 TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI

KHOA CÔNG NGHỆ THÔNG TIN

---------------o0o---------------

Logo

Description automatically generated

**ĐỒ ÁN TỐT NGHIỆP**

ĐỀ TÀI

**XÂY DỰNG WEBSITE QUẢN LÝ SINH VIÊN**

|  |  |
| --- | --- |
| Giảng viên hướng dẫn | : Nguyễn Trọng Phúc |
| Sinh viên thực hiện | : Ngô Quang Thắng |
| Mã sinh viên | : 201200337 |
| Lớp | : Công nghệ thông tin 6 |
| Khóa | : K61 |

**Hà Nội - 2024**

# LỜI CẢM ƠN

Trong quá trình học tập và rèn luyện tại Trường Đại học Giao thông vận tải, em xin gửi lời tri ân chân thành đến các thầy, cô giáo đã tạo điều kiện thuận lợi nhất để em có thể tiếp cận và học tập hiệu quả.

Em muốn bày tỏ lòng biết ơn sâu sắc đến tất cả các thầy cô giáo trong Khoa Công nghệ Thông tin và các bộ môn liên quan, nhờ sự nhiệt tình và kiến thức uyên bác của họ, em đã có cơ hội tích lũy và phát triển kiến thức một cách toàn diện trong suốt thời gian học tập tại trường.

Đặc biệt, em xin gửi lời biết ơn đặc biệt đến thầy Nguyễn Trọng Phúc, người đã dành nhiều thời gian và công sức để hướng dẫn em trong quá trình thực hiện đồ án tốt nghiệp. Sự hỗ trợ và chỉ dẫn từ thầy đã giúp em vượt qua những khó khăn và hoàn thành đồ án một cách thành công.

Em chân thành cảm ơn sự đóng góp và sự hỗ trợ của tất cả các thầy cô giáo và bạn bè trong quá trình học tập của mình.

Trân trọng,

Ngô Quang Thắng

# NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

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

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

|  |  |
| --- | --- |
|  | Hà Nội, ngày tháng năm 2024  Giảng viên hướng dẫn |

**Nguyễn Trọng Phúc**

# MỤC LỤC

[LỜI CẢM ƠN 1](#_Toc166707907)

[NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN 2](#_Toc166707908)

[MỤC LỤC 3](#_Toc166707909)

[DANH MỤC CÁC TỪ VIẾT TẮT 4](#_Toc166707910)

[DANH MỤC HÌNH ẢNH 5](#_Toc166707911)

[MỞ ĐẦU 7](#_Toc166707912)

[CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI 8](#_Toc166707913)

[1.1. Khảo sát 9](#_Toc166707914)

[1.2. Nhiệm vụ 10](#_Toc166707917)

[1.3. Giới thiệu về công nghệ 11](#_Toc166707922)

[1.3.1. Ngôn ngữ lập trình php 11](#_Toc166707923)

[1.3.2. Hệ quản trị cơ sở dữ liệu MySQL 12](#_Toc166707924)

[1.3.3. XAMPP 14](#_Toc166707925)

[1.3.4. Visual Studio Code 15](#_Toc166707926)

[1.3.5. JavaScript 16](#_Toc166707926)

[1.3.6. Html & Css 17](#_Toc166707926)

[1.3.7. Mô hình MVC 17](#_Toc166707926)

[CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 18](#_Toc166707928)

[2.1. Phân tích 18](#_Toc166707929)

[2.1.1. Sơ đồ chức năng 18](#_Toc166707930)

[2.1.2. Sơ đồ use-case Admin 19](#_Toc166707932)

[2.1.2. Sơ đồ use-case User 31](#_Toc166707934)

[2.1.2. Sơ đồ hoạt động(Activity) 33](#_Toc166707934)

[2.2. Thiết kế cơ sở dữ liệu 39](#_Toc166707935)

[2.2.1. ERD 39](#_Toc166707936)

[2.2.3. Mô tả các bảng 40](#_Toc166707937)

[CHƯƠNG 3: CÀI ĐẶT CHƯƠNG TRÌNH 45](#_Toc166707938)

[3.1. Xây dựng trang Admin 45](#_Toc166707939)

[3.2. Xây dựng trang User 48](#_Toc166707940)

[KẾT LUẬN 52](#_Toc166707941)

[TÀI LIỆU THAM KHẢO 53](#_Toc166707941)

# DANH MỤC CÁC TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| STT | Viết tắt | Ý nghĩa |
| 1 | CNTT | Công nghệ thông tin |
| 2 | Html | HyperText Markup Language |
| 3 | Css | Cascading Style Sheets |
| 4 | Js | Javascript |
| 5 | Asp | Active Server Pages |
| 6 | Sql | Structured Query Language |
| 7 | ERD | Entity Relationship Diagram |
| 8 | RM | Relationship Model |
| 9 | BFD | Business Function Diagram |
| 10 | DFD | Data Flow Diagram |

# DANH MỤC HÌNH ẢNH

[Hình 1.1. Hình ảnh website qldt.utc.edu.vn 9](#_Toc168090511)

[Hình 2.1. Sơ đồ chức năng 18](#_Toc168090512)

[Hình 2.2. Sơ đồ use-case tổng quát trang admin 19](#_Toc168090513)

[Hình 2.3. Sơ đồ usecase chức năng Đăng nhập 20](#_Toc168090514)

[Hình 2.4. Sơ đồ usecase Quản lý sinh viên 21](#_Toc168090515)

[Hình 2.5. Sơ đồ usecase Quản lý giảng viên 22](#_Toc168090516)

[Hình 2.6. Sơ đồ usecase Quản lý học phần 24](#_Toc168090517)

[Hình 2.7. Sơ đồ usecase Quản lý điểm 25](#_Toc168090518)

[Hình 2.8. Sơ đồ usecase Quản lý khoa 27](#_Toc168090519)

[Hình 2.9. Sơ đồ usecase Quản lý lớp học 28](#_Toc168090520)

[Hình 2.10. Sơ đồ usecase Quản lý môn học 30](#_Toc168090521)

[Hình 2.11. Sơ đồ use-case tổng quát trang User 31](#_Toc168090522)

[Hình 2.12. Sơ đồ hoạt động đăng nhập 33](#_Toc168090523)

[Hình 2.13. Sơ đồ hoạt động đăng ký học 34](#_Toc168090524)

[Hình 2.14. Sơ đồ hoạt động sửa dữ liệu trong danh mục 35](#_Toc168090525)

[Hình 2.15. Sơ đồ hoạt động thêm dữ liệu vào danh mục 36](#_Toc168090526)

[Hình 2.16. Sơ đồ hoạt động Xóa dữ liệu trong danh mục 37](#_Toc168090527)

[Hình 2.17. Sơ đồ hoạt động thêm lớp học phần 38](#_Toc168090527)

[Hình 2.18. Sơ đồ ERD 39](#_Toc168090528)

[Hình 3.1. Giao diện trang Đăng nhập của Admin 45](#_Toc168090529)

[Hình 3.2. Giao diện Quản lý Sinh viên 45](#_Toc168090530)

[Hình 3.3. Giao diện Quản lý Giảng viên 45](#_Toc168090531)

[Hình 3.4. Giao diện Quản lý học kỳ 46](#_Toc168090532)

[Hình 3.5. Giao diện Biểu đồ tổng quan 46](#_Toc168090535)

[Hình 3.6. Giao diện Quản lý lớp học phần 47](#_Toc168090536)

[Hình 3.7. Giao diện Quản lý điểm 47](#_Toc168090537)

[Hình 3.8. Giao diện trang Đăng nhập của sinh viên 48](#_Toc168090538)

[Hình 3.9. Giao diện Trang chủ 49](#_Toc168090539)

[Hình 3.10. Giao diện Đăng ký môn học 49](#_Toc168090540)

[Hình 3.11. Giao diện Kết quả đăng ký học 50](#_Toc168090541)

[Hình 3.12. Giao diện Tra cứu điểm 50](#_Toc168090542)

[Hình 3.13. Giao diện Đổi mật khẩu 51](#_Toc168090543)

[Hình 3.14. Giao diện Thông tin cá nhân 51](#_Toc168090544)

# DANH MỤC BẢNG BIỂU

[Bảng 3.1. Bảng “tbl\_admin” 39](#_Toc167399946)

[Bảng 3.2. Bảng “tbl\_chucvu” 39](#_Toc167399947)

[Bảng 3.3. Bảng “tbl\_class” 39](#_Toc167399948)

[Bảng 3.4. Bảng “tbl\_classroom” 39](#_Toc167399949)

[Bảng 3.5. Bảng “tbl\_class\_section” 39](#_Toc167399950)

[Bảng 3.6. Bảng “tbl\_course” 40](#_Toc167399951)

[Bảng 3.7. Bảng “tbl\_education” 40](#_Toc167399952)

[Bảng 3.8. Bảng “tbl\_enrollmentdetail” 40](#_Toc167399953)

[Bảng 3.9. Bảng “tbl\_enrollments” 40](#_Toc167399954)

[Bảng 3.10. Bảng “tbl\_faculty” 40](#_Toc167399955)

[Bảng 3.11. Bảng “tbl\_grades” 40](#_Toc167399956)

[Bảng 3.12. Bảng “tbl\_gradesdetail” 41](#_Toc167399957)

[Bảng 3.13. Bảng “tbl\_major” 41](#_Toc167399958)

[Bảng 3.14. Bảng “tbl\_posts” 41](#_Toc167399959)

[Bảng 3.15. Bảng “tbl\_schedule” 41](#_Toc167399960)

[Bảng 3.16. Bảng “tbl\_semester” 41](#_Toc167399961)

[Bảng 3.17. Bảng “tbl\_semester\_subject” 42](#_Toc167399962)

[Bảng 3.18. Bảng “tbl\_start\_end\_date” 42](#_Toc167399963)

[Bảng 3.19. Bảng “tbl\_student” 42](#_Toc167399964)

[Bảng 3.20. Bảng “tbl\_student\_account” 43](#_Toc167399965)

[Bảng 3.21. Bảng “tbl\_subject” 43](#_Toc167399966)

[Bảng 3.22. Bảng “tbl\_teacher” 43](#_Toc167399967)

[Bảng 3.23. Bảng “tbl\_teacher\_account” 43](#_Toc167399968)

[Bảng 3.24. Bảng “tbl\_teacher\_assignment” 44](#_Toc167399969)

# MỞ ĐẦU

Trong thời đại số hóa ngày nay, việc quản lý thông tin sinh viên trở nên ngày càng phức tạp và quan trọng đối với các cơ sở giáo dục. Một hệ thống quản lý sinh viên hiệu quả không chỉ giúp tiết kiệm thời gian và công sức của nhân viên quản lý mà còn cung cấp một cơ sở dữ liệu đáng tin cậy và dễ dàng truy cập cho cả sinh viên và cán bộ quản lý.

Website quản lý sinh viên là một công cụ mạnh mẽ, mang lại sự thuận tiện và hiệu quả trong việc tổ chức, theo dõi và quản lý thông tin của sinh viên. Từ việc đăng ký môn học, lên kế hoạch học tập, đến quản lý thông tin cá nhân và kết quả học tập, website này cung cấp một nền tảng toàn diện để tối ưu hóa quá trình quản lý sinh viên.

Em tin rằng việc phát triển một hệ thống quản lý sinh viên hoàn hảo không chỉ là một ưu tiên mà còn là một bước quan trọng để nâng cao chất lượng giáo dục. Em tự hào giới thiệu website quản lý sinh viên này, hy vọng rằng nó sẽ đóng góp vào sự phát triển bền vững của cộng đồng giáo dục.

Bố cục của đề tài gồm: phần mở đầu, phần thân (3 chương), phần kết luận và tài liệu tham khảo. Kết quả chính của đề tài tập trung ở chương 2 và chương 3. Cụ thể:

Chương 1: Trình bày tổng quan về đề tài, mục tiêu và phạm vi của đề tài. Tham khảo một số website, khảo sát nghiệp vụ. Nêu các công nghệ và công cụ được sử dụng.

Chương 2: Phân tích và thiết kế các chức năng cho website, trình bày sơ đồ mô hình hoá đối tượng như: Sơ đồ phân rã chức năng, sơ đồ use – case và sơ đồ hành động. Phân tích mối quan hệ giữa các thực thể và thiết kế CSDL vậy lý.

Chương 3: Cài đặt và kiểm thử chương trình. Nêu ra các kết quả đã đạt được sau khi đã phân tích thiết kế.

Kết luận: Trình bày tóm tắt được những kết quả mà đề tài đạt được, chỉ ra những điểm hạn chế, đưa ra những hướng phát triển trong tương lai.

# CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

## Khảo sát

### Website <https://qldt.utc.edu.vn/>



#### Hình 1.1. Hình ảnh website qldt.utc.edu.vn

Là một hệ thống quản lý thông tin sinh viên của Trường Đại học Giao thông Vận tải (UTC). Dưới đây là một số ưu và nhược điểm của trang web này:

* Ưu điểm:
* Dễ dàng truy cập thông tin: Sinh viên có thể dễ dàng truy cập thông tin cá nhân, kế hoạch học tập, kết quả học tập và thông báo từ trường qua website này.
* Tiện ích đa dạng: Trang web cung cấp nhiều tính năng hữu ích như đăng ký môn học, xem thời khóa biểu, tra cứu điểm, và thậm chí là đặt câu hỏi trực tuyến.
* Tính tương tác: Sinh viên có thể tương tác trực tiếp với các dịch vụ và thông tin trên website, giúp họ tiết kiệm thời gian và nâng cao trải nghiệm của mình.
* Cập nhật thông tin nhanh chóng: Thông tin được cập nhật thường xuyên, giúp sinh viên và nhân viên có được thông tin mới nhất và chính xác nhất.
* Tăng tính minh bạch và độ tin cậy: Website giúp tăng cường minh bạch và độ tin cậy trong quản lý thông tin sinh viên của trường.
* Nhược điểm:
* Hệ thống bảo mật cần được cải thiện: Có thể gặp phải những vấn đề về bảo mật thông tin cá nhân nếu hệ thống không được bảo vệ và duy trì đúng cách.
* Giao diện không thân thiện: Một số người dùng có thể gặp khó khăn khi sử dụng do giao diện không được thiết kế đơn giản và dễ sử dụng.

## Nhiệm vụ

### Mục đích

Cung cấp một nền tảng trực tuyến để quản lý thông tin sinh viên một cách hiệu quả và thuận tiện. Dưới đây là một số mục đích cụ thể của dự án này:

Quản lý thông tin cá nhân của sinh viên: Website cung cấp một cơ sở dữ liệu để lưu trữ thông tin cá nhân của sinh viên như tên, số điện thoại, địa chỉ email, lớp, khoa, và các thông tin khác.

Quản lý điểm số và kết quả học tập: Website cho phép giáo viên và quản trị viên nhập điểm số và theo dõi kết quả học tập của sinh viên theo từng môn học. Điều này giúp cải thiện quá trình đánh giá và phản hồi cho sinh viên.

Quản lý lịch trình học tập: Website có thể cung cấp thông tin về lịch học, và các sự kiện khác để giúp sinh viên và giáo viên tổ chức thời gian học tập một cách hiệu quả.

### Yêu cầu

Xây dựng được website gồm các chức năng:

* Dành cho sinh viên
* **Đăng nhập:**

Cho phép sinh viên của trường đăng nhập vào hệ thống bằng tài khoản và mật khẩu cá nhân.

* **Quản lý thông tin sinh viên:**

Hiển thị thông tin cá nhân của sinh viên như họ tên, lớp, khoa, số điện thoại, email, địa chỉ, v.v.

Cập nhật thông tin cá nhân khi cần thiết.

* **Quản lý điểm số và kết quả học tập:**

Hiển thị điểm số và kết quả học tập của sinh viên qua các kỳ học.

Cho phép sinh viên xem và in bảng điểm cá nhân.

* **Đăng ký môn học và thời khóa biểu:**

Cho phép sinh viên đăng ký các môn học trong từng kỳ học.

Hiển thị thời khóa biểu cá nhân của sinh viên.

* **Thông báo và tin tức:**

Cung cấp thông báo và tin tức về các sự kiện, hoạt động, thay đổi chính sách của trường.

* Dành cho quản trị viên:
* Quản lý thông tin sinh viên, giảng viên: Thêm, sửa và xoá
* Quản lý điểm
* Quản lý lớp học phần
* Quản lý khoa: Thêm, sửa và xoá.
* Quản lý môn học: Thêm, sửa và xoá.
* Quản lý học kỳ: Thêm, sửa và xoá.
* Quản lý lịch học: Thêm, sửa và xoá.
* Biểu đồ thống kê

### Môi trường phát triển

Trong quá trình tìm hiểu và xây dựng đồ án, em đã chọn thực hiện website trên nền tảng ứng dụng mã nguồn mở. Phía Front-end em sử dụng Html, Css, Javascript và các Framework như Bootstrap, Jquery để phục vụ việc xây dựng giao diện cho website. Về phía Back-end em sử dụng cơ sở dữ liệu MySql, Php và Framework Laravel. Đó đều là những công nghệ em đã được trang bị trong thời gian còn ngồi trên ghế nhà trường.

Môi trường phát triển: Visual Studio code, MySql.

Ngôn ngữ lập trình: Html, Css, Javascript và Php.

Hệ quản trị cơ sở dữ liệu: MySql.

## Giới thiệu về công nghệ

### Ngôn ngữ lập trình php

* PHP là gì: PHP là 1 ngôn ngữ lập trình mã nguồn mở được sử dụng chủ yếu để phát triển ứng dụng web động. Nó hoạt động trên phía máy chủ, và khi 1 trình duyệt web yêu cầu 1 trang web PHP, máy chủ sẽ xử lý mã PHP và sau đó gửi kết quả HTML đến trình duyệt để hiển thị. PHP có khả năng tương tác với cơ sở dữ liệu, xử lý biểu mẫu web, tạo và quản lý cookie, và thực hiện nhiều nhiệm vụ khác nhau liên quan đến phát triển ứng dụng web.
* Ưu điểm
* Dễ học và sử dụng

PHP có cú pháp dễ hiểu và gần gũi với ngôn ngữ tự nhiên, điều này làm cho việc học và sử dụng PHP trở nên dễ dàng cho người mới bắt đầu trong lĩnh vực lập trình. Ngoài ra vì độ phổ biến của nó mà các developer dễ dàng tìm được những cộng đồng sử dụng ngôn ngữ lập trình này và học hỏi kinh nghiệm của nhau.

* Thích hợp để xây dựng web

Được thiết kế đặc biệt cho việc phát triển web, PHP có nhiều thư viện và framework mạnh mẽ như Laravel, Symfony và WordPress. Nhờ vào các công cụ này, việc xây dựng các trang web, web app và hệ thống CMS trở nên dễ dàng và nhanh chóng.

* Tương thích đa nền tảng

Khả năng tương thích của PHP là một trong những đặc điểm mạnh của ngôn ngữ này. PHP có thể chạy trên hầu hết các hệ điều hành phổ biến như Windows, Linux, macOS và các hệ điều hành UNIX khác. Điều này cho phép phát triển ứng dụng PHP trên nhiều nền tảng khác nhau mà không gặp phải rào cản về hệ điều hành.

* Hỗ trợ đa dạng cơ sở dữ liệu

PHP có thể tương tác với các hệ quản trị cơ sở dữ liệu phổ biến như MySQL, PostgreSQL, Oracle và nhiều hơn nữa. Ngoài ra, PHP cũng hỗ trợ giao tiếp với các dịch vụ web và các nguồn dữ liệu khác như API, RSS feeds, và SOAP.

* Tính linh hoạt

Với PHP có thể tự do xây dựng các ứng dụng web từ đơn giản đến phức tạp. PHP không ràng buộc người lập trình vào một cấu trúc nghiêm ngặt, mà cho phép linh hoạt trong việc tổ chức mã nguồn và cấu trúc dự án, có thể tự do chọn các thư viện, framework và công cụ phù hợp để phát triển ứng dụng theo ý muốn.

* Tiết kiệm chi phí

PHP là một ngôn ngữ web mã nguồn mở, do đó hoàn toàn miễn phí. Không có chi phí liên quan đến việc mua giấy phép hoặc phần mềm đắt tiền. Nó có thể hoạt động hiệu quả với các cơ sở dữ liệu khác nhau, chẳng hạn như MySQL, Apache và PostgreSQL. Nhờ vậy có thể phát triển một trang web với chi phí thấp.

* Nhược điểm
* Tính bảo mật kém

Lịch sử của PHP có nhiều vấn đề liên quan đến bảo mật. Vì nó là ngôn ngữ mã nguồn mở, có nghĩa là mã nguồn của nó có thể bị xem bởi bất kỳ ai. Điều này có thể tạo ra rủi ro bảo mật khi những lỗ hổng và lỗi được khai thác bởi kẻ tấn công.

* Không phù hợp với ứng dụng quá phức tạp

Mặc dù PHP thích hợp cho phát triển ứng dụng web, nó không phải lựa chọn tốt cho các ứng dụng phức tạp khác như ứng dụng di động. Người dùng càng cần sử dụng nhiều tính năng thì hiệu suất của PHP càng giảm. Trong các trường hợp này, ngôn ngữ lập trình khác có thể hiệu quả hơn.

* Hiệu suất thấp

Một trong những hạn chế chính của PHP là hiệu suất không cao so với một số ngôn ngữ khác như Java hoặc C++. PHP có thể chậm hơn trong việc xử lý các tác vụ phức tạp và đòi hỏi nhiều tài nguyên hơn.

* Khả năng giới hạn

Chắc chắn PHP là một ngôn ngữ linh hoạt và có thể được mở rộng. Tuy nhiên, nó có một phạm vi cốt lõi. Do đặc điểm này nên không thể đưa ra bất kỳ thay đổi hoặc sửa đổi nào đối với chức năng cơ bản hoặc bộ khung của ứng dụng.

Vì vậy, khi một ứng dụng được phát triển với một chức năng cụ thể, phạm vi của nó sẽ vẫn bị giới hạn trong nhiệm vụ cụ thể đó.

* Tham khảo thêm tại: <https://vi.wikipedia.org/wiki/PHP>

A blue and black logo

Description automatically generated

Hình 1.2 Hình ảnh minh họa ngôn ngữ lập trình php

### Hệ quản trị cơ sở dữ liệu MySQL

* MySQL là cơ sở dữ liệu tốc độ cao, ổn định hoạt động trên nhiều hệ điều hành, cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Đặc biệt, hệ quản trị cơ sở dữ liệu MySQL hoàn toàn miễn phí nên người dung có thể thoải mái tải về từ trang chủ. MySQL được sử dụng cho việc hỗ trợ PHP, Perl và nhiều ngôn ngữ khác. Là nơi lưu trữ những thông tin trên các trang web viết bằng framework PHP hay Perl …
* Ưu điểm
* Dễ sử dụng: MySQL là cơ sở dữ liệu tốc độ cao, ổn định, dễ sử dụng và hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.
* Độ bảo mật cao: MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên Internet khi sở hữu nhiều nhiều tính năng bảo mật thậm chí là ở cấp cao.
* Đa tính năng: MySQL hỗ trợ rất nhiều chức năng SQL được mong chờ từ một hệ quản trị cơ sở dữ liệu quan hệ cả trực tiếp lẫn gián tiếp.
* Khả năng mở rộng và mạnh mẽ: MySQL có thể xử lý rất nhiều dữ liệu và hơn thế nữa nó có thể được mở rộng nếu cần thiết.
* Nhanh chóng: Việc đưa ra một số tiêu chuẩn cho phép MySQL để làm việc rất hiệu quả và tiết kiệm chi phí, do đó nó làm tăng tốc độ thực thi.
* Nhược điểm
* Giới hạn: Theo thiết kế, MySQL không có ý định làm tất cả và nó đi kèm với các hạn chế về chức năng mà một vào ứng dụng có thể cần.
* Độ tin cậy: Cách các chức năng cụ thể được xử lý với MySQL (ví dụ tài liệu tham khảo, các giao dịch, kiểm toán,…) làm cho nó kém tin cậy hơn so với một số hệ quản trị cơ sở dữ liệu quan hệ khác.
* Dung lượng hạn chế: Nếu số bản ghi của bạn lớn dần lên thì việc truy xuất dữ liệu của bạn là khá khó khăn, khi đó chúng ta sẽ phải áp dụng nhiều biện pháp để tăng tốc độ truy xuất dữ liệu như là chia tải database này ra nhiều server, hoặc tạo cache MySQL.
* Tham khảo thêm tại: <https://vi.wikipedia.org/wiki/MySQL>

A dolphin and text on a black background

Description automatically generated

Hình 1.3 Hình ảnh minh họa về MySQL

### XAMPP

* Xampp là 1 ứng dụng mã nguồn mở miễn phí được sử dụng để tạo môi trường phát triển web trên máy tính các nhân. Tên ‘XAMPP’ là viết tắt của các thành phần chính mà nó kết hợp: **X** (đa nền tảng), A (Apache – máy chủ web), M ( MySQL – hệ quản trị cơ sở dữ liệu), P (PHP – ngôn ngữ lập trình), và một P bổ sung ( thường là Perl, nhưng có thể thay thế bằng Python hoặc Ruby).
* Ưu điểm
* Có thể chạy được linh hoạt được trên mọi hệ điều hành: Cross-platform, Windows, MacOS, Linux.
* Có cấu hình đơn giản nhưng mang đến nhiều chức năng hữu ích như: Lập Server giả định, lập Mail Server giả định và hỗ trợ SSL trên localhost.
* Tích hợp được nhiều tính năng với các thành phần quan trọng như: Apache, PHP, MySql. Vì thế, người dùng không cần cài đặt từng phần trên riêng lẻ mà chỉ cần cài XAMPP là có 1 web server hoàn chỉnh.
* Tạo mã nguồn mở: Giao diện quản lý dễ dàng và tiện lợi giúp người dùng luôn chủ động được trong chế độ khởi động lại hay bật/ tắt đối với máy chủ theo thời gian phù hợp nhất.
* Nhược điểm
* Do cấu hình khá đơn giản cho nên Xampp hoàn toàn không nhận được sự hỗ trợ về cấu hình Module, đồng thời cũng không có cả MySQL.
* Dung lượng tương đối nặng (141Mb).
* Không có nhiều phiên bản cho từng thành phần của server như PHP, Apache mà phải cài đặt riêng.
* Tham khảo thêm tại: <https://vi.wikipedia.org/wiki/XAMPP>

A black background with a black square

Description automatically generated with medium confidence

Hình 1.4 Hình ảnh minh họa về XAMPP

### Visual Studio Code

* VS Code hoặc VSC là 1 trình soạn thảo mã nguồn mở phổ biến được phát triển bởi Microsoft. Nó là 1 công cụ phát triển tích hợp mạnh mẽ được sử dụng rộng rãi trong lĩnh vực lập trình và phát triển phần mềm.
* **Những tính năng nổi bật của Visual Studio Code:**
* Hỗ trợ nhiều ngôn ngữ lập trình

Là trình biên tập lập trình code cho nên Visual Studio Code được hỗ trợ khá nhiều ngôn ngữ lập trình: C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript, … Cho nên khi sử dụng, nó dễ dàng phát hiện và đưa ra thông báo nếu chương chương trình xảy ra lỗi.

* Visual Studio Code hỗ trợ đa nền tảng

Visual Studio Code có thể hỗ trợ tên đa nền tảng: Windows, Linux và macOS.

* Cung cấp kho tiện ích mở rộng

Visual Studio Code còn cung cấp kho tiện ích mở rộng khá hữu ích đối với các lập trình viên. Khi họ muốn sử dụng một ngôn ngữ lập trình không nằm trong số các ngôn ngữ Visual Studio hỗ trợ, họ có thể tải xuống tiện ích mở rộng mà không hề làm giảm hiệu năng của phần mềm.

* Kho lưu trữ an toàn

Visual Studio Code được nhận xét là một kho lưu trữ vô cùng an toàn. Người dùng có thể hoàn toàn yên tâm vì nó dễ dàng kết nối với Git hoặc bất kỳ kho lưu trữ hiện có nào.

* Hỗ trợ website

Visual Studio Code hỗ trợ nhiều ứng dụng website. Ngoài ra, nó cũng có một trình soạn thảo và thiết kế website không hề khó sử dụng.

* Lưu trữ dữ liệu dạng phân cấp

Visual Studio Code cung cấp các thư mục cho một số tệp đặc biệt quan trọng. Đặc biệt, trình biên tập này còn có khả năng lưu trữ dữ liệu dạng phân cấp.

* Hỗ trợ viết Code

Đây là một tính năng khá hay và hỗ trợ rất nhiều cho các lập trình viên. Visual Studio Code sẽ đề xuất cho lập trình viên các tùy chọn thay thế nếu có sự thay đổi chút ít.

* Hỗ trợ thiết bị đầu cuối

Visual Studio Code có tích hợp thiết bị đầu cuối, giúp người dùng không cần tab chuyển đổi giữa hai màn hình hoặc trở về thư mục gốc khi thực hiện các thao tác.

* Màn hình đa nhiệm

Người dùng Visual Studio Code có thể mở cùng lúc nhiều tệp tin và thư mục không có liên quan đến nhau.

* Hỗ trợ Git

Visual Studio Code hỗ trợ kéo/sao chép mã trực tiếp từ GitHub. Các mã này có thể thay đổi và lưu lại.

* Tham khảo tại: <https://vi.wikipedia.org/wiki/Visual_Studio_Code>

### JavaScript

* JavaScript là ngôn ngữ lập trình được nhà phát triển sử dụng để tạo trang web tương tác. Từ làm mới bảng tin trên trang mạng xã hội đến hiển thị hình ảnh động và bản đồ tương tác, các chức năng của JavaScript có thể cải thiện trải nghiệm người dùng của trang web.
* Tác dụng của JavaScript

Sử dụng JavaScript, các trình duyệt có thể phản hồi tương tác của người dùng và

thay đổi bố cục của nội dung trên trang web.

* Cách hoạt động của JavaScript

Công cụ JavaScript là một chương trình máy tính chạy mã JavaScript. Các công cụ JavaScript đầu tiên chỉ là bộ diễn giải, nhưng tất cả các công cụ hiện đại đều sử dụng biên dịch tại chỗ hoặc thời gian chạy để cải thiện hiệu suất.

JavaScript phía máy khách đề cập đến cách thức JavaScript hoạt động trong trình duyệt của bạn. Trong trường hợp này, công cụ JavaScript nằm trong mã trình duyệt. Tất cả các trình duyệt web phổ biến đều có các công cụ JavaScript tích hợp riêng.

JavaScript phía máy chủ đề cập đến việc sử dụng ngôn ngữ viết mã trong logic máy chủ back-end. Trong trường hợp này, công cụ JavaScript nằm trực tiếp trên máy chủ.

### HTML & CSS

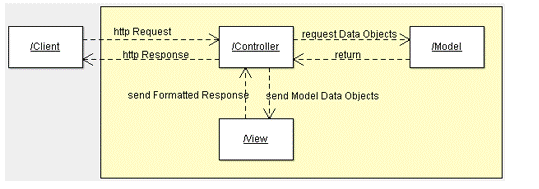
Ngôn ngữ đánh dấu siêu văn bản (HTML) và Ngôn ngữ định kiểu theo tầng (CSS) là hai ngôn ngữ lập trình khác mà các nhà phát triển sử dụng trong việc phát triển frontend. HTML là khối dựng cơ bản của hầu hết các trang web. Tất cả các đoạn văn, phần mục, hình ảnh, tiêu đề và văn bản đều được viết bằng HTML. Nội dung sẽ xuất hiện trên trang web theo thứ tự được viết bằng HTML.

CSS là một ngôn ngữ của các quy tắc định kiểu mà chúng tôi sử dụng để định kiểu cho nội dung HTML của mình. Bạn có thể sử dụng CSS để thiết kế các phần tử của trang web như màu nền, phông chữ, cột và đường viền.

### Mô hình (MVC) Model – View – Controller

Mẫu kiến trúc Model-View-Controller là phương pháp chia nhỏ các các thành phần dữ liệu (data), trình bày (output) và dữ liệu nhập từ người dùng (input) thành những thành phần riêng biệt.

Mô hình MVC (viết tắt của **Model View Controller**) chính cái tên của nó cũng đã thể hiện được khái niệm, MVC là mô hình gồm 3 lớp: Model, View, Controller, ta có thể mô tả lại hoạt động của mô hình MVC thông qua sơ đồ sau:



Hình 1.5 Hình ảnh minh họa về MVC

# CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

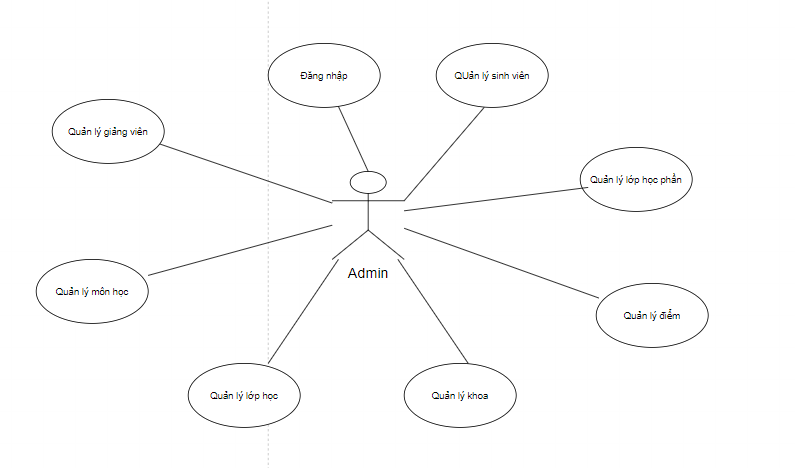
## 2.1. Phân tích

### 2.1.1. Sơ đồ chức năng

#### Hình 2.1. Sơ đồ chức năng

### 2.1.2. Sơ đồ use-case Admin

**Sơ đồ tổng quát**



#### Hình 2.2. Sơ đồ use-case tổng quát trang admin

|  |  |  |
| --- | --- | --- |
| STT | Tên use case | Mô tả |
| 1 | Đăng nhập | Cho phép người dùng đăng nhập vào hệ thống bằng cách cung cấp tên người dùng và mật khẩu. |
| 2 | Quản lý sinh viên | Admin có thể xem, thêm, sửa, xóa sinh viên |
| 3 | Quản lý giảng viên | Admin có thể xem, thêm, sửa, xóa giảng viên |
| 4 | Quản lý học phần | Admin có thể xem, thêm, sửa, xóa lớp học phần |
| 5 | Quản lý điểm | Admin có thể xem, thêm, sửa, xóa điểm |
| 6 | Quản lý khoa | Admin có thể xem, thêm, sửa, xóa khoa |
| 7 | Quản lý lớp học | Admin có thể xem, thêm, sửa, xóa lớp học |
| 8 | Quản lý môn học | Admin có thể xem, thêm, sửa, xóa môn học |

* Chức năng đăng nhập

A diagram of a diagram

Description automatically generated

#### Hình 2.3. Sơ đồ usecase chức năng Đăng nhập

|  |  |
| --- | --- |
| Usecase | Đăng nhập |
| Tác nhân | Admin |
| Mô tả ngắn | Admin đăng nhập vào hệ thống, truy cập hệ thống với đúng quyền của tài khoản |
| Chuỗi sự kiện chính | 1, Người dùng chọn đăng nhập  2, Hệ thống yêu cầu nhập tài khoản và mật khẩu  3, Người dùng nhập tài khoản và nhập khẩu  4. Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 3  5. Lấy quyền truy cập tài khoản  6. Hiển thị giao diện tương ứng với quyền của tài khoản |
| Điều kiện đầu ra | Người dùng truy cập hệ thống với đúng quyền của tài khoản đã đăng nhập |

|  |  |
| --- | --- |
| Usecase | Đăng xuất |
| Tác nhân | Admin |
| Mô tả ngắn | Admin muốn đăng xuất khỏi hệ thống |
| Chuỗi sự kiện chính | 1, Admin chọn đăng xuất  2, Hệ thống hủy lưu tài khoản  3, Hệ thống hiển thị giao diện trang đăng nhập |
| Điều kiện đầu ra | Hủy lưu tài khoản người dùng, hiển thị trang đăng nhập |

* Quản lý sinh viên

A diagram of a diagram

Description automatically generated

#### Hình 2.4. Sơ đồ usecase Quản lý sinh viên

|  |  |
| --- | --- |
| Usecase | Thêm thông tin sinh viên |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện thêm các sinh viên, giúp phân loại các sinh viên dễ dàng |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý sinh viên  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý sinh viên  3, Admin chọn nút Thêm sinh viên  4, Hệ thống hiển thị form thêm sinh viên  5, Admin nhập các trường dữ liệu, nhấn nút Thêm mới  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo thêm sinh viên thành công, và lưu thông tin sinh viên trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin sinh viên được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Sửa sinh viên |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện sửa thông tin sinh viên |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý sinh viên  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý sinh viên  3, Admin chọn danh sách sinh viên, chọn sửa  4, Hệ thống lấy dữ liệu sinh viên theo ID, và hiển thị form sửa sinh viên  5, Admin nhập các trường dữ liệu, nhấn nút Sửa thông tin  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo sửa sinh viên thành công, và cập nhật thông tin sinh viên trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin sinh viên được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Xóa sinh viên |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện xóa các sinh viên không quản lý |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý sinh viên  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý sinh viên  3, Admin chọn danh sách học sinh, chọn Xóa  4, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  5, Hệ thống báo xóa sinh viên thành công, và cập nhật trạng thái xóa sinh viên trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin sinh viên được xóa khỏi cơ sở dữ liệu |

* Quản lý giảng viên

A diagram of a diagram

Description automatically generated

#### Hình 2.5. Sơ đồ usecase Quản lý giảng viên

|  |  |
| --- | --- |
| Usecase | Thêm thông tin giảng viên |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện thêm các giảng viên, giúp phân loại giảng viên dễ dàng |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý giảng viên  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý sinh viên  3, Admin chọn nút Thêm giảng viên  4, Hệ thống hiển thị form thêm giảng viên  5, Admin nhập các trường dữ liệu, nhấn nút Thêm mới  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo thêm giảng viên thành công, và lưu thông tin giảng viên trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin giảng viên được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Sửa thông tin giảng viên |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện sửa thông tin giảng viên |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý giảng viên  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý giảng viên  3, Admin chọn danh sách giảng viên, chọn sửa  4, Hệ thống lấy dữ liệu giảng viên theo ID, và hiển thị form sửa giảng viên  5, Admin nhập các trường dữ liệu, nhấn nút Sửa thông tin  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo sửa giảng viên thành công, và cập nhật thông tin giảng viên trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin giảng viên được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Xóa giảng viên |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện xóa các giảng viên không quản lý |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý giảng viên  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý giảng viên  3, Admin chọn danh sách giảng viên, chọn Xóa  4, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  5, Hệ thống báo xóa giảng viên thành công, và cập nhật trạng thái xóa giảng viên trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin giảng viên được xóa khỏi cơ sở dữ liệu |

* Quản lý học phần

A diagram of a diagram

Description automatically generated

#### Hình 2.6. Sơ đồ usecase Quản lý học phần

|  |  |
| --- | --- |
| Usecase | Thêm thông tin lớp học phần |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện thêm các lớp học phần, giúp phân loại các lớp học phần dễ dàng |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý lớp học phần  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý lớp học phần  3, Admin chọn nút Thêm lớp học phần  4, Hệ thống hiển thị form thêm lớp học phần  5, Admin nhập các trường dữ liệu, nhấn nút Thêm mới  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo thêm lớp học phần thành công, và lưu thông tin lớp học phần trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin lớp học phần được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Sửa thông tin lớp học phần |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện sửa thông tin lớp học phần |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý lớp học phần  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý lớp học phần  3, Admin chọn danh sách lớp học phần, chọn Chi tiết  4, Hệ thống lấy dữ liệu lớp học phần theo ID, và hiển thị form sửa lớp học phần  5, Admin nhập các trường dữ liệu, nhấn nút Sửa thông tin  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo sửa lớp học phần thành công, và cập nhật thông tin lớp học phần trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin lớp học phần được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Xóa lớp học phần |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện xóa các lớp học phần không quản lý |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý lớp học phần  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý lớp học phần  3, Admin chọn danh sách lớp học phần, chọn Xóa  4, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  5, Hệ thống báo xóa lớp học phần thành công, và cập nhật trạng thái xóa lớp học phần trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin lớp học phần được xóa khỏi cơ sở dữ liệu |

* Quản lý điểm

A diagram of a diagram

Description automatically generated with medium confidence

#### Hình 2.7. Sơ đồ usecase Quản lý điểm

|  |  |
| --- | --- |
| Usecase | Thêm điểm của sinh viên |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện thêm điểm của sinh viên |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý điểm  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý lớp học phần  3, Admin chọn nút Mã lớp hoặc Tên lớp  4, Hệ thống hiển thị danh sách điểm môn học đó của tất cả sinh viên  5, Admin chọn nút Quản lý điểm  5, Admin nhập các trường dữ liệu, nhấn nút Cập nhật  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo thêm sinh viên thành công, và lưu thông tin danh mục trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin điểm của sinh viên được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Sửa điểm của sinh viên |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện sửa điểm sinh viên |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý điểm  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý lớp học phần  3, Admin chọn nút Mã lớp hoặc Tên lớp  4, Hệ thống hiển thị danh sách điểm môn học đó của tất cả sinh viên  5, Admin chọn nút Quản lý điểm  5, Admin nhập các trường dữ liệu, nhấn nút Cập nhật  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo thêm sinh viên thành công, và lưu thông tin danh mục trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin điểm của sinh viên được cập nhật vào cơ sở dữ liệu |

* Quản lý khoa

A diagram of a diagram

Description automatically generated

#### Hình 2.8. Sơ đồ usecase Quản lý khoa

|  |  |
| --- | --- |
| Usecase | Thêm khoa |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện thêm các khoa, giúp phân loại các khoa dễ dàng |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý khoa  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý khoa  3, Admin chọn nút Thêm khoa  4, Hệ thống hiển thị form thêm khoa  5, Admin nhập các trường dữ liệu, nhấn nút Thêm mới  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo thêm khoa thành công, và lưu thông tin khoa trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin khoa được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Sửa thông tin khoa |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện sửa thông tin khoa |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý khoa  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý khoa  3, Admin chọn danh sách khoa, chọn sửa  4, Hệ thống lấy dữ liệu khoa theo ID, và hiển thị form sửa khoa  5, Admin nhập các trường dữ liệu, nhấn nút Sửa thông tin  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo sửa khoa thành công, và cập nhật thông tin khoa trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin khoa được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Xóa khoa |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện xóa các khoa không quản lý |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý khoa  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý khoa  3, Admin chọn danh sách khoa, chọn Xóa  4, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  5, Hệ thống báo xóa khoa thành công, và cập nhật trạng thái xóa khoa trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin khoa được xóa khỏi cơ sở dữ liệu |

* Quản lý lớp học

A diagram of a diagram

Description automatically generated

#### Hình 2.9. Sơ đồ usecase Quản lý lớp học

|  |  |
| --- | --- |
| Usecase | Thêm thông tin lớp học |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện thêm các lớp học, giúp phân loại các lớp học dễ dàng |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý lớp học  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý lớp học  3, Admin chọn nút Thêm mới  4, Hệ thống hiển thị form thêm lớp học  5, Admin nhập các trường dữ liệu, nhấn nút Thêm mới  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo thêm lớp học thành công, và lưu thông tin lớp học trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin lớp học được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Sửa thông tin lớp học |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện sửa thông tin sinh viên |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý lớp học  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý lớp học  3, Admin chọn danh sách lớp học, chọn sửa  4, Hệ thống lấy dữ liệu lớp học theo ID, và hiển thị form sửa lớp học  5, Admin nhập các trường dữ liệu, nhấn nút Sửa thông tin  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo sửa lớp học thành công, và cập nhật thông tin lớp học trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin lớp học được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Xóa lớp học |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện xóa các lớp học không quản lý |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý lớp học  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý lớp học  3, Admin chọn danh sách lớp học, chọn Xóa  4, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  5, Hệ thống báo xóa lớp học thành công, và cập nhật trạng thái xóa lớp học trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin lớp học được xóa khỏi cơ sở dữ liệu |

* Quản lý môn học

A diagram of a diagram

Description automatically generated

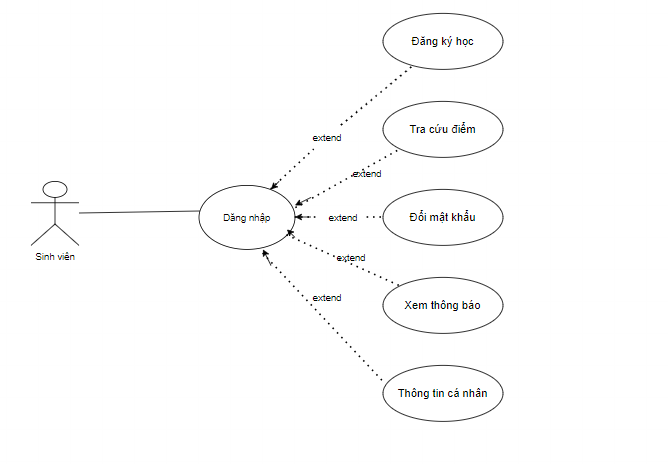
#### Hình 2.10. Sơ đồ usecase Quản lý môn học

|  |  |
| --- | --- |
| Usecase | Thêm môn học |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện thêm các môn học, giúp phân loại các môn học dễ dàng |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý môn học  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý môn học  3, Admin chọn nút Thêm mới  4, Hệ thống hiển thị form thêm môn học  5, Admin nhập các trường dữ liệu, nhấn nút Thêm mới  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo thêm môn học thành công, và lưu thông tin môn học trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin môn học được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Sửa thông tin môn học |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện sửa thông tin môn học |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý môn học  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý môn học  3, Admin chọn danh sách môn học, chọn sửa  4, Hệ thống lấy dữ liệu môn học theo ID, và hiển thị form sửa môn học  5, Admin nhập các trường dữ liệu, nhấn nút Sửa thông tin  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống báo sửa môn học thành công, và cập nhật thông tin môn học trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin môn học được cập nhật vào cơ sở dữ liệu |

|  |  |
| --- | --- |
| Usecase | Xóa môn học |
| Tác nhân | Admin |
| Mô tả ngắn | Admin thực hiện xóa các môn học không quản lý |
| Chuỗi sự kiện chính | 1, Admin đăng nhập hệ thống, chọn quản lý môn học  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý môn học  3, Admin chọn danh sách môn học, chọn Xóa  4, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  5, Hệ thống báo xóa môn học thành công, và cập nhật trạng thái xóa môn học trong cơ sở dữ liệu |
| Điều kiện đầu ra | Các thông tin môn học được xóa khỏi cơ sở dữ liệu |

* + 1. **Sơ đồ use-case User**



#### Hình 2.11. Sơ đồ use-case tổng quát trang User

|  |  |
| --- | --- |
| Usecase | Đăng nhập |
| Tác nhân | User |
| Mô tả ngắn | User đăng nhập vào hệ thống, truy cập hệ thống với đúng quyền của tài khoản |
| Chuỗi sự kiện chính | 1, Người dùng chọn đăng nhập  2, Hệ thống yêu cầu nhập tài khoản và mật khẩu  3, Người dùng nhập tài khoản và nhập khẩu  4. Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 3  5. Lấy quyền truy cập tài khoản  6. Hiển thị giao diện tương ứng với quyền của tài khoản |
| Điều kiện đầu ra | Người dùng truy cập hệ thống với đúng quyền của tài khoản đã đăng nhập |

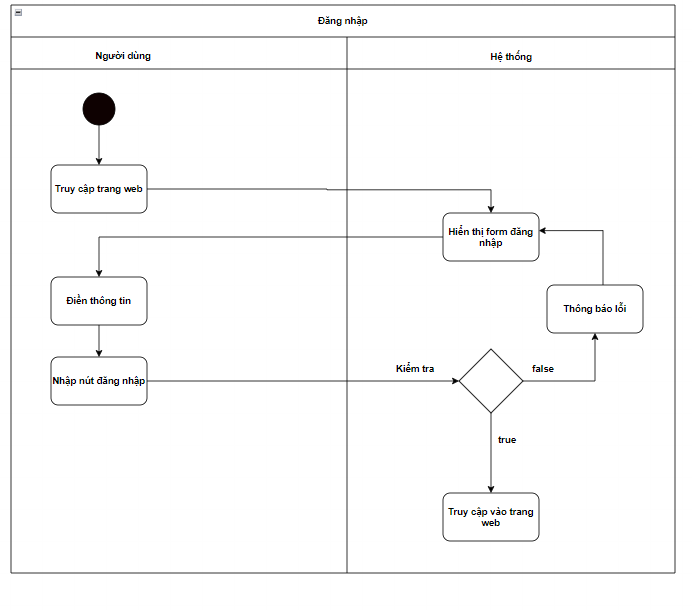
|  |  |
| --- | --- |
| Usecase | Đăng ký học |
| Tác nhân | User |
| Mô tả ngắn | User thực hiện đăng ký học |
| Chuỗi sự kiện chính | 1, User đăng nhập hệ thống, chọn đăng ký học  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang đăng ký học  3, User chọn lớp học phù hợp và chọn nút Đăng ký  4. Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 3  5. Hệ thống hiển thị thống báo đăng ký học thành công |
| Điều kiện đầu ra | User đã đăng ký học thành công và hiển thị danh sách lớp học user đã đăng ký |

|  |  |
| --- | --- |
| Usecase | Tra cứu điểm |
| Tác nhân | User |
| Mô tả ngắn | User thực hiện tra cứu điểm |
| Chuỗi sự kiện chính | 1, User đăng nhập hệ thống, chọn quản lý điểm  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý điểm  3, Hệ thống sẽ hiển thị tất các các điểm của các môn mà user đã thi  4, User chọn học kì  5, Hệ thống sẽ hiển thị danh sách các môn học đã có điểm trong học kỳ đó |
| Điều kiện đầu ra | User xem được điểm của môn học mà user muốn |

|  |  |
| --- | --- |
| Usecase | Đổi mật khẩu |
| Tác nhân | User |
| Mô tả ngắn | User thực hiện đổi mật khẩu |
| Chuỗi sự kiện chính | 1, User đăng nhập hệ thống, chọn quản lý tài khoản  2, Hệ thống kiểm tra tài khoản và quyền, nếu đúng thì truy cập hệ thống, chuyển hướng tới trang quản lý tài khoản  3, User chọn đổi mật khẩu  4, Hệ thống hiển thị form đổi mật khẩu  5, User thực hiện nhập dữ liệu và chọn Thay đổi  6, Hệ thống kiểm tra dữ liệu: Nếu thỏa mãn thực hiện bước tiếp theo. Ngược lại, thông báo lỗi và quay lại bước 5  7, Hệ thống thông báo đổi mật khẩu thành công, và mật khẩu mới được cập nhật trong cơ sở dữ liệu |
| Điều kiện đầu ra | Mật khẩu được cập nhật trong cơ sở dữ liệu |

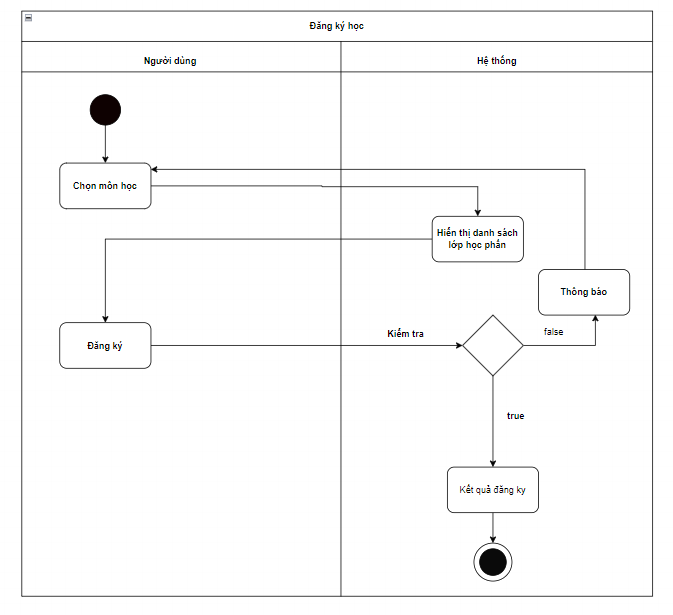
### Sơ đồ hoạt động (Activity)

a, Sơ đồ hoạt động đăng nhập



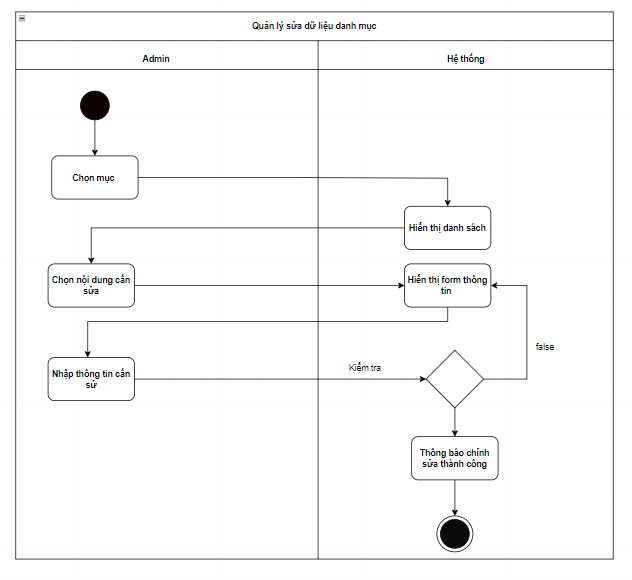
#### Hình 2.12. Sơ đồ hoạt động đăng nhập

b, Sơ đồ hoạt động đăng ký học



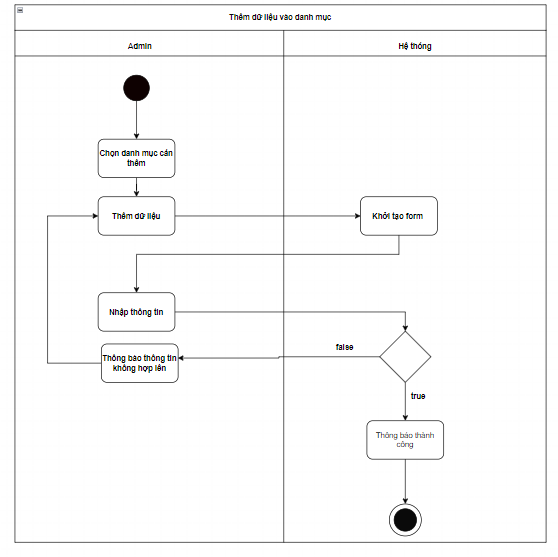
#### Hình 2.13. Sơ đồ hoạt động đăng ký học

c, Sơ đồ hoạt động sửa danh mục

****

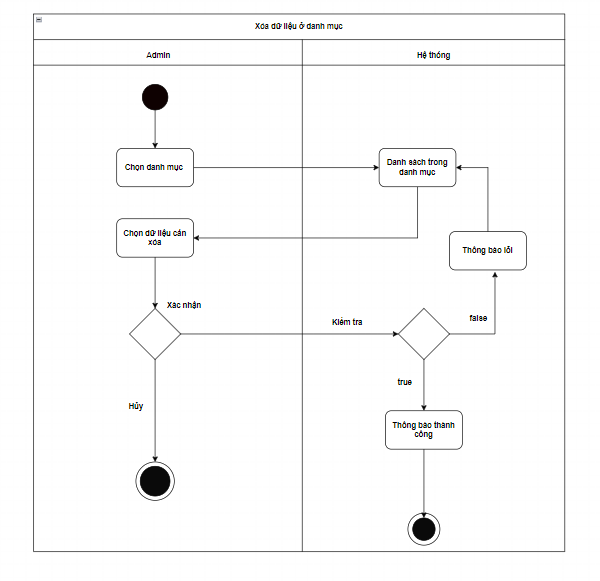
#### Hình 2.14. Sơ đồ hoạt động sửa danh mục

d, Sơ đồ hoạt động thêm danh mục

****

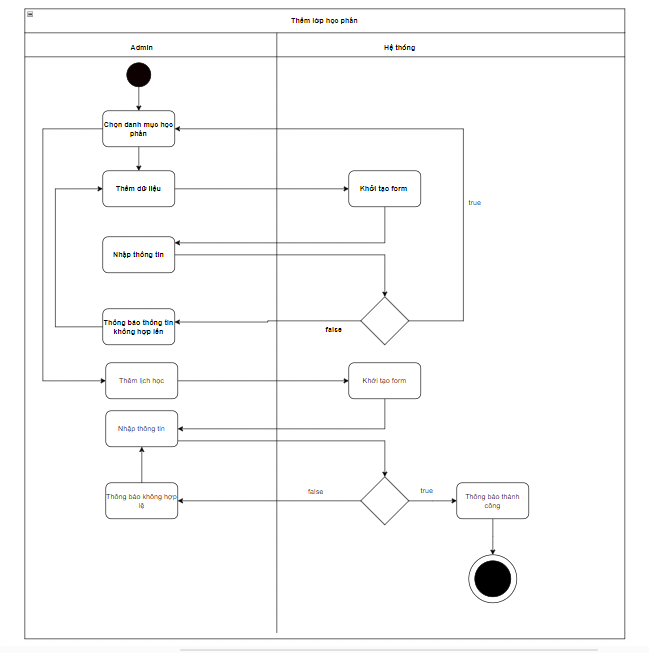
#### Hình 2.15. Sơ đồ hoạt động thêm dữ liệu danh mục

e, Sơ đồ hoạt động xóa dữ liệu trong danh mục

****

#### Hình 2.16. Sơ đồ hoạt động xóa dữ liệu trong danh mục

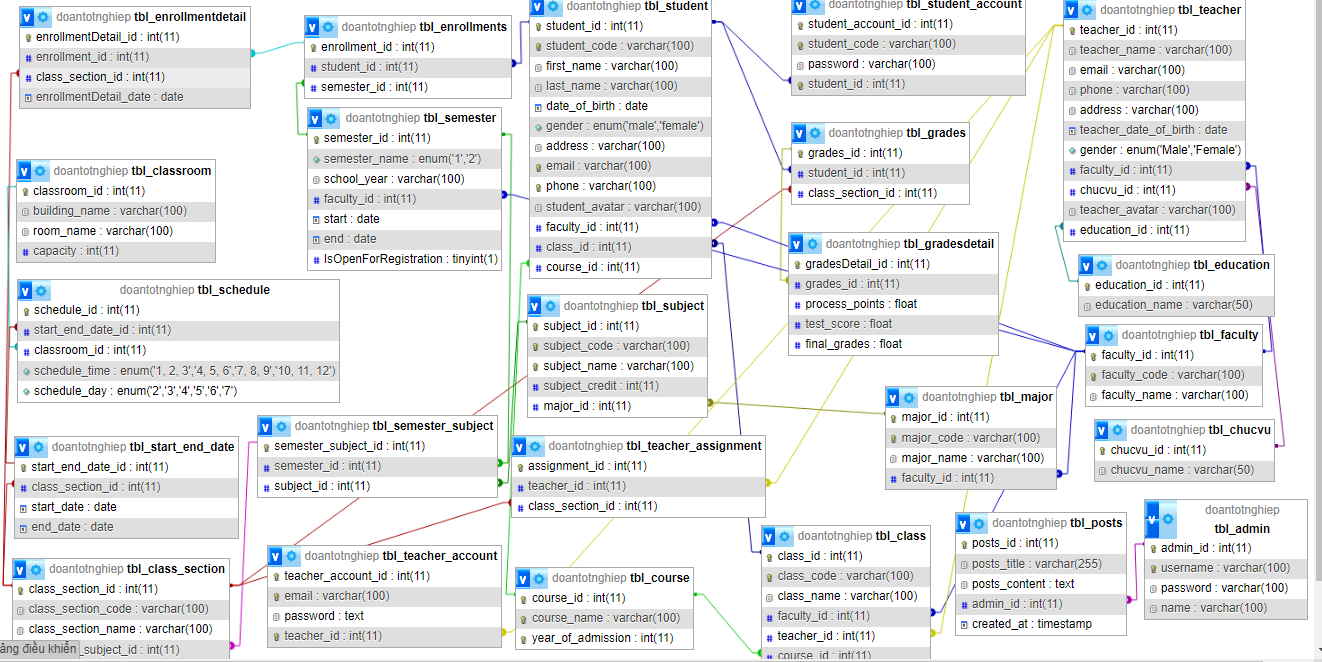
f, Sơ đồ thêm lớp học phần



#### Hình 2.17. Sơ đồ hoạt đông thêm lớp học phần

## Thiết kế cơ sở dữ liệu

### 2.2.1. ERD



#### Hình 2.18. Sơ đồ ERD

### 2.2.2. Mô tả các bảng

##### Bảng 3.1. Bảng “tbl\_admin”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| admin\_id | int(11) | No | PRIMARY KEY – mã admin |
| username | varchar(100) | No | Tên tài khoản |
| password | varchar(100) | No | Mật khẩu |
| name | varchar(100) | No | Họ tên |

##### Bảng 3.2. Bảng “tbl\_chucvu”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| chucvu\_id | int(11) | No | PRIMARY KEY – |
| chucvu\_name | varchar(50) | No | Tên chức vụ |

##### Bảng 3.3. Bảng “tbl\_class”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| class\_id | int(11) | No | PRIMARY KEY – |
| class\_code | varchar(100) | No | Mã lớp học |
| class\_name | varchar(100) | No | Tên lớp học |
| faculty\_id | int(11) | No | Mã khoa |
| Teacher\_id | int(11) | No | Mã giảng viên |
| course\_id | int(11) | Yes | Mã khóa |

##### Bảng 3.4. Bảng “tbl\_classroom”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| classroom\_id | int(11) | No | PRIMARY KEY – |
| building\_name | varchar(100) | No | Tên tòa |
| room\_name | varchar(100) | No | Tên phòng |
| capacity | int(11) | No | Số lương sinh viên có thể chứa |

##### Bảng 3.5. Bảng “tbl\_class\_section”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| class\_section\_id | int(11) | No | PRIMARY KEY – |
| class\_section\_code | varchar(100) | No | Mã Lớp học phần |
| class\_section\_name | varchar(100) | No | Tên lớp học phần |
| semester\_subject\_id | int(11) | No | Mã học kì |
| class\_section\_capacity | int(11) | No | Số lượng sinh viên của lớp |

##### Bảng 3.6. Bảng “tbl\_course”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| course\_id | int(11) | No | PRIMARY KEY – |
| course\_name | varchar(100) | No | Tên khóa |
| year\_of\_admission | int(11) | No | Năm học |

##### Bảng 3.7. Bảng “tbl\_education”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| education\_id | int(11) | No | PRIMARY KEY – |
| education\_name | varchar(50) | No | Tên trình độ |

##### Bảng 3.8. Bảng “tbl\_enrollmentdetail”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| enrollmentDetail\_id | int(11) | No | PRIMARY KEY– |
| enrollment\_id | int(11) | No | Mã ghi danh |
| class\_section\_id | int(11) | No | Mã lớp học phần |
| enrollmentDetail\_date | date | No | Ngày đăng ký |

##### Bảng 3.9. Bảng “tbl\_enrollments”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| enrollment\_id | int(11) | No | PRIMARY KEY – |
| student\_id | int(11) | No | Mã sinh viên |
| semester\_id | Int(11) | No | Mã học kì |

##### Bảng 3.10. Bảng “tbl\_faculty”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| faculty\_id | int(11) | No | PRIMARY KEY – |
| faculty\_code | varchar(100) | No | Mã code khoa |
| faculty\_name | Varchar(100) | No | Tên khoa |

##### Bảng 3.11. Bảng “tbl\_grades”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| grades\_id | int(11) | No | PRIMARY KEY – |
| student\_id | int(11) | No | Mã sinh viên |
| class\_section\_id | int(11) | No | Mã lớp học phần |

##### Bảng 3.12. Bảng “tbl\_gradesdetail”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| gradesDetail\_id | int(11) | No | PRIMARY KEY – |
| grades\_id | Int(11) | No | Mã điểm trung gian |
| process\_points | float | Yes | Điểm thành phần |
| test\_score | float | Yes | Điểm thi |
| final\_grades | float | Yes | Điểm trung bình |

##### Bảng 3.13. Bảng “tbl\_major”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| major\_id | int(11) | No | PRIMARY KEY – |
| major\_code | varchar(100) | Yes | Mã code chuyên ngành |
| major\_name | varchar(100) | No | Tên chuyên ngành |
| faculty\_id | int(11) | No | Mã khoa |

##### Bảng 3.14. Bảng “tbl\_posts”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| posts\_id | int(11) | No | PRIMARY KEY – |
| posts\_title | varchar(255) | No | Tiêu đề bài báo |
| posts\_content | text | No | Nội dung bài báo |

##### Bảng 3.15. Bảng “tbl\_schedule”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| schedule\_id | int(11) | No | PRIMARY KEY - |
| start\_end\_date\_id | int(11) | No | Mã lịch học |
| classroom\_id | int(11) | No | Mã phòng học |
| schedule\_time | enum | No | Tiết học |
| schedule\_day | enum | No | Ngày thứ |

##### Bảng 3.16. Bảng “tbl\_semester”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| semester\_id | int(11) | No | PRIMARY KEY – |
| semester\_name | enum | No | Tên học kì |
| school\_year | varchar(100) | No | Năm học |
| faculty\_id | int(11) | No | Mã khoa |
| start | date | Yes | Ngày bắt đầu mở đăng ký |
| end | date | Yes | Ngày kết thúc đăng ký |
| isopenforregistration | tinyint(1) | No | Trạng thái |

##### Bảng 3.17. Bảng “tbl\_semester\_subject”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| semester\_subject\_id | int(11) | No | PRIMARY KEY – |
| semester\_id | int(11) | No | Mã học kì |
| subject\_id | int(11) | No | Mã môn học |

##### Bảng 3.18. Bảng “tbl\_start\_end\_date”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| start\_end\_date\_id | int(11) | No | PRIMARY KEY – |
| class\_section\_id | int(11) | No | Mã lớp học phần |
| start\_date | date | No | Ngày bắt đầu học |
| end\_date | date | No | Ngày Kết thúc |

##### Bảng 3.19. Bảng “tbl\_student”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| student\_id | int(11) | No | PRIMARY KEY – |
| student\_code | varchar(100) | No | Mã code sinh viên |
| first\_name | varchar(100) | No | Họ sinh viên |
| last\_name | varchar(100) | No | Tên sinh viên |
| date\_of\_birth | date | No | Ngày sinh |
| gender | enum | No | Giới tính |
| address | varchar(100) | No | Địa chỉ |
| email | varchar(100) | No | Email |
| phone | varchar(100) | No | Số điện thoại |
| student\_avatar | varchar(100) | Yes | Ảnh sinh viên |
| class\_id | int(11) | No | Mã lớp học |
| faculty\_id | int(11) | No | Mã khoa |
| course\_id | int(11) | No | Mã khóas |

##### Bảng 3.20. Bảng “tbl\_student\_account”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| student\_account\_id | int(11) | No | PRIMARY KEY – |
| student\_code | varchar(100) | No | Mã code sinh viên |
| password | varchar(100) | No | Mật khẩu |
| student\_id | int(11) | No | Mã sinh viên |

##### Bảng 3.21. Bảng “tbl\_subject”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| subject\_id | int(11) | No | PRIMARY KEY – |
| subject\_code | varchar(100) | No | Mã code môn học |
| subject\_name | varchar(100) | No | Tên môn học |
| subject\_credit | int(11) | No | Tín chỉ |
| major\_id | int(11) | No | Mã chuyên ngành |

##### Bảng 3.22. Bảng “tbl\_teacher”

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| teacher\_id | int(11) | No | PRIMARY KEY – |
| teacher\_code | varchar(100) | No | Mã code giảng viên |
| teacher\_name | varchar(100) | No | Tên giảng viên |
| teacher\_date\_of\_birth | date | No | Ngày inh |
| gender | enum | No | Giói tính |
| address | varchar(100) | No | Địa chỉ |
| email | varchar(100) | No | Email |
| phone | varchar(100) | No | Số điện thoại |
| teacher\_avatar | varchar(100) | No | Ảnh |
| faculty\_id | int(11) | No | Mã khoa |
| chucvu\_id | int(11) | No | Mã chức vụ |
| education\_id | int(11) | No | Mã học vấn |

##### Bảng 3.23. Bảng “tbl\_teacher\_account”

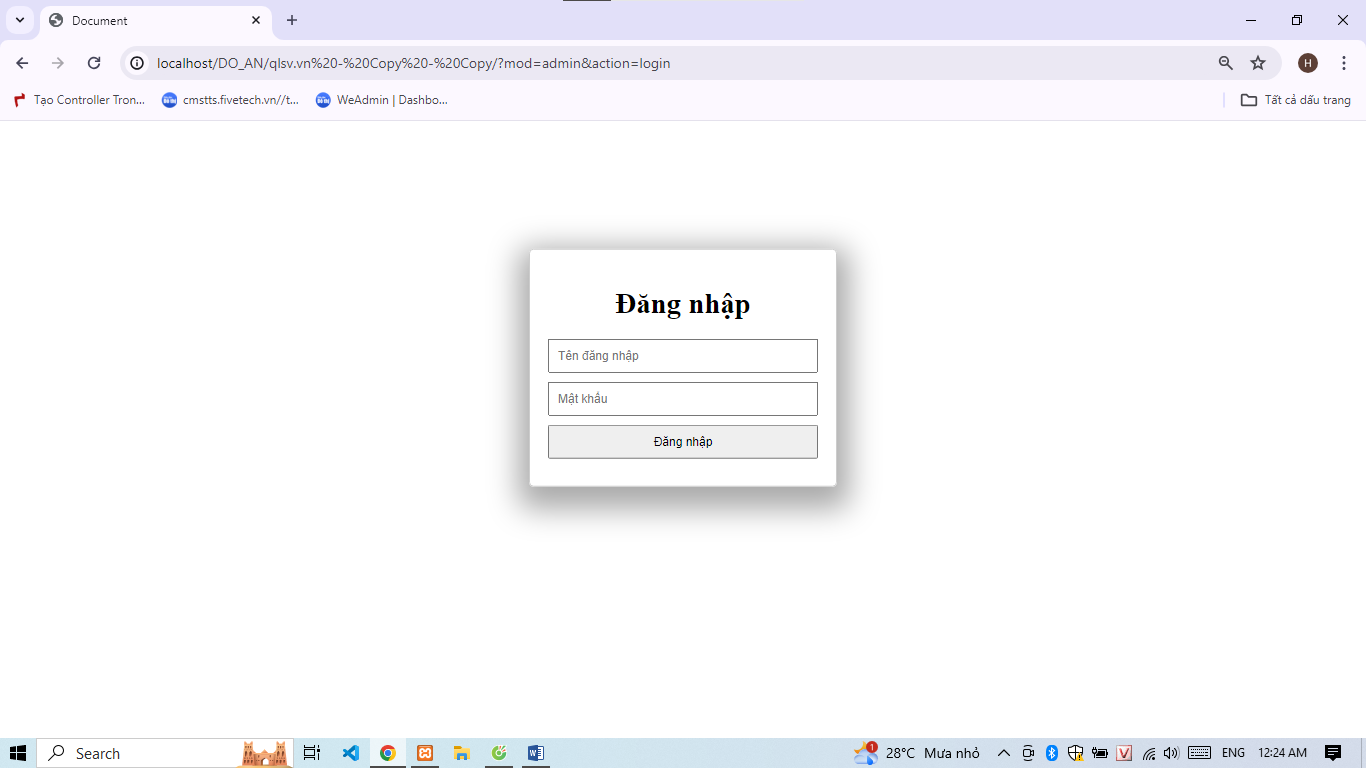
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| teacher\_account\_id | int(11) | No | PRIMARY KEY – |
| email | varchar(100) | No | Tên tài khoản |
| password | text | No | Mật khẩu |
| teacher\_id | int(11) | No | Mã giảng viên |

##### Bảng 3.24. Bảng “tbl\_teacher\_assignment”

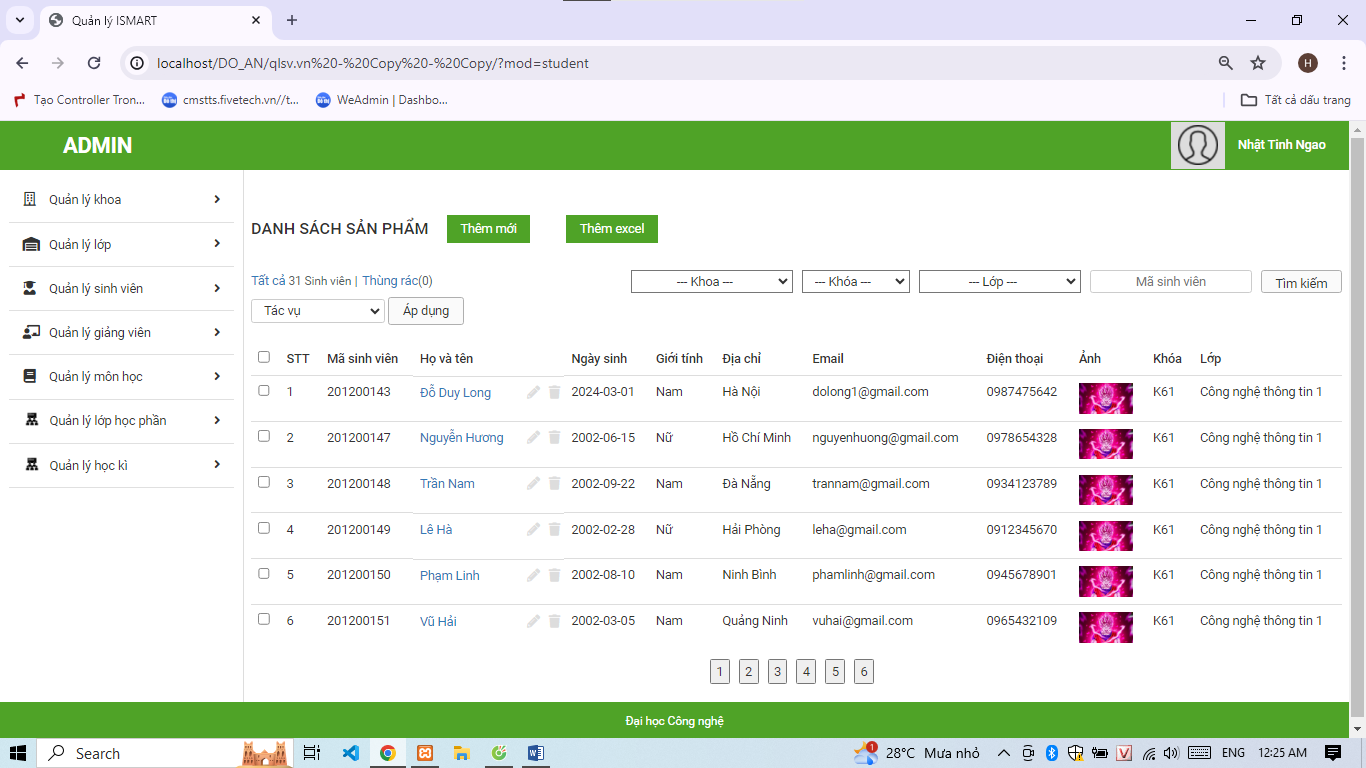
|  |  |  |  |
| --- | --- | --- | --- |
| **Tên cột** | **Kiểu dữ liệu** | **Null** | **Mô tả** |
| assignment\_id | int(11) | No | PRIMARY KEY – |
| teacher\_id | int(11) | No | Mã giảng viên |
| class\_section\_id | int(11) | No | Lớp học phần |

# CHƯƠNG 3: CÀI ĐẶT CHƯƠNG TRÌNH

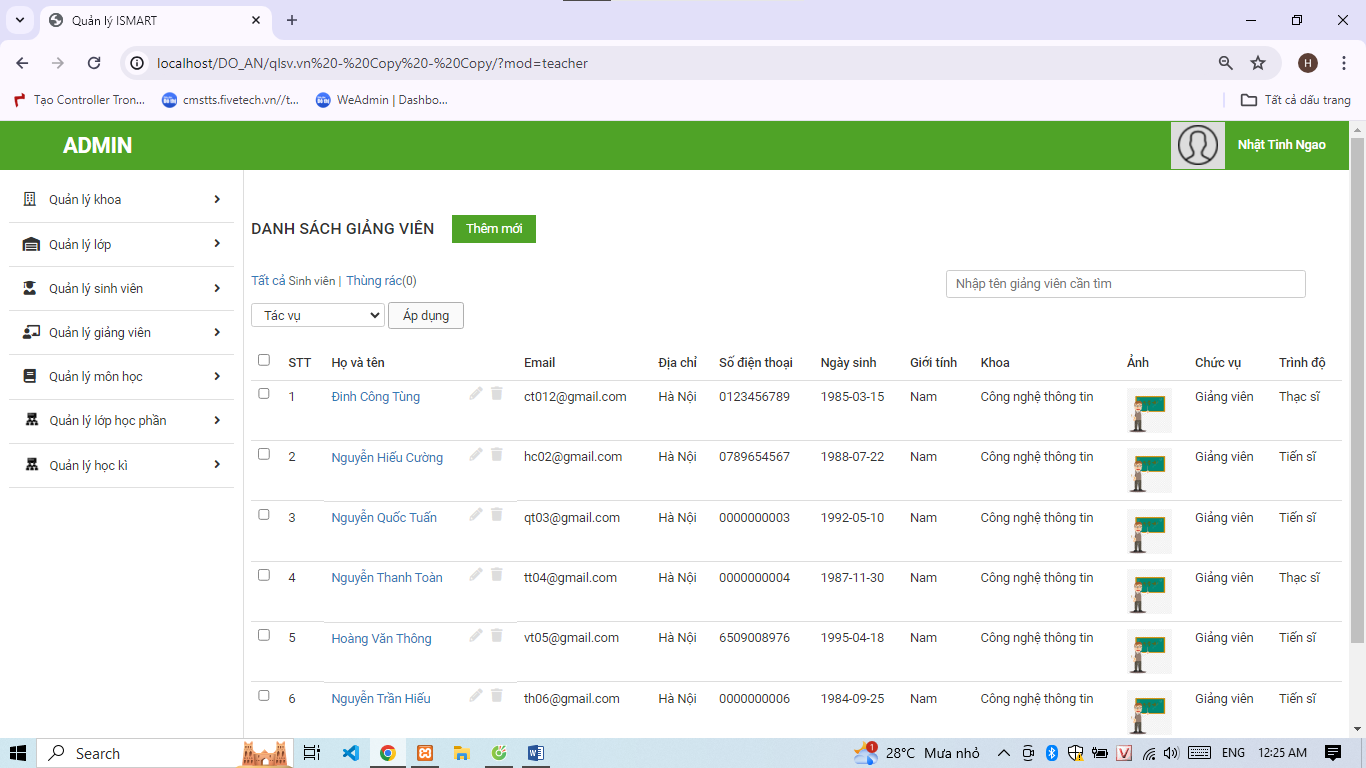
## 3.1. Xây dựng trang Admin



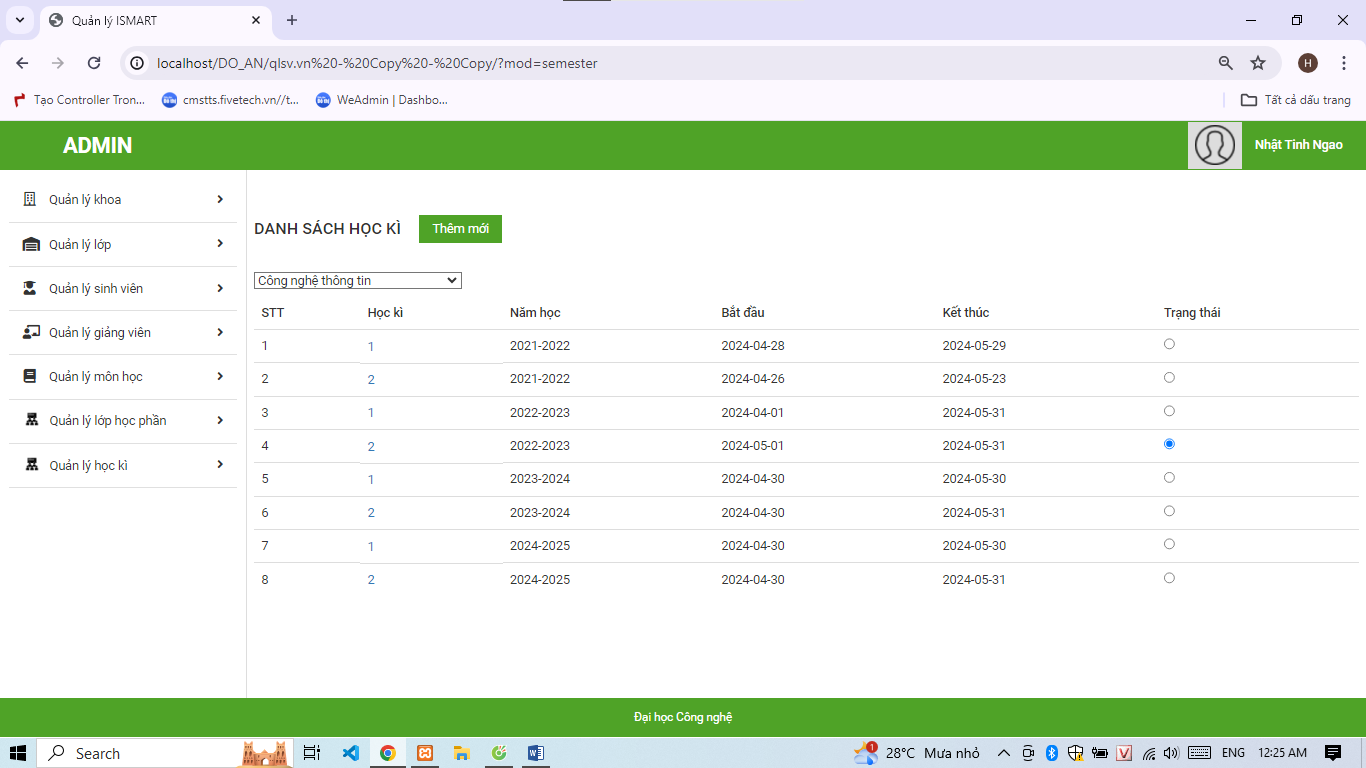
#### Hình 3.1. Giao diện trang Đăng nhập của Admin



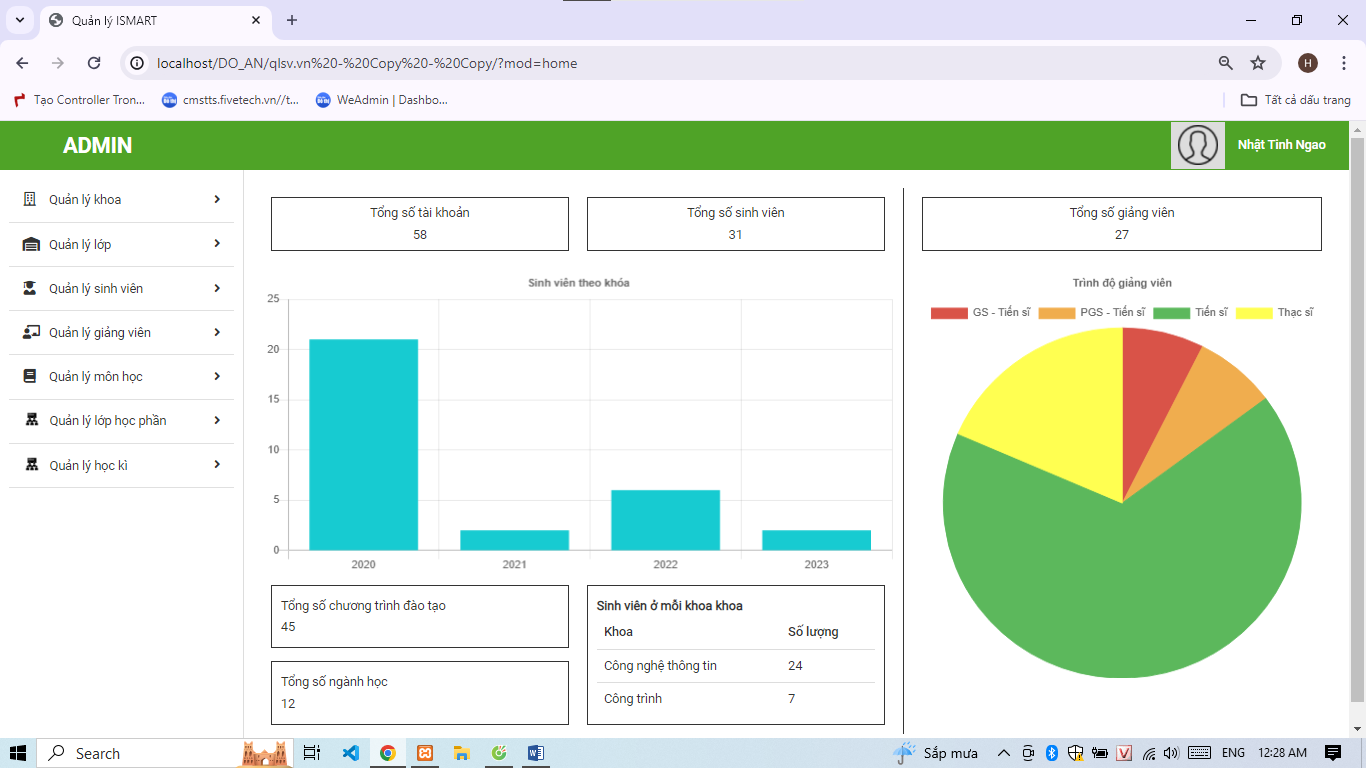
#### Hình 3.2. Giao diện Quản lý Sinh viên



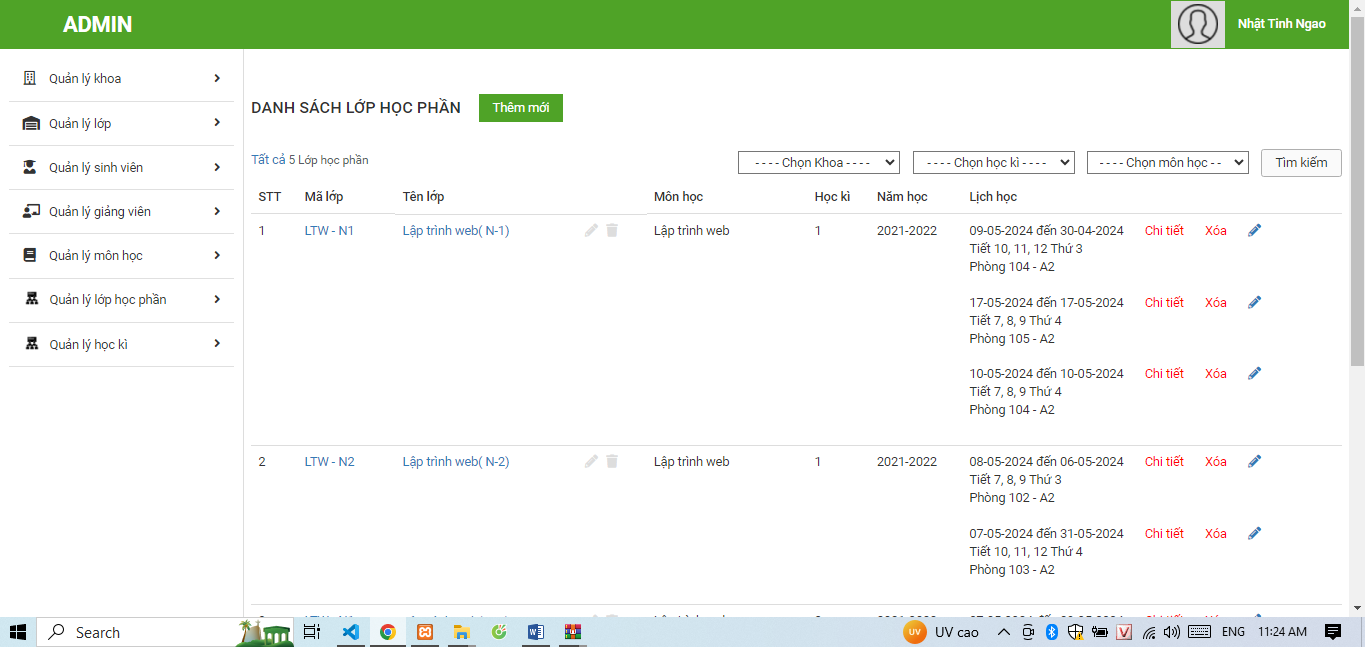
#### Hình 3.3. Giao diện Quản lý Giảng viên



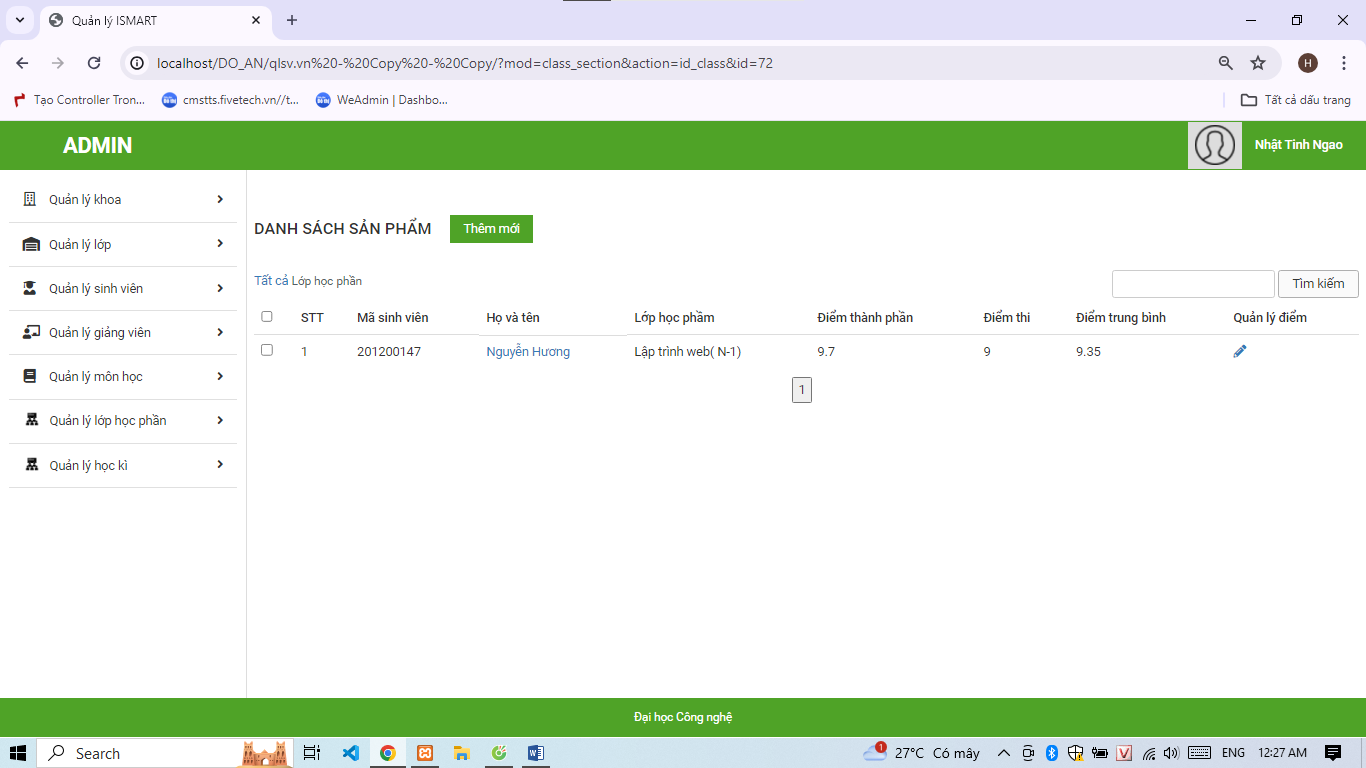
#### Hình 3.4. Giao diện Quản lý học kỳ



#### Hình 3.5. Giao diện Biểu đồ tổng quan

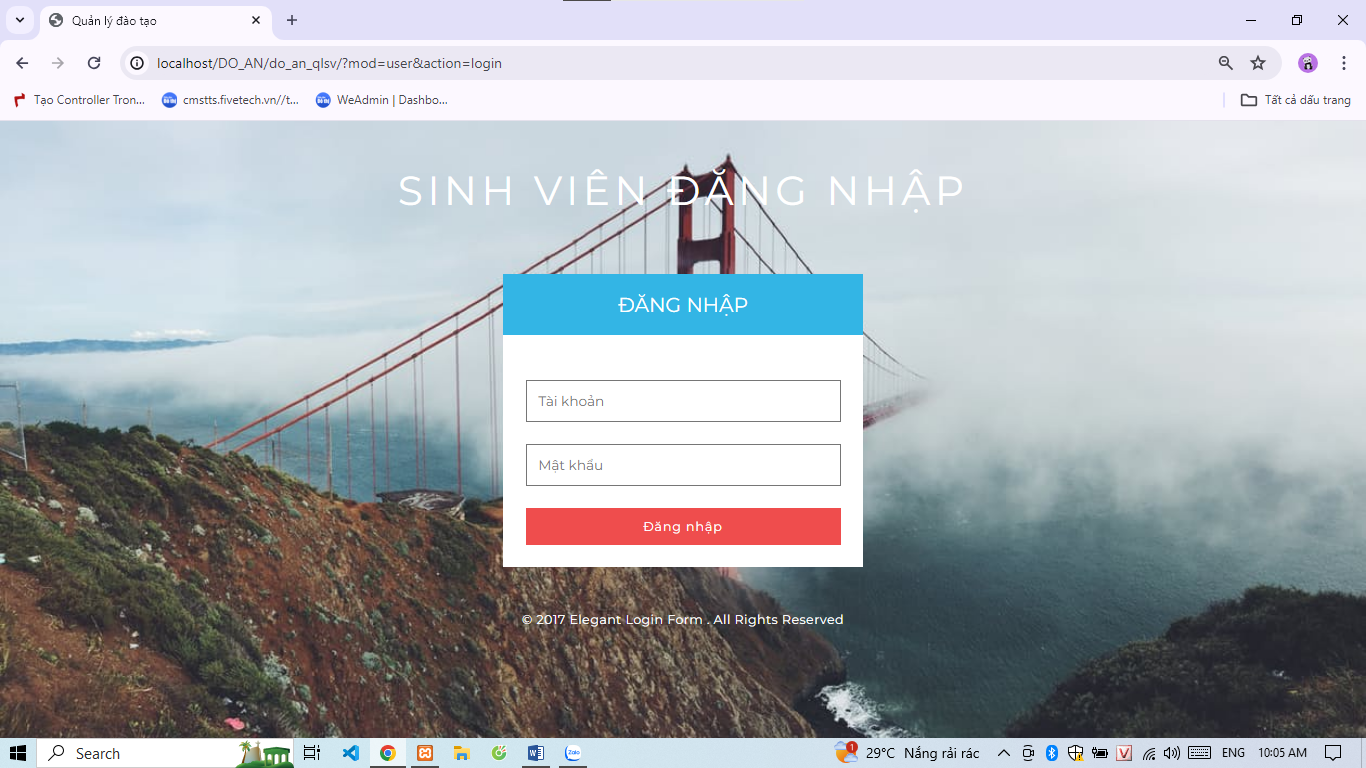


#### Hình 3.6. Giao diện Quản lý lớp học phần

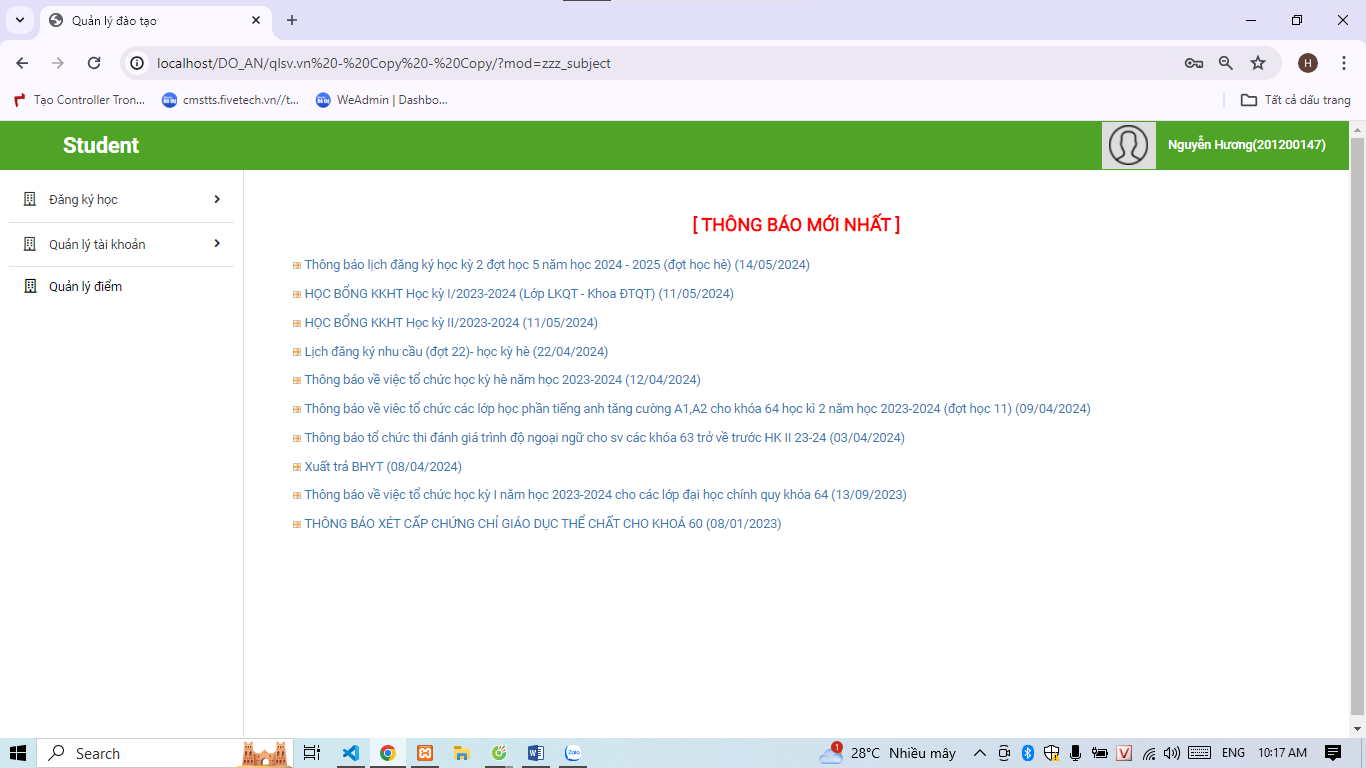


#### Hình 3.7. Giao diện Quản lý điểm

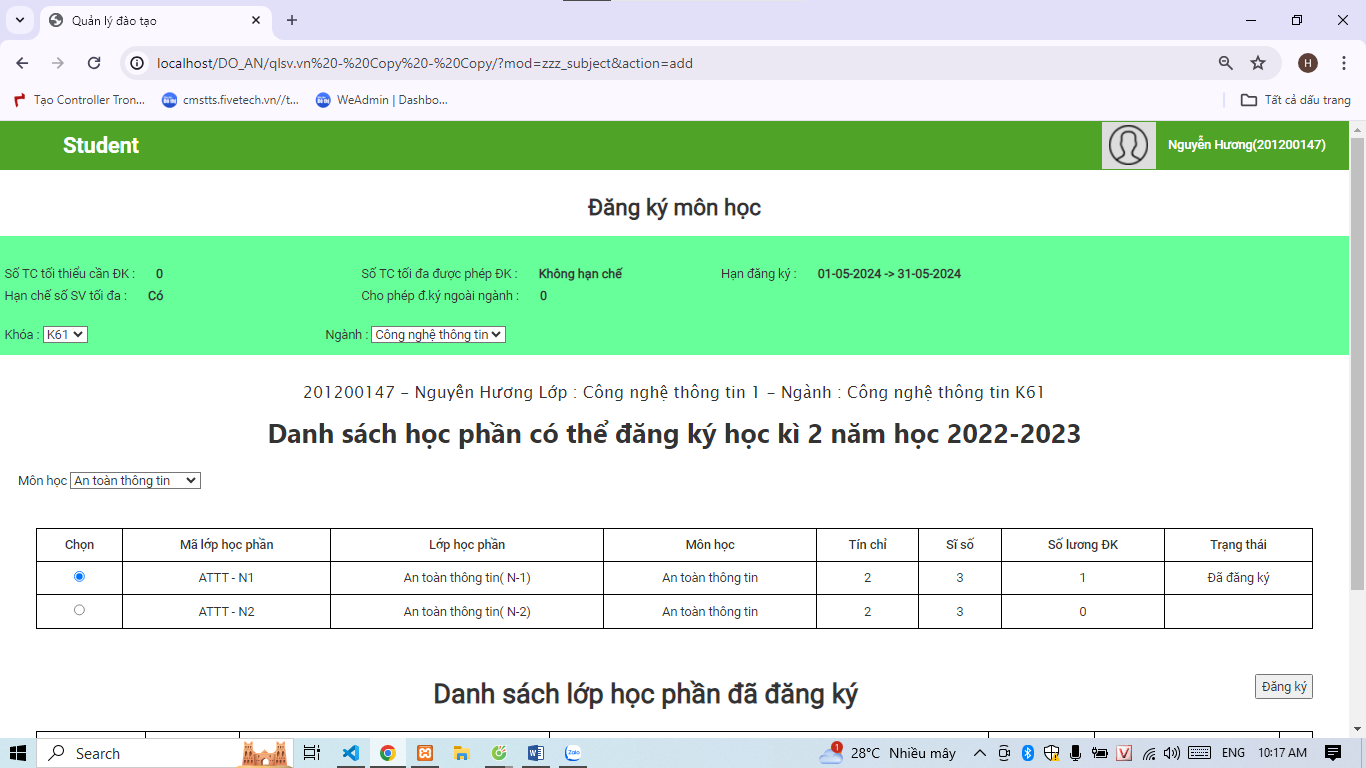
## 3.2. Xây dựng trang User



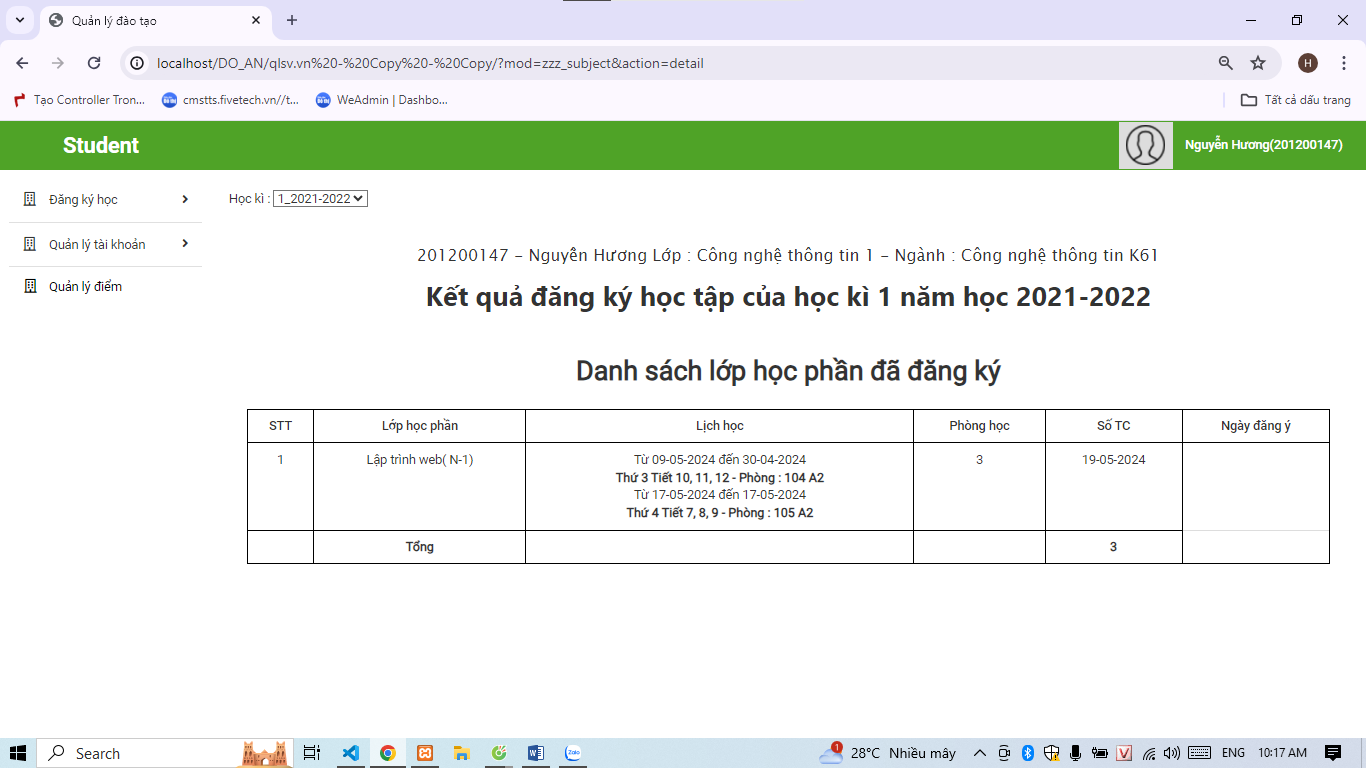
#### Hình 3.8. Giao diện trang Đăng nhập của sinh viên



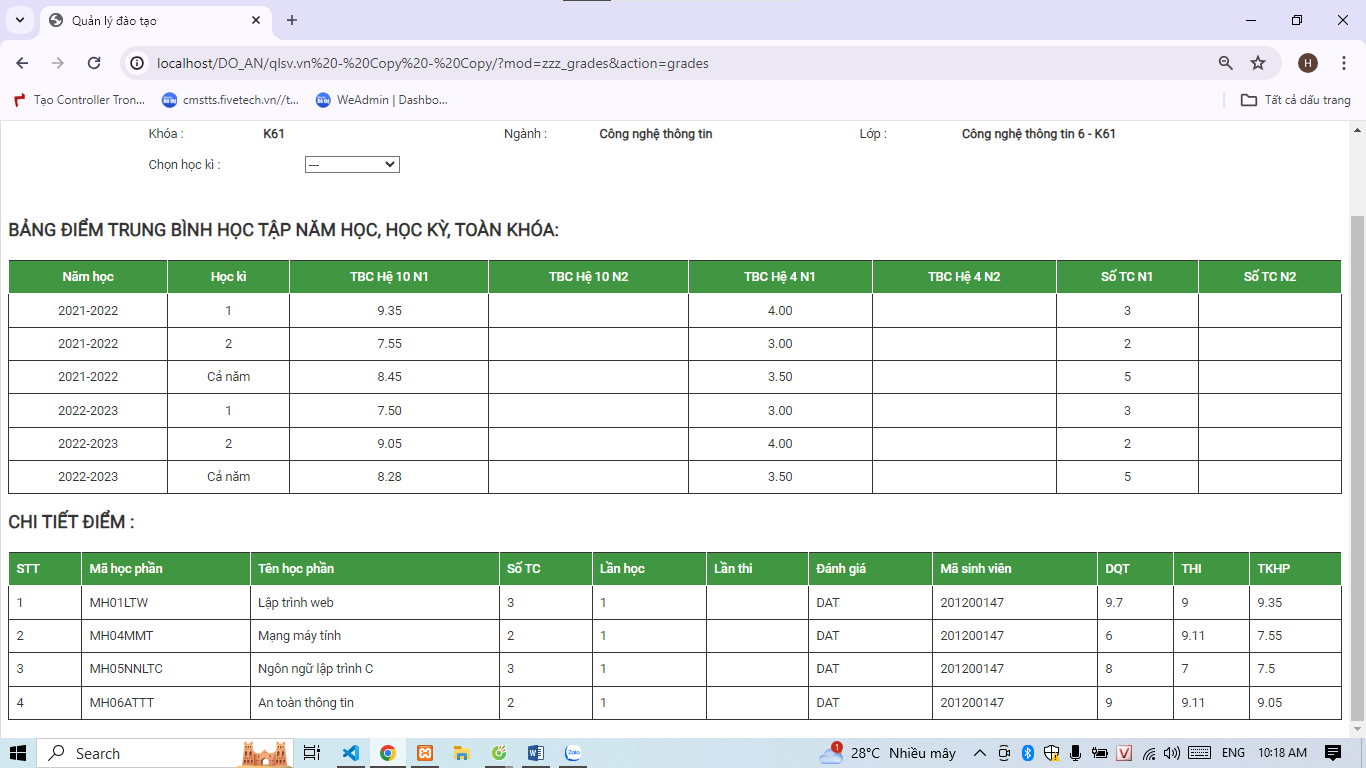
#### Hình 3.9. Giao diện Trang chủ



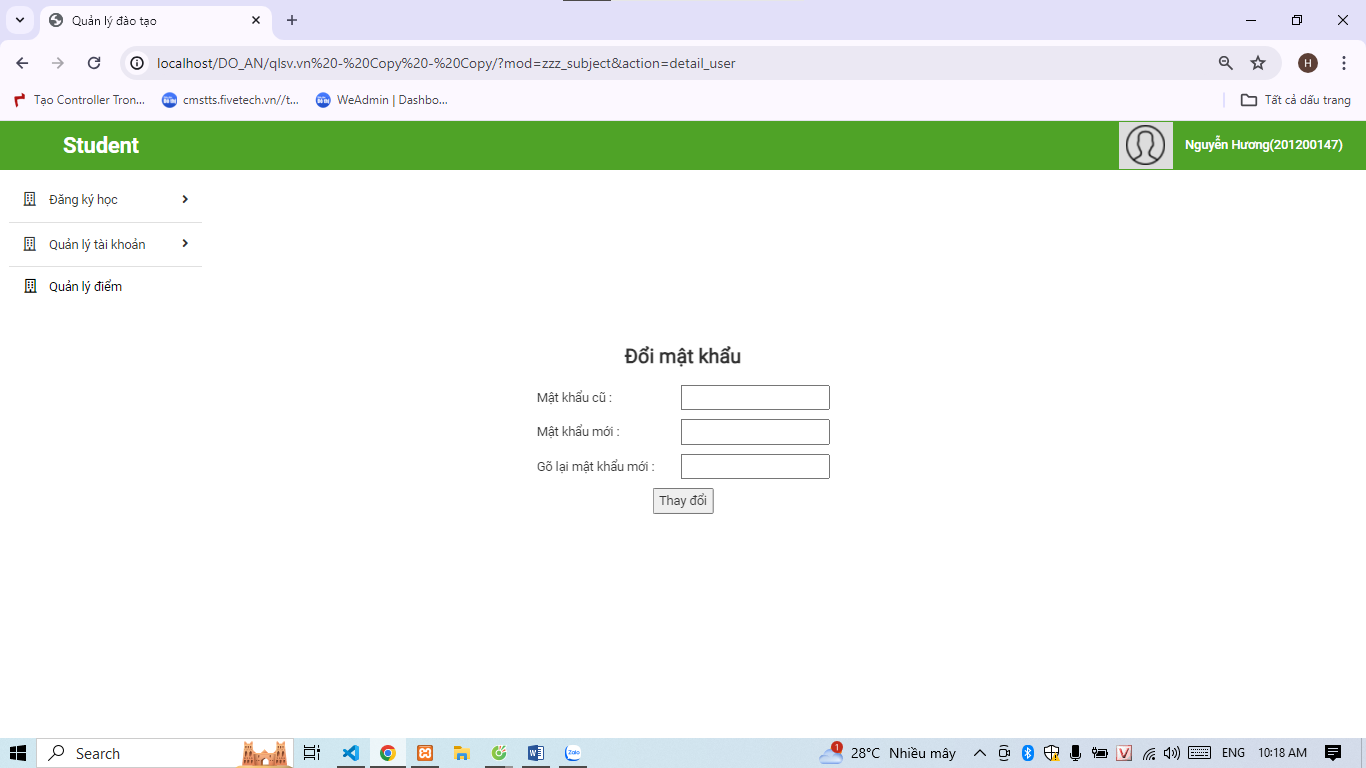
#### Hình 3.10. Giao diện Đăng ký môn học



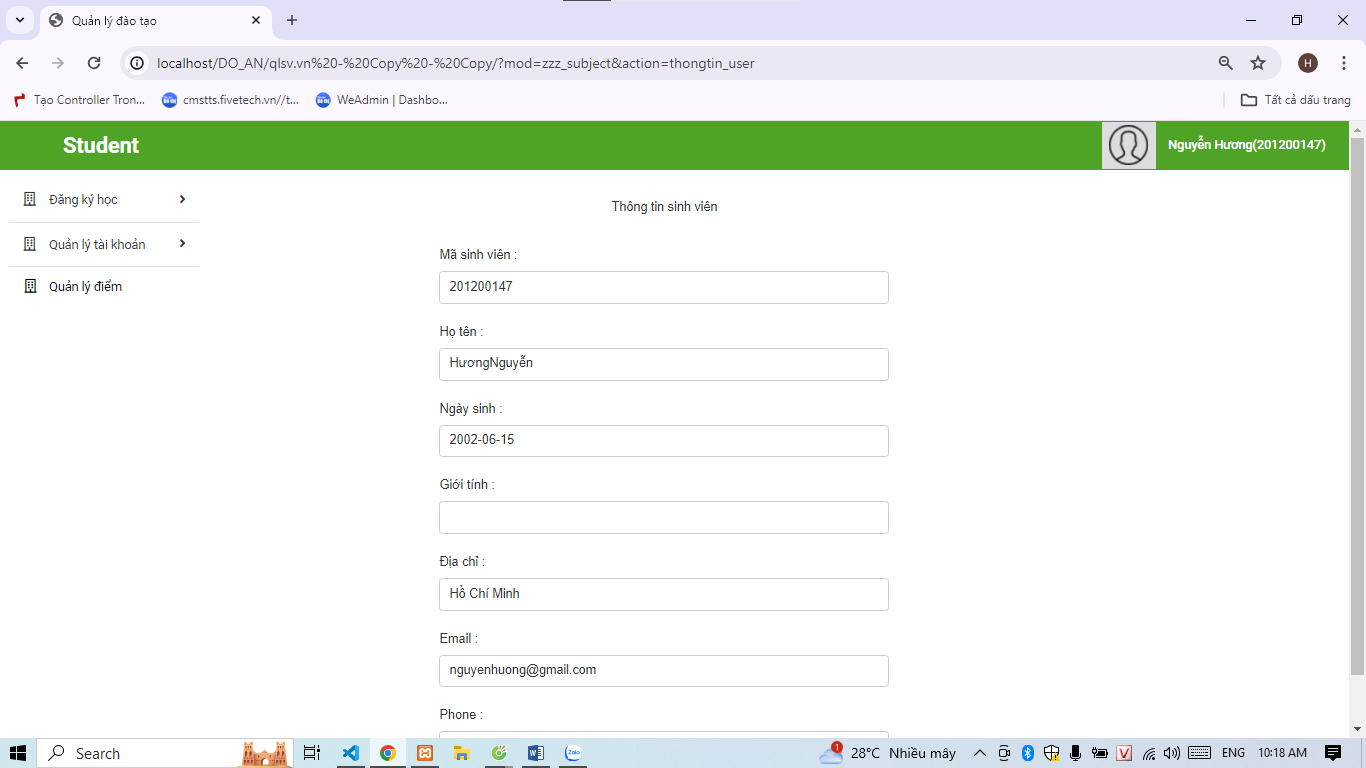
#### Hình 3.11. Giao diện Kết quả đăng ký học



#### Hình 3.12. Giao diện Tra cứu điểm



#### Hình 3.13. Giao diện Đổi mật khẩu



#### Hình 3.14. Giao diện Thông tin cá nhân

# KẾT LUẬN

* **Kết quả đạt được:**
  + Hoàn chỉnh trang web dành cho Admin và User với các trang cơ bản
  + Giao diện khá đẹp, dễ sử dụng.
  + Có thêm kiến thức về MySQL, Jquery, các kĩ thuật liên quan đến Frontend.
  + Biết cách sử dụng git để quản lý mã nguồn
* **Những vấn đề còn tồn tại**
  + Dữ liệu chưa được kiểm soát chặt chẽ: validate chưa hoàn chỉnh.
* **Hướng phát triển**
  + Phát triển web thành ứng dụng trên các thiết bị di động.
  + Thêm chức năng quản trị khác.
  + Biểu đồ thống kê được hiển thị theo thời gian lựa chọn.
  + Thông báo khi có thay đổi.

# TÀI LIỆU THAM KHẢO

[1] Trang web dành cho nhà phát triển PHP: <https://webcoban.vn/php/tai-lieu-hoc-php>

[2] Mô hình MVC(Model-View-Controller): <https://freetuts.net/mvc-php-mo-hinh-mvc-la-gi-354.html>

[3] Tham khảo MySql: <https://freetuts.net/hoc-mysql>

[4] JavaScript : <https://www.w3schools.com/js/>