

**CHƯƠNG TRÌNH HỖ TRỢ ĐÀO TẠO SINH VIÊN**

GVHD: TS. Tôn Long Phước

|  |  |  |
| --- | --- | --- |
| STT | MSSV | Họ tên |
| 1 | 19504781 | Nguyễn Việt Anh |
| 2 | 19507391 | Phạm Nguyễn Văn Trường |

Danh sách thành viên

tp hỒ CHÍ MINH, THÁNG 5 NĂM 2023

Khóa luận tốt nghiệp

Bộ môn Kỹ thuật phần mềm – Khoa Công nghệ thông tin – ĐH Công nghiệp TP. HCM

**PROGRAM SUPPORT TRAINNING STUDENTS OF INDUSTRIAL UNIVERSITY OF HO CHI MINH CITY**

**ABSTRACT:**

**METHODS:**

LỜI CẢM ƠN

Để hoàn thành khóa luận này, chúng tôi xin tỏ lòng biết ơn sâu sắc đến thầy Tôn Long Phước đã tận tình hướng dẫn trong suốt quá trình làm đồ án tốt nghiệp và viết báo cáo.

Chúng tôi chân thành cảm ơn quý thầy, cô trong khoa công nghệ thông tin, Trường Đại Học Công Nghiệp thành phố Hồ Chí Minh đã tận tình truyền đạt kiến thức trong những năm chúng tôi học tập. Với vốn kiến thức được tiếp thu trong quá trình học không chỉ là nền tảng cho quá trình nghiên cứu khóa luận mà còn là hành trang quý báu để chúng tôi bước vào đời một cách vững chắc và tự tin.

Chúng tôi chân thành cảm ơn cô Nguyễn Thị Hạnh đã cho phép và tạo điều kiện thuận lợi để chúng tôi đăng ký tiếp nhận đề tài.

Cuối cùng chúng tôi kính chúc quý thầy, cô dồi dào sức khỏe và thành công trong sự nghiệp cao quý. Đồng kính chúc các cô, chú, anh, chị trong khoa, ngành công nghệ thông tin luôn dồi dào sức khỏe, đạt được nhiều thành công tốt đẹp trong công việc.

Tháng 5 – 2021

Phạm Nguyễn Văn Trường – Nguyễn Việt Anh

# NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

|  |
| --- |
| *Ngày ....... tháng ........ năm .........*  **GIẢNG VIÊN HƯỚNG DẪN**  *(Ký và ghi rõ họ tên)* |

# NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

……………………………………………………………………………………………………….

|  |
| --- |
| *Ngày ....... tháng ........ năm .........*  **GIẢNG VIÊN PHẢN BIỆN**  *(Ký và ghi rõ họ tên)* |

**NHẬT KÝ LÀM VIỆC**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tuần** | **Từ ngày** | **Đến ngày** | **Nội dung** |
| 1 | 27/12/2023 | 03/01/2023 | Lập nhóm, lập kênh giao tiếp, tìm hiểu nghiệp vụ đề tài. Lựa chọn các công nghệ cần thiết cho hệ thống. |
| 2 | 03/01/2023 | 10/01/2023 | Thiết kế các use case và sơ đồ cơ sở dữ liệu. Tạo dự án mẫu, triển khai git. Xây dựng Model ở phía backend. |
| 3 | 10/01/2023 | 17/01/2023 | Thực thi API đăng nhập, xác thực người dùng bằng JWT. Áp dụng Redis tăng tốc độ thực hiện nhiều request cùng lúc. Thực thi một số API quản lý của admin. |
| 4 | 24/01/2023 | 31/01/2023 | Tạo UI tổng quan dành cho sinh viên. |
| 5 | 07/02/2023 | 14/02/2023 | Sửa lỗi cors policy khi đăng nhập ở phía frontend. Họp bàn, sửa đôi chút về sơ đồ cơ sở dữ liệu. Hoàn thiện Model tổng ở backend. Thực thi API đăng ký học phần, danh sách lớp học phần và những môn đã đăng ký theo học kỳ. |
| 6 | 14/02/2023 | 21/02/2023 | Sửa lỗi API đăng ký học phần, bổ sung các API: chi tiết lớp học phần, lịch học sinh viên. Tạo dự án mẫu cho Mobile App. Tạo UI tổng quan cho giáo vụ ( admin ). |
| 7 | 21/02/2023 | 28/02/2023 | Thực thi API thanh toán trực tuyến, test API Momo. Thiết kế giao diện tổng quan dành cho sinh viên ở phía Mobile. |
| 8 | 28/02/2023 | 07/03/2023 | Hoàn thiện chức năng thanh toán trực tuyến. Hoàn thiện giao diện Admin ( giáo vụ ). |
| 9 | 07/03/2023 | 14/03/2023 | Backend: tạo các api liên quan đến quản lý ( giáo vụ ). Hoàn thiện chức năng thanh toán học phí cho sinh viên ở phía giáo vụ và xuất phiếu. Bổ sung vài chức năng in ở sinh viên như: in lịch, in phiếu thu,… |
| 10 | 14/03/2023 | 21/03/2023 | Tạo giao diện tổng quan cho giảng viên. Thực hiện các chức năng xem lịch dạy, import, export excel. Tiếp tục hoàn thiện các chức năng, giao diện bên Mobile. |
| 11 | 21/03/2023 | 28/03/2023 | Thực thi tạo hạ tầng trên Amazon Web Service. Fix lỗi và tiếp tục hoàn thiện lại chức năng import, export. Tiếp tục hoàn thiện các chức năng, giao diện bên Mobile. |
| 12 | 28/03/2023 | 04/04/2023 | Viết báo cáo đồ án, bổ sung một vài chức năng phụ như kiểm tra trùng lịch, xem điểm. Bổ sung các chức năng quản lý ở phía giáo vụ. Tiến hành nhập đa dạng dữ liệu. Thực thi tạo hạ tầng trên Amazon Web Service. |
| 13 | 04/04/2023 | 11/04/2023 | Viết báo cáo đồ án, hoàn thiện Abstract. Thực thi tạo hạ tầng trên Amazon Web Service. Tiếp tục hoàn thiện chức năng phía Mobile. |
| 14 | 11/04/2023 | 18/04/2023 |  |
| 15 | 18/04/2023 | 25/04/2023 |  |
| 16 |  |  |  |
| 17 |  |  |  |

**MỤC LỤC**

[NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN 3](#_Toc132720888)

[NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN 4](#_Toc132720889)

[LỜI MỞ ĐẦU 10](#_Toc132720890)

[1. Tổng quan tình hình nghiên cứu thuộc lĩnh vực của đề tài [V.a] 10](#_Toc132720891)

[2. Ý nghĩa khoa học và thực tiễn của đề tài 11](#_Toc132720892)

[CHƯƠNG 1: GIỚI THIỆU 12](#_Toc132720893)

[1.1. Tổng quan [V.a] 12](#_Toc132720894)

[1.2. Mục tiêu đề tài [V.a] 13](#_Toc132720895)

[1.3. Phạm vi đề tài 13](#_Toc132720896)

[1.4. Mô tả yêu cầu chức năng 13](#_Toc132720897)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 14](#_Toc132720898)

[2.1. Javascript 14](#_Toc132720899)

[2.2. Nodejs [V.a] 14](#_Toc132720900)

[2.3. React framework 14](#_Toc132720901)

[2.4. Redis [V.a] 16](#_Toc132720902)

[CHƯƠNG 3: PHÂN TÍCH 18](#_Toc132720903)

[3.1. Sơ đồ Usecase: 18](#_Toc132720904)

[3.1.1. Danh sách các tác nhân và mô tả: 19](#_Toc132720905)

[3.1.2. Danh sách các Usecase 19](#_Toc132720906)

[3.2. Đặc tả các yêu cầu chứng năng: 21](#_Toc132720907)

[3.2.1. UC001 - Xem danh sach chuyen nganh 21](#_Toc132720908)

[3.2.7. UC007 - Them khoa 35](#_Toc132720909)

[3.2.8. UC008 - Them mon hoc 38](#_Toc132720910)

[3.2.9. UC009 - Xem danh sach mon hoc 41](#_Toc132720911)

[3.2.10. UC010 - Xem danh sach hoc phan 43](#_Toc132720912)

[3.2.11. UC011 - Them hoc phan 45](#_Toc132720913)

[3.2.12. UC012 - Xem danh sach lop hoc phan 48](#_Toc132720914)

[3.2.13. UC013 - Them lop hoc phan 50](#_Toc132720915)

[3.3. Lược đồ cơ sở dữ liệu: 54](#_Toc132720916)

[CHƯƠNG 4: THIẾT KẾ VÀ HIỆN THỰC 55](#_Toc132720917)

[4.1. Bài toán 55](#_Toc132720918)

[4.2. Kiến trúc đề xuất 55](#_Toc132720919)

[4.3. Công nghệ sử dụng 55](#_Toc132720920)

[CHƯƠNG 5: KẾT QUẢ ĐẠT ĐƯỢC VÀ HƯỚNG PHÁT TRIỂN 56](#_Toc132720921)

[5.1. Kết quả đạt được: 56](#_Toc132720922)

[5.2. Hạn chế của đồ án: [V.a] 56](#_Toc132720923)

[5.3. Hướng phát triển: 56](#_Toc132720924)

# LỜI MỞ ĐẦU

## 1. Tổng quan tình hình nghiên cứu thuộc lĩnh vực của đề tài

Cloud Compting hay còn gọi là điện toán máy chủ ảo cung cấp các công nghệ, tài nguyên máy tính liên kết với mạng Internet.Với mô hình Cloud Computing,người dùng sẽ được tiếp cận các tài nguyên từ công nghệ , năng lượng điện toán ,lưu trữ cơ sở dữ liệu đến từ nhà cung cấp dịch vụ Cloud Computing thông qua kết nối mạng nột cách dễ dàng, mọi lúc mọi nơi ,theo yêu cầu.Việc sử dụng Cloud Computing của người dùng rất dễ dàng tại vì nó được cung cấp dưới dạng “dịch vụ”,cho phép người dùng sử dụng khởi tạo và truy cập máy chủ ảo thông qua hệ thống quản lý từ nhà cung cấp dịch vụ Cloud Computing mà không phải có nhiều kiến thức , kinh nghiệm về công nghệ đó , cũng như không cần quan tâm đến các cơ sở hạ tầng phục vụ công nghệ đó.Tuy được tạo ra rất sớm từ những năm cuối thế kỉ XX nhưng phải đến đầu những năm thế kỉ XXI Cloud Computing mới thực sự phát triển mạnh mẽ với nhiều nhà cung cấp và mô hình, sự phổ biến của nó chứng minh bằng “Theo Cloudwards,năm 2021 94% các doanh nghiệp trên toàn cầu sử dụng dịch vụ đám mây ở một hình thức hay nhiều hình thức ,trong đó phần lớn là dịnh vụ SaaS(Software as a Service)”. Với việc dân số thế giới ngày càng tăng việc truy cập Internet trên toàn cầu cũng tăng theo việc phát triển một cơ sở hạ tầng vật lí mà không liên tục nâng cấp sẽ rất khó để đáp ứng nhu cầu này, tuy nhiên Cloud Computing có thể đáp ứng nhu cầu này một cách dễ dàng giúp người dùng không cần lo lắng đến cơ sở hạ tầng và tập trung hơn vào công việc của mình. Về chi phí của Cloud Computing khá là rẻ so với tạo ra và vận hành một cơ sở hạ tầng vật lý, đôi khi Cloud Computing có thể tiết kiệm chi phí người dùng bằng cách tắt những service người dùng không sử dụng.

Hiện tại ,trường đại học Công Nghiệp thành phố Hồ Chí Minh đang sử dụng hệ thống hỗ trợ đào tạo sinh viên bằng Data Center,khi lượng truy cập tại một thời điểm tăng nhanh đột biến thì gây quá tải đôi khi là tắc nghẽn khiến những công việc như đăng kí học phần, xem thời khóa biểu, xem điểm,… của sinh viên bị trễ nãi.Song song đó với việc Việt Nam là một nước có tỷ lệ sử dụng smart phone cao (Có đến 63.1% người Việt Nam sở hữu smart phone .Theo số liệu được cung cấp bởi Statista) nên việc phát triển một ứng dụng mobile là điều cần thiết để sinh viên có thể tiện lợi và đa dạng hóa cách truy cập hơn, nhưng trường lại chưa có ứng dụng phục vụ trên mobile tạo nên nhiều bất tiện khi truy cập web hệ thống của trường trên mobile.

Ngoài ra,đề tài phát triển web cho người quản lý có thể quản lý nhiều đối tượng cuả trường như sinh viên,giảng viên, khoa,... và web cho giảng viên để quản lý các hoạt động như xem lịch giảng dạy ,xem lớp và sinh viên đang dạy ,… giúp nhóm đối tượng này có thể sử dụng các dịch vụ này trên hệ thông luôn mà không cần truy cập hệ thống khác.

## 2. Ý nghĩa khoa học và thực tiễn của đề tài

# CHƯƠNG 1: GIỚI THIỆU

## 1.1. Tổng quan

Với sự bùng nổ của cách mạng công nghiệp 4.0,rất nhiều lĩnh vực đã và đang được số hóa trên môi trường kỹ thuật,bao gồm ngành giáo dục.Cụ thể tại trường Đại Học Công Nghiệp Thành phố Hồ Chí Minh, với số lượng sinh viên nghiên cứu và học tập tại đây ngày càng tăng như cầu về một chương trình hỗ trợ học tập cho sinh viên có tính đa dụng, tính ổn định và sử dụng được trên hai môi trường phổ biến hiện nay là Web và Andrioid.Chúng tôi đã xây dựng chương trình hỗ trợ học tập sinh viên IUH với nhưng tiêu chí trên.Đây là một chương trình có tính ứng dụng cao,bám sát được với như cầu về quản lý học tập của sinh viên IUH cũng như chương trình cũng tạo ra để hỗ trợ người quản lý chương trình đào tạo tại trường (sinh viên,giảng viên,lớp học phần,…) và giảng viên quản lý (xem lịch giảng dạy,xem thông tin lớp học ,..) quan trọng là việc đơn giản hóa việc đăng ký học phần giúp giảm thiểu sự tắc nghẽn dữ liệu.Hệ thống cũng giải quyết các vấn đề còn tồn đọng của của hệ thống website của trường Đại Học Công Nghiệp Thànnh phố Hồ Chí Minh hiện tại như việc đăng kí học phần khi trường cố gắng mở đăng kí học phần từ nửa đêm để giảm thiểu việc truy cập đăng kí học phần của sinh viên nhưng hệ thống vẫn quá tải khiến việc đăng kí học phần của sinh viên khó khăn hơn khi phải thức khuya dậy sớm để đăng ký hoặc tệ nhất là không đăng ký được gì và đôi lúc việc truy cập vào website trường có thể không thực hiện được do số lượng truy cập hàng ngày của sinh viên là rất cao. Một trong những nguyên nhân quan trọng là hệ thống hạ tầng vật lý của trường không còn đáp ứng đủ nhu cầu ngày một tăng cao khi lượng sinh viên của trường mỗi năm ngày càng tăng.Việc nâng cấp hệ thống vật lý này rất khó khăn khi có nhiều vấn đề như: Rủi ro có cao không? Chi phí nâng cấp ? Thời gian nâng cấp? ,… rất nhiều vấn đề đằng sau nữa.

Để giải quyết vấn đề này, chúng tôi đề xuất giải pháp là sử dụng điện toán đám mây (Cloud Computing) nhằm lưu trữ , xử lí các nghiệp vụ để giảm tải cho máy chủ của trường. Cloud Computing có rất nhiều nhà cung cấp dịch vụ lớn như Amazone,Microsoft,Google hay trong nước như Viettel,FPT,.. chuyên cung cấp cho các công ty lớn, vừa và nhỏ . Những hệ thống này khi triển khai trên điện toán đám mây sẽ có chi phí rẻ, cấu hình đơn giản nhưng vẫn an toàn, bảo mật tốt, khả năng nâng cấp bảo trì hệ thống dễ dàng. Hơn hết, trong các ưu điểm trên là khả năng cân bằng tải, đó là ưu điểm tuyệt vời của điện toán đám mây [1]. Trong đó, Scale là dịch vụ có khả năng tự động điều chỉnh số lượng máy chủ để đáp ứng nhu cầu truy cập của khách hàng (tăng khi yêu cầu nhiều và giảm khi yêu cầu ít). Do đó, nhà phát triển hệ thống không còn phải đầu tư quá nhiều chi phí chỉ để giải quyết nhu cầu cho việc sử dụng tài nguyên của hệ thống trong một quãng thời gian ngắn trong năm. Đó cũng là lý do chúng tôi chọn giải pháp này cho bài toán xây dựng hệ thống hỗ trợ đào tạo cho sinh viên.

Để triển khai giải pháp trên, chúng tôi đã triển khai hệ thống web với kiến trúc micro-services. Micro-service là một kỹ thuật phát triển phần mềm, một biến thể của kiến trúc hướng dịch vụ (SOA), trong đó cấu trúc một ứng dụng như một tập hợp các dịch vụ được ghép với nhau [2]. Trong kiến trúc micro-service, các dịch vụ được xử lý tốt và các giao thức rất nhẹ. Lợi ích của việc phân tách một ứng dụng thành các dịch vụ nhỏ hơn là nó cải thiện tính mô đun. Điều này làm cho ứng dụng trở nên gọn nhẹ, dễ hiểu, dễ phát triển và trở nên linh hoạt hơn. Cụ thể, chúng tôi chia hệ thống làm hai phần chính: Front-end và Back-end. Front-end của chúng tôi ở đây chính là ứng dụng di động cho sinh viên, ứng dụng website cho Giáo vụ và sinh viên của nhà trường. Front-end sẽ đảm nhận trách nhiệm nhận dữ liệu, hiển thị dữ liệu cũng như các tương tác cho người dùng. Back-end chúng tôi xây dựng hai API cho hai nền tảng là Web và Android, hệ thống máy sẽ xử lý, lấy dữ liệu từ hệ cơ sở dữ liệu MySQL và trả về dưới định dạng JSON. Ngoài ra, khả năng phát triển song song và tương đối độc lập giữa Front-end và Back-end cũng là ưu điểm cho việc bảo trì và nâng cấp.

Nhằm hỗ trợ cho các giải pháp trên chúng tôi sử dụng Docker một công nghệ cho phép tạo các máy ảo, lập trình viên chỉ cần khai báo đầy đủ những thông tin cấu hình cần thiết qua Dockerfile và docker-compose để Docker cài đặt và chạy chương trình mà không cần phải cài nhiều quá nhiều thứ để chạy chương trình trên máy chủ mà chỉ cần cài đặt Docker .

## 1.2. Mục tiêu đề tài

## 1.3. Phạm vi đề tài

## 1.4. Mô tả yêu cầu chức năng

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## 2.1. Javascript

JavaScript là ngôn ngữ lập trình website phổ biến hiện nay, có rất nhiều framework ra đời được viết bằng ngôn ngữ lập trình Javascript, từ frontend cho tới backend. Các framework thông dụng được xây dựng dựa trên Javascript như: Reactnative - thư viện dùng cho ứng dụng mobile. Node.js - dùng để xây dựng và phát triển ứng dụng realtime từ phía máy chủ. Angular - Dùng để xây dựng ứng dụng Single Page Javascript có cho mình rất nhiều ưu điểm nổi bật có thể kể đến như: JS có thể hoạt động ở trên nhiều nền tảng và các trình duyệt web khác nhau, được đánh giá là một loại ngôn ngữ lập trình nhẹ và nhanh hơn nhiều so với các ngôn ngữ lập trình khác. Giúp thao tác với người dùng phía Client và tách biệt giữa các Client với nhau.



Hình 2.1 Những công nghệ ra đời dựa trên Javascript

## 2.2. Nodejs

## 2.3. React framework

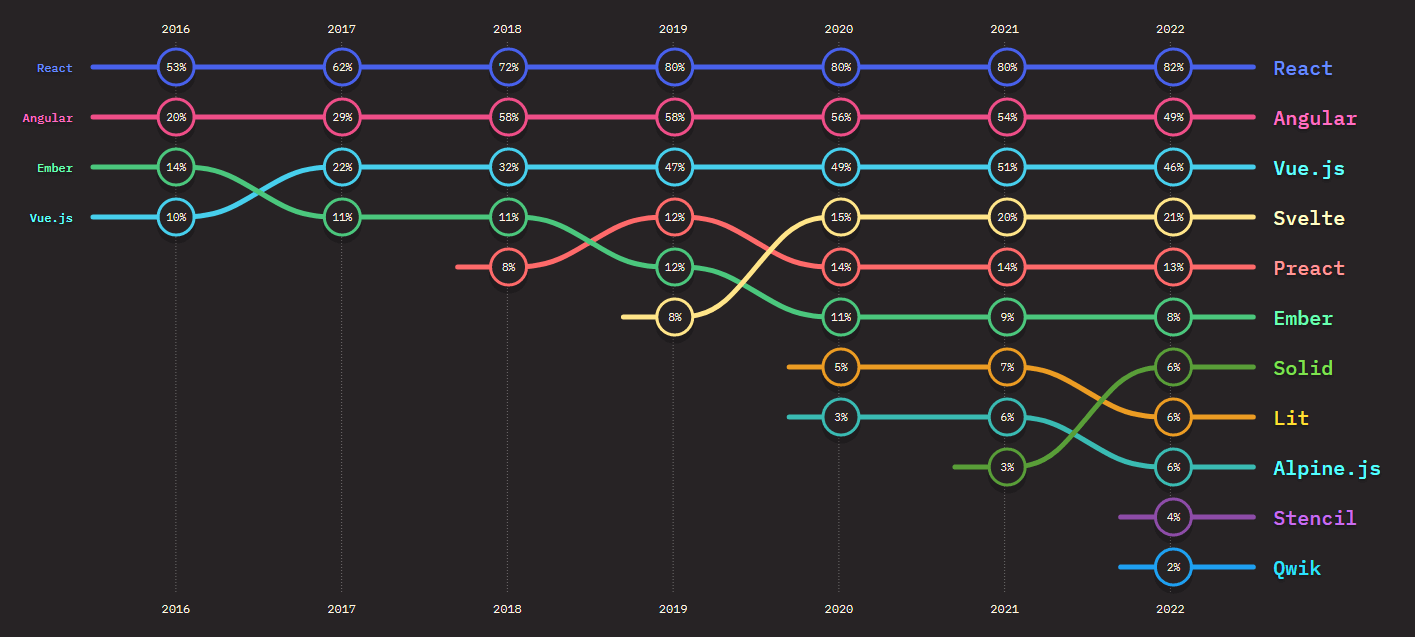
**2.3.1. React Native**

React Native là một framework do công ty công nghệ nổi tiếng Facebook phát triển nhằm mục đích giải quyết bài toán hiệu năng của Hybrid và bài toán chi phí khi mà phải viết nhiều loại ngôn ngữ native cho từng nền tảng di động. React Native giúp lập trình viên chỉ cần phát triển ứng dụng trên một mã nguồn duy nhất mà vẫn có thể tương thích với cả hai hệ điều hành Android và iOS (thậm chí là cả các hệ điều hành cho Ipad và và các dòng tivi). Sau đây là những lợi ích mà nó đem lại cho việc triển khai dự án và giúp chúng tôi có thể trả lời cho câu hỏi có nên dùng React Native không ? Tiết kiệm được nhiều thời gian khi muốn phát triển một ứng dụng nhanh chóng nhưng vẫn đảm bảo hiệu năng tương đối ổn định. Quy mô đội ngũ phát triển nhỏ nhưng vẫn dễ dàng bảo trì, quản lý. Ứng dụng tin cậy và ổn định. Xây dựng cho nhiều hệ điều hành khác nhau với ít native code nhất ( Native App là các ứng dụng được phát triển và build trên chính những công cụ hỗ trợ mà nhà phát triển iOS và Android cung cấp cho người lập trình viên ở đây là XCode và Android ). Trải nghiệm người dùng tốt hơn là hybrid app.

**2.3.2. Reactjs**

Các ứng dụng đơn trang – SPA ( Single Page App ) ngày càng trở nên phổ biến với các framework SPA thế hệ đầu tiên được biết đến như Angular (của Google), Ember, Knockout và Backbone. Sử dụng các framework này làm cho việc xây dựng các ứng dụng web nâng cao một cách dễ dàng hơn là sử dụng thuần JavaScript hay jQuery. React, một giải pháp khác cho các SPA, đã được Facebook cho ra đời vào cuối năm 2013 – đây là một công nghệ giúp tận dụng Javascript một cách triệt để, khi toàn bộ ứng dụng web sẽ được build chỉ dựa trên Javascript và chỉ Javascript. React, cũng như các giải pháp SPA khác, đều mang lại các ưu điểm nổi trội như: giúp lập trình viên dễ dàng quản lý dự án hơn khi tách biệt được 2 phần frontend và backend. Tiết kiệm băng thông cho server, do các dữ liệu có thể chuyển đổi sẽ được thông qua server và client, còn dữ liệu tĩnh chỉ cần được render 1 lần khi người dùng lần đầu vào trang web. Code phía server có thể được tận dụng để dùng cho mobile app. Lập trình viên có thể tái sử dụng tại dữ liệu vì website vẫn có thể cache dữ liệu như bình thường. Đối với người dùng: Người dùng có trải nghiệm mượt mà khi dùng web, không cần mất thời gian đợi trang web tải lại toàn bộ dữ liệu như thông thường.

Vì sao chúng tôi lại lựa chọn Reactjs cho ứng dụng web ? React là một framework đơn giản, dễ học và dễ bắt đầu bởi chúng ta có thể tận dụng Javascript mà không phải là bất cứ ngôn ngữ nào xa lạ. Reactjs giúp xây dựng giao diện người dùng phong phú. React đi kèm với JSX, một phần mở rộng cú pháp tùy chọn, giúp bạn có thể viết các thành phần của riêng mình. Các thành phần này về cơ bản chấp nhận trích dẫn HTML và cũng làm cho tất cả các thành phần con hiển thị trải nghiệm thú vị hơn cho các nhà phát triển. Khi đang xây dựng một ứng dụng phức tạp, tải cao, bắt buộc phải xác định cấu trúc của ứng dụng ngay từ đầu vì nó có thể ảnh hưởng đến hiệu suất của ứng dụng của bạn. Mô hình DOM có cấu trúc dạng cây. Vì vậy, một sửa đổi nhỏ ở lớp cấp cao hơn có thể ảnh hưởng đáng kể đến giao diện người dùng của ứng dụng. Để giải quyết vấn đề này, Facebook đã giới thiệu tính năng DOM ảo. Virtual DOM, như cái tên đã cho thấy đây là đại diện ảo của DOM cho phép thử nghiệm tất cả các thay đổi đối với DOM ảo trước tiên để tính toán rủi ro với mỗi sửa đổi. Do đó, cách tiếp cận này giúp duy trì hiệu suất ứng dụng cao và đảm bảo trải nghiệm người dùng tốt hơn.

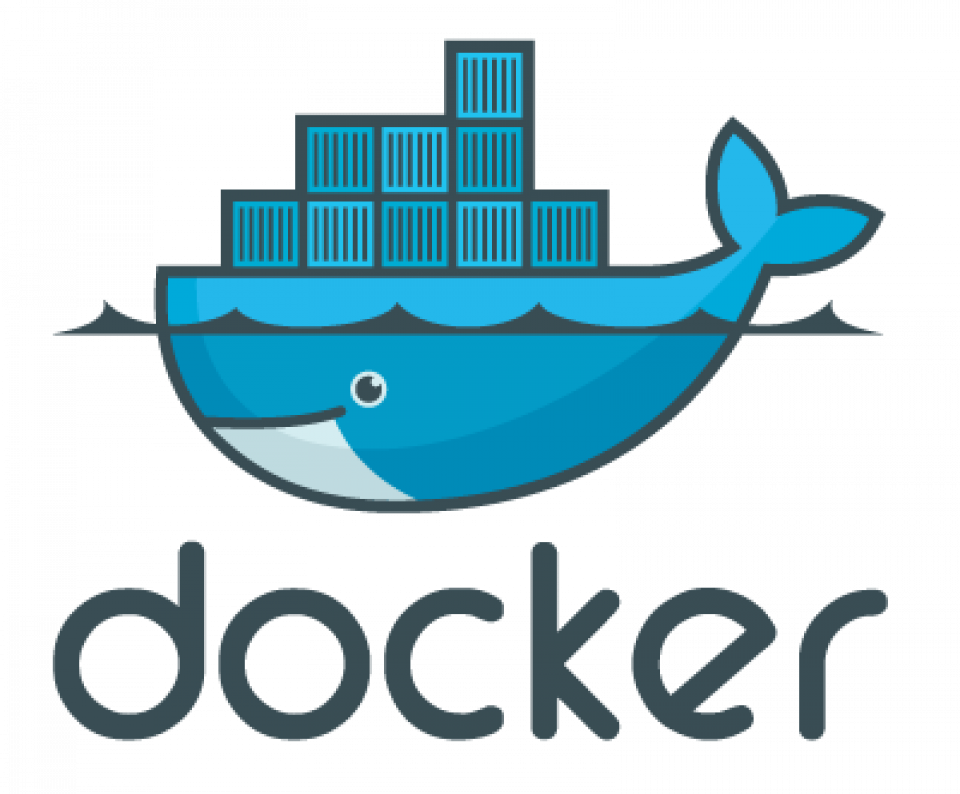


Hình 2.3.2. Biểu đồ phát triển của các Frontend Framework

## 2.4. Redis

## 2.5. Docker

Docker là một nền tảng cung cấp cách để running, building và deloying một cách dễ dàng bằng cách sử dụng các container (trên nền tảng ảo hóa).Docker cung cấp một cách để đóng gói các container và chuyển giao chúng giữa các môi trường khác nhau bao gồm môi trường phát triển và triển khai sản phẩm.



Việc sử dụng Docker giúp cho việc phát triển và triển khai ứng dụng trở nên đơn giản hơn , nhanh chóng và đáng tin cậy hơn. Nó cũng giúp các lập trình viên tiết kiệm thời gian và chi phí trong việc cấu hình các môi trường phát triển và triển khai . Chúng tôi sử dụng docker vào việc triển khai cấu trúc micro-services, đồng thời scale một cách linh hoạt.

# CHƯƠNG 3: PHÂN TÍCH

## 3.1. Sơ đồ Usecase:

**Diagram, schematic

Description automatically generated**

Hình 3.1. Lược đồ usecase của hệ thống AVTEdu

### 3.1.1. Danh sách các tác nhân và mô tả:

|  |  |  |
| --- | --- | --- |
| **Tác nhân** | **Mô tả tác nhân** | **Ghi chú** |
| Sinh viên | Sinh viên là những người đang học tại trường ĐHCN TP.HCM. |  |
| Giảng viên | Giảng viên là những thầy, cô đang dạy, đào tạo cho các sinh viên tại trường ĐHCN TP.HCM |  |
| Giáo vụ | Là những người phụ trách các nghiệp vụ quản lý: thêm, xóa, sửa, nhập thông tin sinh viên, khoa, môn học, học phần,… |  |

### 3.1.2. Danh sách các Usecase

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID** | **Tên Use case** | **Mô tả ngắn gọn Use case** | **Chức năng** | **Ghi chú** |
| UC001 | Xem danh sach chuyen nganh | Cho phép Giáo vụ xem danh sách chuyên ngành trong cơ sở dữ liệu | Xem danh sách chuyên ngành |  |
| UC002 | Them hoc phan cho chuyen nganh | Cho phép Giáo vụ thêm một học phần vào chương trình học của chuyên ngành trong cơ sở dữ liệu | Thêm một học phần vào chương trình học của chuyên ngành |  |
| UC003 | Xem danh sach sinh vien | Cho phép Giáo vụ xem danh sách sinh viên trong cơ sở dữ liệu | Xem danh sách sinh viên |  |
| UC004 | Them sinh vien | Cho phép Giáo vụ thêm 1 sinh viên nào đó vào cơ sở dữ liệu | Thêm sinh viên |  |
| UC005 | Chinh sua thong tin sinh vien | Cho phép Giáo vụ chỉnh sửa thông tin 1 sinh viên nào đó trong cơ sở dữ liệu | Chỉnh sửa thông tin sinh viên |  |
| UC006 | Xem danh sach khoa | Cho phép Giáo vụ xem danh sách khoa khỏi cơ sở dữ liệu | Xem danh sách khoa |  |
| UC007 | Them khoa | Cho phép Giáo vụ thêm 1 khoa nào đó vào cơ sở dữ liệu | Thêm chuyên ngành |  |
| UC008 | Them mon hoc | Cho phép Giáo vụ thêm 1 môn học nào đó vào cơ sở dữ liệu | Thêm môn học |  |
| UC009 | Xem danh sach mon hoc | Cho phép Giáo vụ xem danh sách môn học hiện có trong cơ sở dữ liệu | Xem danh sách môn học |  |
| UC010 | Xem danh sach hoc phan | Cho phép Giáo vụ xem danh sách học phần hiện có trong cơ sở dữ liệu | Xem danh sách học phần |  |
| UC011 | Them hoc phan | Cho phép Giáo vụ thêm mới 1 học phần nào đó vào cơ sở dữ liệu | Thêm học phần |  |
| UC012 | Xem danh sach lop hoc phan | Cho phép Giáo vụ xem danh sách lớp học phần hiện có trong cơ sở dữ liệu | Xem danh sách lớp học phần |  |
| UC013 | Them lop hoc phan | Cho phép Giáo vụ xóa 1 lớp học phần nào đó khỏi cơ sở dữ liệu | Thêm lớp học phần |  |
| UC014 | Dang nhap | Cho phép sinh viên,quản trị viên, giảng viên đăng nhập để thực hiện những chức năng của mình | Đăng nhập |  |
| UC015 | Xem danh sach hoc phan trong hoc ki | Cho phép sinh viên xem danh sách các học phần sẽ mở lớp và đã mở lớp trong học kì | Xem danh sách học phần theo học kì |  |
| UC016 | Xem danh sach lop hoc phan trong hoc ki | Cho phép sinh viên xem danh sách các lớp học phần đang mở trong kì | Xem danh sách lớp học phần trong học kì |  |
| UC017 | Xem thong tin chi tiet lop hoc | Cho phép sinh viên xem thông tin chi tiết của lớp học phần (các buổi học, lịch học, giảng viên,…) | Xem thông tin chi tiết lớp học |  |
| UC018 | Kiem tra trung lich hoc | Cho phép sinh viên kiểm tra xem lịch học của 1 lớp học phần nào đó có bị trùng với lịch đã đăng ký hay không | Kiểm tra trùng lịch học |  |
| UC019 | Xem danh sach lop da dang ky trong hoc ki | Cho phép sinh viên xem danh sách các lớp đã đăng ký trong học kì | Xem danh sách lớp đã đăng ký trong học kì |  |
| UC020 | Dang ky lop hoc phan | Cho phép sinh viên đăng ký lớp học phần | Đăng ký lớp học phần |  |
| UC021 | Xem lich hoc theo tuan | Cho phép sinh viên xem lịch học theo tuần | Xem lịch học theo tuần |  |
| UC022 | Xem ket qua hoc tap | Cho phép sinh viên xem kết quả học tập | Xem kết quả học tập |  |
| UC023 | Thanh toan truc tuyen | Cho phép sinh viên thanh toán học phí hình thức online | Thanh toán trực tuyến |  |
| UC024 | Xem phieu thu tong hop | Cho phép sinh viên xem thông tin về phiếu thu học phí | Xem phiếu thu tổng hợp |  |

## 3.2. Đặc tả các yêu cầu chứng năng:

### 3.2.1. UC001 - Xem danh sach chuyen nganh

**3.2.1.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach chuyen nganh | | UC001 |
| Mục đích: | Xem danh sách chuyên ngành | | |
| Mô tả: | Cho phép Giáo vụ xem danh sách chuyên ngành trong cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công. | | |
| Điều kiện sau: | Hệ thống hiển thị danh sách chuyên ngành hiện có lên giao diện | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút quản lý chuyên ngành | |  | |
|  | | 2. Tải dữ liệu chuyên ngành thành công | |
|  | | 3. Hiển thị bảng danh sách thông tin các chuyên ngành đang có trong cơ sở dữ liệu | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 2.1 Tải dữ liệu chuyên ngành thất bại | |
|  | | 2.2. Thông báo tải dữ liệu thất bại | |

**3.2.1.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.1.3. Sequence Diagram:**

### 3.2.2. UC002 - Them hoc phan cho chuyen nganh

**3.2.2.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Them hoc phan cho chuyen nganh | | UC002 |
| Mục đích: | Thêm một học phần vào chương trình học của chuyên ngành. | | |
| Mô tả: | Cho phép Giáo vụ thêm mới 1 học phần vào chương trình học của chuyên ngành trong cơ sở dữ liệu. | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công, người Giáo vụ đang ở trang quản lý chuyên ngành, chuyên ngành và học phần cần thêm đã tồn tại trong cơ sở dữ liệu. | | |
| Điều kiện sau: | Học phần được thêm mới thành công vào chương trình học của chuyên ngành trong cơ sở dữ liệu và thông báo thành công.  Trường hợp thêm học phần vào chương trình của chuyên ngành thất bại thì thông báo rõ lý do. | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn chuyên ngành muốn thêm vào | |  | |
| 2. Bấm nút thêm | |  | |
|  | | 3. Kiểm tra học phần mới thêm vào hợp lệ | |
|  | | 4. Thêm học phần vào chương trình của chuyên ngành trong cơ sở dữ liệu | |
|  | | 5. Thông báo thêm thành công | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 3.1 Kiểm tra học phần mới không hợp lệ để thêm vào (đã thêm vào trước đó, mã học phần không tồn tại) | |
|  | | 3.2 Thông báo thêm mới thất bại và lý do | |

**3.2.2.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.2.3. Sequence Diagram:**

### 3.2.3. UC003 - Xem danh sach sinh vien

**3.2.3.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach sinh vien | | UC003 |
| Mục đích: | Xem danh sach sinh vien | | |
| Mô tả: | Cho phép Giáo vụ xem danh sách sinh viên trong cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công. | | |
| Điều kiện sau: | Hệ thống hiển thị danh sách sinh viên hiện có lên giao diện | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút quản lý sinh viên | |  | |
|  | | 2. Tải dữ liệu sinh viên thành công | |
|  | | 3. Hiển thị bảng danh sách thông tin các sinh viên đang có trong cơ sở dữ liệu | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 2.1 Tải dữ liệu sinh viên thất bại | |
|  | | 2.2. Thông báo tải dữ liệu thất bại | |

**3.2.3.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.3.3. Sequence Diagram:**

### 3.2.4. UC004 - Them sinh vien

**3.2.4.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Them sinh vien | | UC004 |
| Mục đích: | Thêm sinh viên | | |
| Mô tả: | Cho phép Giáo vụ thêm 1 sinh viên nào đó vào cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công, người Giáo vụ đang ở trang quản lý sinh viên. | | |
| Điều kiện sau: | Sinh viên mới được thêm vào cơ sở dữ liệu và thông báo thành công.  Thông báo khi có lỗi xảy ra. | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút thêm sinh viên | |  | |
|  | | 2. Hiển thị form nhập sinh viên mới | |
| 3. Nhập dữ liệu vào các ô thông tin | |  | |
| 4. Bấm lưu | |  | |
|  | | 5. Xác nhận dữ liệu nhập là hợp lệ | |
|  | | 6. Thêm thành công trong cơ sở dữ liệu | |
|  | | 7. Thông báo thêm thành công | |
|  | | 8. Cập nhật thêm sinh viên mới trên bảng danh sách | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
| 4.1. Bấm hủy | |  | |
|  | | 4.2. Xóa hàng mới ra khỏi bảng sinh viên | |
|  | | 4.3. Quay lại giao diện ban đầu của trang quản lý sinh viên | |
| Luồng sự kiện 2: | | | |
| Tác nhân | | Hệ thống | |
|  | | 5.1 Xác nhận dữ liệu không hợp lệ | |
|  | | 5.2 Hiển thị các lỗi tại các trường nhập dữ liệu chưa hợp lệ | |
| Luồng sự kiện 3: | | | |
| Tác nhân | | Hệ thống | |
|  | | 6.1. Thêm thất bại | |
|  | | 6.2. Thông báo thêm thất bại và lý do | |

**3.2.4.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.4.3. Sequence Diagram:**

### 3.2.5. UC005 - Chinh sua thong tin sinh vien

**3.2.5.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Chinh sua thong tin sinh vien | | UC005 |
| Mục đích: | Chỉnh sửa thông tin sinh viên | | |
| Mô tả: | Cho phép Giáo vụ chỉnh sửa thông tin 1 sinh viên nào đó trong cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công, người Giáo vụ đang ở trang quản lý sinh viên, đối tượng cần chỉnh sửa đã tồn tại trong cơ sở dữ liệu. | | |
| Điều kiện sau: | Sinh viên đã chọn được cập nhật thông tin trong cơ sở dữ liệu và thông báo thành công.  Trường hợp cập nhật thất bại thì thông báo rõ lý do. | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút chỉnh sửa tại hàng cần chỉnh sửa | |  | |
|  | | 2. Chuyển các ô dữ liệu sang trạng thái cho phép nhập liệu | |
| 3. Nhập dữ liệu vào các ô thông tin | |  | |
| 4. Bấm lưu | |  | |
|  | | 5. Xác nhận dữ liệu nhập là hợp lệ | |
|  | | 6. Chỉnh sửa thành công trong cơ sở dữ liệu | |
|  | | 7. Thông báo chỉnh sửa thành công | |
|  | | 8. Cập nhật thông tin mới trên bảng danh sách | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
| 4.1. Bấm hủy | |  | |
|  | | 4.2. Chuyển các ô về lại trạng thái không cho phép nhập thông tin | |
|  | | 4.3. Quay lại giao diện ban đầu của trang quản lý sinh viên | |
| Luồng sự kiện 2: | | | |
| Tác nhân | | Hệ thống | |
|  | | 5.1 Xác nhận dữ liệu không hợp lệ | |
|  | | 5.2 Hiển thị các lỗi tại các trường nhập dữ liệu chưa hợp lệ | |
| Luồng sự kiện 3: | | | |
| Tác nhân | | Hệ thống | |
|  | | 6.1. Chỉnh sửa thất bại | |
|  | | 6.2. Thông báo chỉnh sửa thất bại và lý do | |

**3.2.5.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.5.3. Sequence Diagram:**

### 3.2.6. UC006 - Xem danh sach khoa

**3.2.6.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach khoa | | UC006 |
| Mục đích: | Xem danh sach khoa | | |
| Mô tả: | Cho phép Giáo vụ xem danh sách khoa trong cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công. | | |
| Điều kiện sau: | Hệ thống hiển thị danh sách khoa hiện có lên giao diện | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút quản lý khoa | |  | |
|  | | 2. Tải dữ liệu khoa thành công | |
|  | | 3. Hiển thị bảng danh sách thông tin các khoa đang có trong cơ sở dữ liệu | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 2.1 Tải dữ liệu khoa thất bại | |
|  | | 2.2. Thông báo tải dữ liệu thất bại | |

**3.2.6.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.6.3. Sequence Diagram:**

### 3.2.7. UC007 - Them khoa

**3.2.7.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Them khoa | | UC007 |
| Mục đích: | Thêm khoa | | |
| Mô tả: | Cho phép Giáo vụ thêm 1 khoa nào đó vào cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công, người Giáo vụ đang ở trang quản lý khoa. | | |
| Điều kiện sau: | Khoa mới được thêm vào cơ sở dữ liệu và thông báo thành công. Thông báo khi có lỗi xảy ra, | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút thêm khoa | |  | |
|  | | 2. Hiển thị form thêm mới khoa | |
| 3. Nhập dữ liệu vào các ô thông tin | |  | |
| 4. Bấm lưu | |  | |
|  | | 5. Xác nhận dữ liệu nhập là hợp lệ | |
|  | | 6. Thêm thành công trong cơ sở dữ liệu | |
|  | | 7. Thông báo thêm thành công | |
|  | | 8. Cập nhật thêm khoa mới trên bảng danh sách | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
| 4.1. Bấm hủy | |  | |
|  | | 4.2. Xóa hàng mới ra khỏi bảng khoa | |
|  | | 4.3. Quay lại giao diện ban đầu của trang quản lý khoa | |
| Luồng sự kiện 2: | | | |
| Tác nhân | | Hệ thống | |
|  | | 5.1 Xác nhận dữ liệu không hợp lệ | |
|  | | 5.2 Hiển thị các lỗi tại các trường nhập dữ liệu chưa hợp lệ | |
| Luồng sự kiện 3: | | | |
| Tác nhân | | Hệ thống | |
|  | | 6.1. Thêm thất bại | |
|  | | 6.2. Thông báo thêm thất bại và lý do | |

**3.2.7.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.7.3. Sequence Diagram:**

### 3.2.8. UC008 - Them mon hoc

**3.2.8.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Them mon hoc | | UC008 |
| Mục đích: | Thêm môn học | | |
| Mô tả: | Cho phép Giáo vụ thêm 1 môn học nào đó vào cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công, người Giáo vụ đang ở trang quản lý môn học. | | |
| Điều kiện sau: | Môn học mới được thêm vào cơ sở dữ liệu và thông báo thành công. Thông báo khi có lỗi xảy ra. | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút thêm môn học | |  | |
|  | | 2. Hiển thị form nhập môn học mới | |
| 3. Nhập dữ liệu vào các ô thông tin | |  | |
| 4. Bấm lưu | |  | |
|  | | 5. Xác nhận dữ liệu nhập là hợp lệ | |
|  | | 6. Thêm thành công trong cơ sở dữ liệu | |
|  | | 7. Thông báo thêm thành công | |
|  | | 8. Cập nhật thêm môn học mới trên bảng danh sách | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
| 4.1. Bấm hủy | |  | |
|  | | 4.2. Xóa hàng mới ra khỏi bảng môn học | |
|  | | 4.3. Quay lại giao diện ban đầu của trang quản lý môn học | |
| Luồng sự kiện 2: | | | |
| Tác nhân | | Hệ thống | |
|  | | 5.1 Xác nhận dữ liệu không hợp lệ | |
|  | | 5.2 Hiển thị các lỗi tại các trường nhập dữ liệu chưa hợp lệ | |
| Luồng sự kiện 3: | | | |
| Tác nhân | | Hệ thống | |
|  | | 6.1. Thêm thất bại | |
|  | | 6.2. Thông báo thêm thất bại và lý do | |

**3.2.8.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.8.3. Sequence Diagram:**

### 3.2.9. UC009 - Xem danh sach mon hoc

**3.2.9.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach mon hoc | | UC009 |
| Mục đích: | Xem danh sach môn học | | |
| Mô tả: | Cho phép Giáo vụ xem danh sách môn học trong cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công. | | |
| Điều kiện sau: | Hệ thống hiển thị danh sách môn học hiện có lên giao diện | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút quản lý môn học | |  | |
|  | | 2. Tải dữ liệu môn học thành công | |
|  | | 3. Hiển thị bảng danh sách thông tin các môn học đang có trong cơ sở dữ liệu | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 2.1 Tải dữ liệu môn học thất bại | |
|  | | 2.2. Thông báo tải dữ liệu thất bại | |

**3.2.9.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.9.3. Sequence Diagram:**

### 3.2.10. UC010 - Xem danh sach hoc phan

**3.2.10.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach hoc phan | | UC010 |
| Mục đích: | Xem danh sach học phần | | |
| Mô tả: | Cho phép Giáo vụ xem danh sách học phần trong cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công. | | |
| Điều kiện sau: | Hệ thống hiển thị danh sách học phần hiện có lên giao diện | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút quản lý học phần | |  | |
|  | | 2. Tải dữ liệu học phần thành công | |
|  | | 3. Hiển thị bảng danh sách thông tin các học phần đang có trong cơ sở dữ liệu | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 2.1 Tải dữ liệu học phần thất bại | |
|  | | 2.2. Thông báo tải dữ liệu thất bại | |

**3.2.10.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.10.3. Sequence Diagram:**

### 3.2.11. UC011 - Them hoc phan

**3.2.11.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Them hoc phan | | UC011 |
| Mục đích: | Thêm học phần | | |
| Mô tả: | Cho phép Giáo vụ thêm 1 học phần nào đó vào cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công, người Giáo vụ đang ở trang quản lý học phần. | | |
| Điều kiện sau: | Học phần mới được thêm vào cơ sở dữ liệu và thông báo thành công.  Thông báo khi có lỗi xảy ra. | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút thêm học phần | |  | |
|  | | 2. Hiển thị form nhập học phần mới | |
| 3. Nhập dữ liệu vào các ô thông tin | |  | |
| 4. Bấm lưu | |  | |
|  | | 5. Xác nhận dữ liệu nhập là hợp lệ | |
|  | | 6. Thêm thành công trong cơ sở dữ liệu | |
|  | | 7. Thông báo thêm thành công | |
|  | | 8. Cập nhật thêm học phần mới trên bảng danh sách | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
| 4.1. Bấm hủy | |  | |
|  | | 4.2. Xóa hàng mới ra khỏi bảng học phần | |
|  | | 4.3. Quay lại giao diện ban đầu của trang quản lý học phần | |
| Luồng sự kiện 2: | | | |
| Tác nhân | | Hệ thống | |
|  | | 5.1 Xác nhận dữ liệu không hợp lệ | |
|  | | 5.2 Hiển thị các lỗi tại các trường nhập dữ liệu chưa hợp lệ | |
| Luồng sự kiện 3: | | | |
| Tác nhân | | Hệ thống | |
|  | | 6.1. Thêm thất bại | |
|  | | 6.2. Thông báo thêm thất bại và lý do | |

**3.2.11.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.11.3. Sequence Diagram:**

### 3.2.12. UC012 - Xem danh sach lop hoc phan

**3.2.12.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach lop hoc phan | | UC012 |
| Mục đích: | Xem danh sach lớp học phần | | |
| Mô tả: | Cho phép Giáo vụ xem danh sách lớp học phần trong cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công. | | |
| Điều kiện sau: | Hệ thống hiển thị danh sách lớp học phần hiện có lên giao diện | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút quản lý lớp học phần | |  | |
|  | | 2. Tải dữ liệu lớp học phần thành công | |
|  | | 3. Hiển thị bảng danh sách thông tin các lớp học phần đang có trong cơ sở dữ liệu | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 2.1 Tải dữ liệu lớp học phần thất bại | |
|  | | 2.2. Thông báo tải dữ liệu thất bại | |

**3.2.12.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.12.3. Sequence Diagram:**

### 3.2.13. UC013 - Them lop hoc phan

**3.2.13.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Them lop hoc phan | | UC013 |
| Mục đích: | Thêm lớp học phần | | |
| Mô tả: | Cho phép Giáo vụ thêm 1 lớp học phần nào đó vào cơ sở dữ liệu | | |
| Tác nhân: | Giáo vụ | | |
| Điều kiện trước: | Đăng nhập thành công, người Giáo vụ đang ở trang quản lý lớp học phần. | | |
| Điều kiện sau: | Lớp học phần mới được thêm vào cơ sở dữ liệu và thông báo thành công.  Thông báo khi có lỗi xảy ra. | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm nút thêm lớp học phần | |  | |
|  | | 2. Hiển thị form thêm mới lớp học phần | |
| 3. Nhập dữ liệu vào các ô thông tin | |  | |
| 4. Bấm lưu | |  | |
|  | | 5. Xác nhận dữ liệu nhập là hợp lệ | |
|  | | 6. Thêm thành công trong cơ sở dữ liệu | |
|  | | 7. Thông báo thêm thành công | |
|  | | 8. Cập nhật thêm lớp học phần mới trên bảng danh sách | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
| 4.1. Bấm hủy | |  | |
|  | | 4.2. Xóa hàng mới ra khỏi bảng lớp học phần | |
|  | | 4.3. Quay lại giao diện ban đầu của trang quản lý lớp học phần | |
| Luồng sự kiện 2: | | | |
| Tác nhân | | Hệ thống | |
|  | | 5.1 Xác nhận dữ liệu không hợp lệ | |
|  | | 5.2 Hiển thị các lỗi tại các trường nhập dữ liệu chưa hợp lệ | |
| Luồng sự kiện 3: | | | |
| Tác nhân | | Hệ thống | |
|  | | 6.1. Thêm thất bại | |
|  | | 6.2. Thông báo thêm thất bại và lý do | |

**3.2.13.2. Activity Diagram:**

**Diagram

Description automatically generated**

**3.2.13.3. Sequence Diagram:**

### 3.2.14. UC014 - Dang nhap

**3.2.14.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Dang nhap | | UC014 |
| Mục đích: | Đăng nhập | | |
| Mô tả: | Cho phép quản trị viên hoặc sinh viên sử dụng tài khoản được cung cấp để đăng nhập và thực hiện các chức năng của hệ thống | | |
| Tác nhân: | Quản trị viên, sinh viên | | |
| Điều kiện trước: | Người dùng đang ở trang đăng nhập của ứng dụng web (với quản trị viên) và ứng dụng di động (với sinh viên) | | |
| Điều kiện sau: | Nếu đăng nhập thành công thì chuyển sang trang chủ của ứng dụng.  Nếu đăng nhập thất bại thì thông báo thất bại và lý do | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Nhập tên đăng nhập, mật khẩu | |  | |
| 2. Bấm nút đăng nhập | |  | |
|  | | 3. Kiểm tra tên đăng nhập, mật khẩu đã nhập đầy đủ | |
|  | | 4. Kiểm tra thông tin đăng nhập với cơ sở dữ liệu trùng khớp | |
|  | | 5. Chuyển sang trang chủ | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 3.1 Người dùng chưa nhập đầy đủ tên đăng nhập và mật khẩu | |
|  | | 3.2. Hiển thị lỗi và yêu cầu người dùng nhập đầy đủ | |
| Luồng sự kiện 2: | | | |
| Tác nhân | | Hệ thống | |
|  | | 4.1 Thông tin đăng nhập không chính xác | |
|  | | 4.2 Thông báo đăng nhập thất bại và lý do | |

**3.2.14.3. Sequence Diagram:**

### 3.2.15. UC015 - Xem danh sach hoc phan trong hoc ki

**3.2.15.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach phan theo hoc ki | | UC015 |
| Mục đích: | Xem danh sách học phần theo học kì | | |
| Mô tả: | Cho phép sinh viên xem danh sách các học phần sẽ mở lớp và đã mở lớp trong học kì | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở giao diện đăng ký học phần, danh sách học kì đã tải xong | | |
| Điều kiện sau: | Hiển thị danh sách học phần theo học kì | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn học kì cần xem danh sách học phần | |  | |
|  | | 2. Tải danh sách học phần sẽ hoặc đã mở lóp trong học kì | |
|  | | 3. Hiển thị danh sách học phần lên giao diện | |

**3.2.15.3. Sequence Diagram:**

### 3.2.16. UC016 - Xem danh sach lop hoc phan trong hoc ki

**3.2.16.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach lop hoc phan trong hoc ki | | UC016 |
| Mục đích: | Xem danh sách lớp học phần trong học kì | | |
| Mô tả: | Cho phép sinh viên xem danh sách các lớp học phần đã mở trong học kì | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở giao diện đăng ký học phần, danh sách học kì và danh sách học phần đã tải xong | | |
| Điều kiện sau: | Hiển thị danh sách lớp học phần theo học kì | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn học phần cần xem danh sách lớp | |  | |
|  | | 2. Tải danh sách lớp học phần đã mở trong học kì | |
|  | | 3. Hiển thị danh sách lớp học phần lên giao diện | |

**3.2.16.3. Sequence Diagram:**

### 3.2.17. UC017 - Xem thong tin chi tiet lop hoc

**3.2.17.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem thong tin chi tiet lop hoc | | UC017 |
| Mục đích: | Xem thông tin chi tiết lớp học phần | | |
| Mô tả: | Cho phép sinh viên xem thông tin chi tiết của lớp học phần (các buổi học, lịch học, giảng viên,…) | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở giao diện đăng ký học phần; danh sách lớp học phần đã tải xong.  Lớp học phần cần xem đã tồn tại và có trên giao diện | | |
| Điều kiện sau: | Hiển thị thông tin chi tiết lớp học phần | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Bấm vào 1 lớp học phần cần xem | |  | |
|  | | 2. Tải thông tin chi tiết của lớp | |
|  | | 3. Hiển thị thông tin chi tiết của lớp (các buổi học, giảng viên, phòng học, tiết học,…) | |

**3.2.17.3. Sequence Diagram:**

### 3.2.18. UC018 - Kiem tra trung lich hoc

**3.2.18.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Kiem tra trung lich hoc | | UC018 |
| Mục đích: | Kiểm tra trùng lịch học | | |
| Mô tả: | Cho phép sinh viên kiểm tra xem lịch học của 1 lớp học phần nào đó có bị trùng với lịch đã đăng ký hay không | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở giao diện đăng ký học phần; danh sách học kì, danh sách học phần và danh sách lớp học phần đã được tải.  Lớp học phần cần kiểm tra đã tồn tại và xuất hiện trên giao diện | | |
| Điều kiện sau: | Hiển thị danh sách lớp không trùng lịch | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn 1 lớp học phần cần kiểm tra lịch trùng | |  | |
|  | | 2. Hiển thị thông tin chi tiết của lớp (phòng học, giảng viên, các nhóm thực hành,…) | |
| 3. Chọn nhóm thực hành (nếu có) | |  | |
| 4. Tích chọn vào ô chỉ hiển thị lớp học phần không trùng lịch | |  | |
|  | | 4. So sánh kiểm tra lịch học của lớp học phần vừa chọn với lớp học phần đã đăng ký | |
|  | | 5. Hiển thị danh sách lớp học không trùng lịch | |

**3.2.18.3. Sequence Diagram:**

### 3.2.19. UC019 - Xem danh sach lop da dang ky trong hoc ki

**3.2.19.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem danh sach lop da dang ky trong hoc ki | | UC019 |
| Mục đích: | Xem danh sách lớp học phần đã đăng ký trong học kì | | |
| Mô tả: | Cho phép sinh viên xem danh sách các lớp đã đăng ký trong học kì | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở giao diện đăng ký học phần, danh sách học kì đã tải xong | | |
| Điều kiện sau: | Hiển thị danh sách lớp học phần đã đăng ký trong học kì | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn học kì cần xem | |  | |
|  | | 2. Tải danh sách lớp đã đăng ký trong học kì | |
|  | | 3. Hiển thị danh sách lên giao diện | |

**3.2.19.3. Sequence Diagram:**

### 3.2.20. UC020 - Dang ky lop hoc phan

**3.2.20.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Dang ky lop hoc phan | | UC020 |
| Mục đích: | Đăng ký lớp học phần | | |
| Mô tả: | Cho phép sinh viên đăng ký lớp học phần | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở giao diện đăng ký học phần, danh sách lớp học phần đã tải xong.  Lớp học phần cần đăng ký đã tồn tại và có trên giao diện | | |
| Điều kiện sau: | Đăng ký thành công lớp học phần cho sinh viên vào cơ sở dữ liệu, thông báo thành công và cập nhật kết quả trên giao diện.  Nếu đăng ký không thành công thì thông báo và ghi rõ lý do (lớp đã khóa, vượt quá số lượng,…) | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn lớp học phần cần đăng ký | |  | |
|  | | 2. Hiển thị thông tin chi tiết của lớp (các buổi học, phòng học, giảng viên, các nhóm thực hành,…) | |
| 3. Chọn nhóm thực hành (nếu có) | |  | |
| 4. Bấm nút đăng ký | |  | |
|  | | 5. Kiểm tra thông tin đăng ký lớp học phần hợp lệ | |
|  | | 6. Thêm sinh viên vừa đăng ký vào danh sách lớp học phần | |
|  | | 7. Thông báo đăng ký học phần thành công | |
|  | | 8. Cập nhật danh sách học phần đã đăng ký lên giao diện | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 5.1 Kiểm tra thông tin đăng ký không hợp lệ (trùng lịch học, quá số lượng sinh viên, lớp đã khóa,…) | |
|  | | 5.2 Hiển thị thông báo thất bại và lý do | |

**3.2.20.3. Sequence Diagram:**

### 3.2.21. UC021 - Xem lich hoc theo tuan

**3.2.21.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem lich hoc theo tuan | | UC021 |
| Mục đích: | Xem lịch học theo tuần | | |
| Mô tả: | Cho phép sinh viên xem lịch học theo tuần | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở trang chủ của ứng dụng | | |
| Điều kiện sau: | Hiển thị danh sách lịch học theo tuần | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn mục lịch theo tuần | |  | |
|  | | 2. Tải danh sách lịch học của tuần hiện tại | |
|  | | 3. Hiển thị danh sách lịch học của tuần hiện tại lên giao diện | |
| 4. Chọn tuần khác cần xem lịch học | |  | |
|  | | 5. Tải danh sách lịch học của tuần đã chọn | |
|  | | 6. Hiển thị danh sách lịch học của tuần đã chọn lên giao diện | |

**3.2.21.3. Sequence Diagram:**

### 3.2.22. UC022 - Xem ket qua hoc tap

**3.2.22.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem ket qua hoc tap | | UC022 |
| Mục đích: | Xem kết quả học tập | | |
| Mô tả: | Cho phép sinh viên xem kết quả học tập | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở trang chủ của ứng dụng | | |
| Điều kiện sau: | Hiển thị danh sách kết quả học tập | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn mục bảng điểm ở phần tabbar của ứng dụng | |  | |
|  | | 2. Tải danh sách kết quả học tập | |
|  | | 3. Hiển thị danh sách kết quả học tập lên giao diện | |

**3.2.22.3. Sequence Diagram:**

### 3.2.23. UC023 - Thanh toan truc tuyen

**3.2.23.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Thanh toan truc tuyen | | UC023 |
| Mục đích: | Thanh toán học phí bằng hình thức Online | | |
| Mô tả: | Cho phép sinh viên thanh toán học phí trực tuyến | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở trang chủ của ứng dụng | | |
| Điều kiện sau: | Thanh toán thành công. Hệ thống cập nhật công nợ cho sinh viên, | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn mục thanh toán trục tuyến tại giao diện trang chủ. | |  | |
|  | | 2. Tải danh sách các khoản thu chưa trừ công nợ. | |
| 3. Chọn hình thức thanh toán | |  | |
| 4. Bấm vào nút thanh toán | |  | |
|  | | 5. Hệ thống chuyển sang trang nhập thông tin thanh toán. | |
| 6. Nhập thông tin thanh toán | |  | |
| 7. Click vào thanh toán | |  | |
|  | | 8. Hệ thống yêu cầu nhập OTP. | |
| 9. Nhập OTP | |  | |
|  | | 10. Hệ thống kiểm tra thông tin thanh toán và số dư. | |
|  | | 11. Hệ thống thông báo thanh toán thành công. | |
|  | | 12. Cập nhật công nợ cho sinh viên. | |
| Luồng sự kiện phụ | | | |
| Luồng sự kiện 1: | | | |
| Tác nhân | | Hệ thống | |
|  | | 10.1. Kiểm tra thông tin thanh toán chưa hợp lệ ( Sai thông tin thẻ, sai otp, số tiền không đủ, thẻ bị khóa, …. ) | |
|  | | 10.2. Hiển thị thông báo thất bại và lý do. | |
|  | |  | |

**3.2.23.3. Sequence Diagram:**

### 3.2.24. UC024 - Xem phieu thu tong hop

**3.2.24.1. Đặc tả usecase:**

|  |  |  |  |
| --- | --- | --- | --- |
| Tên use case: | Xem phieu thu tong hop | | UC024 |
| Mục đích: | Xem phiếu thu tổng hợp | | |
| Mô tả: | Cho phép sinh viên xem phiếu thu tổng hợp | | |
| Tác nhân: | Sinh viên | | |
| Điều kiện trước: | Sinh viên đã đăng nhập thành công và đang ở trang chủ của ứng dụng | | |
| Điều kiện sau: | Hiển thị danh sách phiếu thu và chi tiết phiếu thu. | | |
| Luồng sự kiện chính: | | | |
| Tác nhân | | Hệ thống | |
| 1. Chọn mục phiếu thu tổng hợp ở trang chủ. | |  | |
|  | | 2. Tải danh sách các phiếu thu | |
| 3. Chọn vào phiếu thu cần xem chi tiết và nhấn nút chi tiết | |  | |
|  | | 4. Tải danh sách chi tiết phiếu thu tương ứng. | |

**3.2.24.3. Sequence Diagram:**

## 3.3. Lược đồ cơ sở dữ liệu:

****

# CHƯƠNG 4: THIẾT KẾ VÀ HIỆN THỰC

## 4.1. Bài toán

**Giải quyết bài toán lượng người truy cập cùng một thời điểm**

**Vấn đề đặt ra:** Đến thời điểm đăng ký học phần mới cho từng học kỳ. Số lượng sinh viên truy cập lượng lớn vào các thời điểm như đăng ký học phần. Số lượng request xảy ra nhiều nhưng sau đó lại giảm, vì thế cần server sẽ tự scalling giúp giảm chi phí và giúp server không sập lúc cao điểm.

**Hướng giải quyết**

**Performance**

**Cân bằng tải với Load Balancer:** Load Balancer là một công nghệ cho phép cân bằng tải đến nhiều server.

Giả sử ta có 1 server có thể phục vụ 1000 người. Để phục vụ 10000 người, ta có thể chạy 10 server. Người dùng sẽ không trực tiếp truy cập tới server, mà chỉ truy cập tới load balancer. Load Balancer sẽ điều tiết, cân bằng lượng tải trên 10 server này.

**Caching:** Cache là một kĩ thuật để tăng tốc độ đọc dữ liệu, bằng cách lưu dữ liệu sẵn vào cache server để tăng tốc độ đọc ở những lần sau.

Ví dụ mỗi lần gọi API hoặc Database mất 10s, ta chỉ gọi lần đầu và lưu kết quả vào cache. Ở những lần sau, đọc kết quả từ cache chỉ mất 0.5s thôi.

**Scalability**

**AutoScalling** giúp tạo hoặc tắt số lượng server tuỳ vào lượng truy cập vào web/app trong một thời điểm. Sử dụng Amazon EC2 Auto Scaling.

## 4.2. Kiến trúc đề xuất

## 4.3. Công nghệ sử dụng

## 4.4. Một số hình ảnh demo

Công cụ AVTEdu đã có thể triển khai trên hai nền tảng web và mobile app. Chúng tôi xin được giới thiệu trước về nền tảng webapp. Đối với hệ thống web của AVTEdu, chúng tôi xây dựng giao diện, chức năng hướng đến 3 loại người dùng chính: sinh viên, giảng viên và giáo vụ. Đối với sinh viên đang học tập tại trường sẽ được cấp tên đăng nhập ứng với mã số sinh viên và mật khẩu mặc định chung, sinh viên có thể đổi sau. Sinh viên sau khi có được thông tin đăng nhập và tiến hành đăng nhập, trang web sẽ điều hướng đến màn hình trang chủ. Sinh viên có thể thấy tổng quan về số tín chỉ được tích lũy, điểm trung bình học kỳ, các mục tùy chọn như xem lịch theo tuần. xem điểm, đăng ký học phần,…

# CHƯƠNG 5: KẾT QUẢ ĐẠT ĐƯỢC VÀ HƯỚNG PHÁT TRIỂN

## 5.1. Kết quả đạt được:

## 5.2. Hạn chế của đồ án:

Chức năng của giáo vụ khoa vẫn còn hạn chế,chưa đáp ứng được các chức năng hỗ trợ.Chức năng của giảng viên vẫn còn ít chưa hỗ trợ nhiều cho giảng viên.Ứng dụng chưa triển khai được lên các nền tảng cloud.Phần ứng dụng di động cũng chưa triển khai lên các store trên di động.Trong tương lai ,chúng tôi sẽ dành tài nguyên và nhân lực để giải quyết các hạn chế trên .

## 5.3. Hướng phát triển:

Hiện nay, vì là sinh viên nên điều kiện kinh tế còn hạn hẹp cũng như thời gian làm quen với Cloud Computing chưa nhiều nên việc sử dụng các dịch vụ của Amazon Web Services (AWS) rất khó khăn. Chúng tôi chưa thể triển khai kiến trúc hệ thống theo đúng mong muốn được đặt ra từ đầu. Tương lai, chúng tôi muốn triển khai kiến trúc hệ thống theo đúng kiến trúc mong muốn. Cụ thể chúng tôi muốn tách nhỏ hệ thống Back-end thành nhiều dịch vụ khác nhau chạy trên các CPU riêng biệt đúng với tinh thần của kiến trúc Micro-services. Điều này giúp các dịch vụ có tính độc lập cao, vừa giúp khả năng mở rộng tốt vừa tăng tính an toàn cho hệ thống, giả sử một dịch vụ bị lỗi thì các dịch vụ khác vẫn có khả năng hoạt động bình thường. Chúng tôi sẽ tách thành các dịch vụ riêng biệt như Authentication Service (phục vụ việc đăng nhập, phân quyền chức năng), Sign Up Modules Service (phục vụ việc đăng ký học phần),… Giả sử sau này chúng tôi muốn đầu tư và phát triển cho hệ thống dịch vụ đăng ký học phần thì cũng không cần phải lo lắng các dịch vụ khác sẽ bị ảnh hưởng.

Vì điều kiện thiết bị khó khăn, nên chúng tôi vẫn chưa thể thử nghiệm được ứng dụng di động trên nền tảng iOS. Trong tương lai, chúng tôi sẽ thử nghiệm ứng dụng trên điện thoại iPhone. Chúng tôi cũng có kế hoạch đưa ứng dụng di động AVTEdu-App lên CHPlay và Appstore.

Ứng dụng di động AVTEdu-App vẫn còn nhiều hạn chế về mặt tính năng, chưa có tính năng tra cứu công nợ, xem và nhận thông báo, xem chương trình học tập,… Hệ cơ sở dữ liệu cũng chưa thiết kế để đáp ứng được đầy đủ các chức năng này. Vì thế tương lai, chúng tôi sẽ hoàn thiện các tính năng này để đáp ứng đầy đủ nhất nhu cầu cho sinh viên trường ĐH Công nghiệp TP.HCM.

Chúng tôi sẽ triển khai chức năng thông báo thông báo trên ứng dụng di động. Khi lịch thi của một môn học được cập nhật hoặc có các thông báo của nhà trường, cảnh báo học vụ,… thì sinh viên sẽ được thông tin kịp thời.

Về phân hệ chức năng cho các quản trị viên, thời gian sắp tới, chúng tôi hướng đến việc phân quyền chức năng chặt chẽ hơn nữa. Cụ thể chung tôi sẽ phân quyền chức năng theo khoa, quản trị viên của khoa nào thì chỉ có quyền chỉnh sửa những thông tin thuộc quyền quản lý của khoa đó. Và trong mỗi khoa, chúng tôi hướng đến việc phân cấp chức năng kĩ lưỡng hơn nữa. Ví dụ, chỉ có trưởng khoa hoặc trưởng bộ môn mới được quyền chỉnh sửa thông tin môn học.

Ngoài ra, bảo mật cũng là yếu tố cực kì quan trọng trong một hệ thống. Sắp tới, chúng tôi sẽ tập trung nhiều hơn nữa cho vấn đề này.

**TÀI LIỆU THAM KHẢO**

**Các tài liệu Tiếng Anh**

**Các tài liệu từ Internet**