trên quá trình học tập tại trường Đại học Thuỷ Lợi, ta thấy mỗi tiết học đều có rất đông sinh viên, thời lượng tiết học dài, công tác điểm danh sinh viên của giảng viên về cơ bản hầu như vẫn thủ công: đọc tên sinh viên, làm bài tập nộp để điểm danh, tích điểm danh trên trang môn học,... Khiến thời gian điểm danh mất khoảng 5-10p tuỳ thuộc vào mức độ sinh viên. Sinh viên có thể dùng thủ thuật gian lận để qua mặt công tác điểm danh để điểm danh hộ.

Bài toán điểm danh là bài toán không phải mới, đã có nhiều cách giải quyết thậm chí là điểm danh bằng khuôn mặt với độ chính xác cao, tuy nhiên bên cạnh đó chi phí cao nếu lặp đặt cho từng lớp học trên giảng đường.

Vì vậy mục tiêu của đề tài là xây dựng một ứng dụng điểm danh sinh viên với chi phí rẻ, có tính bảo mật tốt, giúp giảm bớt thời gian điểm danh, tăng tính linh hoạt trong việc quản lý điểm danh sinh viên của giảng viên, quản trị viên. Trong đó có sử dụng các công nghệ phổ biến hiện nay như: QRCode, GPS ,….

**CÁC MỤC TIÊU CHÍNH**

***1. Ứng dụng di động với các chức năng chính tương ứng với các Vai trò:***

**Sinh viên**: Sử dụng ứng dụng để điểm danh (dùng QR để quét cho nhanh hoặc nút điểm danh), nhắc lịch học, nhận cảnh báo đi học muộn, nghỉ học, nghỉ quá số buổi, nhận thông báo thay đổi lịch học, phòng học, phản hồi việc nghỉ học quá số buổi để được xem xét.

**Giảng viên**: Sử dụng ứng dụng để tạo lịch học, cập nhật lịch học, phòng học, thời gian. Xem thống kê kết quả điểm danh của sinh viên môn học. Xem xét mở khoá tài khoản sinh viên nghỉ, đi muộn quá số buổi cho phép ( đối với học phần tương ứng). Có thể bổ sung chức năng export ra file excel.

**Quản trị viên**: Sử dụng website để quản lý đối tượng Sinh viên/Giảng viên, lịch học, phòng học, thời gian học. Có thể hỗ trợ import file excel ( theo mẫu có sẵn) để tối giản thời gian nhập dữ liệu.

***2. Các tính năng dự kiến (bổ sung sau khi hoàn thiện chức năng chính):***

**Xác thực**: Để xác minh tính hợp lệ khi Sinh viên quét QR code ( hoặc không quét) để điểm danh thì cần kiểm tra thêm vị trí của người dùng (có hoặc không gần địa điểm đã quy định trước). Có thể sẽ kết hợp sử dụng (FaceID hoặc TouchID) tuỳ dòng máy iPhone người dùng đang sử dụng để dịnh danh.

Giới hạn chỉ 1 tài khoản Sinh viên được phép đăng nhập trên 1 thiết bị trong cùng một thời điểm. (Sinh viên nếu cần đăng nhập trên thiết bị khác cần kích hoạt bởi Quản trị viên ).

***Công nghệ sử dụng trong đồ án****:*

**Mobile**: Sử dụng nền tảng iOS với các tính năng được tích hợp sẵn như: CoreLocation, UIKit, MapKit, VisionKit. Local database Realm để có thể thực hiện tracking location, quét QR code.

**Web**: Sử dụng ReactJS, HTML, CSS, JS để xây dựng giao diện web dành cho đối tượng Quản lý có thể tracking các thông tin về Sinh viên, Giảng viên, các chức năng tạo tài khoản, kích hoạt, xoá tài khoản.

**Server**: Sử dụng NodeJS để xây dựng server, cung cấp API để ứng dụng mobile và web có thể giao tiếp và xử lý thông tin.

**Database**: Sử dụng MySQL hoặc PostgreSQL.

**Authentication**: Sử dụng Firebase: Auth, push notification.

**Deployment**: Heroku cho Web + Server, Appstore cho ứng dụng.

**KẾT QUẢ DỰ KIẾN**

❖ **Giải pháp**: Xây dựng và phát triển được một hệ thống end-to-end từ mobile, server, web, deployment production để cung cấp giải pháp cho đề tài.

❖ **Ứng dụng**: Có giao diện sử dụng rõ ràng, dễ sử dụng, có tính bảo mật (ảnh QRCode được mã hoá)

❖ **Website quản trị**: Có giao diện trực quan dành cho Quản trị viên, quản trị viên có thể import file Excel ( theo mẫu có sẵn) để thuận tiện trong việc nhập, xuất dữ liệu.

❖ **Báo cáo Đồ án tốt nghiệp**

**LỜI CAM ĐOAN**

Em xin cam đoan đây là Đồ án tốt nghiệp của bản thân em. Các kết quả trong Đồ án tốt nghiệp này là trung thực, và tất cả những tham khảo từ các nghiên cứu liên quan đều được nêu nguồn gốc một cách rõ ràng từ danh mục tài liệu tham khảo trong đồ án. Trong đồ án, không có việc sử dụng tài liệu, công trình nghiên cứu của người khác mà không chỉ rõ về tài liệu tham khảo. Các chương trình thực nghiệm đều là do em thực hiện mà có được, không sao chép toàn văn từ bất cứ nguồn nào.

|  |  |
| --- | --- |
|  | **Sinh viên thực hiện**  *Thiêm*  **Nguyễn Cao Thiêm** |

**LỜI CẢM ƠN**

Lời đầu tiên em xin gửi lời cảm ơn đến các thầy cô tại trường Đại Học Thủy Lợi nói chung và các thầy cô trong khoa Công nghệ thông tin nói riêng. Các thầy cô đã truyền đạt những kiến thức, kĩ năng hữu ích, tận tâm giảng dạy và giúp đỡ em suốt 4,5 năm qua, giúp em có đủ điều kiện và kiến thức để hoàn thành đồ án tốt nghiệp này.

Em xin gửi lời cảm ơn chân thành và biết ơn sâu sắc nhất đến cô TS. Nguyễn Quỳnh Diệp người đã tận tình hướng dẫn, chỉ bảo, giúp đỡ em thực hiện đồ án tốt nghiệp trong suốt thời gian vừa qua.

Cuối cùng, em xin chân thành cảm ơn gia đình và bạn bè đã luôn tạo điều kiện, quan tâm, giúp đỡ, trao đổi kiến thức, kinh nghiệm và động viên em trong thời gian học tập và nghiên cứu trong suốt quá trình học tập và hoàn thành đồ án tốt nghiệp.

Mặc dù đã cố gắng hoàn thành đồ án tốt nghiệp trong phạm vi và khả năng cho phép nhưng do vẫn còn nhiều thiếu sót nên em kính mong nhận được sự cảm thông và tận tình chỉ bảo từ phía quý thầy cô và các bạn.

**MỤC LỤC**

[DANH MỤC CÁC HÌNH ẢNH vi](#_Toc122621624)

[DANH MỤC BẢNG BIỂU viii](#_Toc122621625)

[DANH MỤC CÁC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ ix](#_Toc122621626)

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc122621627)

[1.1 Đặt vấn đề 1](#_Toc122621628)

[1.2 Mục tiêu đề tài 2](#_Toc122621629)

[1.3 Phạm vi đề tài 3](#_Toc122621630)

[1.4 Phương pháp nghiên cứu 4](#_Toc122621631)

[CHƯƠNG 2 CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ 5](#_Toc122621632)

[2.1 Cơ sở lý thuyết 5](#_Toc122621633)

[2.1.1 Khái quát chung về công tác quản lý điểm danh sinh viên 5](#_Toc122621634)

[2.1.2 Tổng quan về hệ thống điểm danh dựa trên Sinh trắc học 6](#_Toc122621635)

[2.1.3 Các hình thức điểm danh dựa trên sinh trắc học trong thực tế 6](#_Toc122621636)

[2.1.4 Lợi ích và hạn chế của các hệ thống điểm danh thường gặp 9](#_Toc122621637)

[2.2 Công cụ phát triển 11](#_Toc122621638)

[2.2.1 Nền tảng phát triển hệ thống 11](#_Toc122621639)

[2.2.2 Quản lý mã nguồn 17](#_Toc122621640)

[2.2.3 Cơ sở dữ liệu 19](#_Toc122621641)

[2.2.4 Môi trường triển khai hệ thống 20](#_Toc122621642)

[CHƯƠNG 3 KHẢO SÁT HIỆN TRẠNG VÀ PHÂN TÍCH , THIẾT KẾ HỆ THỐNG 25](#_Toc122621643)

[3.1 Khảo sát thực tế các phần mềm quản lý sinh viên 25](#_Toc122621644)

[3.2 Phân tích người dùng hệ thống 27](#_Toc122621645)

[3.2.1 Các tác nhân trong hệ thống 27](#_Toc122621646)

[3.2.2 Yêu cầu về chức năng chính 27](#_Toc122621647)

[3.2.3 Yêu cầu nghiệp vụ bảo mật của hệ thống 27](#_Toc122621648)

[3.2.4 Yêu cầu nghiệp vụ dành cho người dùng hệ thống 28](#_Toc122621649)

[3.3 Phân tích các ca sử dụng 29](#_Toc122621650)

[3.3.1 Xây dựng biểu đồ tổng quát UseCase 29](#_Toc122621651)

[3.3.2 UseCase quản trị viên đăng nhập hệ thống 30](#_Toc122621652)

[3.3.3 UseCase giảng viên đăng nhập ứng dụng 31](#_Toc122621653)

[3.3.4 UseCase sinh viên đăng nhập ứng dụng 34](#_Toc122621654)

[3.3.5 UseCase giảng viên xem danh sách lớp học 36](#_Toc122621655)

[3.3.6 UseCase sinh viên xem danh sách lớp học 38](#_Toc122621656)

[3.3.7 UseCase giảng viên xem chi tiết lớp học 40](#_Toc122621657)

[3.3.8 UseCase sinh viên xem chi tiết lớp học 42](#_Toc122621658)

[3.3.9 UseCase sinh viên xem thông báo 44](#_Toc122621659)

[3.3.10 UseCase giảng viên cập nhật thông tin điểm danh 46](#_Toc122621660)

[3.3.11 UseCase quản trị viên gửi thông báo nhắc nhở sinh viên 46](#_Toc122621661)

[3.3.12 UseCase sinh viên gửi phản hồi tới giảng viên 48](#_Toc122621662)

[3.3.13 UseCase giảng viên tạo ảnh QR Code để sinh viên điểm danh 51](#_Toc122621663)

[3.3.14 UseCase sinh viên điểm danh nhanh bằng QR Code 51](#_Toc122621664)

[3.3.15 UseCase quản trị viên quản lý sinh viên 54](#_Toc122621665)

[3.3.16 UseCase quản trị viên quản lý giảng viên 59](#_Toc122621666)

[3.3.17 UseCase quản trị viên quản lý môn học 63](#_Toc122621667)

[3.3.18 UseCase quản trị viên quản lý phòng học 67](#_Toc122621668)

[3.3.19 UseCase quản trị viên quản lý lớp học phần 71](#_Toc122621669)

[3.3.20 UseCase quản trị viên quản lý khoa 75](#_Toc122621670)

[3.3.21 UseCase quản trị viên quản lý lớp học biên chế 79](#_Toc122621671)

[3.4 Xây dựng sơ đồ lớp 85](#_Toc122621672)

[3.4.1 Xác định lớp 85](#_Toc122621673)

[3.4.2 Xác định các kiểu dữ liệu và phương thức 86](#_Toc122621674)

[3.4.3 Sơ đồ lớp 88](#_Toc122621675)

[3.5 Mô hình dữ liệu thực thể 88](#_Toc122621676)

[CHƯƠNG 4 THIẾT KẾ CHI TIẾT HỆ THỐNG 89](#_Toc122621677)

[4.1 Bảng cơ sở dữ liệu 89](#_Toc122621678)

[4.2 Thiết kế cơ sở dữ liệu 90](#_Toc122621679)

[4.2.1 Bảng SINHVIEN 90](#_Toc122621680)

[4.2.2 Bảng KHOA 90](#_Toc122621681)

[4.2.3 Bảng LOP 91](#_Toc122621682)

[4.2.4 Bảng GIANGVIEN 91](#_Toc122621683)

[4.2.5 Bảng PHONGHOC 92](#_Toc122621684)

[4.2.6 Bảng CAHOC 92](#_Toc122621685)

[4.2.7 Bảng LOPMONHOC 92](#_Toc122621686)

[4.2.8 Bảng MONHOC 93](#_Toc122621687)

[4.2.9 Bảng GV\_DIEMDANH\_SV 93](#_Toc122621688)

[4.2.10 Bảng MONHOC\_LOPMONHOC 93](#_Toc122621689)

[4.2.11 Bảng CHITIETDD 94](#_Toc122621690)

[4.2.12 Bảng CA\_PHONG\_LOPMONHOC 94](#_Toc122621691)

[4.2.13 Bảng SINHVIEN\_HOC\_LOPMONHOC 94](#_Toc122621692)

[4.3 Thiết kế giao diện 95](#_Toc122621693)

[4.3.1 Giao diện trang đăng nhập của quản trị viên 95](#_Toc122621694)

[4.3.2 Giao diện trang chủ của quản trị viên 95](#_Toc122621695)

[4.3.3 Giao diện trang quản lý sinh viên 96](#_Toc122621696)

[4.3.4 Giao diện trang quản lý giảng viên 96](#_Toc122621697)

[4.3.5 Giao diện trang quản lý môn học 97](#_Toc122621698)

[4.3.6 Giao diện trang quản lý phòng học 97](#_Toc122621699)

[4.3.7 Giao diện trang quản lý lớp học biên chế 98](#_Toc122621700)

[4.3.8 Giao diện trang quản lý lớp học phần 98](#_Toc122621701)

[4.3.9 Giao diện màn hình đăng nhập ứng dụng 99](#_Toc122621702)

[4.3.10 Giao diện màn hình Trang chủ ứng dụng 99](#_Toc122621703)

[4.3.11 Giao diện màn hình Tiện ích ứng dụng 100](#_Toc122621704)

[4.3.12 Giao diện màn hình Góc học tập ứng dụng 100](#_Toc122621705)

[4.3.13 Giao diện màn hình Thông báo ứng dụng 100](#_Toc122621706)

[4.3.14 Giao diện màn hình Cá nhân ứng dụng 100](#_Toc122621707)

[4.3.15 Giao diện màn hình Chi tiết lớp học 100](#_Toc122621708)

[4.3.16 Giao diện màn hình thống kê điểm danh sinh viên ứng dụng 100](#_Toc122621709)

[CHƯƠNG 5 KIỂM THỬ HỆ THỐNG 101](#_Toc122621710)

[5.1 Xây dựng bộ dữ liệu kiểm thử 101](#_Toc122621711)

[5.2 Kế hoạch kiểm thử 101](#_Toc122621712)

[5.3 Kết quả kiểm thử 104](#_Toc122621713)

[**KẾT LUẬN** 105](#_Toc122621714)

[**TÀI LIỆU THAM KHẢO** 106](#_Toc122621715)

# DANH MỤC CÁC HÌNH ẢNH

Hình 3.1 Biểu đồ thị phần di động của các Hệ điều hành 12

Hình 3.1Biểu đồ UseCase tổng quát 29

Hình 3.2 Biểu đồ phân rã UseCase quản trị viên đăng nhập hệ thống 30

Hình 3.3 Biểu đồ hoạt động chức năng quản trị viên đăng nhập hệ thống 31

Hình 3.4 Biểu đồ tuần tự chức năng quản trị viên đăng nhập hệ thống 31

Hình 3.5 Biểu đồ phân rã UseCase giảng viên đăng nhập ứng dụng 31

Hình 3.6 Kịch bản chức năng giảng viên đăng nhập ứng dụng 32

Hình 3.7 Biểu đồ hoạt động chức năng giảng viên đăng nhập ứng dụng 33

Hình 3.8 Biểu đồ tuần tự chức năng giảng viên đăng nhập ứng dụng 33

Hình 3.9 Biểu đồ phân rã UseCase sinh viên đăng nhập ứng dụng 34

Hình 3.10 Biểu đồ hoạt động sinh viên đăng nhập ứng dụng 35

Hình 3.11 Biểu đồ tuần tự chức năng sinh viên đăng nhập ứng dụng 35

Hình 3.12 Biểu đồ phân rã UseCase giảng viên xem danh sách lớp học 36

Hình 3.13 Biểu đồ hoạt động giảng viên xem danh sách lớp học 37

Hình 3.14 Biểu đồ tuần tự chức năng giảng viên xem danh sách lớp học 37

Hình 3.15 Biểu đồ phân rã UseCase sinh viên xem danh sách lớp học 38

Hình 3.16 Biểu đồ hoạt động chức năng sinh viên xem danh sách lớp học 39

Hình 3.17 Biểu đồ tuần tự chức năng sinh viên xem danh sách lớp học 39

Hình 3.18 Biểu đồ phân rã giảng viên xem chi tiết lớp học 40

Hình 3.19 Biểu đồ hoạt động chức năng giảng viên xem chi tiết lớp học 41

Hình 3.20 Biểu đồ tuần tự chức năng giảng viên xem chi tiết lớp học 41

Hình 3.21 Biểu đồ phân rã UseCase sinh viên xem chi tiết lớp học 42

Hình 3.22 Biểu đồ hoạt động chức năng sinh viên xem chi tiết lớp học 43

Hình 3.23 Biểu đồ tuần tự chức năng sinh viên xem chi tiết lớp học 43

Hình 3.27 Biểu đồ phân rã UseCase sinh viên xem thông báo 44

Hình 3.28 Biểu đồ hoạt động chức năng sinh viên xem thông báo 45

Hình 3.29 Biểu đồ tuần tự chức năng sinh viên xem thông báo 45

Hình 3.30 Biểu đồ phân rã UseCase quản trị viên gửi thông báo nhắc nhở sinh viên 46

Hình 3.31 Biểu đồ hoạt động chức năng quản trị viên gửi thông báo nhắc nhở sinh viên 47

Hình 3.32 Biểu đồ tuần tự chức năng quản trị viên gửi thông báo nhắc nhở sinh viên 48

Hình 3.33 Biểu đồ phân rã UseCase sinh viên gửi phản hồi tới giảng viên 48

Hình 3.34 Biểu đồ hoạt động chức năng sinh viên gửi phản hồi tới giảng viên 50

Hình 3.35 Biểu đồ tuần tự chức năng sinh viên gửi phản hồi tới giảng viên 51

Hình 3.36 Biểu đồ phân rã UseCase sinh viên điểm danh nhanh bằng QRCode 51

Hình 3.37 Biểu đồ hoạt động chức năng sinh viên điểm danh nhanh bằng QRCode 53

Hình 3.38 Biểu đồ tuần tự chức năng sinh viên điểm danh nhanh bằng QRCode 54

Hình 3.39 Biểu đồ phân rã UseCase quản trị viên quản lý sinh viên 54

Hình 3.40 Biểu đồ hoạt động chức năng thêm sinh viên 56

Hình 3.41 Biểu đồ tuần tự chức năng thêm sinh viên 57

Hình 3.42 Biểu đồ hoạt động chức năng sửa thông tin sinh viên 57

Hình 3.43 Biểu đồ tuần tự chức năng sửa thông tin sinh viên 58

Hình 3.44 Biểu đồ phân rã UseCase quản trị viên quản lý giảng viên 59

Hình 3.45 Biểu đồ hoạt động chức năng quản trị viên thêm giảng viên 61

Hình 3.46 Biểu đồ tuần tự chức năng quản trị viên thêm giảng viên 61

Hình 3.47 Biểu đồ hoạt động chức năng sửa thông tin giảng viên 62

Hình 3.48 Biểu đồ tuần tự chức năng sửa thông tin giảng viên 62

Hình 3.51 Biểu đồ phân rã UseCase quản trị viên quản lý môn học 63

Hình 3.52 Biểu đồ hoạt động chức năng quản trị viên thêm môn học 65

Hình 3.53 Biểu đồ tuần tự chức năng quản trị viên thêm môn học 65

Hình 3.54 Biểu đồ hoạt động chức năng sửa môn học 66

Hình 3.55 Biểu đồ tuần tự chức năng sửa môn học 66

Hình 3.58 Biểu đồ phân rã UseCase quản trị viên quản lý phòng học 67

Hình 3.59 Biểu đồ hoạt động chức năng quản trị viên thêm phòng học 69

Hình 3.60 Biểu đồ tuần tự chức năng thêm phòng học 69

Hình 3.61 Biểu đồ hoạt động chức năng sửa phòng học 70

Hình 3.62 Biểu đồ tuần tự chức năng sửa phòng học 70

Hình 3.65 Biểu đồ phân rã UseCase quản trị viên quản lý lớp học phần 71

Hình 3.66 Biểu đồ hoạt động chức năng thêm lớp học phần 72

Hình 3.67 Biểu đồ tuần tự chức năng thêm lớp học phần 73

Hình 3.68 Biểu đồ hoạt động chức năng sửa lớp học phần 73

Hình 3.69 Biểu đồ tuần tự chức năng sửa lớp học phần 74

Hình 3.70 Biểu đồ hoạt động chức năng xóa lớp học phần 74

Hình 3.71 Biểu đồ tuần tự chức năng xóa lớp học phần 75

Hình 3.72 Biểu đồ phân rã UseCase quản trị viên quản lý khoa 75

Hình 3.73 Biểu đồ hoạt động chức năng thêm khoa 77

Hình 3.74 Biểu đồ hoạt động chức năng thêm khoa 78

Hình 3.75 Biểu đồ hoạt động chức năng sửa khoa 78

Hình 3.76 Biểu đồ tuần tự chức năng sửa khoa 79

Hình 3.79 Biểu đồ phân rã UseCase quản trị viên quản lý lớp học biên chế 79

Hình 3.80 Biểu đồ hoạt động chức năng thêm lớp học biên chế 82

Hình 3.81 Biểu đồ tuần tự chức năng thêm lớp học biên chế 82

Hình 3.82 Biểu đồ hoạt động chức năng sửa lớp học biên chế 83

Hình 3.83 Biểu đồ tuần tự chức năng sửa lớp học biên chế 83

Hình 3.84 Biểu đồ hoạt động chức năng xóa lớp học biên chế 84

Hình 3.85 Biểu đồ tuần tự chức năng xóa lớp học biên chế 84

Hình 3.93 Biểu đồ lớp của hệ thống 88

Hình 3.94 Mô hình dữ liệu thực thể 88

Hình 4.1 Bảng cơ sở dữ liệu 89

Hình 4.2 Giao diện trang chủ 95

Hình 4.3 Giao diện trang liên hệ 95

Hình 4.4 Giao diện trang đăng ký 96

Hình 4.5 Giao diện trang đăng nhập 96

Hình 4.6 Giao diện trang quản lý 97

Hình 4.7 Giao diện trang đặt hàng 97

Hình 4.8 Giao diện trang xem thông tin sản phẩm 98

Hình 4.9 Giao diện trang quản lý người dùng 98

Hình 4.10 Giao diện trang quản lý danh mục 99

Hình 4.11 Giao diện trang quản lý sản phẩm 99

Hình 4.12 Giao diện trang quản lý bài viết 100

Hình 4.13 Giao diện trang chủ người dùng 100

DANH MỤC BẢNG BIỂU

Bảng 3.1 Kịch bản chức năng quản trị viên đăng nhập hệ thống 30

Bảng 3.2 Kịch bản sinh viên đăng nhập ứng dụng 35

Bảng 3.3 Kịch bảng giảng viên xem danh sách lớp học 36

Bảng 3.4 Kịch bản chức năng sinh viên xem danh sách lớp học 38

Bảng 3.5 Kịch bản chức năng giảng viên xem chi tiết lớp học 40

Bảng 3.6 Kịch bản chức năng sinh viên xem chi tiết lớp học 42

Bảng 3.8 Kịch bản chức năng sinh viên xem thông báo 44

Bảng 3.9 Kịch bản chức năng quản trị viên gửii thông báo nhắc nhở sinh viên 47

Bảng 3.10 Kịch bản chức năng sinh viên gửi phản hồi tới giảng viên 49

Bảng 3.11 Kịch bản chức năng sinh viên điểm danh nhanh bằng QRCode 52

Bảng 3.12 Kịch bản chức năng quản trị viên quản lý sinh viên 56

Bảng 3.13 Kịch bản chức năng quản trị viên quản lý giảng viên 60

Bảng 3.16 Kịch bản chức năng quản trị viên quản lý môn học 64

Bảng 3.17 Kịch bản chức năng quản trị viên quản lý phòng học 68

Bảng 3.18 Kịch bản chức năng quản trị viên quản lý lớp học phần 72

Bảng 3.19 Kịch bản chức năng quản trị viên quản lý khoa 77

Bảng 3.20 Kịch bản chức năng quản trị viên quản lý lớp học biên chế 81

Bảng 3.22 Danh sách các lớp của hệ thống 85

Bảng 3.23 Xác định lớp các kiểu dữ liệu và hàm 87

Bảng 4.1 Mô tả bảng SINHVIEN 90

Bảng 4.2 Mô tả bảng KHOA 90

Bảng 4.3 Mô tả bảng LOP 91

Bảng 4.4 Mô tả bảng GiANGVIEN 91

Bảng 4.5 Mô tả bảng PHONGHOC 92

Bảng 4.6 Mô tả bảng CAHOC 92

Bảng 4.7 Mô tả bảng LOPMOCHOC 92

Bảng 4.8 Mô tả bảng MONHOC 93

Bảng 4.9 Mô tả bảng GV\_DIEMDANH\_SV 93

Bảng 4.10 Mô tả bảng MONHOC\_LOPMONHOC 93

Bảng 4.11 Mô tả bảng CHITIETDD 94

Bảng 4.12 Mô tả bảng CA\_PHONG\_LOPMONHOC 94

Bảng 5.1 Bảng kế hoạch kiểm thử 104

DANH MỤC CÁC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ

|  |  |
| --- | --- |
| **Từ viết tắt** | **Viết đầy đủ** |
| **CSDL** | Cơ sở dữ liệu |
| **HTML** | Hypertext Markup Language |
| **MVC** | **Model-View-Controller** |
| **UML** | Unified Modeling Language |

# CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI

## Đặt vấn đề

Dựa trên quá trình học tập tại trường Đại học Thuỷ Lợi, em thấy mỗi tiết học đều có rất đông sinh viên, thời lượng tiết học dài, công tác điểm danh sinh viên của giảng viên về cơ bản hầu như vẫn thủ công: đọc tên sinh viên, làm bài tập nộp để điểm danh, tích điểm danh trên trang môn học. Khiến thời gian điểm danh mất khoảng 5-10p tuỳ thuộc vào mức độ sinh viên. Sinh viên có thể dùng thủ thuật gian lận để qua mặt công tác điểm danh để điểm danh hộ.

Trong thực tế, việc quản lý sinh viên ở nhiều trường Đại học còn mang tính thủ công chưa hiệu quả trong công tác quản lý. Công tác quản lý hiện nay chủ yếu được thực hiện thủ công, mất nhiều thời gian. Sự cần thiết của một hệ thống quản lý sẽ là một lựa chọn tốt nhất để giải quyết tình trạng hiện nay. Qua quá trình tìm hiểu và phân tích, chỉ ra việc chưa hiệu quả của quy trình quản lý hiện đang triển khai, nêu ra những yêu cầu mới đặt ra đối với công tác quản lý điểm danh sinh viên, qua đó đề xuất những phương án để khắc phục những khó khan đang gặp phải nhằm nâng cao hiệu quả trong công tác quản lý điểm danh sinh viên tại trường học và đẩy nhanh tiến trình ứng dụng tin học hóa trong công tác quản lý điểm danh sinh viên.

Bài toán điểm danh là bài toán không phải mới, đã có nhiều cách giải quyết thậm chí là điểm danh bằng khuôn mặt với độ chính xác cao, tuy nhiên bên cạnh đó chi phí cao nếu lặp đặt cho từng lớp học trên giảng đường.

Vì vậy mục tiêu của đề tài là xây dựng một ứng dụng điểm danh sinh viên với chi phí rẻ, có tính bảo mật tốt, giúp giảm bớt thời gian điểm danh, tăng tính linh hoạt trong việc quản lý điểm danh sinh viên của giảng viên, quản trị viên. Trong đó có sử dụng các công nghệ phổ biến hiện nay như: QRCode, GPS.

Nhận thức được những điều trên cùng với như cầu thực tế. Đồng thời dưới sự đồng ý của giáo viên hướng dẫn và sự chấp nhận đề tài từ phía nhà trường em đã nhận đề tài “***Xây dựng ứng dụng điểm danh sinh viên qua QRCode và công nghệ GPS trên nền tảng iOS***”

## Mục tiêu đề tài

Với đề tài “**Xây dựng ứng dụng điểm danh sinh viên qua QRCode và công nghệ GPS trên nền tảng iOS**” được giao, em đã xác định được mục tiêu dưới đây

* Nghiên cứu phương pháp và các vấn đề liên quan để xây dựng một hệ thông end-to-end bao gồm ứng dụng di động, server và website.
* Khảo sát và tìm hiểu những đặc trưng của các ứng dụng thiết bị di động có sử dụng công nghệ QRCode và GPS trên thị trường.
* Phân tích thiết kế hệ thống thông tin.
* Nghiên cứu phương pháp và các xây dựng ứng dụng di động trên nền tảng iOS.
* Nghiên cứu phương pháp và các xây dựng website.
* Nghiên cứu phương pháp và xây dựng server.
* Nâng cao kỹ năng tìm kiếm thông tin, học hỏi và áp dụng công nghệ mới vào việc phát triển các tính năng cho các nền tảng trên.
* Cải thiện khả năng viết báo cáo đồ án và khả năng thuyết trình.

*Kết quả dự kiến:*

* Phát triển được một hệ thống đầy đủ từ ứng dụng di động, web, server và deploy production hoàn chỉnh.
* Hồ sơ phân tích thiết kế hệ thống.
* Ứng dụng có giao diện rõ ràng, dễ sử dụng, tính bảo mật tương đối tốt.
* Server ổn định, có thể hoạt động 24/24. Database được back-up thường xuyên để tránh các sự việc không mong muốn
* Website quản trị có giao diện trực quan, có các chức năng cần thiết cho người dùng

## Phạm vi đề tài

Phạm vi tìm hiểu lý thuyết

* Xây dựng một ứng dụng di động trên nền tảng iOS, định nghĩa, cách thức hoạt động của ứng dụng, sử dụng các chức năng sẵn có của một ứng dụng di động như QR Code, GPS.
* Xây dựng một website sử dụng ReactJS. Tìm hiểu định nghĩa, cách thức hoạt độngvà cài đặt trên môi trường phát triển.
* Xây dựng một server sử dụng NodeJS. Tìm hiểu cách thức hoạt động và cài đặt trên môi trường phát triển.
* Tìm hiểu, nghiên cứu, đăng ký và triển khai toàn bộ hệ thống bao gồm: ứng dụng di động. website, server. Các môi trường triển khai Appstore với ứng dụng di động, Heroku platform với website và back-end server.

Các chức năng được xây dựng trong hệ thống:

**Đối với sinh viên:**

* Đăng nhập vào hệ thống để xem thông tin lớp học, điểm danh lớp học, các môn học mình đã đăng ký, danh sách lớp học của mình.
* Nhận thông báo nhắc lịch học, nhận cảnh báo đi học muộn, nghỉ học, nghỉ quá số buổi.
* Nhận thông báo thay đổi lịch học, phòng học.
* Phản hồi việc nghỉ học quá số buổi để được giáo viện xem xét.
* Đăng nhập trên một thiết bị duy nhất với mỗi tài khoản được cho phép.

**Đối với giảng viên:**

* Sử dụng ứng dụng để xem kết quả điểm danh của sinh viên thuộc một môn học nhất định
* Xem thông kê kết quả điểm danh của sinh viên trong một lớp học bao gồm: danh sách sinh viên, số sinh viên vắng, số sinh viên điểm danh đúng giờ.
* Xem xét mở khóa, xét duyệt sinh viên bị khóa tài khoản do nghỉ học, đi học muộn quá số buổi.

**Đối với quản trị viên:**

* Sử dụng website để quản lý đối tượng Sinh viên/ Giảng viên, lịch học, phòng học, môn học, ca học.
* Có thể import file excel (theo mẫu có sẵn) để thuận tiện cho quá trình nhập dữ liệu.
* Xem thông kê kết quả điểm danh của sinh viên theo từng lớp học.
* Xem xét giúp người dùng reset mặt khẩu về mật khẩu mặc định, reset thiết bị người dùng sử dụng để người dùng có thể đăng nhập lại trên thiết bị đó.

## Phương pháp nghiên cứu

Để hoàn thành được đề tài một cách thuận lợi thì em sẽ nghiên cứu lý thuyết kết hợp nghiên cứu thực nghiệm. Cụ thể như sau:

*Nghiên cứu lý thuyết:*

* Nghiên cứu, quan sát các quy trình nghiệp vụ trong quá trình quản lý điểm danh sinh viên để phân tích các quy tắc cơ bản để thiết kế và xây dựng hệ thống mới.
* Nghiên cứu các tài liệu liên quan đến việc thiết kế và vận hành một hệ thống ứng dụng di động, website và server
* Nghiên cứu các tài liệu liên quan đến định danh người dùng ứng dụng di động đi kèm với sửa dụng tính năng GPS trong việc xác minh người dùng.
* Nghiên cứu các tài liệu liên quan đến việc triển khai hệ thống lên môi trường thực tế, có thể đưa vào sử dụng thử trong một môi trường kiểm thử nhỏ.
* Kết quả: Có được bộ tài liệu hoàn chỉnh, từ đó có thể phát triển một hệ thống hoàn chỉnh thỏa mãn các yêu cầu được nêu ra.

*Nghiên cứu thực nghiệm:*

* Áp dụng lý thuyết để xây dựng một hệ thống hoàn chỉnh
* Đánh giá các lỗ hổng bảo mật có thể xảy ra và phương pháp sửa lỗi nếu gặp
* Đến tận cơ sở thực tế để lấy dữ liệu vị trí (GPS) tương ứng trong địa điểm thật để công đoạn kiểm thử hệ thống đạt hiệu quả chính xác đáng tin cậy hơn.
* Hoàn thiện, triển khai hệ thống như yêu cầu đề ra và đánh giá tính thành công của hệ thông.

# CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

## Cơ sở lý thuyết

### Khái quát chung về công tác quản lý điểm danh sinh viên

Trong thời đại công nghệ kỹ thuật số tiên tiến như hiện nay, có rất nhiều cách để ghi lại thông tin cá nhân của một người tham dự một sự kiện hoặc người đó đã tham dự ở một nơi nào đó. Điểm danh là một yếu tố quan trọnng để xác định liệu người đó đã được tham dự vào một dịp nào đó hay chưa.

Ví dụ trong công tác điểm danh sinh viên, cách truyền thông thống thường được sử dụng để ghi lại những sinh viên đã tham dựng bằng cách gọi tên với lớp học có sức chứa khảong 10-20 sinh viên hoặc lớn hơn không nhiều. Thông thường ở các trường tiểu học và trung học, mỗi lớp chỉ bao gồm một giao viên đứng lớp với số lượng sinh viên thích hợp để giáo viên có thể sử dụng cách điểm danh truyền thống gọi tên từng học sinh.

Tuy nhiên, khi đến trường Đại học hoặc Cao đẳng, nó có thể sẽ không phù hợp với sử dụng cách truyền thống để theo dõi quá trình điểm danh của sinh viên vì sức chứa cho một lớp ở môi trường Đại học / Cao đẳng sẽ lớn và theo dõi chính xác số người tham dự hơn rất nhiều. Đây sẽ là một vấn đề với phương pháp điểm danh truyền thống này do giảng viên cần gọi tên từng sinh viên từ 40-50 sinh viên để đảm bảo không gọi sót sinh viên nào.

Bên cạnh đó, giờ đây nhiều trường Đại học / Cao học đang sử dụng ký tên chữ ký trên giấy điểm danh như một cách để chứng minh sinh viên có tham dự lớp học hoặc phòng thí nghiệm thực tế. Điều này sẽ gây lãng phí nhiều nhân lực và thời gian hơn cho một lớp học lớn hơn với số lượng sinh viên lớn hơn, lãng phí nhiều tài nguyên hơn vào việc ký tên vào tờ giấy điểm danh.

Học sinh, sinh viên thì thông minh và khéo léo hơn rất nhiều, họ có thể giúp đỡ các người bạn của họ bằng cách bắt chước chữ ký cuả họ. Điều này sẽ dẫn đến một vấn đề nghiêm trọng nếu học sinh gian lậnh điểm danh nhưng không may bị vấn đề hay tai nạn ngoài khuôn viên nhà trường, giảng viên và người phụ trách lớp sẽ chịu hoàn toàn trách nhiệm về vấn đề này.

Như vậy chúng ta đã thấy, đây sẽ là một vấn đề nghiêm trọng về giấy tờ, nhân lực và thời gian nếu chúng ta sử dụng các cách điểm danh truyền thống cũ đối với một tập thể đông như lớp Đại học/Cao đẳng.

### Tổng quan về hệ thống điểm danh dựa trên Sinh trắc học

Trong những năm gần đây, cùng với sự phát triển của công nghệ thông tin, đã có nhiều sự cải tiến tuyệt vời trong môi trường của chúng ta như: nhà thông minh, thành phố thông minh, trí tuệ nhân tạo, xe tự hành,… Khi mọi thứ đều đang được cải thiện nâng cao, chúng ta nên tăng cường sử dụng hệ thống theo điểm danh theo hướng hiện đại hoá. Trong thời đại của Internet of things ( IOTs), có thể sử dụng nhiều thiết bị hơn như một cách hiện đại để điểm danh học sinh. Để phân biệt mọi người với nhau, có nhiều phương pháp tiên tiến như: mã QR, nhận dạng vân tay (Fingerprint recognition), nhận dạng khuôn mặt (Face recognition), Iris recognition, hệ thống RFID hay mã vạch.

Hệ thống mã QR là sự kết hợp giữa hai ứng dụng di động để phát triển để lấy thông tin và lưu trữ thông tin điểm danh của người sử dụng. Các hệ thống sinh trắc vân may, mắt hay khuôn mặt là một loại hệ thông thu thập thông tin nhận dạng duy nhất. Bên cạnh đó, hệ thống RFID *(Radio Frequencey Identification-Hệ thống nhận dạng tần số vô tuyến)*

### Các hình thức điểm danh dựa trên sinh trắc học trong thực tế

#### Hệ thống điểm danh dựa trên dấu vân tay

Dấu vân tay mà bất kỳ ai cũng có trên ngón tay của mình chỉ là một trong những dấu vân vân tay trưởng thành và thông dụng nhất trong công nghệ được sử dụng để nhận dạng cá nhân. Công nghệ sinh trắc học xác minh danh tính và các đặc điểm trong phương pháp nhận dạng dấu vân tay là do dấu vân tay đặc biệt như một ID cá nhân, nó ổn định hơn đáng kể và hiệu quả. Hệ thống dựa trên 2 bước đó là Đăng ký và Xác thực, người dùng sẽ đăng ký dấu vân tay của mình như một ID của họ, hệ thống sẽ chụp lại dấu vân tay người đăng ký và lưu vào cơ sở dữ liệu. Trong quá trình xác thực, sinh trắc học của người dùng sẽ được chụp lại và các đặc điểm trích xuất sẽ được so sánh với thống tin sinh trác học hiện có trong cơ sở dữ liệu để xác minh. Sau khi xác minh thành công, điểm danh sẽ đươc ghi lại theo ID của người dùng. Hệ thống này sẽ sử dụng dấu vân tay để làm dữ liệu đầu vào để có được các đặc điểm và phát triển chương trình có nhận dạng vân tay và hệ thống nhận dang bao gồm cơ sở dữ liệu lưu trữ thông tin người dùng.

#### Hệ thống điểm danh dựa trên nhận dạng mống mắt

Công nghệ nhận dạng mống mắt hay còn gọi là công nghệ cảm biến mống mắt là phương pháp sinh trắc học giúp nhận dạng một người nào đó dựa trên mống mắt bao quanh đồng tử. Mỗi mống mắt là duy nhất và hoàn toàn không bị trùng lặp giữa người này và người khác. Do đó, nó trở thành một hình thức sinh trắc học lý tưởng, được đánh giá cao trong lĩnh vực thực thi pháp luật và ứng dụng trong các thiết bị điện tử. Các hoạt động của nhận diện mống mắt là quá trình được thực hiện bởi camera kỹ thuật số chuyên dụng. Thiét bị này chiếu bước sóng nhìn thấy được và tia cận hồng hồng ngoại vào mắt để chụp lại chi tiết và cấu trúc phức tạp của mống mắt. Iris recognition khả dụng ngay cả khi người dùng sử dụng kính áp tròng, kính cận và thậm chí là trong điều kiện môi trường ít ánh sáng.

#### Hệ thống điểm danh dựa trên nhận dạng khuôn mặt

Công nghệ nhận dạng khuôn mặt là công nghệ xác định danh tính của một người qua khuôn mặt của họ. Công nghệ này hoạt động bằng cách xác định và đo lường các đặc điểm khuôn mặt trong hình ảnh. Công nghệ nhận dạng khuôn mặt có thể xác đinh khuôn mặt người trong hình ảnh và video, xác định xem khuôn mặt xuất hiện trong hai hình ảnh có cùng là một người hay không hoặc tìm kiếm khuôn mặt trong một bộ sưu tập trong hình ảnh hiện có. Vd: nhận dạng khuôn mặt có thể xác minh khuôn mặt trong hình ảnh tự chụp bằng camera của dinh động với khuôn mặt trong hình ảnh giấy tờ tùy thân do chính phủ cấp như giấy phép lái xe hoặc hộ chiếu, cũng như xác minh khuôn mặt có trong ảnh tự chụp không khớp với các khuôn mặt trong ảnh tự chụp trước đó.

Các thuật toán xác minh khuôn mặt khớp với hình ảnh tham chiếu rõ ràng, chẳng hạn như giấy phép lái xe hoặc ảnh chụp chân dung, đạt tới điểm chính xác cao. Tuy nhiên, mức độ chính xác này chỉ có thể xảy ra trong những điều kiện sau:

* Điều kiện vị trí và ánh sáng nhất quán
* Đặc điểm khuôn mặt rõ ràng và không bị che khuất
* Màu sắc và khung cảnh phía sau được kiểm soát
* Chất lượng camera và độ phân giải hình ảnh

#### Hệ thống điểm danh dựa trên công nghệ RFID

Công nghệ RFID là viết tắt của *Radio Frequency Identification* là công nghệ nhận dạng đối tượng bằng sóng vô tuyến. Khi đó cả hai thiết bị hoạt động thu phat sóng trong cùng tần số và tần số đó thường được sử dụng trong RFID là 125khz và 900mhz. Một thiết bị RFID được cấu tạo bởi 2 thành phần chính là thiết bị đọc và thiết bị phát mã có gắn chip. Trong đó thiết bị đọc được gắn antenna có chứa một mã số nhất định sao cho không trùng lặp với nhau.

Mỗi học sinh sinh viên sẽ được cấp một thẻ RFID có chứa mã số, thông tin HSSV đó. Các thẻ này có thể được thiết kế dưới dạng thẻ trường học, thẻ HSSV, gắn lên cặp, quần áo hoặc vòng đeo cho HSSV. Các đầu đọc RFID và anten được gắp ở cổng trường và những nơi quan trọng, đảm bảo phủ sóng toàn bộ khuôn viên trường cần quản lý HSSV.

Thông qua hệ thống phần mềm quản lý sẽ giúp Quản trị viên và những bộ phân liên quan biết được việc có mặt hay vắng mặt của HSSV, biết được các HSSV đi trễ hay vị trí của HSSV từ đó tăng hiệu quả hoạt động của nhà trường.

### Lợi ích và hạn chế của các hệ thống điểm danh thường gặp

|  |  |  |  |
| --- | --- | --- | --- |
| Hệ thống | Thuận lợi | Hạn chế | Ghi chú |
| Nhận dạng vân tay | - Độ tin cậy cao  - Tính bảo mật cao | - Chi phí lắp đặt lớn  - Cần người có chuyên môn tốt để lắp đặt và triển khai hệ thống | Đề xuất sử dụng hệ thống này trong trường hợp cần mức độ an ninh cao vì sinh trắc học là duy nhất và không thể làm giả. Tuy nhiên nó bao gồm độ chắc chắn không cao là do các vấn đề về kĩ thuật và hạn ché. Đây cũng bao gồm chi phí cho thiết bị phần cứng. |
| Nhận dạng mống mắt |
| Nhận dạng khuôn mặt |
| Nhận dạng bằng RFID | Nhanh chóng và thuận tiện trong việc xác minh thông tin. Độ chính xác tương đối ổn. | Chí phí lắp đặt cao, chi phí bảo trì. Ít an toàn và ít chính xác hơn so với công nghệ nhận diện khác, | Hệ thống này có lợi thế trong việc điểm danh cần tốc độ truy xuất nhanh nhưng nó lại kém an toàn hơn so với các hệ thống điểm danh khác. |
| Nhận dạng bằng QR Code | Dễ sử dụng, dể triển khai, chi phí rẻ. Độ chính xác tương đối. | Vì lý do dễ triển khai nên hệ thống này cần kết hợp với một vài hệ thống xác định, nhận diện khác như GPS hay vân tay. Bởi vì chỉ cần người dùng sử dụng ảnh là có thể quét được QR Code mà không cần phải đến tận nơi yêu cầu, | Hệ thống này có lợi trong việc điểm danh cần sự linh hoạt, không yêu cầu một ví trí cố định. Có thể triển khai trên thiết bị di động ( một phần không thể thiếu của chúng ta hôm nay). Phù hợp với việc triển khai ở nhiều nơi, vị trí khác nhau |

* **Tổng kết:** Trong bài toán quản lý điểm danh sinh viên, để đáp ứng được mong muốn một hệ thống điểm danh có hệ thống bảo mật tương đối tốt, linh hoạt, dễ sử dụng, dễ triển khai và chi phí rẻ thì giải pháp Hệ thống điểm danh sinh viên có sử dụng QRCode + GPS trên nền tảng ứng dụng đi động là một giái pháp đúng đắn.

## Công cụ phát triển

### Nền tảng phát triển hệ thống

Hiện nay có rất nhiều công cụ giúp cho việc phát triển một hệ thống ứng dụng di động, website, server một cách dễ dàng nhưng trong đề tài này em đã sử dụng các công nghệ dưới đây để phát triển sản phẩm của mình:

* ***Ứng dụng di động***: iOS Development Kit ( bộ công cụ phát triển ứng dụng di động của Apple)
* ***Website***: ReacJS
* ***Server***: NodeJS + ExpressJS

Trên đây là những công nghệ rất phổ biến trong thời gian gần đây, tuy nhiên để phù hợp với hiệu quả công việc cũng như thời gian làm đồ án, em cũng đã lựa chọn những nền tảng phát triển hệ thống hoàn toàn miễn phí và có tính linh động cao, không cần thời gian cũng như phần cứng hỗ trợ quá nhiều).

Ngoài ra để vẽ các biểu đồ em dùng phần mềm là StarUML. **StarUML** là một mô hình nền tảng, **là phần mềm hỗ trợ UML** (Unified Modeling Language). Nó hỗ trợ các phương pháp tiếp cận MDA (Model Driven Architecture) bằng cách hỗ trợ các khái niệm hồ sơ **UML**. Tức là StarUML hỗ trợ phân tích và thiết kế hệ thống một điều mà bất cứ dự án nào đều cần có. Ngoài ra dùng **StarUML**sẽ đảm bảo tối đa hóa năng suất và chất lượng của các dự án phần mềm. Vì nó cho phép mô hình hóa nên sẽ không phụ thuộc vào người code, ngôn ngữ code hay nền tảng sử dụng.

#### **Mobile**

Trong khoảng thời gian gần đây, lập trình di động là một ngành hot. Các mẫu tin tuyển dụng gần đây thường tuyển Android Developer, iOS Developer, .. với mức lương rất hấp dẫn.

Trên thị trường ứgn dụng hiện nay, 3 hệ điều hành chiếm thị phần cao nhất là: Android, iOS, Windown phone

Chart, line chart

Description automatically generated

Hình . Biểu đồ thị phần di động của các Hệ điều hành

Theo biểu đồ, ta dễ thấy Android luôn chiếm hơn 70% thị trường của mảng di động. **Ứng dụng Android** được viết bằng ngôn ngữ **Java.** Lập trình Android cũng đang là mục tiêu được các công ty săn đón. Android có nhiều thiết bị với nhiều kích cỡ màn hình, cùng với vô số phiên bản. Android là một hệ điều hành dựa trên nền tảng Linux, được thiết kế dành cho các thiết bị di động có màn hình cảm ứng như điện thoại thông minh hay máy tính bảng. Android có mã nguồn mở và Google phát hành mã nguồn mở theo Giấy phép Apache. Chính mã nguồn mở và những giấy phép không có quá nhiểu ràng buộc đã tạo điều kiện thuận lợi để Android phát triển.

Bên cạnh đó, một đối thủ cạnh tranh của Android, iOS cũng là một hệ điều hành cực kì mạnh mẽ mạnh dù thị phần ít hơn nhiều so với Android vì đặc tính bảo mật cao của hệ điều hành iOS. Lập trình IOS là công việc xây dựng và phát triển bằng ngôn ngữ lập trình **Swift** trên ứng dụng của thiết bị iPhone, iPad, .. tương thích với hệ điều hành iOS của thương hiệu Apple.

Được ra đời vào năm 2007 với tên gọi iPhoneOS và đổi tên lần đầu tiên vào năm 2010. Như một làn gió mới thổi vào giới công nghệ lúc bấy giờ, iOS đi theo hướng phát triển tối giản, giao diện thân thiện với người dùng và được phát triển trên tất cả các sản phẩm của nhà Táo khuyết

*Sự khác nhau giữa lập trình iOS và lập trình Android:*

|  |  |  |
| --- | --- | --- |
| Hệ điều hành | Android | iOS |
| Ngôn ngữ lập trình | Java / Kotlin | Swift / Objective-C |
| Mã nguồn | Mã nguồn mở, public với các nhà phát triển phần mềm | Mã nguồn đóng, bảo mật |
| Delegate và Adapter | Sử dụng từ khóa Adapter trong Delegation Pattern [\*] | Sử dụng từ khóa Delegate trong Delegation Pattern |
| Đồ họa và giao diện | Sử dụng file XML để xây dựng lên giao diện người dùng. Android chú trọng vào sự linh động cũng như tính tương thích với nhiều thiết bị khác | Sử dụng file XIB và StoryBoard, giao diện với animation mượt, dẹp mắt và chú trọng vào trải nghiệm người dùng |
| Platform | Android được lấy cảm hứng từ Windows Metro, các giao diện của Android được thay đổi nhiều lần cho tới hiện nay và dựa nhiều hơn vào tính sáng tạo của Lập trình viên | iOS là phiên bản rút gọn của MacOS. Apple chú trọng tới giao diện người dùng và tạo ra iOS với platform hoàn mỹ. Việc phát triển nhanh chóng cùng những lần update liên tục giúp cho iOS luôn có các version mới và đẹp mắt |

[\*] *Delegation pattern: là mộ Design pattern hoạt động dựa trên tính kế kế thừa trong lập trình hướng đối tượng. Bên cạnh đó Delegation pattern có nhiều tính mở rộng hơn là: kế thừa copy hoàn toàn một class còn delegate chỉ coy một phần tính năng của class, delegate thường dùng để coply tính năng của class*

#### **Front-End**

Website hay còn gọi trang web (có thể nhầm lẫn với “web page”) hoặc trang mạng, là một tập hợp các trang web, thường chỉ nằm trong một tên miền hoặc tên miền phụ trên World Wide Web của Internet. Website có 2 dạng cơ bản là website tĩnh và website động:

* Website tĩnh: Quản trị viên không thể tùy ý thay đôi nội dung và hình ảnh. Dạng web này cần lập trình viên có kiến thức về HTML cơ bản. Website tĩnh viết trên nền tảng HTML, CSS và Javascript
* Website động: Là website có bộ công cụ quản trị để tùy biến nội dung hình ảnh cho người quản trị. Người quản trị có thể tùy ý thay đổi nội dung và hình ảnh. Ngôn ngữ xây dựng website động thường là PHP, Javascript, ASP .NET, Java, ..

Đối với một web developer, chỉ dừng lại ở HTML, CSS và Javascript là chưa đủ bởi vì những yêu cầu hiện nay đòi hỏi phải biết sử dụng thêm những framework hay thư viện để tăng tốc quá trình phát triển cũng như maintain website hơn nữa.

**ReactJS** là một opensource được phát triển bởi Facebook, ra mắt vào năm 2013, bản thân nó là một thư viện Javascript được dùng để xây dựng các tương tác với các thành phần trên website. Một trong những đặc điểm nổi bật nhất của ReactJS đó là việc render dữ liệu trong chỉ được thực hiện ở tầng Server nữa mà còn ở dưới tầng Client

Trước khi có ReactJS, lập trình viên thường gặp rất nhiều khó khăn trong việc sử dụng Javascipt thuần và Jquery để xây dựng UI. Điều đó đồng nghĩa với quá trình phát triển sẽ lâu hơn và gặp nhiều rủi ro hơn. Vì vậy vào năm 2011, Jordan Walke - một nhân viên của facebook đã khởi tạo ReactJS với mục đích cải thiện quá trình phát triển UI.

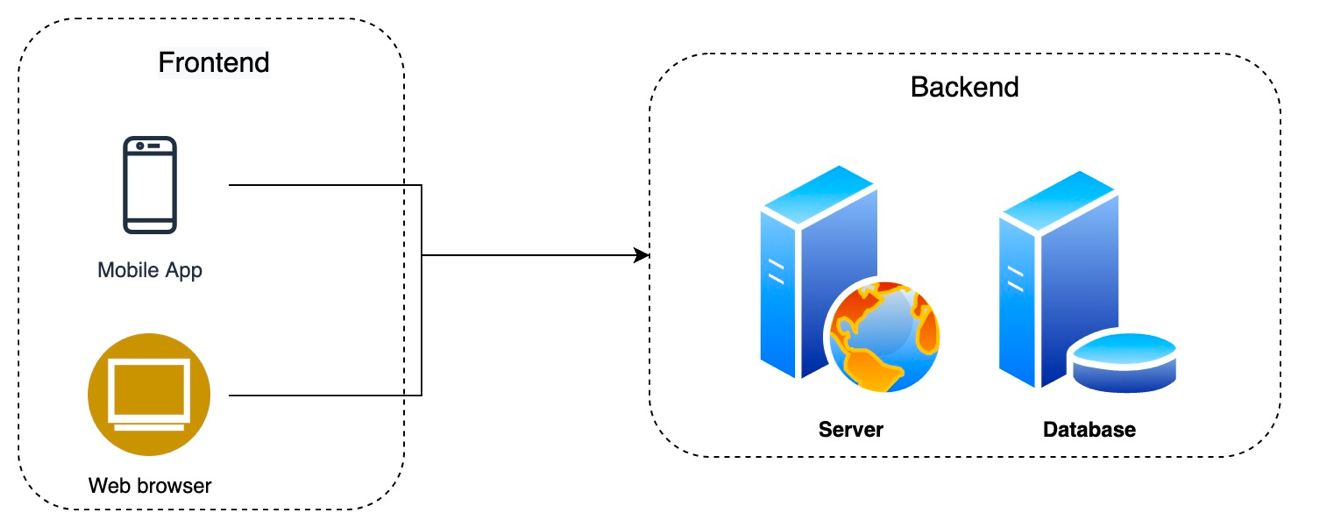
ReactJS là một thư viện Javascript chuyên giúp các nhà phát triển xây dựng giao diện người dùng hay UI. Trong lập trình ứng dụng front-end, lập trình viên thường sẽ phải việc chính trên 2 thành phần: UI và xử lý tương tác của người dùng. UI là tập hợp những thành phần mà bạn nhìn thấy được trên bất kỳ một ứng dụng web nào, ví dụ có thể kể đến bao gồm: menu, thanh tìm kiếm, thanh nút nhấn… Giả sử bạn đang lập trình một website thương mại điện tử, sau khi người dùng chọn được sản phẩm ưng ý rồi nhần vào nút “Thêm vào giỏ hàng”, thì việc tiếp theo mà bạn phải làm đó là thêm sản phẩm được chọn vào giỏ hàng và hiển thị lại sản phẩm đó sau khi người dùng đã xem => xử lý tương tác.

#### **Back-End**

Lập trình BackEnd là lập trình phía server để xử lý, lưu trữ và phục vụ dữ liệu cho hía Front-end. Thông thường phía backend sẽ bao gồm database ( cơ sở dữ liệu) và các chương trình ( application hoặc service) chạy trên cùng một hoặc nhiều servers cùng kết nối, tương tác với nhau. Các application trên backend chính là “xương sống” cho tất cả các ứng dụng front-end ngày nay.

Từ đó các công việc của một lập trình viên backend là viết ra các application, serice giao tiếp với các hệ thống database. Các application trên backend thường là webserver, API haowjc một dịch vụ lưu trữ hình ảnh hay xử lý dữ liệu nào đó.

BackEnd là các application service chạy trên các server có hình thấp hơn đáng kể ( RAM, CPU, DISK ). Tuy nhiên BackEnd developer không cần phải lập trình giao diện cho người dùng cuối nhưng bù lại thì service của họ lại phải chịu lượng truy xuất khổng lồ từ phía front-end. Việc phải xử lý lượng lớn truy cập như vậy khiến các Backend developer luôn phải tư duy tối ưu hiệu năng thực thi và sử dụng ít tài nguyên máy chủ nhất có thể.

****

**NodeJS** là một môi trường runtime chạy Javascript đa nền tảng và có mã nguồn mở, được sử dụng để chạy các ứng dụng web bên ngoài trình duyệt của client. Nền tảng này được phả triển bởi Ryan Dahl vào năm 2009, được xem là một giải pháp hoàn hảo cho các ứng dụng sử dụng nhiều dữ liệu nhờ vào mô hình hướng sự kiện ( event-driven) không đồng bộ.

**NodeJS** có một số ưu và nhược điểm cơ bản như sau:

* NodeJS có IO hướng sự kiện đồng bộ, cho phép xử lý nhiều yêu cầu đồng thời
* Sử dụng Javascript, một nguô ngữ lập trình khá dễ tiếp cận với nhiều lập trình viên, chia sẻ cùng code ở cả phía client và server

Tuy nhiên NodeJS lại ít có khả năng mở rộng, vì vậy không thể tận dụng được lợi thế mô hình đa lõi trong các phần cứng cung cấp server hiện nay, cần có kiến thức tốt về Javascript, không phù hợp với các tác dùng đòi hỏi nhiều CPU.

### Quản lý mã nguồn

Git là một hệ thống quản lý phân tán cho phép người quản lý mã nguồn một cách dễ dàng, trực tuyến và nhiều người có thể sử dụng chung tài nguyên.

* Là công cụ quản lý source code được tổ chức theo dạng dữ liệu phân tán ( distributed)
* Giúp đồng bộ source code của nhóm làm việc chung với nhau lên 1 server (remote)
* Hỗ trợ các thao tác kiểm tra source code của developer trong quá trình làm việc

*Cơ chế làm việc của Git khi sử dụng quản lý dự án*

Mô hình hoạt động của Git cần nắm một số khái niệm:

* Repository: là nơi chứa source code
* Remote: server online
* Local: server local trên máy
* Working copy: nơi làm việc trực tiếp trên máy local

Chúng ta tổ quản lý source code trên máy sẽ được lưu trữ trong local repository. Khi có internet chúng ta có thể “**đồng bộ**” local repo lên online, tức là sync lên remote repo. Khi trong team hoàn thành các tính năng và đã sync lên cho server online ( remote repo) thông qua các thao tác ( commit & push).

Các hệ thống quản lý phiên bản khác như: Subversion, Perforce, Bazaar,... coi thông tin được lưu trữ như là một tập hợp các tập tin và các thay đổi được thực hiện trên mỗi tập tin theo thời gian.

Git không nghĩ hoặc xử lý dữ liệu theo cách này. Mà thay vào đó Git coi dữ liệu của nó giống như một tập hợp các "ảnh" (snapshot) của một hệ thống tập tin nhỏ. Mỗi lần bạn "commit", hoặc lưu lại trạng thái hiện tại của dự án trong Git, về cơ bản Git "chụp một bức ảnh" ghi lại nội dung của tất cả các tập tin tại thời điểm đó và tạo ra một tham chiếu tới "ảnh" đó. Để hiệu quả hơn, nếu như tập tin không có sự thay đổi nào, Git không lưu trữ tập tin đó lại một lần nữa mà chỉ tạo một liên kết tới tập tin gốc đã tồn tại trước đó.

Phần lớn các thao tác/hoạt động trong Git chỉ cần yêu cầu các tập tin hay tài nguyên cục bộ - thông thường nó sẽ không cần bất cứ thông tin từ máy tính nào khác trong mạng lưới của bạn. Nếu như bạn quen với việc sử dụng các hệ thống quản lý phiên bản tập trung nơi mà đa số hoạt động đều chịu sự ảnh hưởng bởi độ trễ của mạng, thì với Git đó lại là một thế mạnh. Bởi vì toàn bộ dự án hoàn toàn nằm trên ổ cứng của bạn, các thao tác được thực hiện gần như ngay lập tức.

Ví dụ, khi bạn muốn xem lịch sử của dự án, Git không cần phải lấy thông tin đó từ một máy chủ khác để hiển thị, mà đơn giản nó được đọc trực tiếp từ chính cơ sở dữ liệu cục bộ của bạn. Điều này có nghĩa là bạn có thể xem được lịch sử thay đổi của dự án gần như ngay lập tức. Nếu như bạn muốn so sánh sự thay đổi giữa phiên bản hiện tại của một tập tin với phiên bản của một tháng trước, Git có thể tìm kiếm tập tin cũ đó trên máy cục bộ rồi sau đó so sánh sự khác biệt cho bạn. Thay vì việc phải truy vấn từ xa hoặc "kéo về" (pull) phiên bản cũ của tập tin đó từ máy chủ trung tâm rồi mới thực hiện so sánh cục bộ.

Điều này còn đồng nghĩa với có rất ít việc mà bạn không thể làm được khi không có kết nối Internet hoặc VPN bị ngắt. Nếu bạn muốn làm việc ngay cả khi ở trên máy bay hoặc trên tầu, bạn vẫn có thể commit bình thường cho tới khi có kết nối Internet để đồng bộ hoá. Nếu bạn đang ở nhà mà VPN lại không thể kết nối được, bạn cũng vẫn có thể làm việc bình thường. Trong rất nhiều hệ thống khác, việc này gần như là không thể hoặc rất khó khăn. Ví dụ trong Perforce, bạn gần như không thể làm gì nếu như không kết nối được tới máy chủ; trong Subversion và CVS, bạn có thể sửa tập tin nhưng bạn không thể commit các thay đổi đó vào cơ sở dữ liệu (vì cơ sở dữ liệu của bạn không được kết nối). Đây có thể không phải là điều gì đó lớn lao, nhưng bạn sẽ ngạc nhiên về sự thay đổi lớn mà nó có thể làm được.

Mỗi tập tin trong Git được quản lý dựa trên ba trạng thái: committed, modified, và staged.

* Committed: dữ liệu đã được lưu trữ một cách an toàn trong cơ sở dữ liệu.
* Modified: đã thay đổi tập tin nhưng chưa commit vào cơ sở dữ liệu.
* Staged: đã đánh dấu sẽ commit phiên bản hiện tại của một tập tin đã chỉnh sửa trong lần commit sắp tới.

Điều này tạo ra ba phần riêng biệt của một dự án sử dụng Git: thư mục Git, thư mục làm việc, và khu vực tổ chức (staging area).

### Cơ sở dữ liệu

Cơ sở dữ liệu (CSDL) là tập hợp dữ liệu được tổ chức có cấu trúc liên quan với nhau và được lưu trữ trong máy tính.

CSDL được thiết kế, xây dựng cho phép người dùng lưu trữ dữ liệu, truy xuất thông tin hoặc cập nhật dữ liệu.

CSDL được tổ chức có cấu trúc: Các dữ liệu được lưu trữ có cấu trúc thành các bản ghi (record), các trường dữ liệu (field). Các dữ liệu lưu trữ có mối quan hệ (relation) với nhau

CSDL được cấu trúc để dễ dàng truy cập, quản lý và cập nhật.

Các loại mô hình dữ liệu thông dụng hiện nay:

* Mô hình dữ liệu file
* Mô hình dữ liệu phân cấp
* Mô hình dữ liệu mạng
* Mô hình dữ liệu quan hệ
* Mô hình dữ liệu hướng đối tượng

**Giới thiệu hệ quản trị cơ sở dữ liệu MySQL**

MySQL là một phần mềm quản trị CSDL mã nguồn mở, miễn phí nằm trong nhóm LAMP (Linux – Apache – MySQL – PHP). MySQL là phần mềm ổn định, an toàn. Hiện nay trong số các Website có lưu lượng truy cập lớn trên lnternet, có rất nhiều Website sử dụng LAMP và LAMP đang được coi là một đối trọng với các sản phẩm mã đóng của Microsoft (Website, IIS, SQL, Server, ASP/ASP. NET.).

**Một số đặc điểm CSDL**

MySQL là một phần mềm quản trị CSDL dạng server - based (gần tương đương với SQL Server của Microsoft), MySQL quản lý dữ liệu thông qua các CSDL, mỗi điểm CSDL có thể có nhiều bảng quan hệ chứa dữ liệu.

MySQL có cơ chế phân quyền người sử dụng riêng, mỗi người có thể được quản lý một hoặc nhiều CSDL khác nhau, mỗi người dung có một tên truy cập và mật khẩu tương ứng dụng để truy xuất đến CSDL. Khi ta truy vấn tới CSDL MySQL, ta phải cung cấp tên và mật khẩu của tài khoản có quyền sử dụng CSDL đó.

### Môi trường triển khai hệ thống

#### **Appstore**

App Store là một kho ứng dụng dành riêng cho các thiết bị sử dụng hệ điều hành iOS (iPhone, iPad, iPod, v.v...). App Store cung cấp hàng triệu ứng dụng miễn phí lẫn có phí. Các ứng dụng này rất đa dạng và phong phú, đáp ứng tất cả các nhu cầu của chúng ta từ giải trí cho tới công việc: các ứng dụng game, đọc sách, xem phim, nghe nhạc, bản đồ, học nấu ăn, đọc báo, v.v...[App Store là gì? Hướng dẫn chi tiết cách tải ứng dụng từ App Store cho iPhone](https://thanhtrungmobile.vn/huong-dan-chi-tiet-cach-tai-ung-dung-tu-appstore-cho-iphone-p25201.html)[](https://thanhtrungmobile.vn/)

Ứng dụng trên App Store đều đã qua kiểm duyệt nghiêm ngặt trước khi được đăng lên, giúp đảm bảo người dùng sẽ có được những ứng dụng hữu ích và đặc biệt an toàn với thiết bị.

**Cách triển khai ứng dụng lên Appstore**

Nhà phát triển cần đăng ký chương trình nhà phát triển của Apple, (Apple Developer Program) ứng dụng của bạn tuân theo Nguyên tắc đánh giá ứng dụng và giao diện con người của Apple (Apple’s App Review and Human Interface Guidelines) và ứng dụng của bạn đã sẵn sàng để đăng lên sàn của Apple.

Hồ sơ cung cấp phát triển (development provisioning profile) và chứng chỉ phát triển (development certificate) mà bạn đang sử dụng chỉ dành cho các thiết bị cụ thể. Để phân phối ứng dụng của bạn cho người thử nghiệm beta hoặc cho người dùng thông qua App Store, bạn sẽ cần hồ sơ cấp phép phân phối và chứng chỉ phân phối riêng.

Cách dễ nhất để làm điều này là thông qua Xcode. Nếu tính năng signing ( ký tên) tự động được bật, Xcode sẽ tạo và quản lý chứng chỉ, xác nhận danh tính và xử lý đăng ký thiết bị cho bạn. Nếu tính năng signing tự động đã được bật hoặc nếu bạn không cần trợ giúp về việc này, bạn có thể chuyển sang bước hai .

(Trong một số trường hợp, bạn có thể thích Signing (ký tên) thủ công hơn. Đây là hướng dẫn chuyên sâu về cách ký tên ứng dụng của bạn theo cách thủ công. Hãy nhớ rằng tất cả các mục tiêu trong một gói phải sử dụng cùng một phương pháp signing.)

1. Trước tiên, hãy thêm tài khoản Chương trình nhà phát triển của bạn vào Xcode nếu bạn chưa có. Từ menu trên cùng, chọn Xcode, sau đó chọn Tùy chọn.
2. Nhấp vào Tài khoản . Ở góc dưới cùng bên trái của cửa sổ, nhấn vào dấu + , sau đó Thêm Apple ID… .
3. Nhập ID Apple và mật khẩu bạn sử dụng cho Chương trình nhà phát triển Apple, sau đó nhấp vào Đăng nhập (Sign In).
4. Tiếp theo, kích hoạt tính năng ký tên (signing) tự động. Từ Trình chỉnh sửa dự án (Project Editor), chọn một mục tiêu và chọn Chung (General).
5. Cuộn xuống phần Signing (“Ký tên”) và nhấp vào biểu tượng hình tam giác để mở rộng cài đặt.
6. Nhấp vào hộp để Tự động quản lý việc ký tên (Automatically manage signing). Chọn nhóm của bạn.

Khi bạn kết nối thiết bị mới với máy Mac, Xcode sẽ tự động phát hiện và đăng ký thiết bị đó vào hồ sơ cấp phép nhóm của bạn. Lưu ý rằng để khởi chạy ứng dụng của bạn trên một thiết bị, thiết bị đó cần được đăng ký trên hồ sơ cấp phép nhóm của bạn.

#### **Heroku**

**Heroku** là gì? Nó là nền tảng đám mây cho phép các lập trình viên xây dựng, triển khai, quản lý và mở rộng ứng dụng (PaaS – Platform as a service).

Nó rất linh hoạt và dễ sử dụng, cung cấp cho một con đường đơn giản nhất để đưa sản phẩm tiếp cận người dùng. Nó giúp các nhà phát triển tập trung vào phát triển sản phẩm mà không cần quan tâm đến việc vận hành máy chủ hay phần cứng…

Sau khái niệm Heroku là gì thì hãy cùng nhau tìm hiểu các hoạt động của nó. Heroku chạy các ứng dụng trong dynos – nó là một máy ảo mà có thể tăng giảm sức mạnh dự vào độ lớn của ứng dụng.

Ngoài ra chúng ta còn được cung cấp Database, SSL miễn phí, hỗ trợ mạnh làm việc team cũng như liên kết với Github một cách đơn giản.

Khuyết điểm thì tất nhiên là có, việc miễn phí sẽ chỉ có giới hạn là 550 giờ mỗi tháng. Nếu muốn tăng lên 1000 giờ thì bạn cần cài đặt phương thức thanh toán trên đó. Tuy nhiên để kiểm nghiệm một ý tưởng hay một trang web nhỏ thì nhiêu đó là quá đủ để thu về kết quả.

Sau 2 đến 3 giờ nếu server không có người truy cập thì server sẽ chuyển sang trạng thái ngủ. Về việc server bị tắt khi không có traffic, cách đơn giản nhất là tự tạo traffic cho nó.

**Các tính năng của Heroku**

* Heroku runtime
* Heroku Teams
* Scale
* Add-ons
* Code/data rollback
* App metric
* CI/CD

**Cách triển khai application lên Heroku**

***Bước 1: Cài đặt Heroku***

Bạn cài đặt theo link hướng dẫn dưới đây cho hệ điều hành mình đang dùng, do mình sử dụng MacOS nên sẽ cài thông qua homebrew

brew install heroku/brew/heroku

Sau đó xác định phiên bản của nó bằng command:

heroku –version

***Bước 2: Cài đặt ứng dụng Laravel***

Mở terminal và sử dụng command sau:

composer create-project laravel/laravel herokularavel --prefer-dist

***Bước 3: Tạo một Procfile***

Bên trong thư mục gốc của Laravel (root), tạo một file gọi là Procfile – chứa câu lệnh mà heroku sẽ chạy. Sau đó bỏ dòng này vào procfile:

# Procfile

web: vendor/bin/heroku-php-apache2 public/

***Bước 4: Khởi tạo dự án với Git repo***

git init

***Bước 5: Đăng nhập vào Heroku từ terminal***

heroku login

***Bước 6: Tạo một ứng dụng Heroku***

Gõ lệnh sau tại thư mục chứa code, Heroku sẽ yêu cầu bạn nhập tài khỏan, rồi tạo một “app” cho bạn, đồng thời thiết lập git để sẵn sàng “đẩy code lên Heroku”

heroku create

Creating app... done, ⬢ ironman-spiderman-12345

https://ironman-spiderman-12345.herokuapp.com/ | https://git.heroku.com/ironman-spiderman-12345.git

Nó sẽ tự động tạo một ứng dụng với tên random, ví dụ như: ironman-spiderman-12345. Bạn sẽ có một url là ironman-spiderman-12345.herokuapp.com

***Bước 7: Setting một Laravel encryption key***

Gõ lệnh sau:

php artisan key:generate --show

Nó sẽ show một key và bạn copy vào cho vào lệnh sau:

heroku config:set APP\_KEY={Your copied key}

***Bước 8: Push sự thay đổi lên Heroku***

Sau đó add các file vào Git:

git add .

Commit các thay đổi:

git commit -m "laravel deploy to heroku"

git push heroku master

# KHẢO SÁT HIỆN TRẠNG VÀ PHÂN TÍCH , THIẾT KẾ HỆ THỐNG

## Khảo sát thực tế các phần mềm quản lý sinh viên

1. ***Phần mềm quản lý học sinh – sinh viên Mona eLMS***

Mona eLMS là phần mềm e-learning được phát triển với công ty Mona Software được ứng dụng phổ biến trong quy trình quản lý học viên, giáo viên, khoá học, tài liệu giảng dạy, quản lý đào tạo, tài chính, đào tạo trực tuyến… ở các cơ sở giáo dục hiện nay. Hiện có hơn 50 trung tâm đang sử dụng và hơn 100 cá nhân đang sử dụng để dạy học trực tuyến, phát triển khoá học.

**Phần mềm Mona eLMS** là phần mềm được tích hợp trên nền tảng web-app và các thiết bị ngoại vi khác. Đây cũng cũng là phần mềm được đánh giá cao về bảo mật thông tin tốt nhất hiện nay.

Phần mềm Mona eLMS phù hợp với những cơ sở giáo dục nào:

* Trung tâm ngoại ngữ tiếng Anh, Nhật, Hàn, Trung, Pháp….
* Trường học muốn ứng dụng các phần mềm vào quản lý.
* Các trung tâm dạy học trực tuyến, trung tâm gia sư.
* Bán khoá học dạy thêm, video.
* Các trung tâm năng khiếu và các kỹ năng khác.
* Các trường dạy lái xe.

Các tính năng nổi bật của phần mềm Mona eLMS

* Quản lý học viên như thông tin học viên, quá trình học tập, kết quả học tập.
* Quản lý khoá học, đào tạo, thời khoá biểu.
* Kiểm soát tuyển sinh, đăng ký.
* Điểm danh học viên trên lớp, học viên nào vắng mặt.
* Quản lý tương tác.
* Hỗ trợ giáo viên giảng dạy.
* Quản lý giáo viên: thông tin giáo viên, lịch học…
* Lưu trữ dữ liệu liên quan đến giáo viên, học viên, khoá học.
* Thống kê, báo cáo.
* Quản lý học phí, tính lương cho cán bộ giáo viên một cách chính xác

1. ***Phần mền quản lý giáo dục Misa – QLTH.VN***

Phần mềm Misa QLTH.VN được xây dựng và phát triển theo công nghệ điện toán đám mây vì vậy có thể sử dụng mọi lúc, mọi nơi. Có thể truy cập bằng các thiết bị máy tín, máy tính bảng, điện thoại…chỉ cần có internet thì bạn có thể sử dụng bất kỳ yêu cầu nào.

Phần mềm Misa phù hợp với:

* Nhà trường
* Phòng giáo dục
* Sở giáo dục

Tính năng của phần mềm QLTH.VN!

* Quản lý giảng dạy
* Quản lý trang thiết bị, cơ sở vật chất
* Quản lý giáo viên, học sinh, sinh viên
* Quản lý các khoản thu, chi….

Lợi ích của phần mềm QLTH.VN

* Ứng dụng trí tuệ nhân tạo trong quản lý
* Ứng dụng công nghệ quét mã vạch
* Thanh toán tiền học trực tuyến
* Tuyển sinh trực tuyến
* Quản lý thời khoá biểu
* Điểm danh học viên….

## Phân tích người dùng hệ thống

### Các tác nhân trong hệ thống

* Quản trị viên
* Sinh viên
* Giảng viên

### Yêu cầu về chức năng chính

* Quản lý tài khoản người dùng
* Quản lý điểm danh
* Quản lý môn học
* Quản lý lớp học
* Quản lý thông báo
* Xem thống kê sinh viên tham dự các môn học

### Yêu cầu nghiệp vụ bảo mật của hệ thống

* Đối với sinh viên sử dụng ứng dụng, chỉ cho phép **1 thiết bị điện thoại đăng nhập 1 tài khoản sinh viên duy nhất** và **1 tài khoản chỉ được đăng nhập trên 1 thiết bị**. Nếu tài khoản sinh viên đã từng được đăng nhập trên bất kì thiết bị nào, thiết bị và tài khoản sinh viên đó sẽ được liên kết.
* Chỉ quản trị viên mới có quyền **reset mật khẩu**, **reset thiết bị** sử dụng của tài khoản sinh viên.
* Sinh viên, giảng viên chỉ có quyền **đổi mật khẩu tài khoản,** không có quyền **tạo**, **xoá tài khoản** cá nhân của mình. Mọi hành động tạo, xoá tài khoản cần liên hệ Quản trị viên.
* Ứng dụng yêu cầu kết nối **Wifi/Mạng di động,** thiết bị điện thoạikhông root, jailbreak.
* Ứng dụng cần quyền bật bảo mật **FaceID / TouchID**.
* Khi điểm danh ứng dụng sinh viên cần bật **GPS** với chế độ **Always** (luôn luôn) để ứng dụng tracking vị trí sinh viên và so sánh vị trí của sinh viên với vị trí lớp học. Khi sử dụng ứng dụng bình thường, sinh viên không cần bật GPS để tránh hao pin thiết bị.
* Chỉ chấp nhận cho phép điểm danh khi **vị trí sinh viên** và **vị trí lớp học** có khoảng cách < **50m** ( đây là kích thước của một hội trường lớn).
* Đối với tính năng **Điểm danh nhanh** bằng **QRCode,** ứng dụng sinh viên cần yêu cầu quyền truy cập Camera thiết bị.
* Ứng dụng cần có nghiệp vụ chống **fake-location (giả mạo vị trí)**

### Yêu cầu nghiệp vụ dành cho người dùng hệ thống

*Quản trị viên*

* Sử dụng website quản trị, để đăng nhập hệ thống
* Quản lý tài khoản các đối tượng người dùng ( giảng viên, sinh viên)
* Quản lý các thông tin như: danh sách phòng học, danh sách lớp học, danh sách môn học.
* Để phục vụ công tác quản lý, người bán có thể thống kê danh sách các lớp học đang mở, số lượng sinh viên tham dự.
* Các thông tin cần nhập liệu số lượng lớn, quản trị viên có thể sử dụng cách import file Excel để tăng tốc thời gian xử lý.

*Giảng viên*

* Sử dụng ứng dụng di động
* Có thể xem thông tin các môn học mình đang quản lý, xem số lượng điểm danh của các sinh viên mỗi buổi học
* Giảng viên có quyền khoá điểm danh trong môn học đó ( do sinh viên nghỉ học quá số buổi cho phép)
* Xem phản hồi của sinh viên nếu có ( sinh viên có thể phản hồi lại giảng viên lý do điểm danh muộn)

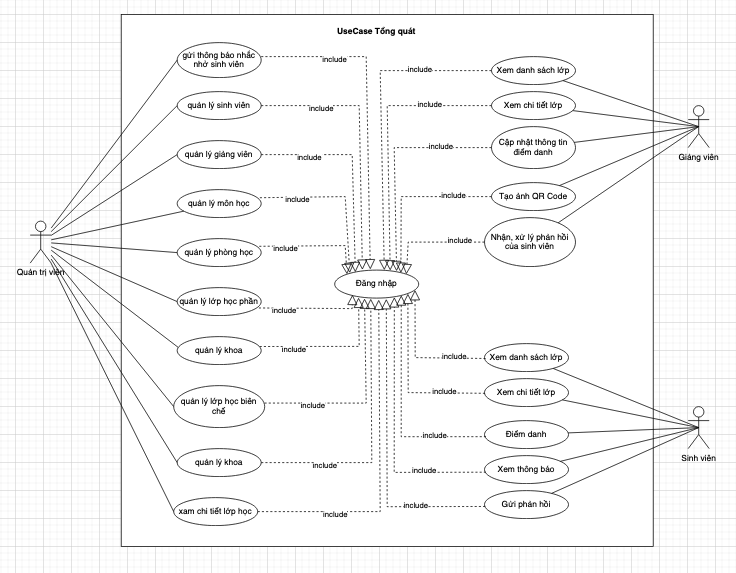
*Sinh viên*

* Sử dụng ứng dụng di động
* Điểm danh (bằng QR hoặc thủ công), nhắc lịch học, nhận cảnh báo đi học muộn nhiều lần, nghỉ học, nghỉ quá số buổi.
* Xem thông báo tương ứng của lớp học phần hoặc thông báo của lớp biên chế
* Phản hồi lại giảng viên buổi học mình muộn (để được xem xét)

## Phân tích các ca sử dụng

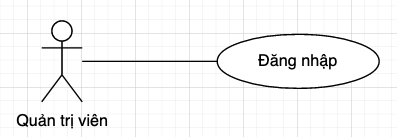
### Xây dựng biểu đồ tổng quát UseCase

Từ mô tả bài toán có thể thấy, ta thấy có 3 actor chính : quản trị viên , giảng viên, sinh viên. Tất cả các actor đều có chức năng chung : cần đăng nhập vào hệ thống để sử dụng .Quan hệ giao tiếp ( Association ) được dùng để mô tả mối quan hệ giữa Actor và Use Case với nhau. Quan hệ Include mô tả việc một Use Case lớn được chia ra thành các Use Case nhỏ để dễ cài đặt . Generalization được sử dụng để thể hiện quan hệ thừa kế giữa các Actor hoặc giữa các Use Case với nhau. Actor quản trị viên kế thừa toàn bộ quyền của actor giảng viên, sinh viên. Từ các phân tích trên , em xây dựng được biểu đồ tổng quát UseCase như sau :



Hình .Biểu đồ UseCase tổng quát

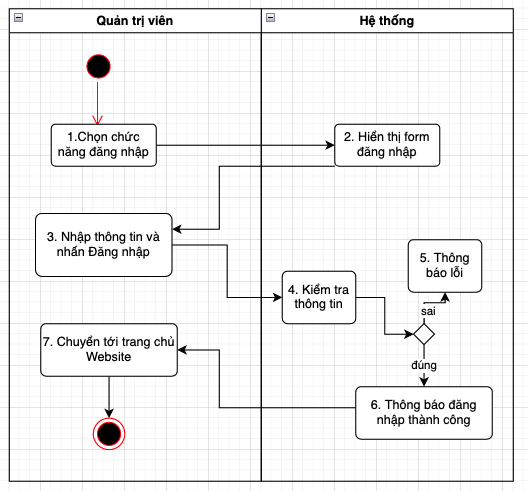
### UseCase quản trị viên đăng nhập hệ thống



Hình . Biểu đồ phân rã UseCase quản trị viên đăng nhập hệ thống

|  |  |
| --- | --- |
| **Tên UC** | **Quản trị viên đăng nhập hệ thống** |
| **Mục đích** | Cho phép người dùng quản trị viên đăng nhập vào hệ thống. |
| **Tác nhân** | Quản trị viên |
| **Tiền điều kiện** | Quản trị viên chọn chức năng đăng nhập |
| **Luồng sự kiện chính** | 1. Quản trị viên chọn chức năng đăng nhập  2. Hệ thống hiển thị giao diện đăng nhập  3. Quản trị viên nhập thông tin theo các trường và ấn đăng nhập  4. Hệ thống Kiểm tra thông tin đăng nhập, nếu chính xác thì đi tới bước 6, ngược lại thì chuyển sang bước 5  5. Hiển thị thông báo lỗi, yêu cầu quản trị viên nhập lại.  6. Hệ thống hiển thị thông báo đăng nhập thành công và chuyển đến trang chủ. |
| **Luồng sự kiện ngoại lệ** | Luồng sự kiện ngoại A1  1. Quản trị viên nhập thông tin sai, bỏ trống  2. Hệ thống kiểm tra thông tin nhập và thông báo lỗi, yêu cầu nhập lại.  Hệ thống kiểm tra thông tin , yêu cầu nhập email khác |
| **Đảm bảo thành công** | Quản tri viên truy cập vào trang chủ của hệ thống |

Bảng . Kịch bản chức năng quản trị viên đăng nhập hệ thống



Hình . Biểu đồ hoạt động chức năng quản trị viên đăng nhập hệ thống

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng quản trị viên đăng nhập hệ thống

### UseCase giảng viên đăng nhập ứng dụng

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase giảng viên đăng nhập ứng dụng

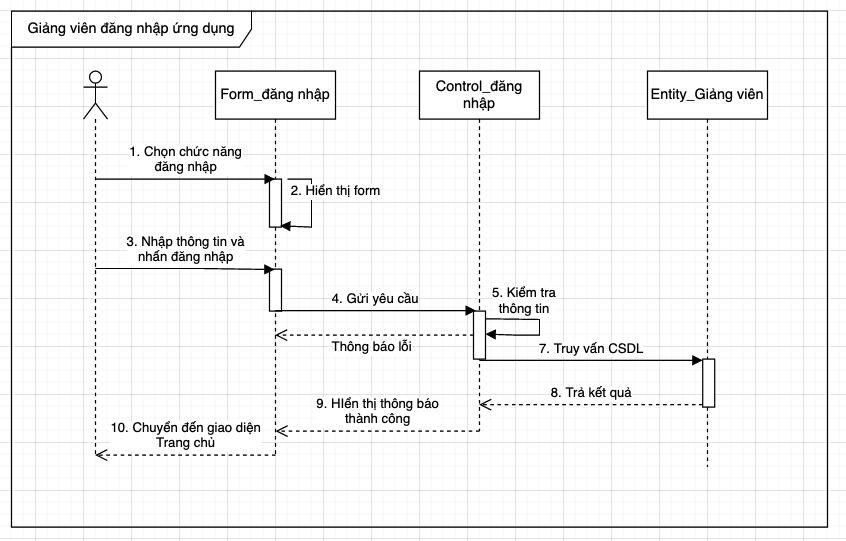
|  |  |
| --- | --- |
| **Tên UC** | **Giảng viên đăng nhập ứng dụng** |
| **Mục đích** | Cho phép người dùng là giảng viên đăng nhập vào ứng dụng |
| **Tác nhân** | Giảng viên |
| **Tiền điều kiện** | Giảng viên chọn chức năng đăng nhập |
| **Luồng sự kiện chính** | 1. Giảng viên chọn chức năng đăng nhập, chọn mục Giảng viên  2. Ứng dụng hiển thị giao diện đăng nhập  3. Giảng viên nhập thông tin theo các trường và ấn đăng nhập  4. Ứng dụng kiểm tra thông tin đăng nhập, nếu chính xác thì đi tới bước 6, ngược lại thì chuyển sang bước 5  5. Hiển thị thông báo lỗi, yêu cầu giảng viên nhập lại.  6. Hệ thống hiển thị thông báo đăng nhập thành công và chuyển đến trang chủ. |
| **Luồng sự kiện ngoại lệ** | Giảng viên đăng nhập không thành công  1. giảng viên nhập sai tên , hoặc bỏ trống trường  2. Hệ thống yêu cầu nhập đủ  3. Gỉảng viên chọn mode đăng nhập là Sinh viên |
| **Đảm bảo thành công** | Người dùng đăng nhập thành công vào ứng dụng |

Hình . Kịch bản chức năng giảng viên đăng nhập ứng dụng

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng giảng viên đăng nhập ứng dụng



Hình . Biểu đồ tuần tự chức năng giảng viên đăng nhập ứng dụng

### UseCase sinh viên đăng nhập ứng dụng

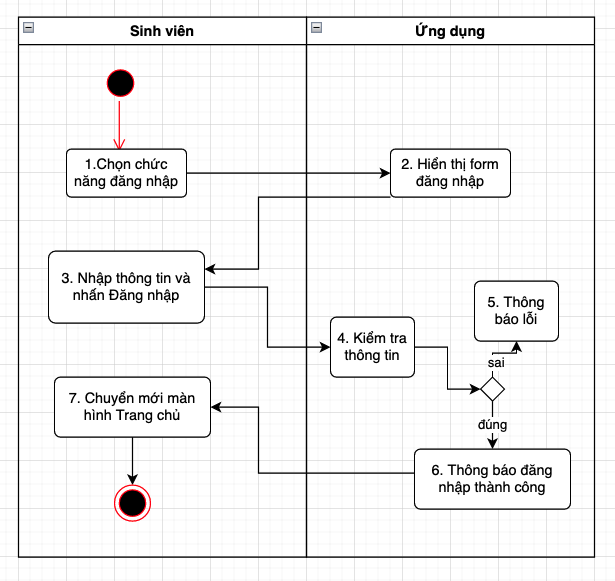
Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase sinh viên đăng nhập ứng dụng

|  |  |
| --- | --- |
| **Tên UC** | **Sinh viên đăng nhập ứng dụng** |
| **Mục đích** | Cho phép người dùng là sinh viên đăng nhập vào ứng dụng |
| **Tác nhân** | Sinh viên |
| **Tiền điều kiện** | Sinh viên chọn chức năng đăng nhập |
| **Luồng sự kiện chính** | 1. Sinh viên chọn chức năng đăng nhập, chọn mục Giảng viên  2. Ứng dụng hiển thị giao diện đăng nhập  3. Sinh viên nhập thông tin theo các trường và ấn đăng nhập  4. Ứng dụng kiểm tra thông tin đăng nhập, nếu chính xác thì đi tới bước 6, ngược lại thì chuyển sang bước 5  5. Hiển thị thông báo lỗi, yêu cầu sinh viên nhập lại.  6. Hệ thống hiển thị thông báo đăng nhập thành công và chuyển đến trang chủ. |
| **Luồng sự kiện ngoại lệ** | Giảng viên đăng nhập không thành công  1. Sinh viên nhập sai tên , hoặc bỏ trống trường  2. Hệ thống yêu cầu nhập đủ  3. Sinh viên chọn mode đăng nhập là Giảng viên  4. Sinh viên đăng nhập trên thiết bị đã từng được đăng nhập một tài khoản khác  5. Tài khoản của sinh viên đã được liên kết với một thiết bị khác nên sinh viên không thể đăng nhập được vào ứng dụng ngoài thiết bị đã được liên kết |
| **Đảm bảo thành công** | Sinh viên đăng nhập thành công vào ứng dụng |

Bảng . Kịch bản sinh viên đăng nhập ứng dụng



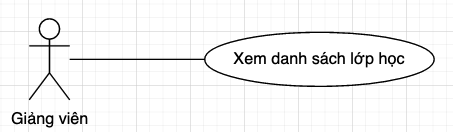
Hình . Biểu đồ hoạt động sinh viên đăng nhập ứng dụng

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sinh viên đăng nhập ứng dụng

### UseCase giảng viên xem danh sách lớp học



Hình . Biểu đồ phân rã UseCase giảng viên xem danh sách lớp học

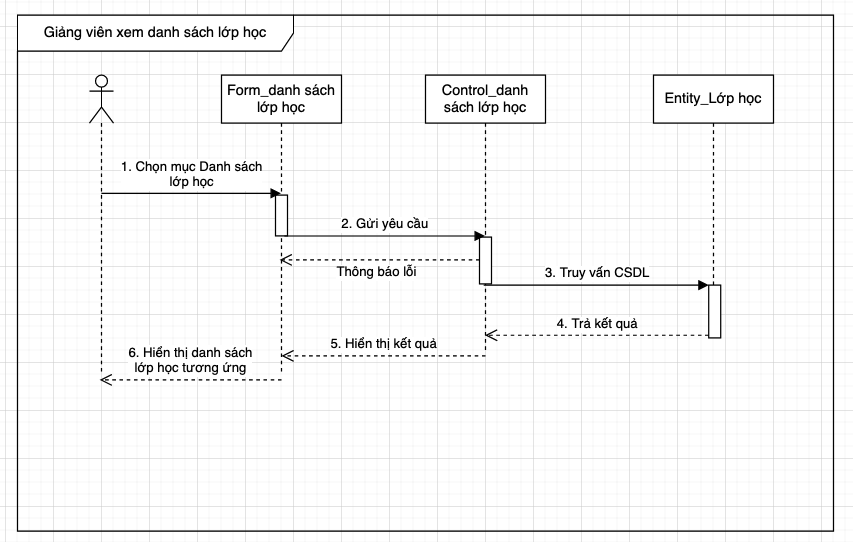
|  |  |
| --- | --- |
| **Tên UC** | **Giảng viên xem danh sách lớp học** |
| **Mục đích** | Giảng viên truy cập vào tab Trang chủ trong ứng dụng, tại đó, giảng viên có thể xem được trong 3 ngày: ngày nay, ngày mai, ngày kia có tiết dạy nào của mình hay không, thời gian, địa điểm là ở đâu, hiện tại đang có bao nhiêu sinh viên điểm danh / tổng số sinh viên. |
| **Tác nhân** | Giảng viên |
| **Tiền điều kiện** | Giảng viên đã đăng nhập hệ thống |
| **Luồng sự kiện chính** | 1. Giảng viên chọn tab Trang chủ trong ứng dụng  2. Ứng dụng hiển thị danh sách các lớp học theo 3 ngày: hôm nay, ngày mai, ngày kia  3. Giảng viên xem danh sách lớp, xem được thông tin phòng học và thời gian học |
| **Luồng sự kiện ngoại lệ** | 1. Back-End trả về mã lỗi, timeout dẫn tới ứng dụng không hiểu thị đúng với thông tin trong thực tế  2. Hiển thị thông báo lỗi tới người: “Mất kết nối đến hệ thống. Vui lòng thử lại sau”. |
| **Đảm bảo thành công** | Danh sách lớp học hiển thị đầy đủ thông tin |

Bảng . Kịch bảng giảng viên xem danh sách lớp học

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động giảng viên xem danh sách lớp học



Hình . Biểu đồ tuần tự chức năng giảng viên xem danh sách lớp học

### UseCase sinh viên xem danh sách lớp học

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase sinh viên xem danh sách lớp học

|  |  |
| --- | --- |
| **Tên UC** | **Sinh viên xem danh sách lớp học** |
| **Mục đích** | Sinh viên truy cập vào tab Trang chủ trong ứng dụng, tại đó, sinh viên có thể xem được trong 3 ngày: nay, mai, ngày kia có môn nào mình học hay không, thời gian, địa điểm là ở đâu. |
| **Tác nhân** | Sinh viên |
| **Tiền điều kiện** | Sinh viên đã đăng nhập vào hệ thống |
| **Luồng sự kiện chính** | 1. Sinh viên chọn tab Trang chủ trong ứng dụng  2. Ứng dụng hiển thị danh sách các lớp học theo 3 ngày: hôm nay, ngày mai, ngày kia  3. Sinh viên xem danh sách lớp, xem được thông tin phòng học và thời gian học |
| **Luồng sự kiện ngoại lệ** | 1. Back-End trả về mã lỗi, timeout dẫn tới ứng dụng không hiểu thị đúng với thông tin trong thực tế  2. Hiển thị thông báo lỗi tới người: “Mất kết nối đến hệ thống. Vui lòng thử lại sau”. |
| **Đảm bảo thành công** | Danh sách lớp học hiển thị đầy đủ thông tin |

Bảng . Kịch bản chức năng sinh viên xem danh sách lớp học

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng sinh viên xem danh sách lớp học

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sinh viên xem danh sách lớp học

### UseCase giảng viên xem chi tiết lớp học

Diagram

Description automatically generated

Hình . Biểu đồ phân rã giảng viên xem chi tiết lớp học

|  |  |
| --- | --- |
| **Tên UC** | **Giảng viên xem chi tiết lớp học** |
| **Mục đích** | Giảng viên cần các thông tin của lớp học như: tên môn học, phòng học, giờ học, danh sách sinh viên trong lớp. Xem được thông tin hiện tại đang có bao nhiêu sinh viên điểm danh, thời gian điểm danh của từng sinh viên. |
| **Tác nhân** | Giảng viên |
| **Tiền điều kiện** | Giảng viên đã đăng nhập vào hệ thống |
| **Luồng sự kiện chính** | 1. Giảng viên truy cập vào tab Trang chủ  2. Giảng viên chọn lớp học để xem thông tin |
| **Luồng sự kiện ngoại lệ** | 1. Back-End trả về mã lỗi, timeout dẫn tới ứng dụng không hiểu thị đúng với thông tin trong thực tế  2. Hiển thị thông báo lỗi tới người: “Mất kết nối đến hệ thống. Vui lòng thử lại sau”. |
| **Đảm bảo thành công** | Chi tiết lớp học hiển thị đầy đủ thông tin |

Bảng . Kịch bản chức năng giảng viên xem chi tiết lớp học

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng giảng viên xem chi tiết lớp học

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng giảng viên xem chi tiết lớp học

### UseCase sinh viên xem chi tiết lớp học

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase sinh viên xem chi tiết lớp học

|  |  |
| --- | --- |
| **Tên UC** | **Sinh viên xem chi tiết lớp học** |
| **Mục đích** | Sinh viên cần các thông tin của lớp học như: tên môn học, phòng học, giờ học, danh sách sinh viên trong lớp. Có nút điểm danh khi có thể điểm danh, có nút phản hồi giảng viên khi có thông tin cần trao đổi.. |
| **Tác nhân** | Sinh viên |
| **Tiền điều kiện** | Sinh viên đã đăng nhập vào hệ thống |
| **Luồng sự kiện chính** | 1. Sinh viên truy cập vào tab Trang chủ  2. Sinh viên chọn lớp học để xem thông tin |
| **Luồng sự kiện ngoại lệ** | 1. Back-End trả về mã lỗi, timeout dẫn tới ứng dụng không hiểu thị đúng với thông tin trong thực tế  2. Hiển thị thông báo lỗi tới người: “Mất kết nối đến hệ thống. Vui lòng thử lại sau”. |
| **Đảm bảo thành công** | Chi tiết lớp học hiển thị đầy đủ thông tin |

Bảng . Kịch bản chức năng sinh viên xem chi tiết lớp học

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng sinh viên xem chi tiết lớp học

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sinh viên xem chi tiết lớp học

### UseCase sinh viên xem thông báo

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase sinh viên xem thông báo

|  |  |
| --- | --- |
| **Tên UC** | **Sinh viên xem thông báo** |
| **Mục đích** | Sinh viên có thể xem thông báo được gửi từ hệ thống, thông báo được chia thành 2 tab: Thông báo của lớp biên chế, thông báo của lớp học phần, thông báo chung của nhà trường. |
| **Tác nhân** | Sinh viên |
| **Tiền điều kiện** | Sinh viên đã đăng nhập hệ thống |
| **Luồng sự kiện chính** | 1. Sinh viên truy cập vào tab Trang chủ  2. Sinh viên chọn icon “Thông báo”  3. Hệ thống hiển thị thông báo theo 3 tab: Thông báo lớp học phần, thông báo lớp biên chế, thông báo chung.  4. Sinh viên có thể xem thông báo trực tiếp trên ứng dụng. |
| **Luồng sự kiện ngoại lệ** | 1. Back-End trả về mã lỗi, timeout dẫn tới ứng dụng không hiểu thị đúng với thông tin trong thực tế  2. Hiển thị thông báo lỗi tới người: “Mất kết nối đến hệ thống. Vui lòng thử lại sau”. |
| **Đảm bảo thành công** | Sinh viên xem được thông báo |

Bảng . Kịch bản chức năng sinh viên xem thông báo

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng sinh viên xem thông báo

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sinh viên xem thông báo

### UseCase giảng viên cập nhật thông tin điểm danh

### UseCase quản trị viên gửi thông báo nhắc nhở sinh viên

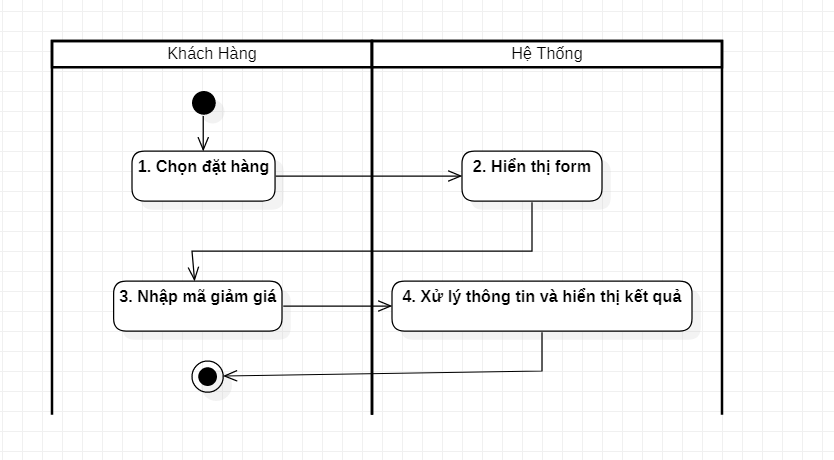
Diagram

Description automatically generated

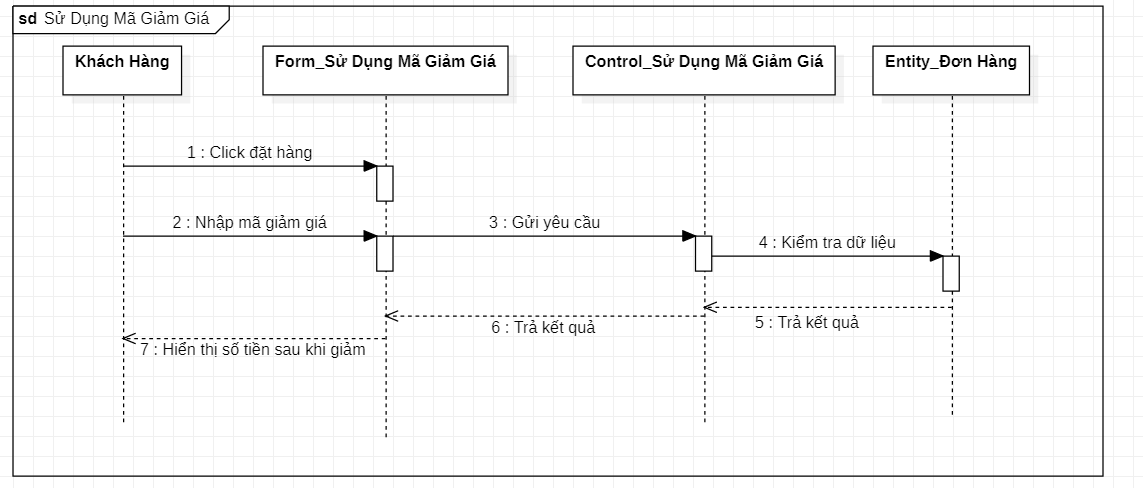
Hình . Biểu đồ phân rã UseCase quản trị viên gửi thông báo nhắc nhở sinh viên

|  |  |
| --- | --- |
| **Tên UC** | **Quản trị viên gửi thông báo nhắc nhở tới sinh viên** |
| **Mục đích** | Trong trường hợp sinh viên sắp nghỉ học quá số buổi quy định, nguy cơ bị cấm thi thì hệ thống cần có cơ chế gửi thông báo tới sinh viên.  Trong trường hợp quản trị viên có đổi lịch học, đổi phòng học hay đổi ca học thì hệ thông cũng có cơ chế gửi thông báo tới sinh viên. |
| **Tác nhân** | Quản trị viên |
| **Tiền điều kiện** | Quản trị viên đã đăng nhập hệ thống |
| **Luồng sự kiện chính** | 1. Quản trị viên đăng nhập hệ thống  2. Truy cập vào trang Quản lý lớp học  3. Thực hiện chọn lớp học, xem chi tiết lớp học  4. Ấn chỉnh sửa  5. Hệ thống hiển thị portal cho phép chỉnh sửa  6. Chỉnh sửa thành công  7. Ấn lưu  (vd: Trường hợp quản trị viên sửa thông tin lớp học) |
| **Luồng sự kiện ngoại lệ** | Cơ chế gửi thông báo ( Push Notification) thông qua firebase nên có thể trong một trường hợp hi hữu thông báo bị gửi chậm. |
| **Đảm bảo thành công** | Quản trị viên gửi thành công thông báo đến sinh viên |

Bảng . Kịch bản chức năng quản trị viên gửii thông báo nhắc nhở sinh viên



Hình . Biểu đồ hoạt động chức năng quản trị viên gửi thông báo nhắc nhở sinh viên



Hình . Biểu đồ tuần tự chức năng quản trị viên gửi thông báo nhắc nhở sinh viên

### UseCase sinh viên gửi phản hồi tới giảng viên

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase sinh viên gửi phản hồi tới giảng viên

|  |  |
| --- | --- |
| **Tên UC** | **Sinh viên gửi phản hồi tới giảng viên** |
| **Mục đích** | Trong trường hợp sinh viên có lý do cho việc điểm danh muộn, đã bị khoá điểm danh cần gửi tới giảng viên, ứng dụng cần có cơ chế hỗ trợ sinh viên báo cáo lý do để được giảng viên xem xét. |
| **Tác nhân** | Sinh viên |
| **Tiền điều kiện** | Sinh viên đã đăng nhập hệ thống |
| **Luồng sự kiện chính** | 1. Sinh viên vào màn hình Trang chủ  2. Chọn môn học trong phần Lịch học  3. Ứng dụng hiển thị Thông tin lớp học  4. Chọn gửi phản hồi tới giảng viên  5. Ấn gửi |
| **Luồng sự kiện ngoại lệ** | 1. Back-End trả về mã lỗi, timeout dẫn tới ứng dụng không hiểu thị đúng với thông tin trong thực tế  2. Hiển thị thông báo lỗi tới người: “Mất kết nối đến hệ thống. Vui lòng thử lại sau”.  3. Sinh viên nhập nội dung quá dài ( chỉ giới hạn 200 ký tứ) dẫn đến ứng dụng thông báo lỗi.  4. Sinh viên bỏ trống, không nhập gì dẫn tới ứng dụng thông báo lỗi |
| **Đảm bảo thành công** | Sinh viên gửi phản hồi tới giảng viên thành công |

Bảng . Kịch bản chức năng sinh viên gửi phản hồi tới giảng viên

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng sinh viên gửi phản hồi tới giảng viên

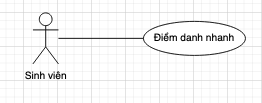
A picture containing diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sinh viên gửi phản hồi tới giảng viên

### UseCase giảng viên tạo ảnh QR Code để sinh viên điểm danh

### UseCase sinh viên điểm danh nhanh bằng QR Code



Hình . Biểu đồ phân rã UseCase sinh viên điểm danh nhanh bằng QRCode

|  |  |
| --- | --- |
| **Tên UC** | **Sinh viên điểm danh nhanh bằng QR Code** |
| **Mục đích** | Giống như các ứng dụng ngân hàng có chức năng quét QRCode nhanh để chuyển tiền, sinh viên có thể chọn option quét QR nhanh để điểm danh lớp học. |
| **Tác nhân** | Sinh viên |
| **Tiền điều kiện** | Sinh viên đã đăng nhập hệ thống, mã QR Code sinh viên điểm danh hợp lệ |
| **Luồng sự kiện chính** | 1. Sinh viên đăng nhập hệ thống  2. Chọn phần “Điểm danh nhanh bằng QRCode”  3. Ứng dụng kiểm qua trạng thái sẵn sàng ( đã bật GPS, kiểm tra GPS)  4. Ứng dụng thực hiện nghiệp vụ điểm danh bằng QR dựa trên thông tin của mã QRCode  5. Hiển thị popip điểm danh thành công |
| **Luồng sự kiện ngoại lệ** | Mã QRCode không hợp lệ, ứng dụng hiển thị thông báo QRCode không hợp lệ. |
| **Đảm bảo thành công** | Sinh viên điểm danh nhanh bằng QRCode thành công |

Bảng . Kịch bản chức năng sinh viên điểm danh nhanh bằng QRCode

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng sinh viên điểm danh nhanh bằng QRCode

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sinh viên điểm danh nhanh bằng QRCode

### UseCase quản trị viên quản lý sinh viên

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase quản trị viên quản lý sinh viên

|  |  |
| --- | --- |
| **Tên UC** | **Quản trị viên quản lý sinh viên** |
| **Mục đích** | Quản trị viên muốn thêm/sửa/xoá sinh viên vào cơ sở dữ liệu |
| **Tác nhân** | Quản trị viên |
| **Tiền điều kiện** | Quản trị viên đã đăng nhập thành công vào hệ thống và chọn giao diện quản lý sinh viên |
| **Luồng sự kiện chính** | 1. Quản trị viên chọn chức năng thêm sinh viên đơn lẻ:  + Điền đầy đủ thông tin sinh viên  + Hệ thống kiểm tra thông tin  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu sinh viên vào CSDL và hiển thị danh sách sinh viên  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  2. Quản trị viên chọn chức năng thêm sinh viên bằng file CSV  + Tải bản CSV mẫu từ hệ thống  + Điền đầy đủ thông tin sinh viên trong file CSV  + Tiến hành upload file CSV lên hệ thống  + Hệ thống đọc từ file CSV, điền dữ liệu vào bản danh sách tạm.  + Quản trị viên kiểm tra thông tin sinh viên trong danh sách  + Ấn xác nhận  + Hệ thống kiểm tra thông tin sinh viên  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống danh sách sinh viên vào CSDL và hiển thị danh sách sinh viên  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  3. Quản trị viên chọn chức năng sửa sinh viên  + Điền thông tin sinh viên cần sửa  + Hệ thống kiểm tra thông tin  + Nếu thông tin sinh viên được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu sinh viên vào CSDL và hiển thị danh sách sinh viên  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  4. Quản trị viên chọn chức năng xoá sinh viên  + Chọn sinh viên cần xoá  + Hiển thị popup xác nhận xoá thông tin sinh viên  + Hệ thống xoá thông tin sinh viên đã chọn khỏi CSDL. |
| **Luồng sự kiện ngoại lệ** | Không |
| **Đảm bảo thành công** | Quản trị viên thêm/sửa /xoá thông tin sinh viên thành công |

Bảng 3.12 Kịch bản chức năng quản trị viên quản lý sinh viên

**A. UseCase Thêm sinh viên**

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng thêm sinh viên

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng thêm sinh viên

**B. UseCase Sửa Thông Tin sinh viên**

Diagram, schematic

Description automatically generated

Hình . Biểu đồ hoạt động chức năng sửa thông tin sinh viên

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sửa thông tin sinh viên

### UseCase quản trị viên quản lý giảng viên

Diagram

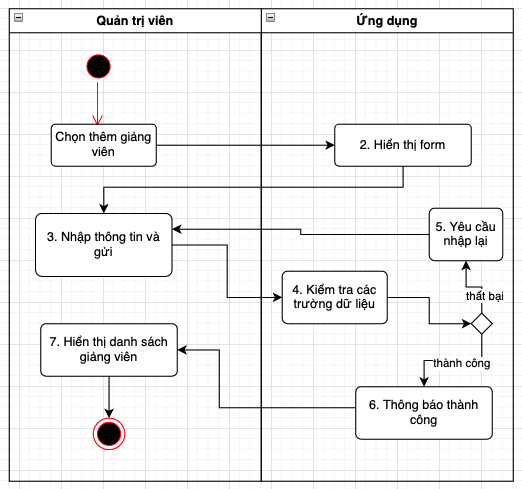
Description automatically generated

Hình . Biểu đồ phân rã UseCase quản trị viên quản lý giảng viên

|  |  |
| --- | --- |
| **Tên UC** | **Quản trị viên quản lý giảng viên** |
| **Mục đích** | Quản trị viên muốn thêm/sửa/xoá giảng viên vào cơ sở dữ liệu |
| **Tác nhân** | Quản trị viên |
| **Tiền điều kiện** | Quản trị viên đã đăng nhập thành công vào hệ thống và chọn giao diện quản lý giảng viên |
| **Luồng sự kiện chính** | 1. Quản trị viên chọn chức năng thêm giảng viên đơn lẻ:  + Điền đầy đủ thông tin giảng viên  + Hệ thống kiểm tra thông tin  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu giảng viên vào CSDL và hiển thị danh sách giảng viên  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  2. Quản trị viên chọn chức năng thêm giảng viên bằng file CSV  + Tải bản CSV mẫu từ hệ thống  + Điền đầy đủ thông tin giảng viên trong file CSV  + Tiến hành upload file CSV lên hệ thống  + Hệ thống đọc từ file CSV, điền dữ liệu vào bản danh sách tạm.  + Quản trị viên kiểm tra thông tin giảng viên trong danh sách  + Ấn xác nhận  + Hệ thống kiểm tra thông tin giảng viên  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống danh sách vào CSDL và hiển thị danh sách giảng viên  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  3. Quản trị viên chọn chức năng sửa giảng viên  + Điền thông tin giảng viên cần sửa  + Hệ thống kiểm tra thông tin  + Nếu thông tin giảng viên được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu giảng viên vào CSDL và hiển thị danh sách giảng viên  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  4. Quản trị viên chọn chức năng xoá giảng viên  + Chọn giảng viên cần xoá  + Hiển thị popup xác nhận xoá thông tin giảng viên  + Hệ thống xoá thông tin giảng viên đã chọn khỏi CSDL. |
| **Luồng sự kiện ngoại lệ** | Không |
| **Đảm bảo thành công** | Quản trị viên thêm/sửa /xoá thông tin giảng viên thành công |

Bảng 3.13 Kịch bản chức năng quản trị viên quản lý giảng viên

**A. UseCase thêm giảng viên**



Hình . Biểu đồ hoạt động chức năng quản trị viên thêm giảng viên

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng quản trị viên thêm giảng viên

**B. UseCase sửa thông tin giảng viên**

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng sửa thông tin giảng viên

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sửa thông tin giảng viên

### UseCase quản trị viên quản lý môn học

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase quản trị viên quản lý môn học

|  |  |
| --- | --- |
| **Tên UC** | **Quản trị viên quản lý môn học** |
| **Mục đích** | Quản trị viên muốn thêm/sửa/xoá môn học vào cơ sở dữ liệu |
| **Tác nhân** | Quản trị viên |
| **Tiền điều kiện** | Quản trị viên đã đăng nhập thành công vào hệ thống và chọn giao diện quản lý môn học |
| **Luồng sự kiện chính** | 1. Quản trị viên chọn chức năng thêm môn học đơn lẻ:  + Điền đầy đủ thông tin môn học  + Hệ thống kiểm tra thông tin  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu môn học vào CSDL và hiển thị danh sách môn học  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  2. Quản trị viên chọn chức năng thêm môn học bằng file CSV  + Tải bản CSV mẫu từ hệ thống  + Điền đầy đủ thông tin môn học trong file CSV  + Tiến hành upload file CSV lên hệ thống  + Hệ thống đọc từ file CSV, điền dữ liệu vào bản danh sách tạm.  + Quản trị viên kiểm tra thông tin môn học trong danh sách  + Ấn xác nhận  + Hệ thống kiểm tra thông tin môn học  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống danh sách vào CSDL và hiển thị danh sách môn học  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  3. Quản trị viên chọn chức năng sửa môn học  + Điền thông tin môn học cần sửa  + Hệ thống kiểm tra thông tin  + Nếu thông tin môn học được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu môn học vào CSDL và hiển thị danh sách môn học  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  4. Quản trị viên chọn chức năng xoá môn học  + Chọn môn học cần xoá  + Hiển thị popup xác nhận xoá thông tin môn học  + Hệ thống xoá thông tin môn học đã chọn khỏi CSDL. |
| **Luồng sự kiện ngoại lệ** | Không |
| **Đảm bảo thành công** | Quản trị viên thêm/sửa /xoá thông tin môn học thành công |

Bảng . Kịch bản chức năng quản trị viên quản lý môn học

**A. UseCase thêm môn học**

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng quản trị viên thêm môn học

Hình . Biểu đồ tuần tự chức năng quản trị viên thêm môn học

**B. UseCase sửa môn học**

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng sửa môn học

Hình . Biểu đồ tuần tự chức năng sửa môn học

### UseCase quản trị viên quản lý phòng học

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase quản trị viên quản lý phòng học

|  |  |
| --- | --- |
| **Tên UC** | **Quản trị viên quản lý phòng học** |
| **Mục đích** | Quản trị viên muốn thêm/sửa/xoá phòng học vào cơ sở dữ liệu |
| **Tác nhân** | Quản trị viên |
| **Tiền điều kiện** | Quản trị viên đã đăng nhập thành công vào hệ thống và chọn giao diện quản lý phòng học |
| **Luồng sự kiện chính** | 1. Quản trị viên chọn chức năng thêm phòng học đơn lẻ:  + Điền đầy đủ thông tin phòng học  + Hệ thống kiểm tra thông tin  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu phòng học vào CSDL và hiển thị danh sách phòng học  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  2. Quản trị viên chọn chức năng thêm phòng học bằng file CSV  + Tải bản CSV mẫu từ hệ thống  + Điền đầy đủ thông tin môn học trong file CSV  + Tiến hành upload file CSV lên hệ thống  + Hệ thống đọc từ file CSV, điền dữ liệu vào bản danh sách tạm.  + Quản trị viên kiểm tra thông tin phòng học trong danh sách  + Ấn xác nhận  + Hệ thống kiểm tra thông tin môn học  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu danh sách vào CSDL và hiển thị danh sách phòng học  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  3. Quản trị viên chọn chức năng sửa môn học  + Điền thông tin môn học cần sửa  + Hệ thống kiểm tra thông tin  + Nếu thông tin môn học được nhập đ đủ và chính xác theo yêu cầu: Hệ thống lưu phòng học vào CSDL và hiển thị danh sách phòng học  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  4. Quản trị viên chọn chức năng xoá phòng học  + Chọn phòng học cần xoá  + Hiển thị popup xác nhận xoá thông tin phòng học + Hệ thống xoá thông tin phòng học đã chọn khỏi CSDL. |
| **Luồng sự kiện ngoại lệ** | Không |
| **Đảm bảo thành công** | Quản trị viên thêm/sửa /xoá thông tin phòng học thành công |

Bảng . Kịch bản chức năng quản trị viên quản lý phòng học

**A. UseCase thêm phòng học**

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng quản trị viên thêm phòng học

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng thêm phòng học

**B. UseCase sửa phòng học**

Diagram, schematic

Description automatically generated

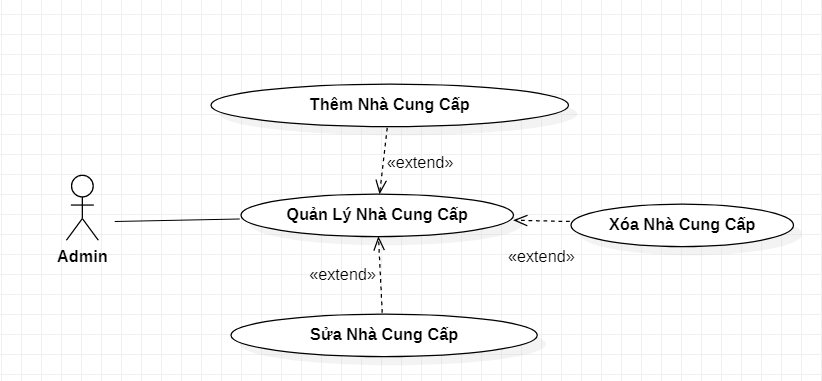
Hình . Biểu đồ hoạt động chức năng sửa phòng học

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sửa phòng học

### UseCase quản trị viên quản lý lớp học phần

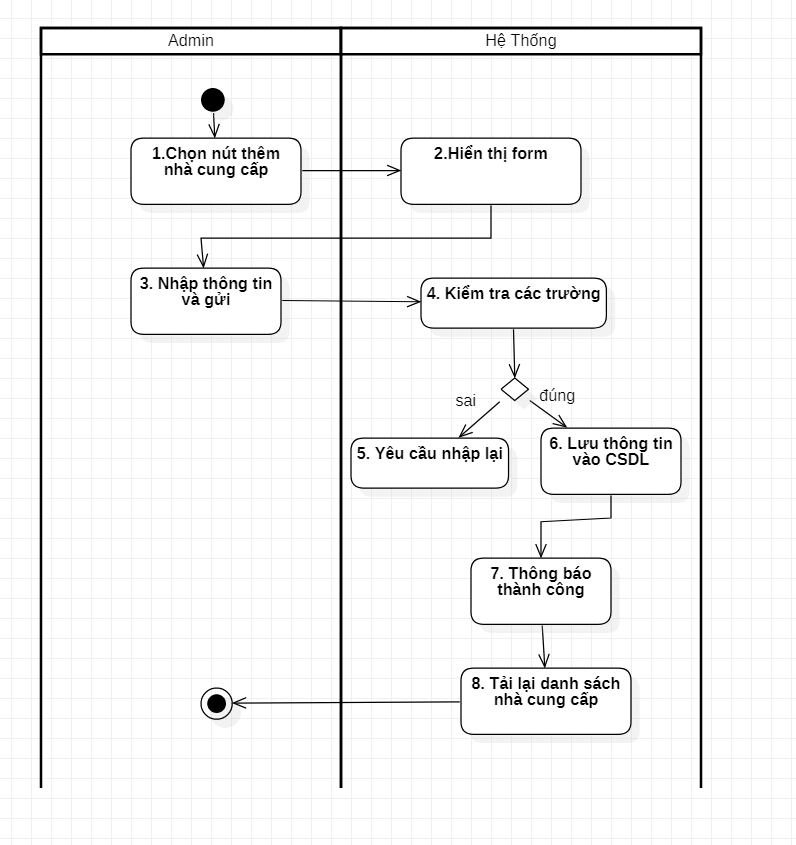


Hình . Biểu đồ phân rã UseCase quản trị viên quản lý lớp học phần

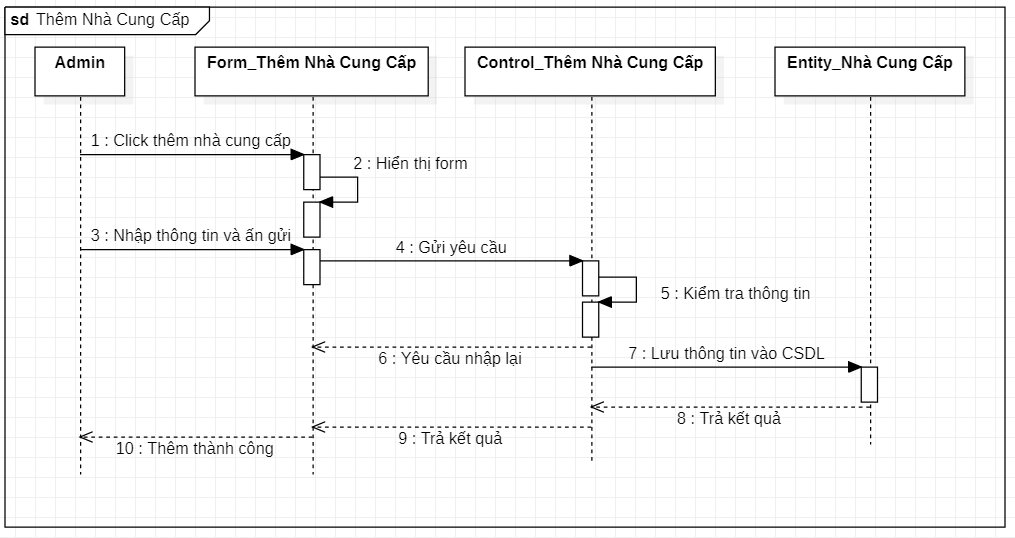
|  |  |
| --- | --- |
| **Tên UC** | **Quản Lý Nhà Cung Cấp** |
| **Mục đích** | Admin muốn thêm/sửa/xóa nhà cung cấp trong cơ sở dữ liệu |
| **Tác nhân** | Admin |
| **Tiền điều kiện** | Admin đã đăng nhập vào hệ thống và và chọn chức năng quản lý nhà cung cấp |
| **Luồng sự kiện chính** | 1. Admin chọn chức năng xoá nhà cung cấp  + Chọn nhà cung cấp cần xoá  + Xác nhận xoá  + Nếu đồng ý xoá: Hệ thống sẽ xoá nhà cung cấp khỏi CSDL  + Nếu từ chối: Hệ thống bỏ thông báo xác nhận  2. Admin chọn chức năng thêm nhà cung cấp  + Hệ thống hiển thị form  + Điền đầy đủ thông tin nhà cung cấp và gửi  + Hệ thống kiểm tra thông tin  + Nếu thông tin nhà cung cấp được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu nhà cung cấp vào CSDL  + Nếu thông tin nhà cung cấp nhập không đầy đủ: Hệ thống thông báo lỗi  3. Admin chọn chức năng sửa nhà cung cấp  + Hệ thống hiển thị form  + Điền đầy đủ thông tin nhà cung cấp và gửi  + Hệ thống kiểm tra thông tin  + Nếu thông tin nhà cung cấp được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu nhà cung cấp vào CSDL  + Nếu thông tin nhà cung cấp nhập không đầy đủ: Hệ thống thông báo lỗi |
| **Luồng sự kiện ngoại lệ** | Không |
| **Đảm bảo thành công** | Admin thêm/sửa/xóa nhà cung cấp thành công |

Bảng . Kịch bản chức năng quản trị viên quản lý lớp học phần

**A.UseCase thêm lớp học phần**

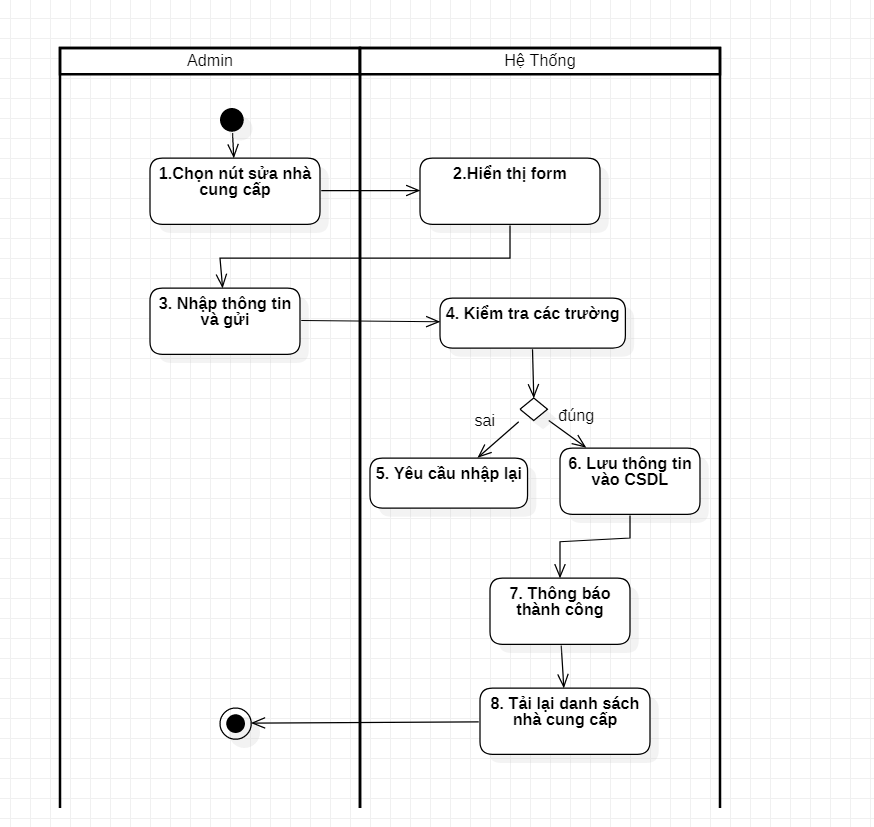


Hình . Biểu đồ hoạt động chức năng thêm lớp học phần

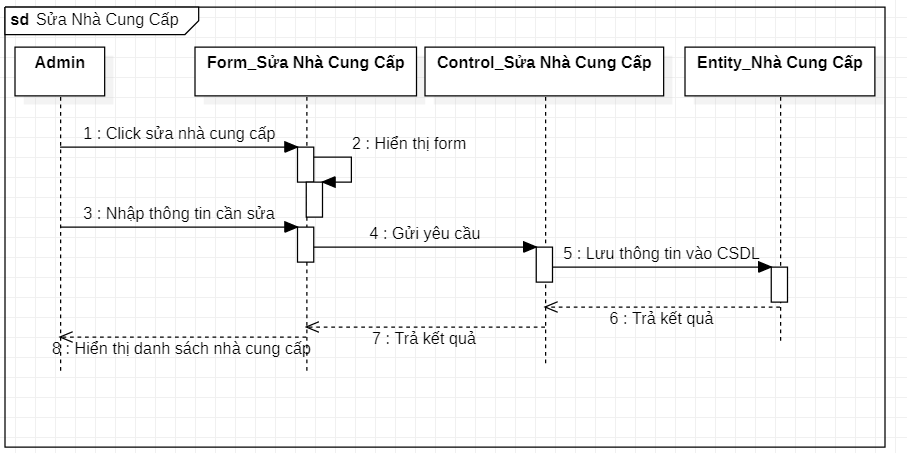


Hình . Biểu đồ tuần tự chức năng thêm lớp học phần

**B. UseCase sửa lớp học phần**

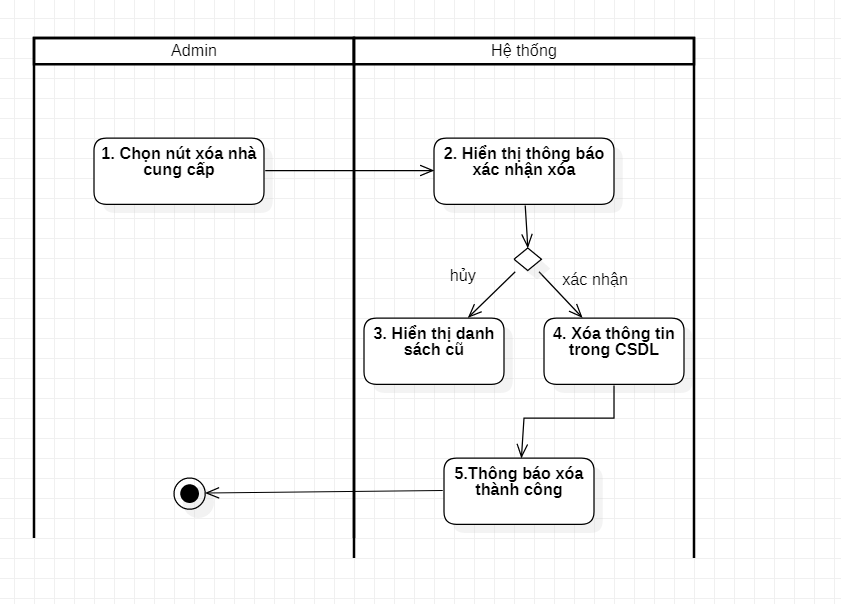


Hình . Biểu đồ hoạt động chức năng sửa lớp học phần

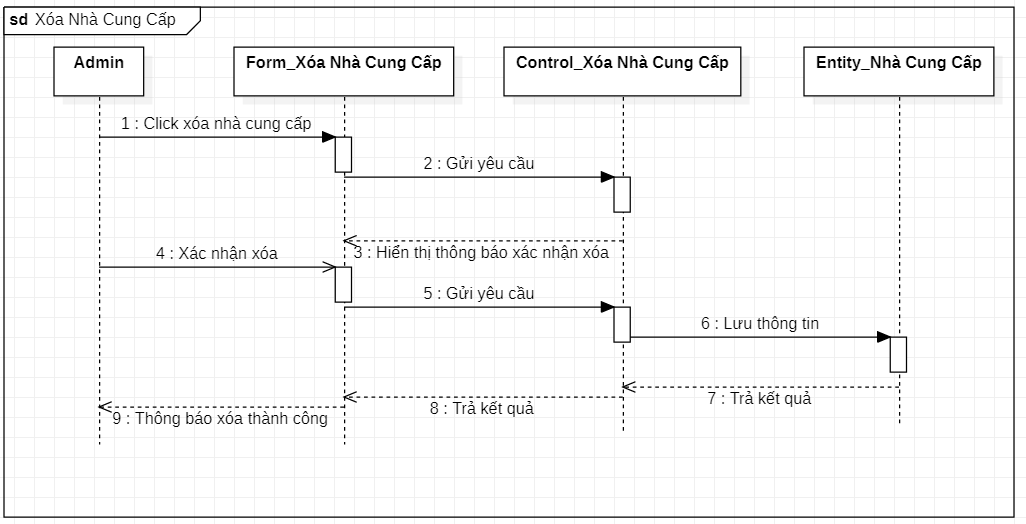


Hình . Biểu đồ tuần tự chức năng sửa lớp học phần

**C. UseCase xóa lớp học phần**



Hình . Biểu đồ hoạt động chức năng xóa lớp học phần



Hình . Biểu đồ tuần tự chức năng xóa lớp học phần

### UseCase quản trị viên quản lý khoa

Diagram

Description automatically generated

Hình . Biểu đồ phân rã UseCase quản trị viên quản lý khoa

|  |  |
| --- | --- |
| **Tên UC** | **Quản trị viên quản lý khoa** |
| **Mục đích** | Quản trị viên muốn thêm/sửa/xoá khoa vào cơ sở dữ liệu |
| **Tác nhân** | Quản trị viên |
| **Tiền điều kiện** | Quản trị viên đã đăng nhập thành công vào hệ thống và chọn giao diện quản lý khoa |
| **Luồng sự kiện chính** | 1. Quản trị viên chọn chức năng thêm khoa đơn lẻ:  + Điền đầy đủ thông tin khoa  + Hệ thống kiểm tra thông tin  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu khoa vào CSDL và hiển thị danh sách khoa  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  2. Quản trị viên chọn chức năng thêm khoa bằng file CSV  + Tải bản CSV mẫu từ hệ thống  + Điền đầy đủ thông tin khoa trong file CSV  + Tiến hành upload file CSV lên hệ thống  + Hệ thống đọc từ file CSV, điền dữ liệu vào bản danh sách tạm.  + Quản trị viên kiểm tra thông tin khoa trong danh sách  + Ấn xác nhận  + Hệ thống kiểm tra thông tin khoa  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu danh sách vào CSDL và hiển thị danh sách khoa  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  3. Quản trị viên chọn chức năng sửa khoa  + Điền thông tin khoa cần sửa  + Hệ thống kiểm tra thông tin  + Nếu thông tin khoa được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu khoa vào CSDL và hiển thị danh sách khoa  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  4. Quản trị viên chọn chức năng xoá khoa  + Chọn khoa cần xoá  + Hiển thị popup xác nhận xoá thông tin khoa  + Hệ thống xoá thông tin khoa đã chọn khỏi CSDL. |
| **Luồng sự kiện ngoại lệ** | Không |
| **Đảm bảo thành công** | Quản trị viên thêm/sửa /xoá thông tin khoa thành công |

Bảng . Kịch bản chức năng quản trị viên quản lý khoa

**A. UseCase thêm khoa**

Diagram, schematic

Description automatically generated

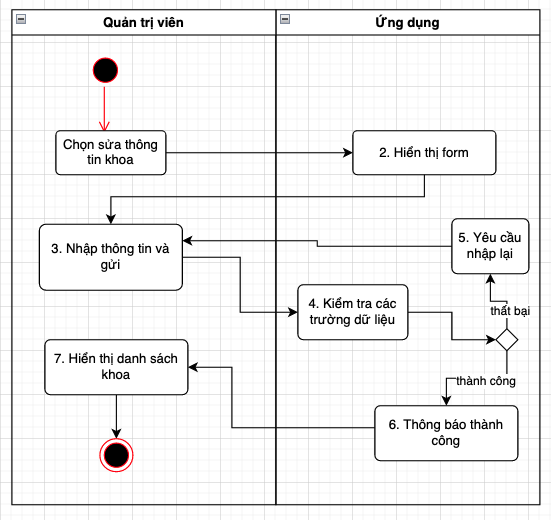
Hình . Biểu đồ hoạt động chức năng thêm khoa

Diagram

Description automatically generated

Hình . Biểu đồ hoạt động chức năng thêm khoa

**B. UseCase sửa khoa**



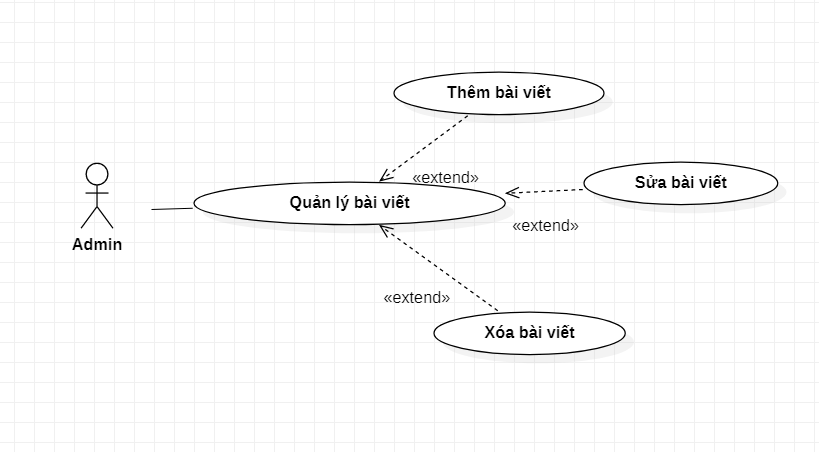
Hình . Biểu đồ hoạt động chức năng sửa khoa

Diagram

Description automatically generated

Hình . Biểu đồ tuần tự chức năng sửa khoa

### UseCase quản trị viên quản lý lớp học biên chế

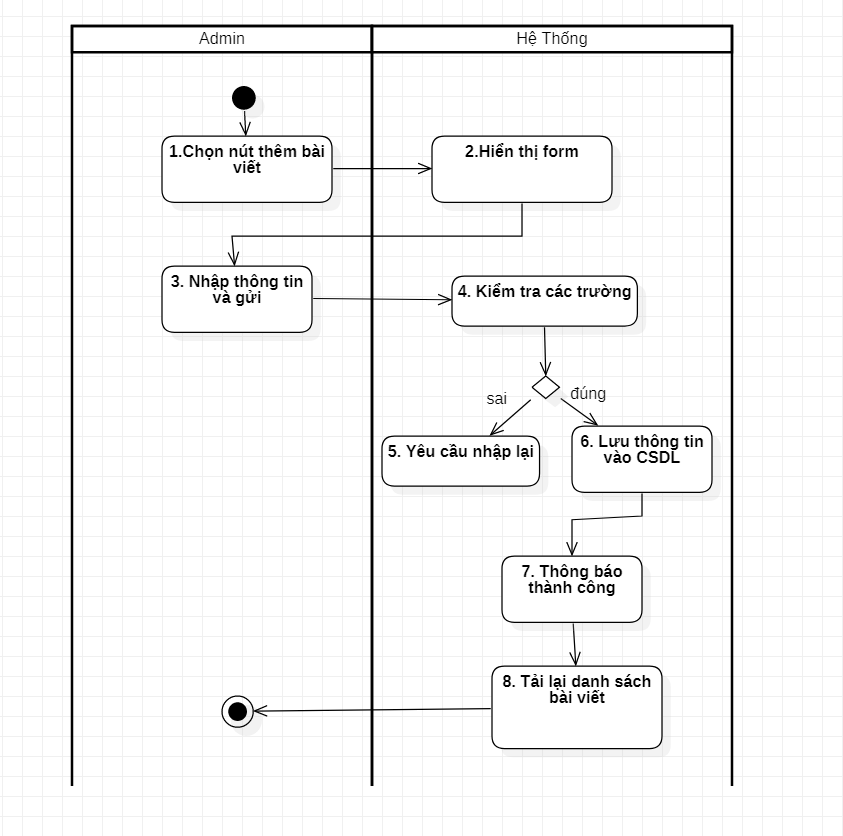


Hình . Biểu đồ phân rã UseCase quản trị viên quản lý lớp học biên chế

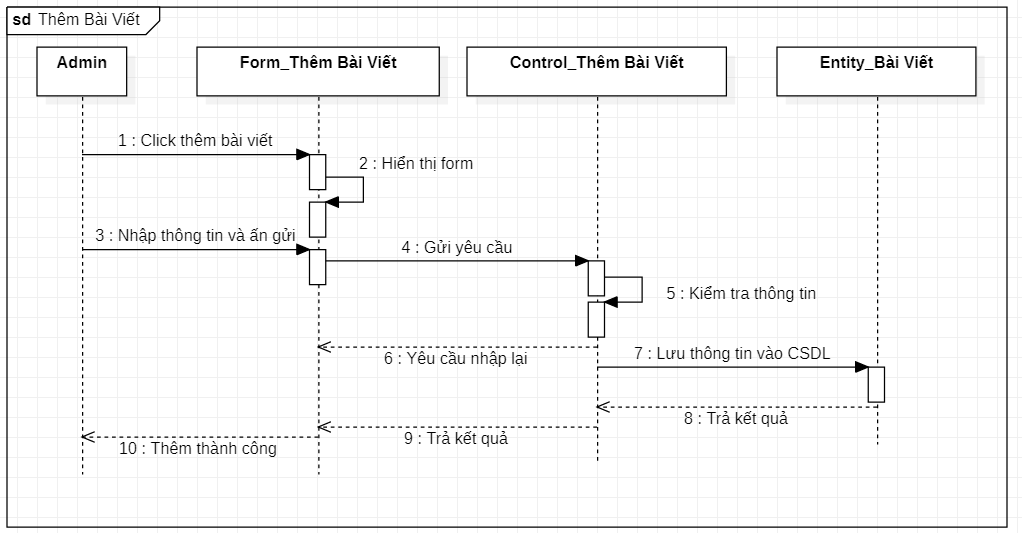
|  |  |
| --- | --- |
| **Tên UC** | **Quản trị viên quản lý lớp học biên chế** |
| **Mục đích** | Quản trị viên muốn thêm/sửa/xoá lớp học biên chế vào cơ sở dữ liệu |
| **Tác nhân** | Quản trị viên |
| **Tiền điều kiện** | Quản trị viên đã đăng nhập thành công vào hệ thống và chọn giao diện quản lý lớp học biên chế |
| **Luồng sự kiện chính** | 1. Quản trị viên chọn chức năng thêm lớp học biên chế đơn lẻ:  + Điền đầy đủ thông tin lớp học biên chế  + Hệ thống kiểm tra thông tin  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu lớp học biên chế vào CSDL và hiển thị danh sách lớp học biên chế  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  2. Quản trị viên chọn chức năng thêm lớp học biên chế bằng file CSV  + Tải bản CSV mẫu từ hệ thống  + Điền đầy đủ thông tin lớp học biên chế trong file CSV  + Tiến hành upload file CSV lên hệ thống  + Hệ thống đọc từ file CSV, điền dữ liệu vào bản danh sách tạm.  + Quản trị viên kiểm tra thông tin lớp học biên chế trong danh sách  + Ấn xác nhận  + Hệ thống kiểm tra thông tin lớp học biên chế  + Nếu thông tin được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu danh sách vào CSDL và hiển thị danh sách lớp học biên chế  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  3. Quản trị viên chọn chức năng sửa lớp học biên chế  + Điền thông tin lớp học biên chế cần sửa  + Hệ thống kiểm tra thông tin  + Nếu thông tin lớp học biên chế được nhập đầy đủ và chính xác theo yêu cầu: Hệ thống lưu lớp học biên chế vào CSDL và hiển thị danh sách lớp học biên chế  + Nếu thông tin nhập không đầy đủ: Hệ thống thông báo lỗi  4. Quản trị viên chọn chức năng xoá lớp học biên chế  + Chọn lớp học biên chế cần xoá  + Hiển thị popup xác nhận xoá thông tin lớp học biên chế  + Hệ thống xoá thông tin lớp học biên chế đã chọn khỏi CSDL. |
| **Luồng sự kiện ngoại lệ** | Không |
| **Đảm bảo thành công** | Quản trị viên thêm/sửa /xoá thông tin lớp học biên chế thành công |

Bảng . Kịch bản chức năng quản trị viên quản lý lớp học biên chế

**A. UseCase thêm lớp học biên chế**

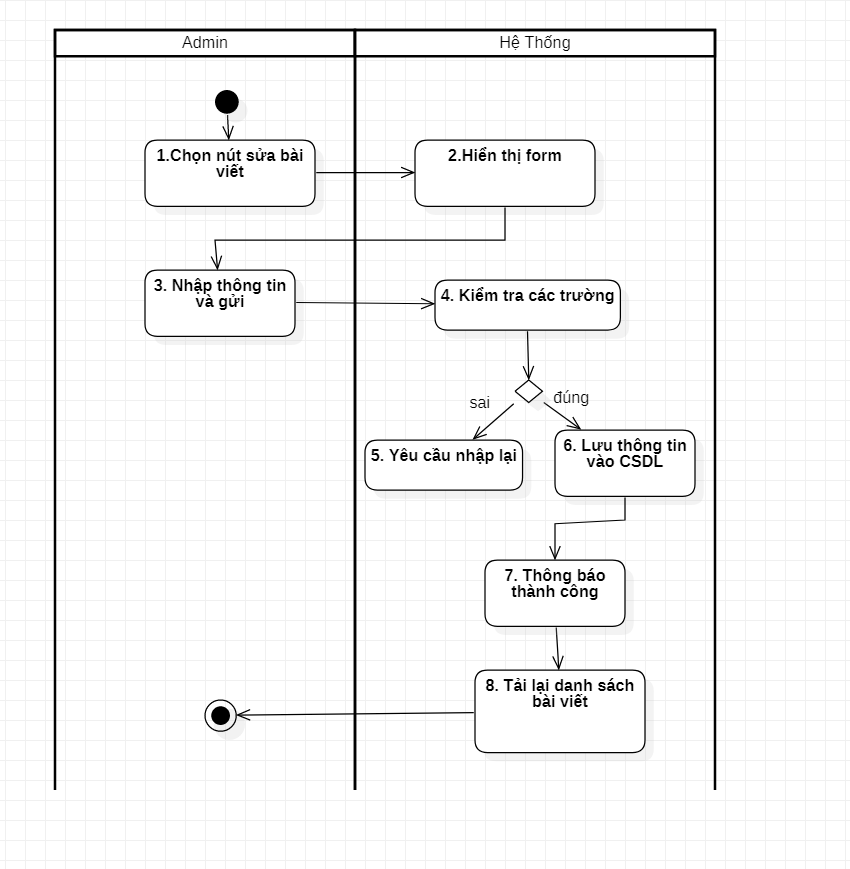


Hình . Biểu đồ hoạt động chức năng thêm lớp học biên chế

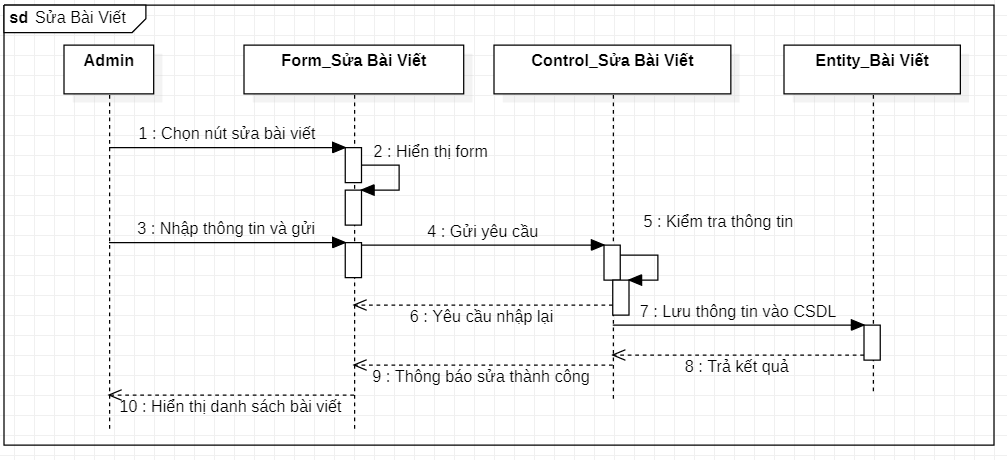


Hình . Biểu đồ tuần tự chức năng thêm lớp học biên chế

**B. UseCase sửa lớp học biên chế**

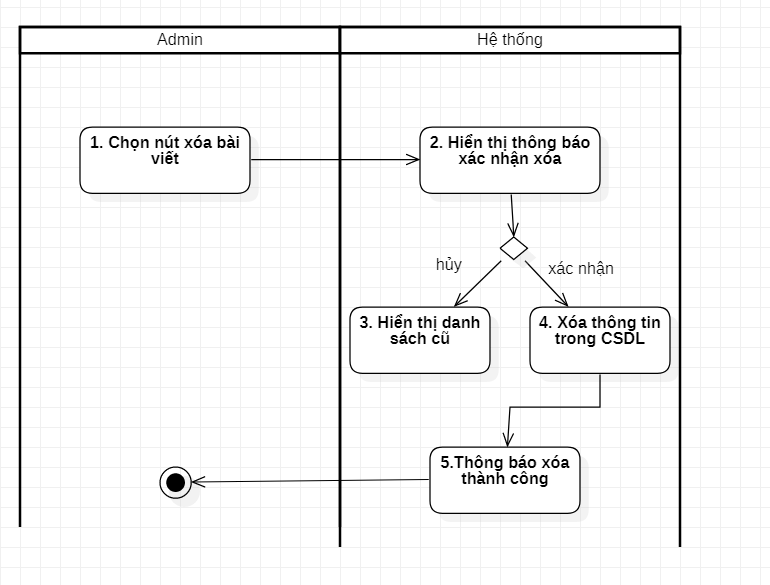


Hình . Biểu đồ hoạt động chức năng sửa lớp học biên chế

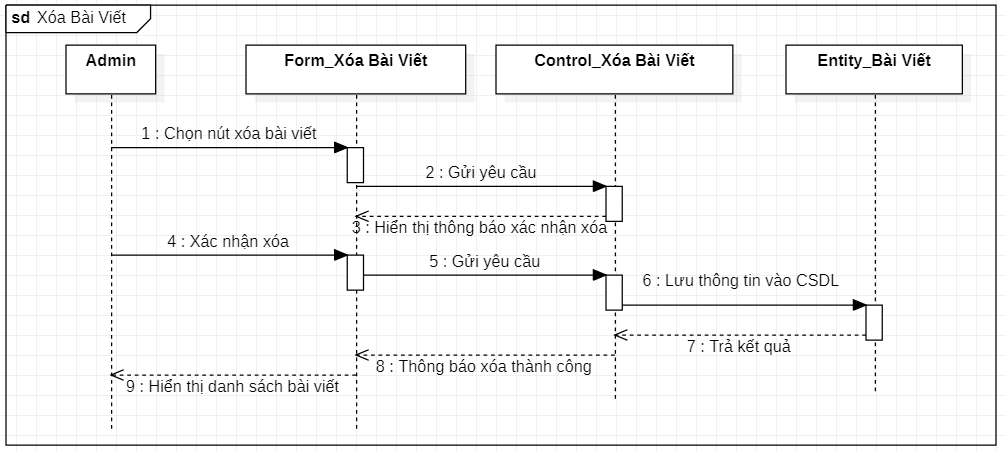


Hình . Biểu đồ tuần tự chức năng sửa lớp học biên chế

**C. UseCase xóa lớp học biên chế**



Hình . Biểu đồ hoạt động chức năng xóa lớp học biên chế



Hình . Biểu đồ tuần tự chức năng xóa lớp học biên chế

## Xây dựng sơ đồ lớp

### Xác định lớp

|  |  |  |
| --- | --- | --- |
| STT | Tên lớp | Ý nghĩa |
| 1 | sinhvien | Lưu thông tin của sinh viên |
| 2 | giangvien | Lưu thông tin của giảng viên |
| 3 | khoa | Lưu thông tin các khoa trong trường |
| 4 | lop | Lưu thông tin của các lớp |
| 5 | cahoc | Lưu thông tin các ca học trong ngày |
| 6 | lopmonhoc | Lưu thông tin các lớp môn học |
| 7 | monhoc | Lưu thông tin các môn học |
| 8 | gv\_diemdanh\_sv | Lưu thông tin về các buổi học giảng viên điểm sinh viên |
| 9 | monhoc\_lopmonhoc | Lưu thông tin về các lớp môn học thuộc môn học |
| 10 | chitietdiemdanh | Lưu thông tin về chi tiết thông tin điểm danh của sinh viên trong từng buổi học có điểm danh |
| 11 | ca\_phong\_lopmonhoc | Lưu thông tin về lớp môn học ở phòng học nào và ca học nào |
| 12 | sinhvien\_hoc\_lopmonhoc | Lưu thông tin về sinh viên học các lớp môn học |
| 13 | phonghoc | Lưu thông tin về phòng học |

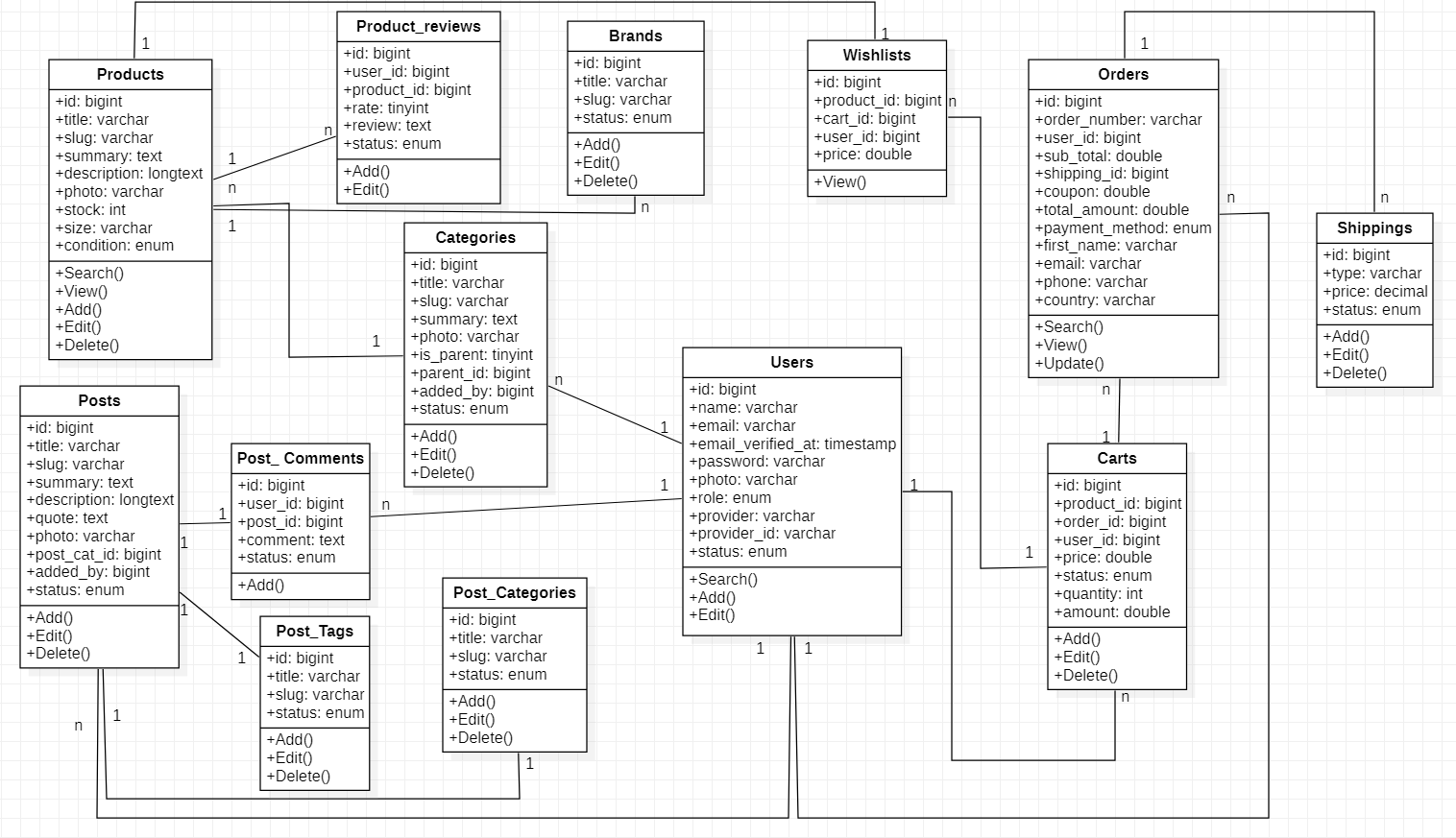
Bảng . Danh sách các lớp của hệ thống

### Xác định các kiểu dữ liệu và phương thức

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Lớp | Thuộc tính | Phương Thức |
| 1 | sinhvien | id: int  ma\_sv: varchar  ma\_lop: varchar  hovaten: varchar  ngaysinh: date  gioitinh: varchar  ma\_thiet\_bi: varchar  ma\_fcm: varchar  matkhau: varchar | Add()  Edit()  Delete() |
| 2 | khoa | id: int  ma\_khoa: varchar  ten\_khoa: varchar | Add()  Edit()  Delete() |
| 3 | giangvien | id: int  ma\_gv: varchar  ma\_khoa: varchar  hovaten: varchar  ngaysinh: date  gioitinh: varchar  ma\_thiet\_bi: varchar  ma\_fcm: varchar  matkhau: varchar | Add()  Edit()  Delete() |
| 4 | lop | id: int  ma\_lop: bigint  ten\_lop: bigint | Add()  Edit()  Delete() |
| 5 | phonghoc | id: int  ma\_phonghoc: varchar  ten\_phonghoc: varchar  dia\_chi: varchar | Add()  Edit()  Delete() |
| 6 | cahoc | id: int  ma\_cahoc: varchar  gio\_bat\_dau: varchar  gio\_ket\_thuc: varchar | Add()  Edit()  Delete() |
| 7 | lopmonhoc | id: int  ma\_lopmonhoc: varchar  ma\_gv: varchar  ten\_lopmonhoc: bigint | Add()  Edit()  Delete() |
| 8 | monhoc | id: int  ma\_monhoc: varchar  ten\_monhoc: varchar  so\_tinchi: varchar | Add()  Edit()  Delete() |
| 9 | monhoc\_lopmonhoc | id: int  ma\_monhoc: varchar  ma\_lopmonhoc: varchar  namhoc: varchar  hocky: varchar | Add()  Edit()  Delete() |
| 10 | gv\_diemdanh\_sv | id: int  ma\_gv: varchar  ma\_lopmonhoc: varchar  ngay\_diemdanh: date  gio\_diemdanh: varchar | Add()  Edit()  Delete() |
| 11 | chitiet\_diemdanh | id: int  ma\_diemdanh: int  ma\_sv: varchar  thoigian\_diemdanh: date  lydo: varchar | Add()  Edit()  Delete() |
| 12 | ca\_phong\_lopmonhoc | id: int  ma\_cahoc: varchar  ma\_phonghoc: varchar  ma\_lopmonhoc: varchar | Add()  Edit()  Delete() |
| 13 | sinhvien\_hoc\_lopmonhoc | id: int  ma\_lopmonhoc: varchar  ma\_sv: varchar  kichhoat: boolean | Add()  Edit()  Delete() |

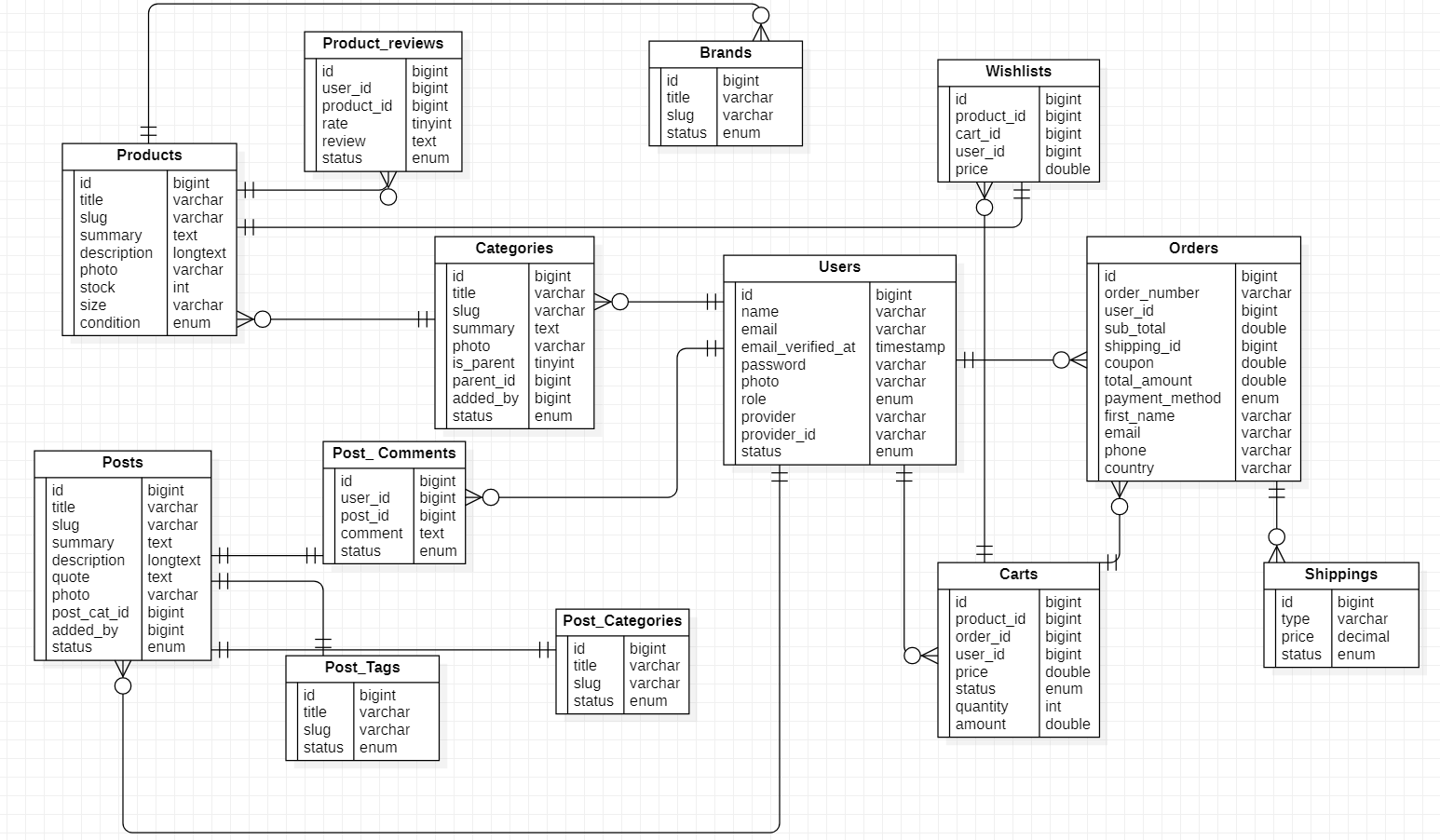
Bảng . Xác định lớp các kiểu dữ liệu và hàm

### Sơ đồ lớp



Hình . Biểu đồ lớp của hệ thống

## Mô hình dữ liệu thực thể



Hình . Mô hình dữ liệu thực thể

# THIẾT KẾ CHI TIẾT HỆ THỐNG

## Bảng cơ sở dữ liệu

**Ảnh có chứa văn bản, trong nhà, ảnh chụp màn hình

Mô tả được tạo tự động**

Hình . Bảng cơ sở dữ liệu

## Thiết kế cơ sở dữ liệu

### Bảng SINHVIEN

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_sv | varchar | Mã sinh viên | Không null, Duy nhất |
| ma\_lop | int | Mã lớp biên chế | Khóa ngoại |
| hoten | varchar | Họ và tên sinh viên |  |
| ngaysinh | date | Ngày sinh |  |
| gioitinh | varchar | Giới tính |  |
| ma\_thietbi | varchar | Mã thiết bị | Duy nhất |
| ma\_fcm | varchar | Mã fcm | Duy nhất |
| matkhau | varchar | Mật khẩu tài khoản |  |

Bảng . Mô tả bảng SINHVIEN

### Bảng KHOA

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_khoa | varchar | Mã khoa | Không null, Duy nhất |
| ten\_khoa | varchar | Tên khoa |  |

Bảng . Mô tả bảng KHOA

### Bảng LOP

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_lop | varchar | Mã lớp | Không null, Duy nhất |
| ten\_lop | varchar | Tên lớp |  |

Bảng . Mô tả bảng LOP

### Bảng GIANGVIEN

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | Int | Id | Khóa chính |
| ma\_gv | varchar | Mã giảng viên | Không null, Duy nhất |
| ma\_khoa | varchar | Mã khoa | Khóa ngoại |
| hoten | varchar | Họ và tên giảng viên |  |
| ngaysinh | date | Ngày sinh |  |
| gioitinh | varchar | Giới tính |  |
| ma\_thietbi | varchar | Mã thiết bị | Duy nhất |
| ma\_fcm | varchar | Mã fcm | Duy nhất |
| matkhau | varchar | Mật khẩu tài khoản |  |

Bảng . Mô tả bảng GiANGVIEN

### Bảng PHONGHOC

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_phonghoc | varchar | Mã phòng học | Không null, Duy nhất |
| ten\_phonghoc | varchar | Tên phòng học |  |
| diachi | varchar | Địa chỉ phòng học theo GPS |  |

Bảng . Mô tả bảng PHONGHOC

### Bảng CAHOC

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_cahoc | varchar | Mã ca học | Không null, Duy nhất |
| gio\_batdau | varchar | Giờ bắt đầu |  |
| gio\_ketthuc | varchar | Giờ kết thúc |  |

Bảng . Mô tả bảng CAHOC

### ***Bảng LOPMONHOC***

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | Int | Id | Khóa chính |
| ma\_lopmonhoc | varchar | Mã lớp môn hoc | Khóa ngoại |
| ma\_gv | varchar | Mã giảng viên | Khóa ngoại |
| ten\_lopmonhoc | varchar | Tên lớp môn hoc |  |

Bảng . Mô tả bảng LOPMOCHOC

### Bảng MONHOC

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_monhoc | varchar | Mã môn hoc | Không null, Duy nhất |
| ten\_monhoc | varchar | Tên môn học |  |
| sotinchi | varchar | Số tín chỉ môn học |  |

Bảng . Mô tả bảng MONHOC

### Bảng GV\_DIEMDANH\_SV

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_gv | varchar | Mã giảng viên | Khóa ngoại |
| ma\_lopmonhoc | varchar | Mã lớp môn học | Khóa ngoại |
| ngay\_diemdanh | varchar | Ngày điểm danh |  |
| gio\_diemdanh | varchar | Thời gian điểm danh |  |

Bảng . Mô tả bảng GV\_DIEMDANH\_SV

### Bảng MONHOC\_LOPMONHOC

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_monhoc | varchar | Mã môn học | Khóa ngoại |
| ma\_lopmonhoc | varchar | Mã lớp môn học | Khóa ngoại |
| namhoc | varchar | Năm học |  |
| hocky | varchar | Học kỳ |  |

Bảng . Mô tả bảng MONHOC\_LOPMONHOC

### Bảng CHITIETDD

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | Int | Id | Khóa chính |
| ma\_diemdanh | varchar | Mã điểm danh | Không null, Duy nhất |
| ma\_sv | varchar | Mã sinh viên | Khóa ngoại |
| thoigian\_diemdanh | varcar | Thời gian điểm danh |  |
| lydo | varchar | Lý do điểm danh muộn ( nếu có ) |  |

Bảng . Mô tả bảng CHITIETDD

### Bảng CA\_PHONG\_LOPMONHOC

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_cahoc | varchar | Mã ca học | Khóa ngoại |
| ma\_phonghoc | varchar | Mã phòng học | Khóa ngoại |
| ma\_lopmonhoc | varchar | Mã lớp môn học | Khóa ngoại |

Bảng . Mô tả bảng CA\_PHONG\_LOPMONHOC

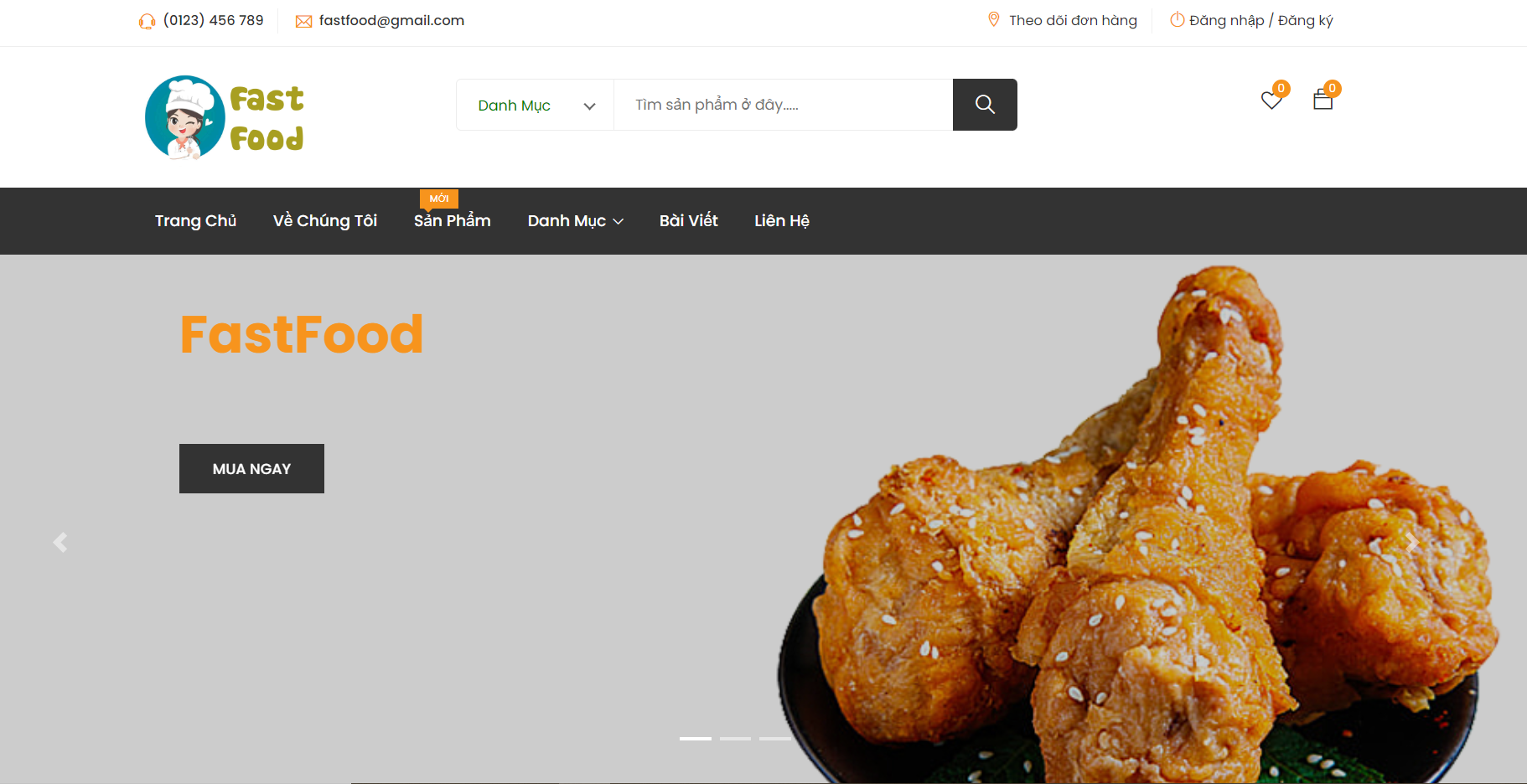
### Bảng SINHVIEN\_HOC\_LOPMONHOC

|  |  |  |  |
| --- | --- | --- | --- |
| Thuộc tính | Kiểu dữ liệu | Mô tả | Ràng buộc |
| id | int | Id | Khóa chính |
| ma\_lopmonhoc | varchar | Mã lớp môn học | Khóa ngoại |
| ma\_sv | varchar | Mã sinh viên | Khóa ngoại |
| kichhoat | boolean | Trạng thái của sinh viên trong lớp môn học đó |  |

Bảng .13 Mô tả bảng CA\_PHONG\_LOPMONHOC

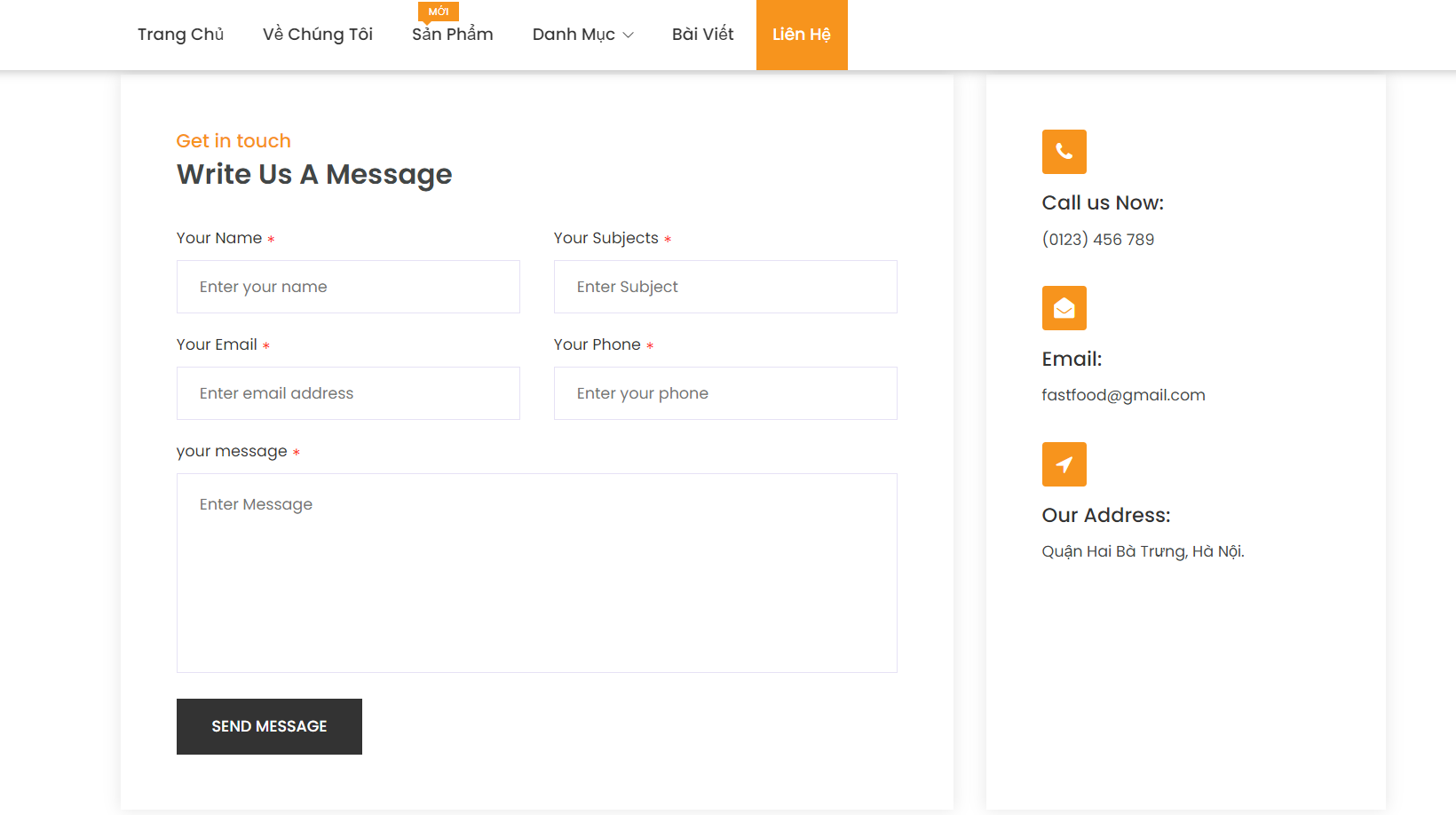
## Thiết kế giao diện

### Giao diện trang đăng nhập của quản trị viên



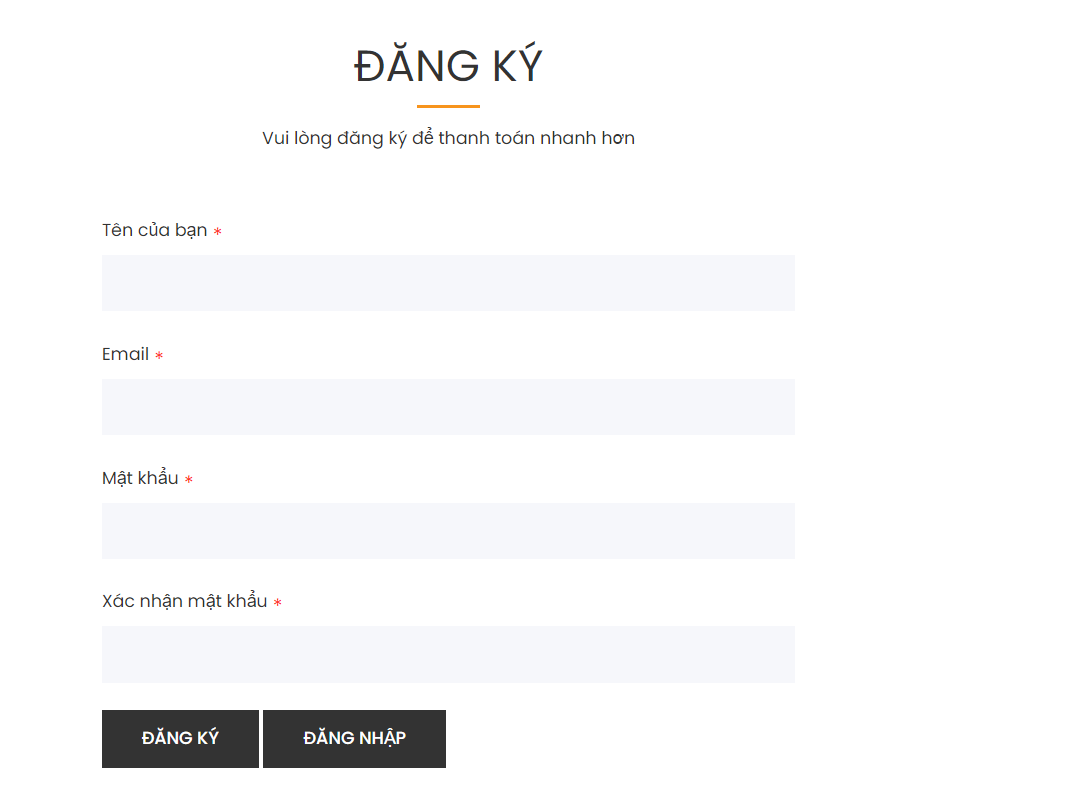
Hình . Giao diện trang chủ

### Giao diện trang chủ của quản trị viên



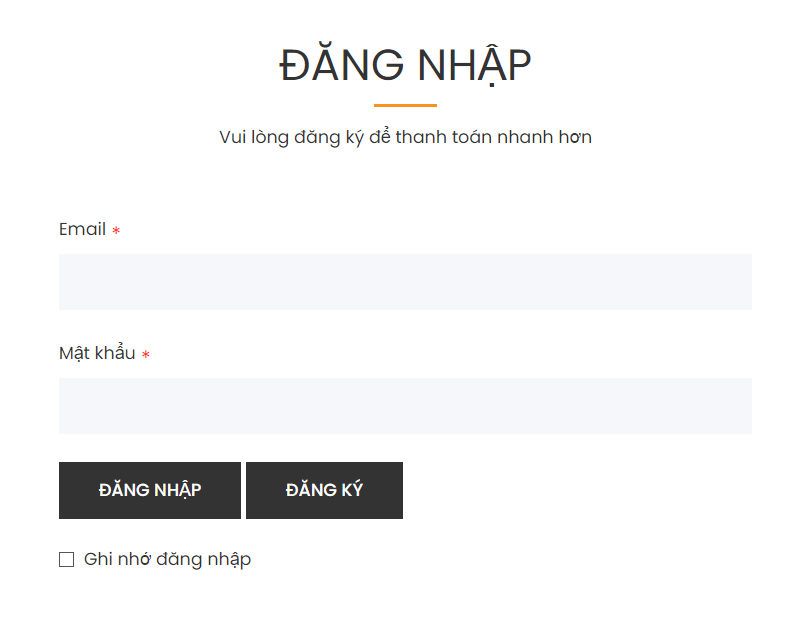
Hình . Giao diện trang liên hệ

### Giao diện trang quản lý sinh viên



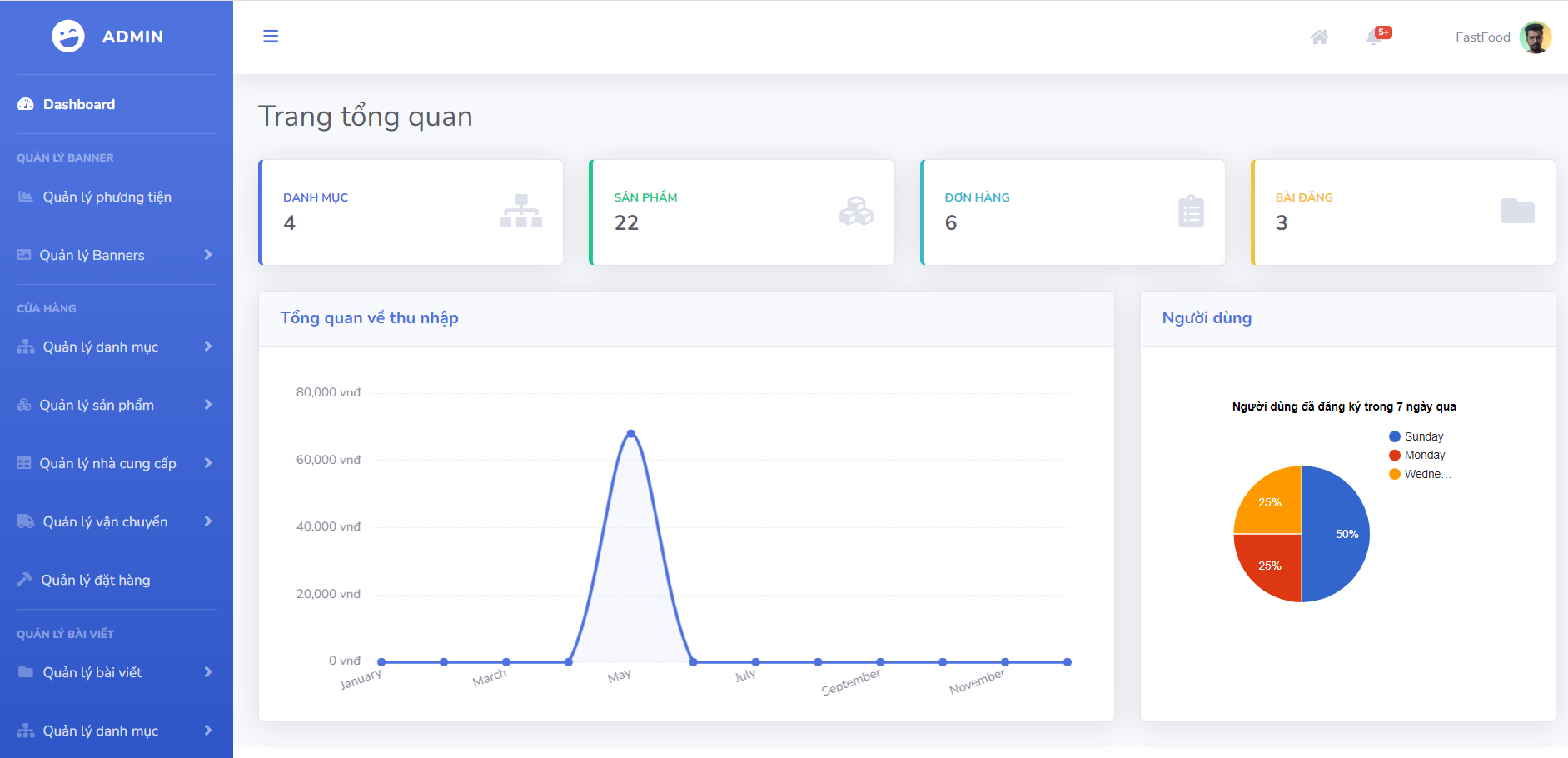
Hình . Giao diện trang đăng ký

### Giao diện trang quản lý giảng viên



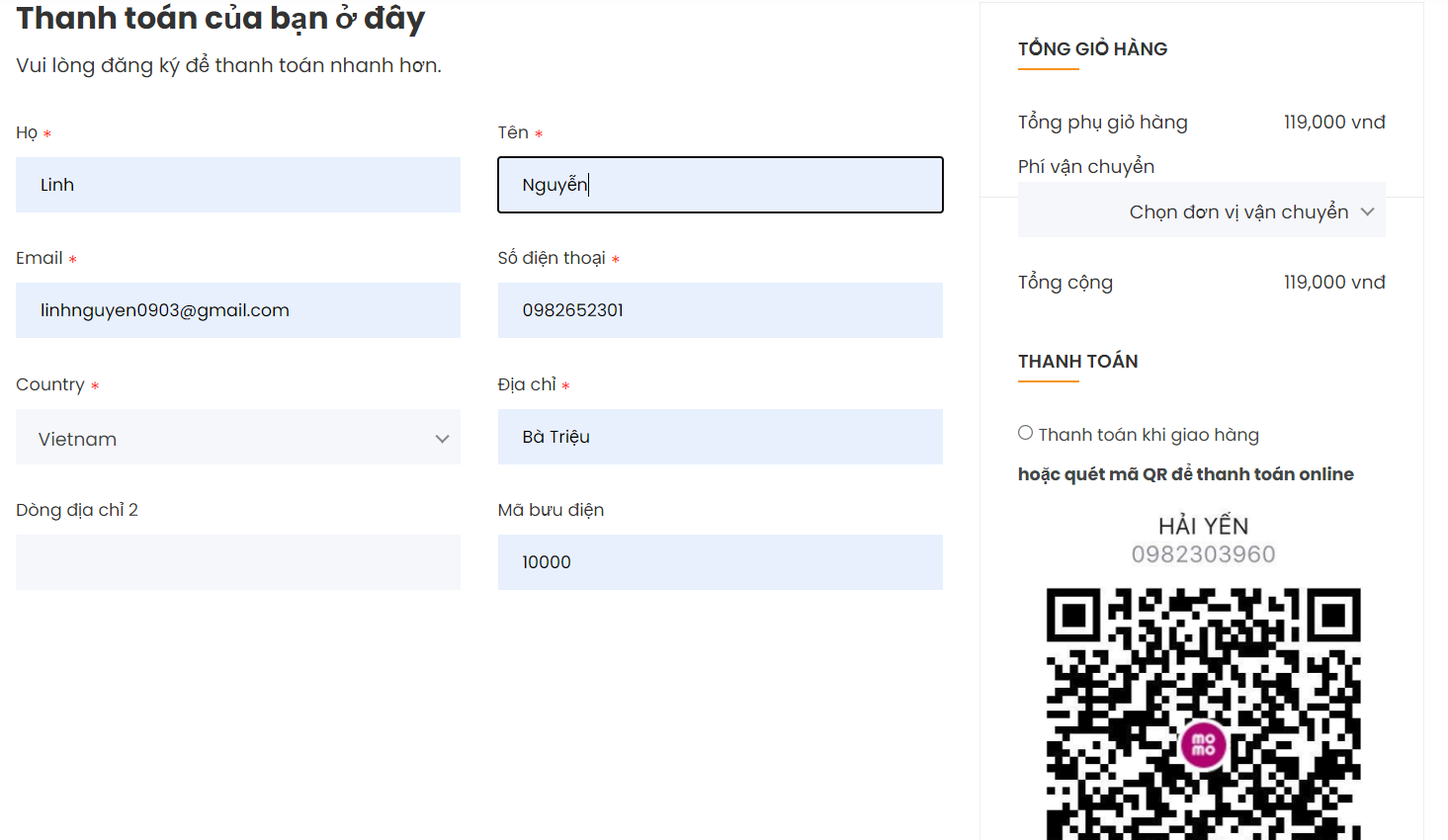
Hình . Giao diện trang đăng nhập

### Giao diện trang quản lý môn học



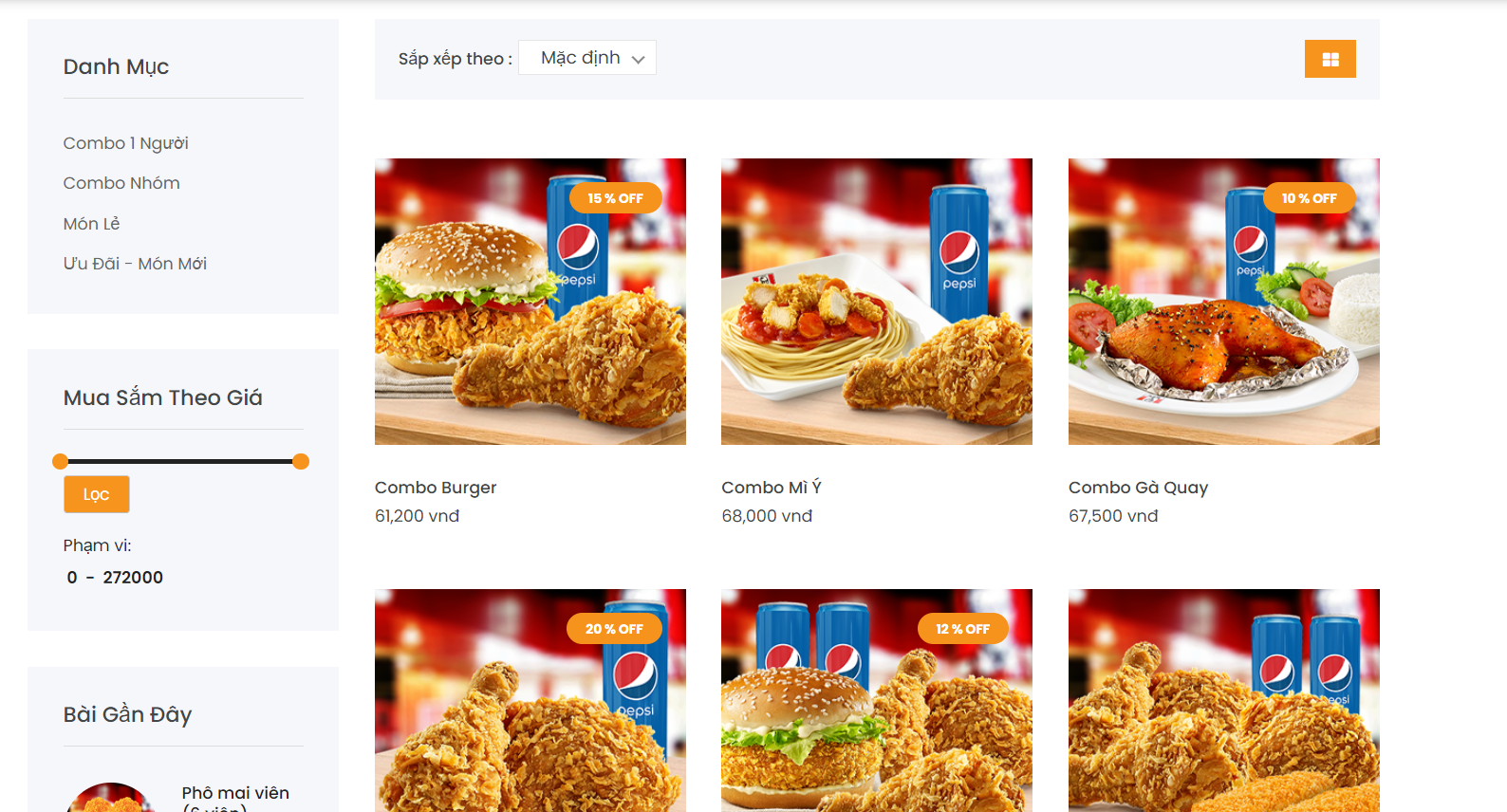
Hình . Giao diện trang quản lý

### Giao diện trang quản lý phòng học



Hình . Giao diện trang đặt hàng

### Giao diện trang quản lý lớp học biên chế



Hình . Giao diện trang xem thông tin sản phẩm

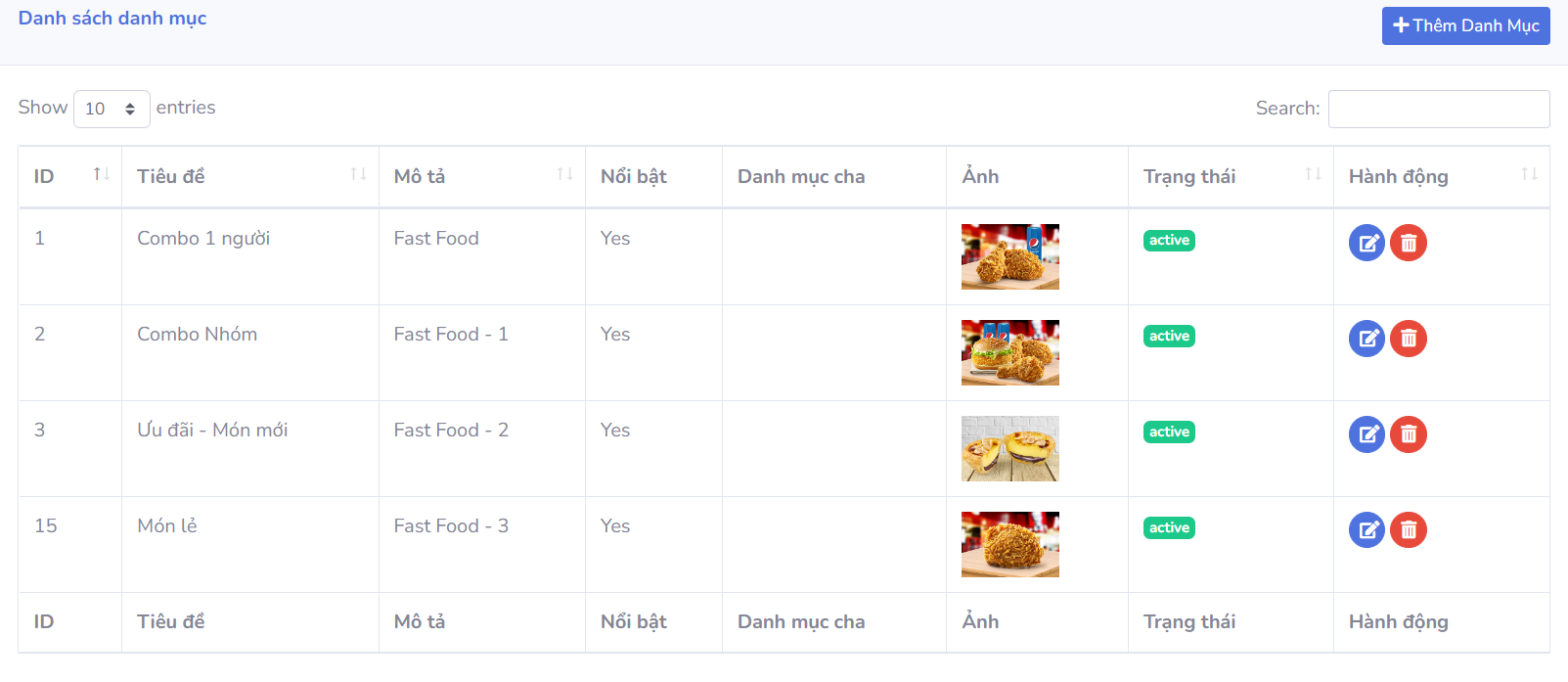
### Giao diện trang quản lý lớp học phần

Ảnh có chứa văn bản

Mô tả được tạo tự động

Hình . Giao diện trang quản lý người dùng

### Giao diện màn hình đăng nhập ứng dụng



Hình . Giao diện trang quản lý danh mục

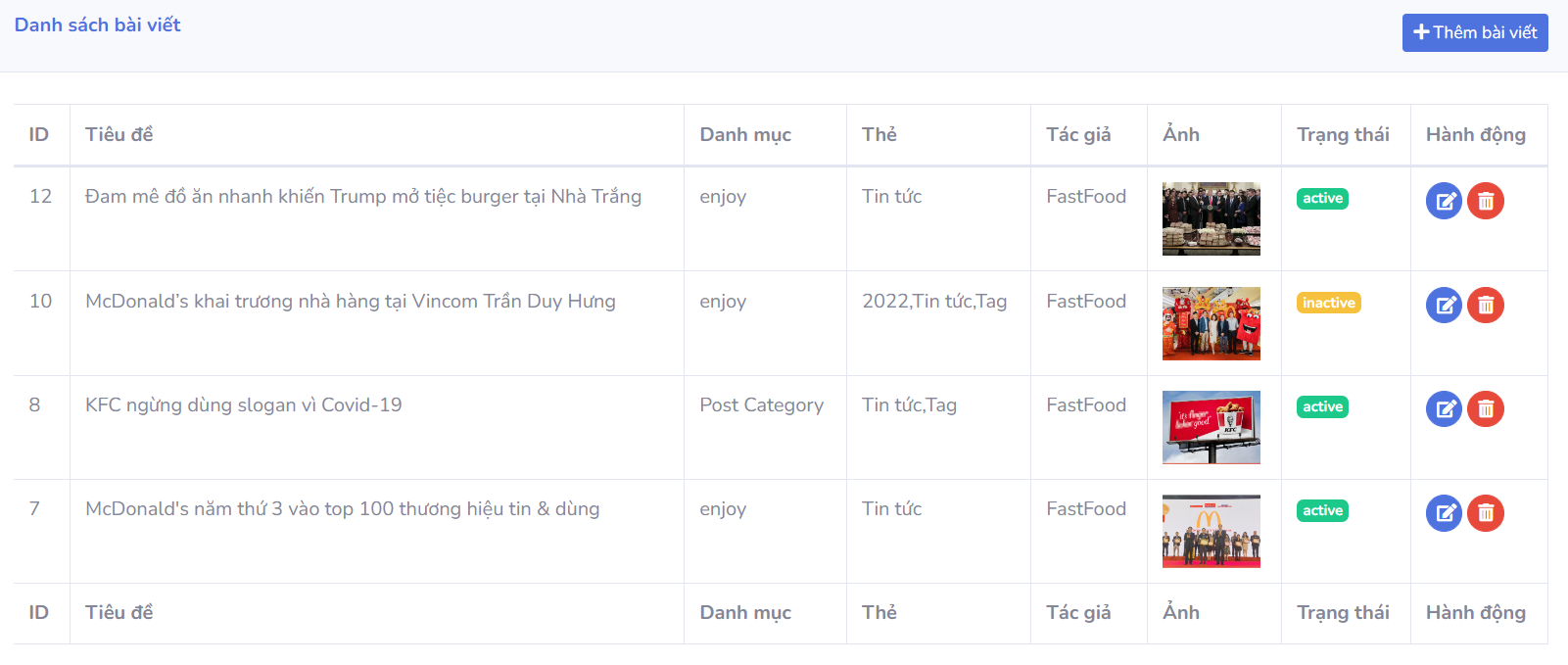
### Giao diện màn hình Trang chủ ứng dụng



Hình . Giao diện trang quản lý sản phẩm

### Giao diện màn hình Tiện ích ứng dụng

### Giao diện màn hình Góc học tập ứng dụng



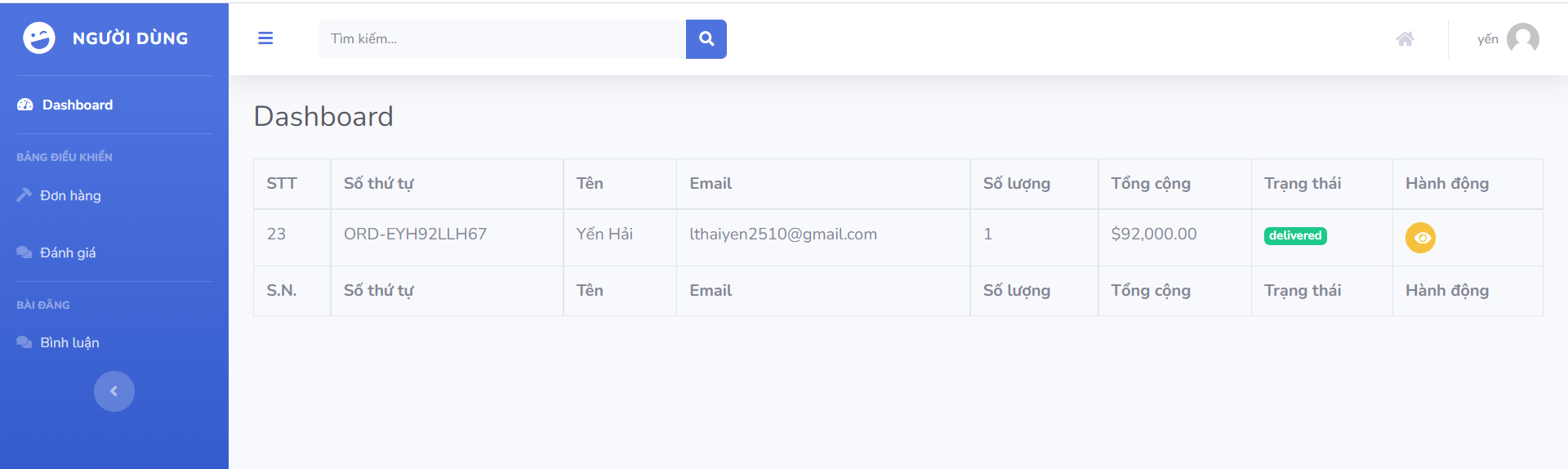
Hình . Giao diện trang quản lý bài viết

### Giao diện màn hình Thông báo ứng dụng

### Giao diện màn hình Cá nhân ứng dụng

### Giao diện màn hình Chi tiết lớp học

### Giao diện màn hình thống kê điểm danh sinh viên ứng dụng



Hình . Giao diện trang chủ người dùng

# KIỂM THỬ HỆ THỐNG

## Xây dựng bộ dữ liệu kiểm thử

Thông qua việc tìm hiểu công tác quản lý điểm danh sinh viên cũng như tham khảo lịch học của sinh khoa Công nghệ thông tin trường Đại học Thủy Lợi, em đã bổ sung dữ liệu lớp học, năm học, môn học, lịch học để làm dữ liệu đầu vào cho quá trình kiểm thử.

Với bộ dữ liệu thu được. em thực hiện bổ sung thêm thông tin vị trí phòng học (vị trí thực tế các phòng học tại trường được em đến tận nơi để tracking GPS) để có thể chạy các chức năng của hệ thống.

## Kế hoạch kiểm thử

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Mục đích kiểm thử** | **Các bước thực hiện** | **Đầu vào** | **Kết quả** |
| **1** | *Kiểm thử trên thiết bị di động:*  Chức năng đăng nhập với tài khoản chưa đăng nhập bao giờ trên thiết bị mới. | 1. Nhập tài khoản, mật khẩu vào màn hình Đăng nhập trên ứng dụng  2. Nhấn vào button Đăng nhập | Tên tài khoản, mật khẩu | Đăng nhập thành công |
| **2** | *Kiểm thử trên thiết bị di động:*  Chức năng đăng nhập với tài khoản chưa đăng nhập bao giờ trên thiết bị đã từng được đăng nhập. | 1. Nhập tài khoản, mật khẩu vào màn hình Đăng nhập trên ứng dụng  2. Nhấn vào button Đăng nhập | Tên tài khoản, mật khẩu | Đăng nhập không thành công và hiển thị thông báo “Thiết bị đã được sử dụng trên tài khoản khác” |
| **3** | *Kiểm thử trên thiết bị di động:*  Chức năng đăng nhập với tài khoản đã đăng tập trên thiết bị mới. | 1. Nhập tài khoản, mật khẩu vào màn hình Đăng nhập trên ứng dụng  2. Nhấn vào button Đăng nhập | Tên tài khoản, mật khẩu | Đăng nhập không thành công và hiển thị thông báo “Đăng nhập trên thiết bị lạ” |
| **4** | *Kiểm thử trên thiết bị di động:*  Chức năng điểm danh bằng QRCode | 1. Sinh viên chọn buổi học cần điểm danh  2. Xác định trạng thái điểm danh đang “Chưa điểm danh”  3. Bật chức năng quét QR để ứng dụng xác định vị trí hiện tại của bạn là trong lớp học  4. Chọn button quét QR | Ảnh QR được cung cấp bởi giảng viên | Sau khi điểm danh hiển thị trạng thái “Đã điểm danh” và không hiển thị mục GPS và button Điểm danh |
| **5** | *Kiểm thử trên thiết bị di động:*  Chức năng xem thông báo | 1. Đăng nhập vào ứng dụng với tài khoản và email của sinh viên  2.Chọn icon quả chuông ở gói phải trên màn hình | Không | Sinh viên có thể xem thông báo theo 2 mục có sẵn: Thông báo lớp học phần, thông báo lớp biên chế |
| **6** | *Kiểm thử trên thiết bị di động*:  Chức năng xem thông tin lớp đổi với Giảng viên | 1. Đăng nhập vào ứng dụng với tài khoản Giảng viên  2. Chọn môn học ( theo buổi học) | Không | Màn hình hiển thị thông tin buổi học hôm đó, có hiển thị QR Code và chức năng lưu ảnh về máy, xem được số lượng sinh viên đã điểm danh, trạng thái điểm danh của từng sinh viên đang có trong lớp |
| **7** | *Kiểm thử trên thiết bị di động:*  Chức năng mở khóa tài khoản sinh viên bị cấm thi do nghỉ quá số buổi quy định | 1. Đăng nhập vào ứng dụng với tài khoản Giảng viên  2. Chọn xem thông tin lớp học phần  3. Kiểm tra danh sách sinh viên của lớp học phần đó  4. Tiến hành sửa trạng thái của sinh viên trong lớp học đó ( sửa lại buổi điểm danh mà sinh viên bị điểm danh muộn / vắng) | Không | Màn hình hiển thị thông báo Cập nhật thành công, vào tài khỏa sinh viên đó sẽ được điểm danh môn học đó |
| **8** | *Kiểm thử trên web*:  Chức năng đăng nhập trang quản lý với tài khoản admin | 1. Vào trang web quản trị viên  2. Đăng nhập với tài khoản + mật khẩu quản trị viên | Tên tài khoản, mật khẩu | Màn hình hiển thị đăng nhập thành công và redirect sang màn quản trị |
| **9** | *Kiểm thử trên web:*  Chức năng xem danh sách sinh viên, giảng viên, phòng học, môn học, ... | 1. Vào trang web quản trị viên  2. Chọn đề mục thông tin tương ứng để xem danh sách | Không | Màn hình hiển thị danh sách các thông tin tương ứng với yêu cầu của người quản trị.  vd: Chọn Sinh viên sẽ hiển thị danh sách sinh viên... |
| **10** | *Kiểm thử trên web:*  Chức năng thêm sinh viên bằng import file CSV hợp lệ | 1. Vào trang web quản trị viên  2. Vào mục Sinh viên  3. Chọn button Thêm, Thêm với file csv  4. Tài file mẫu, điền thông tin vào file mẫu.  5. Đính kèm file mẫu vào form và nhấn button Xác nhận | File csv danh sách sinh viên | Màn hình thông báo thêm xx sinh viên thành công ( xx tương ứng với số lượng sinh viên đã thêm trong danh sách ) |
| **11** | *Kiểm thử trên web:*  Chức năng xóa thông tin thiết bị của người dùng ( trong trường hợp người dùng không sử dụng thiết bị cũ được nữa) | 1. Vào trang web quản trị viên  2. Vào mục Danh sách sinh viên  3. Nhập thông tin sinh viên vào ô tìm kiếm (mã sinh viên, tên sinh viên, ..)  4. Ấn buttonSửa  5. Thực hiện xóa thông tin Mã thiết bị  6. Ấn button Lưu | Mã sinh viên cần chỉnh sửa | Màn hình hiển thị thông báo thành công, người dùng có thể đăng nhập với thiết bị đó |
| **12** | *Kiểm thử trên server*  Kiểm tra tính bảo mật của server khi có client gọi API | 1. Sử dụng browser bất kì, chưa đăng nhập vào hệ thống.  2. Thực hiện gọi API trực tiếp trên thanh URL | Không | Không có dữ liệu trả về, trả về mã lỗi (do không có token đính kèm theo header của https requests) |
| **13** | *Kiểm thử trên server*  Chức năng gửi thông báo đến sinh viên khi lịch học thay đổi | 1. Vào trang web quản trị viên  2. Vào chi tiết lớp học  3. Ấn chỉnh sửa lịch học  4. Ấn Lưu | Không | Màn hình danh sách thông báo của các sinh viên trong lớp đó hiển có hiển thị thông báo lịch học thay đổi |

Bảng . Bảng kế hoạch kiểm thử

## 5.3 Kết quả kiểm thử

* Tất cả các ca kiểm thử đúng với kế hoạch đề ra
* Tất cả các ca kiểm thử cho kết quả ổn định (thành công hoặc không thành công ) sau các lần tiến hành thực hiện
* Tổng số ca kiểm thử: 13
* Số ca kiểm thử pass: 13

**KẾT LUẬN**

*Kết quả đạt được*

Sau 3 tháng tìm hiểu, khảo sát quy trình nghiệp vụ, nghiên cứu, tìm hiểu và xây dựng ứng dụng điểm danh sinh viên bằng QR Code có tích hợp GPS trên nền tảng iOS, em đạt được một số kết quả sau:

* Nghiên cứu phương pháp và các vấn đề liên quan để xây dựng ứng dụng trên nền tảng iOS, cách xây dựng website bằng ReactJS và server bằng NodeJS
* Xây dựng được một hệ thống End-To-End từ khâu phân tích thiết kế hệ thống thông tin, phát triển. kiểm thử và triển khai lên môi trường thực tế. Hệ thống đã cơ bản thực hiện được các chức năng yêu cầu của một hệ thống quản lý điểm danh sinh viên, có chức năng thông kê, bảo mật, xác minh người dùng.
* Phân tích được các quy trình hoạt động và các chức năng của hệ thống biểu diễn qua các biểu đồ UML và xây dựng các lớp chi tiết của hệ thống.

*Hướng phát triển*

Do thời gian có hạn, kinh nghiệm và vốn hiểu biết của em còn nhiều hạn chế, thiếu kiến thức thực tế nên đồ án còn nhiều thiết sót. Trong tương lai, cần có một hướng phát triển mới để để tài ngày càng hoàn thiện hơn:

* Mong muốn ứng dụng từng bước trở thành một ứng dụng dành cho sinh viên, không chỉ có chức năng điểm danh mà còn các chức năng khác như: xem điểm học tập, lịch học, xem các thông tin từ nhà trường, các dịch vụ, xử lý giấy tờ để bớt phần nào công tác quản lý sinh viên đơn giản hơn theo phương hướng chuyển đổi số đang rất phát triển hiện nay.
* Tiếp tục phát triển ứng dụng di động trên môi trường Android, vì hiện nay thị phần sử dụng điện thoại di động HĐH Android vẫn cực kì lớn.
* Chính vì vậy em mong cô nhận xét và hướng dẫn để để tài của em được hoàn thiện và tốt hơn.

**TÀI LIỆU THAM KHẢO**

[1] “Công tác quản lý điểm danh sinh viên ”

https://tech-buzz.net/top-5-phan-mem-quan-ly-diem-danh/

[2] "Các đặc trưng trong công tác quản lý điểm danh sinh viên” https://tapchigiaoduc.moet.gov.vn/vi/magazine/so-dac-biet-thang-7-383/23-doi-moi-cong-tac-quan-li-sinh-vien-truong-dai-hoc-lao-dong-xa-hoi-thuc-trang-va-giai-phap-6865.html

[3] “Các bài toán điểm danh nhân sự công ty, doanh nghiệp”

https://diendandoanhnghiep.vn/giai-bai-toan-nhan-su-khi-doanh-nghiep-tro-lai-duong-dua-so-221894.html

[4] "iOS là gì, lập trình ios từ cơ bản đến nâng cao ?” https://codegym.vn/blog/2021/03/26/6-tai-lieu-tu-hoc-lap-trinh-ios-co-ban-nang-cao-hay-nhat/

[5] “Hướng dẫn lập trình web cơ bản bằng ReactJS có sử dụng server bằng NodeJS”

https://vntalking.com/xay-dung-ung-dung-web-voi-nodejs-expressjs.html

[6] "Hướng dẫn triển khai ứng dụng lên host bằng heroku”

https://topdev.vn/blog/heroku-la-gi/

Source code đồ án

https://github.com/thiemnc-datn