TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH**

**HỌC KỲ I, NĂM HỌC 2025-2026**

**PHÁT TRIỂN HỆ THỐNG QUẢN LÝ**

**KẾ HOẠCH HỌC TẬP VÀ ĐĂNG KÝ HỌC PHẦN**

**TẠI TRƯỜNG ĐẠI HỌC TRÀ VINH**

|  |  |
| --- | --- |
| *Giảng viên hướng dẫn:*  ThS.Nguyễn Ngọc Đan Thanh | *Sinh viên thực hiện:*  Họ Tên: Triệu Chanh Đa  Mssv: 110122043  Lớp: DA22TTA |

**Vĩnh Long, tháng 12 năm 2025**

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH**

**HỌC KỲ I, NĂM HỌC 2025-2026**

**PHÁT TRIỂN HỆ THỐNG QUẢN LÝ**

**KẾ HOẠCH HỌC TẬP VÀ ĐĂNG KÝ HỌC PHẦN**

**TẠI TRƯỜNG ĐẠI HỌC TRÀ VINH**

|  |  |
| --- | --- |
| *Giảng viên hướng dẫn:*  ThS.Nguyễn Ngọc Đan Thanh | *Sinh viên thực hiện:*  Họ Tên: Triệu Chanh Đa  Mssv: 110122043  Lớp: DA22TTA |

**Vĩnh Long, tháng 12 năm 2025**

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

*Vĩnh Long, ngày … tháng … năm 2025*

**Giảng viên hướng dẫn**

*(Ký tên và ghi rõ họ tên)*

NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG

*Vĩnh Long, ngày … tháng … năm 2025*

**Thành viên hội đồng**

*(Ký tên và ghi rõ họ tên)*

LỜI CẢM ƠN

Đầu tiên, chúng em xin gửi lời cảm ơn chân thành đến Ban Giám hiệu Trường Kỹ thuật và Công nghệ cùng quý thầy cô Khoa Công nghệ Thông tin đã tạo điều kiện thuận lợi và môi trường học tập tốt nhất cho chúng em trong suốt quá trình học tập và nghiên cứu tại trường. Đặc biệt, chúng em xin bày tỏ lòng biết ơn sâu sắc đến Cô Nguyễn Ngọc Đan Thanh, giảng viên hướng dẫn trực tiếp chuyên ngành. Cô đã tận tình truyền đạt kiến thức, định hướng tư duy và đưa ra những nhận xét quý báu giúp nhóm hoàn thiện đồ án này. Sự hướng dẫn của Cô không chỉ giúp chúng em hoàn thành nhiệm vụ môn học mà còn hiểu rõ hơn về quy trình thiết kế lấy người dùng làm trung tâm. Cuối cùng, cảm ơn gia đình và bạn bè đã luôn động viên, hỗ trợ tinh thần để nhóm có thể hoàn thành tốt đề tài này. Mặc dù đã rất cố gắng, nhưng do giới hạn về thời gian và kiến thức, đồ án khó tránh khỏi những thiếu sót. Chúng em rất mong nhận được sự đóng góp ý kiến từ Cô để đề tài được hoàn thiện hơn. Xin chân thành cảm ơn!

MỤC LỤC

[1 Lý do chọn đề tài 1](#_Toc217902801)

[2 Mục đích nghiên cứu 1](#_Toc217902802)

[3 Đối tượng và phạm vi nghiên cứu 2](#_Toc217902803)

[4 Phương pháp nghiên cứu 2](#_Toc217902804)

[5 Cấu trúc báo cáo 3](#_Toc217902805)

[CHƯƠNG 1. TỔNG QUAN NGHIÊN CỨU 4](#_Toc217902806)

[1.1 Bối cảnh và bài toán đặt ra 4](#_Toc217902807)

[1.2 Thực trạng quản lý kế hoạch học tập và đăng ký học phần hiện nay 4](#_Toc217902808)

[1.3 Những hạn chế của những phương pháp hiện nay 4](#_Toc217902809)

[1.4 Các hệ thống quản lý kế hoạch học tập và đăng ký học phần hiện có 5](#_Toc217902810)

[1.5 Đánh giá ưu và nhược của các hệ thống hiện có 5](#_Toc217902811)

[1.6 Định hướng giải pháp của đề tài 5](#_Toc217902812)

[1.7 Kết luận chương 6](#_Toc217902813)

[CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT 7](#_Toc217902814)

[2.1 Tổng quan về Web và hệ thông Web 7](#_Toc217902815)

[2.1.1 Khái niệm Web 7](#_Toc217902816)

[2.1.2 Ứng dụng Web và hệ thống Web 7](#_Toc217902817)

[2.1.3 Vai trò của hệ thống Web trong xây dựng các hệ thống quản lý 8](#_Toc217902818)

[2.1.4 Kiến trúc Client – Server 9](#_Toc217902819)

[2.1.4.1 Tổng quan kiến trúc Client – Server 9](#_Toc217902820)

[2.1.4.2 Mô hình Client – Server trong hệ thống 10](#_Toc217902821)

[2.1.4.3 Lợi ích của kiến trúc Client – Server đối với đề tài 10](#_Toc217902822)

[2.2 Các công nghệ và công cụ sử dụng trong đề tài 11](#_Toc217902823)

[2.2.1 Công nghệ Front-end 11](#_Toc217902824)

[2.2.1.1 HTML 11](#_Toc217902825)

[2.2.1.2 CSS 14](#_Toc217902826)

[2.2.1.3 JavaScript 16](#_Toc217902827)

[2.2.2 Công nghệ Back-end 18](#_Toc217902828)

[2.2.2.1 PHP 19](#_Toc217902829)

[2.2.2.2 MySQL 22](#_Toc217902830)

[2.2.2.3 Tương tác giữa PHP và MySQL 26](#_Toc217902831)

[2.2.3 Các công cụ và môi trường hỗ trợ phát triển 28](#_Toc217902832)

[2.2.3.1 Môi trường máy chủ cục bộ XAMPP 28](#_Toc217902833)

[2.2.3.2 Trình soạn thảo mã nguồn Visual Studio Code 28](#_Toc217902834)

[2.2.3.3 Nền tảng quản lý mã nguồn GitHub 29](#_Toc217902835)

[2.2.3.4 Trình duyệt Web 30](#_Toc217902836)

[2.3 Kết luận chương 30](#_Toc217902837)

[CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU 32](#_Toc217902838)

[3.1 Mô tả bài toán 32](#_Toc217902839)

[3.1.1 Mục tiêu hệ thống 32](#_Toc217902840)

[3.1.2 Quy trình nghiệp vụ tổng quát 32](#_Toc217902841)

[3.1.3 Đối tượng sử dụng 33](#_Toc217902842)

[3.2 Phân tích yêu cầu hệ thống 33](#_Toc217902843)

[3.2.1 Tác nhân hệ thống 34](#_Toc217902844)

[3.2.2 Yêu cầu chức năng 34](#_Toc217902845)

[3.2.3 Yêu cầu phi chức năng 35](#_Toc217902846)

[3.3 Phân tích và thiết kế hệ thống 36](#_Toc217902847)

[3.3.1 Sơ đồ Use Case 36](#_Toc217902848)

[3.3.2 Sơ đồ hoạt động 37](#_Toc217902849)

[3.3.2.1 Sơ đồ hoạt động của Sinh Viên 37](#_Toc217902850)

[3.3.2.2 Sơ đồ hoạt động của Cố vấn học tập 38](#_Toc217902851)

[3.3.2.3 Sơ đồ hoạt động của Admin 38](#_Toc217902852)

[3.4 Thiết kế Cơ sở dữ liệu 39](#_Toc217902853)

[3.4.1 Sơ đồ thực thể kết hợp 39](#_Toc217902854)

[3.4.2 Thiết kế chi tiết các bảng 36](#_Toc217902855)

[3.5 Thiết kế giao diện 40](#_Toc217902856)

[3.5.1 Thiết kế bố cục chung 40](#_Toc217902857)

[3.5.2 Giao diện Đăng nhập 40](#_Toc217902858)

[3.5.3 Giao diện Sinh viên 41](#_Toc217902859)

[3.5.4 Giao diện Cố vấn học tập 43](#_Toc217902860)

[3.5.5 Giao diện Admin 46](#_Toc217902861)

[3.6 Quy trình xây dựng ứng dụng 49](#_Toc217902862)

[3.6.1 Môi trường phát triển 49](#_Toc217902863)

[3.6.1.1 Môi trường máy chủ cục bộ và bộ công cụ XAMPP 49](#_Toc217902864)

[3.6.1.2 Môi trường thực thi phía máy khách 49](#_Toc217902865)

[3.6.2 Công cụ hỗ trợ 50](#_Toc217902866)

[3.6.2.1 Môi trường phát triển tích hợp Visual Studio Code 50](#_Toc217902867)

[3.6.2.2 Công cụ quản trị Cơ sở dữ liệu phpMyAdmin 50](#_Toc217902868)

[3.6.2.3 Khung giao diện Bootstrap 5 Framework 50](#_Toc217902869)

[3.6.2.4 Hệ thống kiểm soát phiên bản GitHub 50](#_Toc217902870)

[3.6.2.5 Công cụ gỡ lỗi Browser Developer Tool 51](#_Toc217902871)

[3.6.3 Cấu trúc tổ chức mã nguồn 51](#_Toc217902872)

[3.6.3.1 Nhóm điều khiển Controllers 51](#_Toc217902873)

[3.6.3.2 Nhóm Mô hình Models 52](#_Toc217902874)

[3.6.3.3 Nhóm Giao diện Views 52](#_Toc217902875)

[3.6.3.4 Tập tin hệ thống và cấu hình 54](#_Toc217902876)

[3.6.4 Các giai đoạn thực hiện 54](#_Toc217902877)

[3.6.4.1 Giai đoạn 1: Khởi tạo môi trường và Cơ sở dữ liệu nền tảng 54](#_Toc217902878)

[3.6.4.2 Giai đoạn 2: Xây dựng Phân hệ Quản trị 55](#_Toc217902879)

[3.6.4.3 Giai đoạn 3: Xây dựng Phân hệ Người dùng 56](#_Toc217902880)

[3.6.4.4 Giai đoạn 4: Hoàn thiện Giao diện và Kiểm thử luồng nghiệp vụ 56](#_Toc217902881)

[3.7 Kết luận chương 57](#_Toc217902882)

[CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU 58](#_Toc217902883)

[4.1 Kết quả triển khai hệ thống 58](#_Toc217902884)

[4.1.1 Môi trường triển khai Xampp 58](#_Toc217902885)

[4.1.2 Đánh giá hiệu quả sử dụng tài nguyên 58](#_Toc217902886)

[4.2 Kết quả giao diện chức năng người dùng 60](#_Toc217902887)

[4.2.1 Giao diện đăng nhập và đổi mật khẩu 60](#_Toc217902888)

[4.2.2 Giao diện của Sinh viên 61](#_Toc217902889)

[4.2.3 Giao diện của Cố vấn học tập 63](#_Toc217902890)

[4.2.4 Giao diện của Admin 65](#_Toc217902891)

[4.3 Đánh giá hệ thống 68](#_Toc217902892)

[4.3.1 Đánh giá về mặt kỹ thuật 68](#_Toc217902893)

[4.3.2 Tự so sánh với mục tiêu ban đầu 69](#_Toc217902894)

[4.4 Kết luận chương 70](#_Toc217902895)

[CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 71](#_Toc217902896)

[5.1 Kết luận 71](#_Toc217902897)

[5.2 Hướng phát triển 72](#_Toc217902898)

**DANH MỤC HÌNH ẢNH**

[Hình 2.1 Cách hoạt động của mô hình Client – Server 9](#_Toc217902899)

[Hình 2.2 Mô hình kiến trúc Client – Server của hệ thống 10](#_Toc217902900)

[Hình 3.1 Sơ đồ Use Case 36](#_Toc217902901)

[Hình 3.2 Sơ đồ hoạt động của Sinh viên 37](#_Toc217902902)

[Hình 3.3 Sơ đồ hoạt động của Cố vấn học tập 38](#_Toc217902903)

[Hình 3.4 Sơ đồ hoạt động của Admin 39](#_Toc217902904)

[Hình 3.5 Sơ đồ ERD 35](#_Toc217902905)

[Hình 3.6 Sơ đồ bố cục chung 40](#_Toc217902906)

[Hình 3.7 Giao diện đăng nhập 41](#_Toc217902907)

[Hình 3.8 Giao diện trang chủ Sinh viên 42](#_Toc217902908)

[Hình 3.9 Giao diện Lập kế hoạch học tập 42](#_Toc217902909)

[Hình 3.10 Giao diện Xem thời khóa biểu 43](#_Toc217902910)

[Hình 3.11 Giao diện trang chủ Cố vấn học tập 44](#_Toc217902911)

[Hình 3.12 Giao diện duyệt Kế hoạch học tập 44](#_Toc217902912)

[Hình 3.13 Giao diện Xem danh sách sinh viên 45](#_Toc217902913)

[Hình 3.14 Giao diện Duyệt kết quả học tập 46](#_Toc217902914)

[Hình 3.15 Giao diện Danh sách học phần 47](#_Toc217902915)

[Hình 3.16 Giao diên Danh sách sinh viên 47](#_Toc217902916)

[Hình 3.17 Giao diện Danh sách Cố vấn học tập 48](#_Toc217902917)

[Hình 3.18 Giao diện Danh sách lớp 48](#_Toc217902918)

[Hình 4.1 Kết quả chạy xampp 58](#_Toc217902919)

[Hình 4.2 Giao diện đăng nhập 60](#_Toc217902920)

[Hình 4.3 Giao diện đổi mật khẩu 61](#_Toc217902921)

[Hình 4.4 Giao diện trang chủ sinh viên 62](#_Toc217902922)

[Hình 4.5 Giao diện đăng ký kế hoạch học tập 62](#_Toc217902923)

[Hình 4.6 Giao diện xem thời khóa biểu 63](#_Toc217902924)

[Hình 4.7 Giao diện trang chủ cố vấn học tập 64](#_Toc217902925)

[Hình 4.8 Giao diện danh sách sinh viên 64](#_Toc217902926)

[Hình 4.9 Giao diện xét kết quả học tập 65](#_Toc217902927)

[Hình 4.10 Giao diện quản lý học phần 66](#_Toc217902928)

[Hình 4.11 Giao diện quản lý sinh viên 66](#_Toc217902929)

[Hình 4.12 Giao diện quản lý cố vấn học tập 67](#_Toc217902930)

[Hình 4.13 Giao diện quản lý lớp học 68](#_Toc217902931)

DANH MỤC BẢNG BIỂU

[Bảng 3.1 Bảng SINHVIEN 36](#_Toc217902932)

[Bảng 3.2 Bảng KEHOACH 36](#_Toc217902933)

[Bảng 3.3 Bảng CHITIETKEHOACH 36](#_Toc217902934)

[Bảng 3.4 Bảng HOCPHAN 37](#_Toc217902935)

[Bảng 3.5 Bảng THOIKHOABIEU 37](#_Toc217902936)

[Bảng 3.6 Bảng KETQUAHOCTAP 37](#_Toc217902937)

[Bảng 3.7 Bảng DIEUKIENTIENQUYET 38](#_Toc217902938)

[Bảng 3.8 Bảng LOP 38](#_Toc217902939)

[Bảng 3.9 Bảng CHUONGTRINHDAOTAO 38](#_Toc217902940)

[Bảng 3.10 Bảng COVANHOCTAP 39](#_Toc217902941)

[Bảng 3.11 Bảng TAIKHOAN 39](#_Toc217902942)

[Bảng 3.12 Bảng LOCHHOC 39](#_Toc217902943)

TÓM TẮT ĐỒ ÁN

Trong bối cảnh đào tạo theo hệ thống tín chỉ, việc sinh viên chủ động lập kế hoạch học tập và Cố vấn học tập theo dõi, phê duyệt là một nghiệp vụ cốt lõi. Tuy nhiên, quy trình này tại Trường Đại học Trà Vinh hiện còn gặp nhiều khó khăn, tốn thời gian và dễ xảy ra sai sót khi thực hiện thủ công hoặc qua các công cụ phân tán.

Đề tài “Phát triển Hệ thống Quản lý Kế hoạch Học tập và Đăng ký Học phần tại Trường Đại học Trà Vinh” được thực hiện nhằm mục tiêu xây dựng một hệ thống web tập trung để giải quyết vấn đề trên. Hệ thống được phát triển dựa trên ngôn ngữ lập trình PHP và hệ quản trị cơ sở dữ liệu MySQL, áp dụng các công nghệ frontend như HTML, CSS, Bootstrap và JavaScript.

Hệ thống cung cấp đầy đủ các chức năng cho ba nhóm đối tượng chính:

Quản trị viên: Quản lý dữ liệu nền tảng như: chương trình đào tạo, học phần, điều kiện tiên quyết, tài khoản người dùng.

Sinh viên: Lập kế hoạch học tập, xem tiến độ học tập (trực quan hóa bằng Chart.js), và nhận các cảnh báo tự động về các ràng buộc (điều kiện tiên quyết, trùng lịch học, giới hạn tín chỉ).

Cố vấn học tập: Theo dõi, kiểm tra, thêm ghi chú, và thực hiện phê duyệt kế hoạch học tập của sinh viên trực tuyến.

Sau khi đồ án hoàn thiện giúp số hóa quy trình lập và duyệt kế hoạch học tập, đảm bảo tính chính xác của dữ liệu, tiết kiệm thời gian cho cả sinh viên và cố vấn học tập, giảm thiểu sai sót, và góp phần nâng cao hiệu quả quản lý học vụ, hỗ trợ công tác chuyển đổi số tại Trường.

MỞ ĐẦU

## Lý do chọn đề tài

Trong đào tạo theo học chế tín chỉ, việc lập kế hoạch học tập đóng vai trò quan trọng, giúp sinh viên chủ động sắp xếp lộ trình học tập phù hợp với chương trình đào tạo và khả năng cá nhân. Tuy nhiên, việc lập kế hoạch học tập của sinh viên hiện nay chủ yếu được thực hiện thủ công, thiếu công cụ hỗ trợ đối chiếu với danh sách học phần đăng ký thực tế, dẫn đến tình trạng đăng ký thiếu môn, trùng học phần hoặc học vượt kế hoạch.

Bên cạnh đó, công tác theo dõi và tư vấn học tập của cố vấn học tập gặp nhiều khó khăn do thiếu hệ thống hỗ trợ tổng hợp và quản lý dữ liệu học tập của sinh viên theo từng học kỳ. Việc kiểm tra tiến độ học tập, phát hiện kịp thời các học phần chưa hoàn thành hoặc nợ môn còn phụ thuộc nhiều vào thao tác thủ công, gây mất thời gian và dễ xảy ra sai sót.

Xuất phát từ những hạn chế nêu trên, việc xây dựng một hệ thống hỗ trợ quản lý kế hoạch học tập và theo dõi đăng ký học phần là cần thiết. Hệ thống không chỉ giúp sinh viên lập và theo dõi kế hoạch học tập một cách khoa học, mà còn hỗ trợ cố vấn học tập trong công tác quản lý, tư vấn và định hướng học tập, góp phần nâng cao hiệu quả quản lý đào tạo theo học chế tín chỉ tại Trường Đại học Trà Vinh.

## Mục đích nghiên cứu

Mục đích của đề tài là nghiên cứu và xây dựng một hệ thống hỗ trợ quản lý kế hoạch học tập và theo dõi đăng ký học phần của sinh viên theo học chế tín chỉ. Hệ thống nhằm giúp sinh viên chủ động lập kế hoạch học tập phù hợp với chương trình đào tạo, đồng thời hỗ trợ cố vấn học tập trong việc kiểm tra, phê duyệt và theo dõi tiến độ học tập của sinh viên qua từng học kỳ.

Bên cạnh đó, đề tài hướng đến việc hỗ trợ đối chiếu giữa kế hoạch học tập dự kiến và danh sách học phần đăng ký thực tế, qua đó kịp thời phát hiện các học phần chưa đăng ký, đăng ký ngoài kế hoạch hoặc các môn học còn nợ. Thông qua việc số hóa và chuẩn hóa quy trình quản lý kế hoạch học tập, đề tài góp phần nâng cao hiệu quả công tác tư vấn học tập và quản lý đào tạo tại Trường Đại học Trà Vinh.

## Đối tượng và phạm vi nghiên cứu

**Đối tượng nghiên cứu**

Đối tượng nghiên cứu của đề tài là quy trình lập kế hoạch học tập cá nhân của sinh viên theo học chế tín chỉ, bao gồm việc xây dựng kế hoạch học tập cho từng học kỳ dựa trên chương trình đào tạo. Bên cạnh đó, đề tài còn nghiên cứu quy trình duyệt kế hoạch, theo dõi tiến độ học tập và tư vấn học tập của cố vấn học tập nhằm đảm bảo kế hoạch học tập phù hợp với quy định đào tạo.

Ngoài ra, cấu trúc cơ sở dữ liệu phục vụ cho việc quản lý học phần, học kỳ, chương trình đào tạo và kết quả học tập của sinh viên cũng là một nội dung nghiên cứu quan trọng, nhằm đảm bảo tính nhất quán và khả năng mở rộng của hệ thống.

**Phạm vi nghiên cứu**

Phạm vi nghiên cứu của đề tài tập trung vào việc xây dựng một hệ thống quản lý kế hoạch học tập và theo dõi đăng ký học phần dưới dạng website. Hệ thống được áp dụng cho đối tượng là sinh viên và cố vấn học tập tại Trường Đại học Trà Vinh, hỗ trợ quản lý và theo dõi tiến độ học tập theo từng học kỳ.

Đề tài không nhằm thay thế hệ thống đăng ký học phần chính thức của nhà trường, mà chỉ đóng vai trò là công cụ hỗ trợ quản lý, đối chiếu và theo dõi kế hoạch học tập nhằm nâng cao hiệu quả công tác tư vấn và quản lý đào tạo.

## Phương pháp nghiên cứu

Đề tài được thực hiện dựa trên sự kết hợp giữa phương pháp nghiên cứu lý thuyết và phương pháp nghiên cứu thực nghiệm.

Phương pháp nghiên cứu lý thuyết được sử dụng nhằm tìm hiểu các khái niệm và mô hình liên quan đến quản lý học tập theo học chế tín chỉ, quy trình lập kế hoạch học tập cá nhân và vai trò của cố vấn học tập trong công tác tư vấn học tập. Đồng thời, đề tài nghiên cứu các cơ sở lý thuyết về phân tích và thiết kế hệ thống thông tin, bao gồm mô hình hóa nghiệp vụ, sơ đồ luồng dữ liệu và sơ đồ thực thể – liên kết, làm nền tảng cho quá trình thiết kế hệ thống.

Phương pháp nghiên cứu thực nghiệm được áp dụng thông qua việc phân tích yêu cầu thực tế, thiết kế và xây dựng hệ thống quản lý kế hoạch học tập và đăng ký học phần. Hệ thống sau khi hoàn thiện được triển khai thử nghiệm với dữ liệu mẫu, đánh giá mức độ đáp ứng yêu cầu và hiệu quả hỗ trợ sinh viên cũng như cố vấn học tập trong quá trình quản lý và theo dõi tiến độ học tập.

## Cấu trúc báo cáo

Báo cáo đồ án được trình bày trong 5 chương, bao gồm:

Chương 1. Tổng quan nghiên cứu: Phân tích bối cảnh bài toán, đánh giá các giải pháp hiện có và đề xuất giải pháp của đề tài.

Chương 2. Nghiên cứu lý thuyết: Trình bày cơ sở lý thuyết về các công nghệ được sử dụng.

Chương 3. Hiện thực hóa nghiên cứu: Phân tích yêu cầu, thiết kế hệ thống và quy trình xây dựng ứng dụng.

Chương 4. Kết quả nghiên cứu: Trình bày các kết quả đạt được, hình ảnh giao diện và đánh giá hệ thống.

Chương 5. Kết luận và hướng phát triển: Tổng kết ưu, nhược điểm và định hướng mở rộng trong tương lai.

# TỔNG QUAN NGHIÊN CỨU

## Bối cảnh và bài toán đặt ra

Trong bối cảnh chuyển đổi số mạnh mẽ trong lĩnh vực giáo dục, các cơ sở đào tạo đại học tại Việt Nam đang từng bước ứng dụng công nghệ thông tin vào công tác quản lý đào tạo nhằm nâng cao hiệu quả và chất lượng giảng dạy. Đặc biệt, mô hình đào tạo theo học chế tín chỉ đòi hỏi sinh viên phải chủ động hơn trong việc lập kế hoạch học tập, đăng ký học phần và theo dõi tiến độ học tập cá nhân.

Tuy nhiên, việc quản lý kế hoạch học tập và đăng ký học phần hiện nay vẫn còn nhiều bất cập. Sinh viên gặp khó khăn trong việc xây dựng lộ trình học tập phù hợp với chương trình đào tạo, trong khi cố vấn học tập thiếu công cụ hỗ trợ để theo dõi và tư vấn kịp thời. Từ thực tiễn đó, bài toán đặt ra là cần xây dựng một hệ thống hỗ trợ quản lý kế hoạch học tập và đăng ký học phần một cách khoa học, hiệu quả và thuận tiện.

## Thực trạng quản lý kế hoạch học tập và đăng ký học phần hiện nay

Hiện nay, tại nhiều cơ sở đào tạo, việc lập kế hoạch học tập của sinh viên chủ yếu được thực hiện thủ công thông qua các biểu mẫu hoặc bảng kế hoạch cá nhân. Sinh viên thường tự đối chiếu chương trình đào tạo và đăng ký học phần dựa trên kinh nghiệm cá nhân, dẫn đến tình trạng thiếu đồng bộ và dễ sai sót.

Bên cạnh đó, hệ thống đăng ký học phần hiện có chủ yếu tập trung vào chức năng đăng ký môn học theo từng học kỳ, chưa hỗ trợ hiệu quả việc lập kế hoạch học tập dài hạn. Việc theo dõi tiến độ học tập, số tín chỉ tích lũy và các học phần còn nợ thường được thực hiện rời rạc, gây khó khăn cho cả sinh viên và cố vấn học tập trong quá trình quản lý và tư vấn học tập.

## Những hạn chế của những phương pháp hiện nay

Các phương pháp quản lý kế hoạch học tập và đăng ký học phần hiện nay còn tồn tại nhiều hạn chế. Thứ nhất, việc lập kế hoạch học tập thủ công dễ dẫn đến sai sót trong việc lựa chọn học phần, không đảm bảo điều kiện tiên quyết hoặc trùng lặp môn học. Thứ hai, thiếu công cụ hỗ trợ đối chiếu giữa kế hoạch học tập dự kiến và danh sách học phần đăng ký thực tế.

Ngoài ra, cố vấn học tập gặp khó khăn trong việc theo dõi tiến độ học tập của sinh viên do dữ liệu phân tán, không được tổng hợp một cách trực quan. Điều này làm giảm hiệu quả công tác tư vấn và hỗ trợ sinh viên, đặc biệt là trong việc phát hiện sớm các vấn đề liên quan đến tiến độ học tập.

## Các hệ thống quản lý kế hoạch học tập và đăng ký học phần hiện có

Hiện nay, một số hệ thống quản lý đào tạo đã được triển khai tại các cơ sở giáo dục nhằm hỗ trợ công tác đăng ký học phần và quản lý thông tin học tập của sinh viên. Các hệ thống này thường cung cấp các chức năng cơ bản như đăng ký môn học, tra cứu kết quả học tập và theo dõi số tín chỉ đã tích lũy.

Tuy nhiên, phần lớn các hệ thống hiện có chưa tập trung vào việc hỗ trợ lập kế hoạch học tập cá nhân một cách toàn diện. Chức năng lập kế hoạch học tập dài hạn, theo dõi tiến độ so với kế hoạch và hỗ trợ cố vấn học tập trong việc duyệt và tư vấn kế hoạch học tập còn hạn chế hoặc chưa được triển khai đầy đủ.

## Đánh giá ưu và nhược của các hệ thống hiện có

Các hệ thống quản lý đào tạo hiện có có ưu điểm là hỗ trợ tốt công tác đăng ký học phần và quản lý thông tin học tập cơ bản của sinh viên. Việc số hóa dữ liệu giúp giảm bớt thao tác thủ công và nâng cao tính chính xác trong quản lý.

Tuy nhiên, nhược điểm lớn nhất của các hệ thống này là chưa hỗ trợ hiệu quả việc lập và quản lý kế hoạch học tập cá nhân. Thiếu sự liên kết giữa kế hoạch học tập và đăng ký học phần thực tế, cũng như thiếu các công cụ hỗ trợ phân tích và theo dõi tiến độ học tập, khiến cho sinh viên và cố vấn học tập chưa khai thác được tối đa lợi ích của hệ thống.

## Định hướng giải pháp của đề tài

Xuất phát từ những hạn chế nêu trên, đề tài định hướng xây dựng một hệ thống hỗ trợ quản lý kế hoạch học tập và theo dõi đăng ký học phần dành cho sinh viên theo học chế tín chỉ. Hệ thống cho phép sinh viên lập kế hoạch học tập theo từng học kỳ, đối chiếu với chương trình đào tạo và theo dõi tiến độ thực hiện kế hoạch.

Bên cạnh đó, hệ thống hỗ trợ cố vấn học tập trong việc kiểm tra, phê duyệt và theo dõi kế hoạch học tập của sinh viên, góp phần nâng cao hiệu quả công tác tư vấn và quản lý đào tạo. Giải pháp hướng đến tính trực quan, dễ sử dụng và có khả năng mở rộng trong tương lai.

## Kết luận chương

Chương 1 đã trình bày tổng quan về đề tài nghiên cứu, làm rõ bối cảnh và bài toán đặt ra trong công tác quản lý kế hoạch học tập và đăng ký học phần theo học chế tín chỉ. Trên cơ sở đó, chương đã phân tích thực trạng quản lý hiện nay, chỉ ra những hạn chế của các phương pháp và hệ thống đang được áp dụng.

Bên cạnh đó, chương đã tổng hợp và đánh giá ưu điểm, nhược điểm của các hệ thống quản lý kế hoạch học tập và đăng ký học phần hiện có, từ đó xác định định hướng giải pháp mà đề tài hướng tới. Các nội dung được trình bày trong chương này đóng vai trò làm cơ sở lý luận và thực tiễn cho các chương tiếp theo, đặc biệt là chương nghiên cứu lý thuyết và chương phân tích, thiết kế hệ thống.

# NGHIÊN CỨU LÝ THUYẾT

## Tổng quan về Web và hệ thông Web

### Khái niệm Web

Web là mạng lưới thông tin toàn cầu, một hệ thống cho phép người dùng truy cập và trao đổi thông tin thông qua mạng Internet. Web hoạt động dựa trên cơ chế liên kết các tài nguyên thông tin bằng siêu liên kết. Cơ chế này giúp người dùng dễ dàng di chuyển giữa các trang thông tin khác nhau. [1]

Về bản chất, Web là một môi trường cung cấp thông tin phân tán. Dữ liệu được lưu trữ tại các máy chủ và được truy cập thông qua các thiết bị đầu cuối như máy tính hoặc điện thoại thông minh. Người dùng sử dụng trình duyệt Web để gửi yêu cầu truy cập. Máy chủ tiếp nhận yêu cầu, xử lý và trả về nội dung tương ứng dưới dạng các trang Web.

Các trang Web được xây dựng chủ yếu bằng ngôn ngữ đánh dấu siêu văn bản. Các công nghệ hỗ trợ khác được sử dụng để tạo giao diện trực quan và tăng khả năng tương tác với người dùng. Thông qua Web, thông tin có thể được cập nhật thường xuyên và truy cập vào bất kỳ thời điểm nào khi có kết nối Internet.

Trong lĩnh vực công nghệ thông tin, Web không chỉ được sử dụng để cung cấp thông tin. Web còn là nền tảng để xây dựng các hệ thống quản lý và các dịch vụ trực tuyến. Đối với các hệ thống quản lý đào tạo, Web giúp kết nối người dùng với hệ thống một cách thuận tiện. Qua đó, hiệu quả quản lý và khai thác thông tin được nâng cao.

### Ứng dụng Web và hệ thống Web

**Ứng dụng Web** là các phần mềm được xây dựng và vận hành trên nền tảng Web. Người dùng có thể truy cập và sử dụng ứng dụng Web thông qua trình duyệt mà không cần cài đặt thêm phần mềm trên thiết bị. Ứng dụng Web cho phép thực hiện nhiều chức năng khác nhau như nhập dữ liệu, xử lý thông tin và hiển thị kết quả cho người dùng.

**Hệ thống Web** là tập hợp các thành phần phần mềm phối hợp với nhau để cung cấp các chức năng và dịch vụ thông qua môi trường Web. Một hệ thống Web thường bao gồm giao diện người dùng, phần xử lý nghiệp vụ và phần lưu trữ dữ liệu. Các thành phần này hoạt động đồng bộ nhằm đảm bảo hệ thống vận hành ổn định và đáp ứng yêu cầu của người sử dụng.

So với các ứng dụng cài đặt truyền thống, ứng dụng Web có ưu điểm là dễ triển khai và dễ bảo trì. Việc cập nhật hệ thống được thực hiện tập trung trên máy chủ. Người dùng luôn sử dụng phiên bản mới nhất của hệ thống. Điều này giúp giảm chi phí quản lý và nâng cao hiệu quả sử dụng.

Trong phạm vi các hệ thống quản lý, ứng dụng Web cho phép tổ chức và quản lý dữ liệu một cách tập trung. Thông tin có thể được truy cập và cập nhật theo thời gian thực. Điều này đặc biệt phù hợp với các hệ thống quản lý trong lĩnh vực giáo dục. Ứng dụng Web giúp kết nối người dùng với hệ thống một cách linh hoạt và thuận tiện.

**Web có 2 loại chính:**

Web tĩnh là loại website không có hệ thống quản lý nội dung, nội dung được lưu cố định trong các tệp HTML, CSS và JavaScript. Người dùng thông thường không thể chỉnh sửa nội dung mà phải thay đổi trực tiếp trong mã nguồn. Web tĩnh thường có tốc độ truy cập nhanh, chi phí xây dựng thấp và thân thiện với công cụ tìm kiếm. Tuy nhiên, loại web này khó quản lý, khó cập nhật và mở rộng, nên hiện nay chủ yếu được sử dụng cho các trang giới thiệu nhỏ, ít thay đổi nội dung hoặc trang one-page đơn giản. [2]

Web động là website có hệ thống quản lý nội dung cho phép người dùng dễ dàng cập nhật và chỉnh sửa thông tin trên trang. Loại website này được xây dựng bằng HTML, CSS, JavaScript kết hợp với ngôn ngữ lập trình phía server như PHP, NodeJS cùng cơ sở dữ liệu như MySQL hoặc MongoDB. Web động dễ quản lý, mở rộng và hỗ trợ tương tác người dùng, phù hợp cho các website thương mại điện tử, tin tức, blog hoặc hệ thống quản lý thông tin. Tuy nhiên, chi phí và yêu cầu kỹ thuật cao hơn so với web tĩnh.

### Vai trò của hệ thống Web trong xây dựng các hệ thống quản lý

Website giữ vai trò là cầu nối giữa hệ thống và người dùng, giúp truy cập, quản lý và trao đổi thông tin một cách nhanh chóng, trực quan và hiệu quả. Trong một hệ thống hiện đại, website hoạt động như giao diện tương tác chính, nơi người dùng thực hiện các chức năng như đăng nhập, quản lý dữ liệu hoặc trao đổi thông tin.

Bên cạnh đó, website còn là kênh truyền thông và quảng bá thông tin của tổ chức, giúp giới thiệu dịch vụ, sản phẩm và hình ảnh đến người dùng trên Internet. Nhờ khả năng truy cập mọi lúc, mọi nơi, website góp phần tự động hóa quy trình làm việc, tăng hiệu quả quản lý, giảm chi phí vận hành và nâng cao trải nghiệm người dùng.

Trong hệ thống quản lý dự án và công việc nhóm, website đóng vai trò trung tâm, giúp kết nối các thành viên, quản lý nhiệm vụ, theo dõi tiến độ và lưu trữ dữ liệu một cách tập trung, đảm bảo hoạt động nhóm diễn ra hiệu quả và đồng bộ.

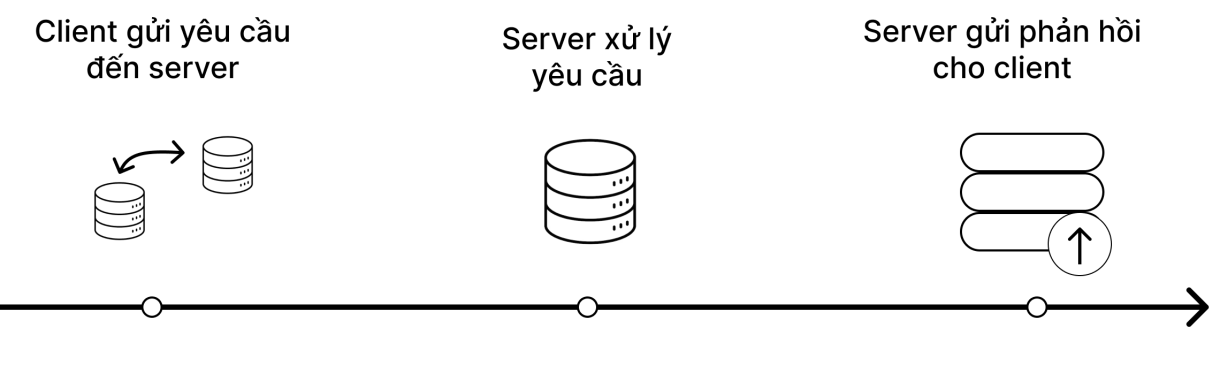
### Kiến trúc Client – Server

#### Tổng quan kiến trúc Client – Server

Kiến trúc Client – Server là mô hình kiến trúc trong đó hệ thống được chia thành hai thành phần chính là phía người dùng và phía máy chủ. Phía người dùng chịu trách nhiệm hiển thị giao diện và tiếp nhận thao tác của người sử dụng. Phía máy chủ đảm nhiệm việc xử lý nghiệp vụ, quản lý dữ liệu và cung cấp dịch vụ cho phía người dùng thông qua mạng. [3]

Trong mô hình này, Client gửi các yêu cầu đến Server thông qua các giao thức truyền thông. Server tiếp nhận yêu cầu, thực hiện xử lý cần thiết và trả kết quả về cho Client. Nhờ sự phân tách rõ ràng giữa các thành phần, kiến trúc Client – Server giúp hệ thống dễ mở rộng, dễ bảo trì và nâng cao hiệu quả quản lý.

Kiến trúc Client – Server được sử dụng rộng rãi trong các ứng dụng Web hiện đại. Mô hình này đặc biệt phù hợp với các hệ thống quản lý trực tuyến nhờ khả năng phục vụ nhiều người dùng đồng thời và quản lý dữ liệu tập trung.



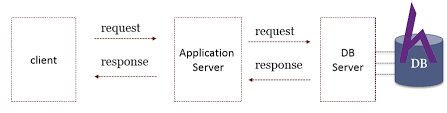
Hình 2.1 Cách hoạt động của mô hình Client – Server

#### Mô hình Client – Server trong hệ thống

Trong hệ thống hỗ trợ lập kế hoạch học tập và đăng ký học phần, phía Client được triển khai dưới dạng ứng dụng Web. Client có nhiệm vụ hiển thị giao diện, tiếp nhận thao tác của sinh viên và cố vấn học tập. Các thao tác này bao gồm lập kế hoạch học tập, theo dõi tiến độ học tập và gửi yêu cầu xử lý đến phía máy chủ.

Phía Server được xây dựng để xử lý các nghiệp vụ của hệ thống. Server thực hiện các chức năng như xác thực người dùng, xử lý kế hoạch học tập, quản lý thông tin học phần và truy vấn cơ sở dữ liệu. Dữ liệu được lưu trữ tập trung trong hệ quản trị cơ sở dữ liệu. Kết quả xử lý được Server trả về cho Client để hiển thị cho người dùng.

Việc áp dụng mô hình Client – Server giúp hệ thống vận hành ổn định và đảm bảo tính nhất quán của dữ liệu trong quá trình sử dụng.



Hình 2.2 Mô hình kiến trúc Client – Server của hệ thống

#### Lợi ích của kiến trúc Client – Server đối với đề tài

Việc áp dụng kiến trúc Client – Server giúp hệ thống tách biệt rõ ràng giữa giao diện người dùng và phần xử lý nghiệp vụ. Điều này giúp quá trình phát triển và bảo trì hệ thống trở nên đơn giản hơn. Khi có sự thay đổi về giao diện hoặc nghiệp vụ, hệ thống có thể được cập nhật mà không ảnh hưởng đến toàn bộ cấu trúc.

Kiến trúc Client – Server cho phép hệ thống mở rộng linh hoạt khi số lượng người dùng tăng lên. Dữ liệu được quản lý tập trung tại máy chủ giúp nâng cao tính an toàn và độ tin cậy của hệ thống. Mô hình này tạo nền tảng phù hợp để xây dựng hệ thống quản lý học tập trực tuyến, đáp ứng nhu cầu sử dụng trong môi trường đào tạo theo học chế tín chỉ.

## Các công nghệ và công cụ sử dụng trong đề tài

### Công nghệ Front-end

Công nghệ giao diện người dùng là nhóm các công nghệ được sử dụng để xây dựng phần giao diện của hệ thống Web. Đây là phần mà người dùng trực tiếp quan sát và tương tác trong quá trình sử dụng hệ thống. Giao diện người dùng có vai trò quan trọng trong việc truyền đạt thông tin, hỗ trợ thao tác và nâng cao trải nghiệm sử dụng.

Trong hệ thống Web, Front-end đảm nhiệm việc hiển thị nội dung, tiếp nhận dữ liệu đầu vào và phản hồi các thao tác của người dùng. Các thao tác này có thể bao gồm nhập thông tin, lựa chọn chức năng và theo dõi kết quả xử lý từ hệ thống. Thông tin được hiển thị dưới dạng trực quan giúp người dùng dễ dàng tiếp cận và sử dụng.

Các công nghệ Front-end hoạt động chủ yếu trên trình duyệt Web. Nhờ đó, người dùng không cần cài đặt thêm phần mềm mà vẫn có thể sử dụng đầy đủ các chức năng của hệ thống. Điều này giúp hệ thống có tính linh hoạt cao và dễ dàng triển khai trên nhiều thiết bị khác nhau.

Trong phạm vi đề tài, công nghệ giao diện người dùng được sử dụng để xây dựng các chức năng như đăng nhập hệ thống, lập kế hoạch học tập, theo dõi tiến độ học tập và hiển thị thông tin học phần. Việc lựa chọn và sử dụng các công nghệ Front-end phù hợp giúp hệ thống có giao diện rõ ràng, dễ sử dụng và đáp ứng tốt nhu cầu của người dùng.

Các nội dung chi tiết về từng công nghệ giao diện người dùng sẽ được trình bày trong các mục tiếp theo. Các công nghệ này bao gồm ngôn ngữ đánh dấu siêu văn bản, ngôn ngữ định kiểu và ngôn ngữ kịch bản phía người dùng.

#### HTML

Ngôn ngữ đánh dấu siêu văn bản (HTML) là ngôn ngữ cơ bản được sử dụng để xây dựng cấu trúc của các trang Web. HTML cho phép mô tả nội dung và cách sắp xếp các thành phần trên trang như văn bản, hình ảnh, bảng biểu và biểu mẫu. Các trình duyệt Web dựa vào mã HTML để hiển thị nội dung cho người dùng.

HTML không phải là ngôn ngữ lập trình mà là ngôn ngữ đánh dấu. Mỗi thành phần trong trang Web được xác định thông qua các thẻ đánh dấu. Các thẻ này giúp trình duyệt hiểu được vai trò và vị trí của từng nội dung trên trang. Nhờ đó, HTML đóng vai trò nền tảng trong việc xây dựng giao diện Web.

Trong hệ thống Web, HTML chịu trách nhiệm tổ chức bố cục và nội dung hiển thị. HTML thường được kết hợp với ngôn ngữ định kiểu và ngôn ngữ kịch bản để tạo nên giao diện hoàn chỉnh. Tuy nhiên, bản thân HTML vẫn giữ vai trò cốt lõi trong việc hình thành cấu trúc trang Web.

**Cấu trúc cơ bản của một tài liệu HTML**

Một tài liệu HTML được tổ chức theo cấu trúc nhất định nhằm đảm bảo trình duyệt có thể hiểu và hiển thị đúng nội dung. Cấu trúc này bao gồm phần khai báo, phần đầu trang và phần thân trang.

Ví dụ cấu trúc cơ bản của một trang HTML:

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<title>Trang Web</title>

</head>

<body>

Nội dung trang Web

</body>

</html>

Trong đó, thẻ html xác định phạm vi của tài liệu HTML. Thẻ head chứa các thông tin mô tả trang Web. Thẻ body chứa toàn bộ nội dung hiển thị cho người dùng.

**Các thẻ HTML cơ bản**

HTML cung cấp nhiều thẻ để xây dựng nội dung trang Web. Một số thẻ thường được sử dụng bao gồm thẻ tiêu đề, thẻ đoạn văn và thẻ liên kết. Các thẻ này giúp trình bày nội dung một cách rõ ràng và có cấu trúc.

Ví dụ một số thẻ HTML cơ bản:

<h1>Kế hoạch học tập</h1>

<p>Trang này dùng để lập và theo dõi kế hoạch học tập cá nhân.</p>

<a href="dangkyhocphan.html">Đăng ký học phần</a>

Việc sử dụng đúng các thẻ HTML giúp nội dung trang Web dễ đọc và dễ hiểu. Điều này đặc biệt quan trọng đối với các hệ thống quản lý có nhiều thông tin cần hiển thị.

**Biểu mẫu HTML trong hệ thống Web**

Biểu mẫu HTML được sử dụng để thu thập dữ liệu từ người dùng. Trong các hệ thống quản lý, biểu mẫu đóng vai trò quan trọng trong việc nhập và xử lý thông tin. Biểu mẫu cho phép người dùng nhập dữ liệu và gửi dữ liệu đó đến máy chủ để xử lý.

Ví dụ biểu mẫu nhập kế hoạch học tập:

<form method="post" action="xuly\_kehoach.php">

<label>Học kỳ:</label>

<input type="text" name="hocky">

<label>Tên học phần:</label>

<input type="text" name="hocphan">

<label>Số tín chỉ:</label>

<input type="number" name="tinchi">

<button type="submit">Lưu kế hoạch</button>

</form>

Trong ví dụ trên, biểu mẫu cho phép sinh viên nhập thông tin học kỳ, học phần và số tín chỉ. Dữ liệu được gửi đến phía máy chủ để xử lý và lưu trữ.

**Vai trò của HTML trong đề tài**

Trong đề tài xây dựng hệ thống hỗ trợ lập kế hoạch học tập và đăng ký học phần, HTML được sử dụng để xây dựng giao diện hiển thị thông tin cho người dùng. HTML giúp tổ chức các trang như trang đăng nhập, trang lập kế hoạch học tập và trang theo dõi tiến độ học tập.

Nhờ HTML, hệ thống có thể trình bày thông tin học tập một cách rõ ràng và có cấu trúc. Điều này giúp người dùng dễ dàng thao tác và giảm thiểu sai sót trong quá trình sử dụng hệ thống.

**Đánh giá ưu điểm và hạn chế của HTML**

HTML có ưu điểm là cú pháp đơn giản và dễ tiếp cận. HTML được hỗ trợ bởi tất cả các trình duyệt Web hiện nay. Điều này giúp các trang Web có khả năng tương thích cao trên nhiều thiết bị khác nhau.

Tuy nhiên, HTML chỉ đảm nhiệm vai trò xây dựng cấu trúc nội dung. HTML không hỗ trợ xử lý logic hay tương tác phức tạp. Do đó, HTML cần được kết hợp với các công nghệ khác để xây dựng hệ thống Web hoàn chỉnh.

#### CSS

Ngôn ngữ định kiểu (CSS) là công nghệ được sử dụng để thiết kế và trình bày giao diện cho các trang Web. CSS cho phép kiểm soát cách hiển thị của các thành phần HTML như màu sắc, kích thước, vị trí và bố cục. Nhờ CSS, giao diện Web trở nên trực quan và dễ sử dụng hơn.

CSS hoạt động dựa trên nguyên tắc tách biệt nội dung và hình thức trình bày. Nội dung trang Web được xây dựng bằng HTML. Hình thức hiển thị được kiểm soát bằng CSS. Việc tách biệt này giúp mã nguồn dễ quản lý và thuận tiện trong việc thay đổi giao diện mà không ảnh hưởng đến cấu trúc nội dung.

Trong hệ thống Web, CSS đóng vai trò quan trọng trong việc nâng cao trải nghiệm người dùng. Một giao diện rõ ràng và thống nhất giúp người dùng dễ dàng thao tác và tiếp cận thông tin. Điều này đặc biệt cần thiết đối với các hệ thống quản lý có nhiều biểu mẫu và dữ liệu hiển thị.

**Cách áp dụng CSS trong trang Web**

CSS có thể được áp dụng vào trang Web theo nhiều cách khác nhau. Phổ biến nhất là sử dụng tập tin CSS riêng và liên kết với trang HTML. Cách làm này giúp mã nguồn gọn gàng và dễ bảo trì.

Ví dụ liên kết tập tin CSS với trang HTML:

<link rel="stylesheet" href="style.css">

Trong đó, tập tin style.css chứa toàn bộ các quy tắc định kiểu cho trang Web.

**Cú pháp cơ bản của CSS**

CSS được xây dựng dựa trên các quy tắc định kiểu. Mỗi quy tắc gồm bộ chọn, thuộc tính và giá trị. Bộ chọn xác định phần tử HTML cần áp dụng định kiểu. Thuộc tính và giá trị xác định cách hiển thị của phần tử đó.

Ví dụ cú pháp CSS cơ bản:

body {

font-family: Arial;

background-color: #f5f5f5;

}

Quy tắc trên giúp thiết lập phông chữ và màu nền cho toàn bộ trang Web.

**Một số thuộc tính CSS thường sử dụng**

CSS cung cấp nhiều thuộc tính để thiết kế giao diện. Các thuộc tính này giúp kiểm soát màu sắc, kích thước và bố cục của trang Web.

Ví dụ một số thuộc tính CSS thường dùng:

h1 {

color: #2c3e50;

text-align: center;

}

label {

font-weight: bold;

}

input {

width: 100%;

padding: 8px;

margin-bottom: 10px;

}

Các thuộc tính trên giúp tiêu đề rõ ràng hơn và các trường nhập liệu dễ thao tác hơn.

**Ứng dụng CSS trong đề tài**

Trong đề tài xây dựng hệ thống hỗ trợ lập kế hoạch học tập và đăng ký học phần, CSS được sử dụng để thiết kế giao diện các trang chức năng. CSS giúp bố trí hợp lý các biểu mẫu nhập liệu, bảng thông tin học phần và các nút chức năng.

Ví dụ định kiểu cho biểu mẫu lập kế hoạch học tập:

.form-kehoach {

width: 60%;

margin: auto;

background-color: #ffffff;

padding: 20px;

border-radius: 5px;

}

Đoạn mã trên giúp biểu mẫu hiển thị gọn gàng và tập trung vào nội dung chính. Điều này giúp người dùng dễ dàng nhập thông tin và theo dõi kế hoạch học tập.

**Đánh giá ưu điểm và hạn chế của CSS**

CSS có ưu điểm là giúp giao diện Web linh hoạt và dễ thay đổi. CSS được hỗ trợ rộng rãi trên các trình duyệt Web hiện nay. Việc sử dụng CSS giúp giảm trùng lặp mã nguồn và tăng tính nhất quán của giao diện.

Tuy nhiên, CSS không xử lý được các chức năng tương tác phức tạp. CSS chỉ đảm nhiệm vai trò trình bày giao diện. Do đó, CSS cần được kết hợp với các công nghệ khác để xây dựng hệ thống Web hoàn chỉnh.

#### JavaScript

Ngôn ngữ kịch bản phía người dùng (JavaScript) là ngôn ngữ được sử dụng để xử lý các tương tác trên trang Web. JavaScript cho phép trang Web phản hồi linh hoạt trước các thao tác của người dùng như nhấp chuột, nhập dữ liệu và gửi biểu mẫu. Nhờ JavaScript, các trang Web trở nên sinh động và có tính tương tác cao hơn. [4]

JavaScript được thực thi trực tiếp trên trình duyệt Web của người dùng. Điều này giúp giảm tải cho máy chủ và nâng cao tốc độ phản hồi của hệ thống. JavaScript thường được sử dụng kết hợp với HTML và CSS để xây dựng giao diện Web hoàn chỉnh.

Trong các hệ thống quản lý trực tuyến, JavaScript đóng vai trò quan trọng trong việc kiểm tra dữ liệu đầu vào và hỗ trợ tương tác tức thời. Việc xử lý này giúp hạn chế sai sót trước khi dữ liệu được gửi đến phía máy chủ.

**Cách nhúng JavaScript vào trang Web**

JavaScript có thể được nhúng trực tiếp trong trang HTML hoặc được đặt trong tập tin riêng. Việc sử dụng tập tin JavaScript riêng giúp mã nguồn rõ ràng và dễ bảo trì hơn.

Ví dụ nhúng tập tin JavaScript vào trang HTML:

<script src="script.js"></script>

**Cú pháp cơ bản của JavaScript**

JavaScript sử dụng cú pháp đơn giản và dễ tiếp cận. Các câu lệnh JavaScript được viết dưới dạng từng dòng và có thể tương tác trực tiếp với các phần tử HTML trên trang Web.

Ví dụ khai báo biến và hiển thị thông báo:

let hocKy = "Học kỳ 1";

alert("Lập kế hoạch cho " + hocKy);

Đoạn mã trên giúp hiển thị thông báo cho người dùng khi thực hiện thao tác trên hệ thống.

**Xử lý sự kiện trong JavaScript**

JavaScript cho phép xử lý các sự kiện phát sinh từ thao tác của người dùng. Các sự kiện thường gặp bao gồm nhấn nút, thay đổi nội dung và gửi biểu mẫu. Việc xử lý sự kiện giúp hệ thống phản hồi kịp thời và thân thiện hơn.

Ví dụ xử lý sự kiện khi nhấn nút lưu kế hoạch học tập:

<button onclick="luuKeHoach()">Lưu kế hoạch</button>

<script>

function luuKeHoach() {

alert("Kế hoạch học tập đã được lưu");

}

</script>

Khi người dùng nhấn nút, hệ thống sẽ hiển thị thông báo xác nhận thao tác.

**Kiểm tra dữ liệu biểu mẫu bằng JavaScript**

Trong các hệ thống quản lý học tập, việc kiểm tra dữ liệu trước khi gửi là rất cần thiết. JavaScript cho phép kiểm tra dữ liệu ngay trên trình duyệt để đảm bảo thông tin nhập vào là hợp lệ.

Ví dụ kiểm tra dữ liệu biểu mẫu lập kế hoạch học tập:

function kiemTraDuLieu() {

let tinChi = document.getElementById("tinchi").value;

if (tinChi <= 0) {

alert("Số tín chỉ phải lớn hơn 0");

return false;

}

return true;

}

Đoạn mã trên giúp hạn chế việc nhập dữ liệu không hợp lệ trước khi gửi về máy chủ.

**Vai trò của JavaScript trong đề tài**

Trong đề tài xây dựng hệ thống hỗ trợ lập kế hoạch học tập và đăng ký học phần, JavaScript được sử dụng để xử lý các tương tác phía người dùng. JavaScript hỗ trợ kiểm tra dữ liệu nhập, hiển thị thông báo và cải thiện trải nghiệm sử dụng hệ thống.

Nhờ JavaScript, người dùng có thể nhận phản hồi ngay lập tức khi thao tác trên hệ thống. Điều này giúp quá trình lập kế hoạch học tập trở nên thuận tiện và hiệu quả hơn.

**Đánh giá ưu điểm và hạn chế của JavaScript**

JavaScript có ưu điểm là khả năng xử lý tương tác nhanh và linh hoạt. JavaScript được hỗ trợ trên hầu hết các trình duyệt Web hiện nay. Việc sử dụng JavaScript giúp tăng tính thân thiện và hiệu quả cho các ứng dụng Web.

Tuy nhiên, JavaScript chỉ xử lý dữ liệu phía trình duyệt. JavaScript không thay thế được các xử lý nghiệp vụ phía máy chủ. Do đó, JavaScript cần được kết hợp với các công nghệ xử lý phía máy chủ để xây dựng hệ thống Web hoàn chỉnh.

### Công nghệ Back-end

Công nghệ xử lý phía máy chủ là nhóm các công nghệ được sử dụng để xây dựng phần xử lý nghiệp vụ và quản lý dữ liệu của hệ thống Web. Đây là thành phần không trực tiếp hiển thị cho người dùng nhưng đóng vai trò cốt lõi trong việc đảm bảo hệ thống vận hành chính xác và ổn định.

Trong hệ thống Web, Back-end chịu trách nhiệm tiếp nhận các yêu cầu từ phía người dùng, xử lý logic nghiệp vụ và tương tác với cơ sở dữ liệu. Các chức năng như xác thực người dùng, lưu trữ thông tin, xử lý kế hoạch học tập và truy xuất dữ liệu đều được thực hiện tại phía máy chủ.

Các công nghệ Back-end thường hoạt động theo kiến trúc Client – Server. Phía máy chủ tiếp nhận yêu cầu từ Client, thực hiện xử lý cần thiết và trả kết quả về cho Client. Cách tổ chức này giúp dữ liệu được quản lý tập trung và đảm bảo tính nhất quán trong quá trình sử dụng hệ thống.

Trong phạm vi đề tài, công nghệ xử lý phía máy chủ được sử dụng để xây dựng các chức năng cốt lõi của hệ thống hỗ trợ lập kế hoạch học tập và đăng ký học phần. Việc lựa chọn công nghệ Back-end phù hợp giúp hệ thống hoạt động hiệu quả, an toàn và đáp ứng được nhu cầu của người dùng.

Các nội dung chi tiết về từng công nghệ xử lý phía máy chủ sẽ được trình bày trong các mục tiếp theo. Các công nghệ này bao gồm ngôn ngữ xử lý phía máy chủ và hệ quản trị cơ sở dữ liệu.

#### PHP

Ngôn ngữ kịch bản phía máy chủ (PHP) là ngôn ngữ được sử dụng để xây dựng các chức năng xử lý nghiệp vụ trong hệ thống Web. PHP được thực thi trên máy chủ và chịu trách nhiệm xử lý dữ liệu trước khi trả kết quả về cho phía người dùng. Nhờ khả năng tích hợp tốt với Web, PHP được sử dụng rộng rãi trong việc phát triển các hệ thống quản lý trực tuyến. [5]

PHP cho phép tiếp nhận dữ liệu từ phía người dùng thông qua các biểu mẫu Web. Sau khi tiếp nhận dữ liệu, PHP tiến hành xử lý theo logic nghiệp vụ và thực hiện các thao tác với cơ sở dữ liệu. Kết quả xử lý được PHP gửi về trình duyệt dưới dạng trang Web hoàn chỉnh.

Trong các hệ thống quản lý, PHP đóng vai trò trung tâm trong việc kiểm soát luồng xử lý. Các chức năng như xác thực người dùng, quản lý thông tin và xử lý nghiệp vụ đều được thực hiện tại phía máy chủ thông qua PHP.

**Cách nhúng PHP vào trang Web**

PHP được nhúng trực tiếp vào các tập tin Web có phần mở rộng là .php. Trong tập tin này, mã PHP được đặt xen kẽ với mã HTML. Máy chủ sẽ xử lý phần mã PHP trước khi gửi nội dung về cho trình duyệt.

Ví dụ cú pháp nhúng PHP:

<?php

echo "Hệ thống quản lý kế hoạch học tập";

?>

Đoạn mã trên giúp hiển thị nội dung văn bản sau khi được xử lý tại phía máy chủ.

**Cú pháp cơ bản của PHP**

PHP sử dụng cú pháp đơn giản và dễ tiếp cận. PHP hỗ trợ khai báo biến, cấu trúc điều khiển và xử lý dữ liệu từ biểu mẫu Web.

Ví dụ khai báo biến và xử lý dữ liệu:

<?php

$hocKy = $\_POST["hocky"];

echo "Kế hoạch học tập cho " . $hocKy;

?>

Đoạn mã trên cho phép PHP tiếp nhận dữ liệu học kỳ từ biểu mẫu và hiển thị kết quả tương ứng.

**Xử lý biểu mẫu bằng PHP**

Trong hệ thống Web, PHP thường được sử dụng để xử lý dữ liệu gửi từ biểu mẫu HTML. PHP tiếp nhận dữ liệu, kiểm tra tính hợp lệ và thực hiện các xử lý cần thiết.

Ví dụ xử lý biểu mẫu lập kế hoạch học tập:

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$hocPhan = $\_POST["hocphan"];

$tinChi = $\_POST["tinchi"];

echo "Đã lưu học phần: " . $hocPhan;

}

?>

Đoạn mã trên minh họa cách PHP xử lý dữ liệu do người dùng nhập và thực hiện phản hồi kết quả.

**Kết nối cơ sở dữ liệu bằng PHP**

PHP hỗ trợ kết nối và làm việc với các hệ quản trị cơ sở dữ liệu. Việc kết nối cơ sở dữ liệu cho phép hệ thống lưu trữ và truy xuất thông tin phục vụ cho các chức năng quản lý.

Ví dụ kết nối cơ sở dữ liệu:

<?php

$conn = mysqli\_connect("localhost", "root", "", "quanlyhoctap");

if (!$conn) {

die("Kết nối cơ sở dữ liệu thất bại");

}

?>

Việc kết nối thành công giúp hệ thống thực hiện các thao tác lưu trữ và truy vấn dữ liệu học tập.

**Vai trò của PHP trong đề tài**

Trong đề tài xây dựng hệ thống hỗ trợ lập kế hoạch học tập và đăng ký học phần, PHP được sử dụng để xử lý các nghiệp vụ cốt lõi. PHP đảm nhiệm các chức năng như xác thực tài khoản, xử lý kế hoạch học tập, quản lý học phần và tương tác với cơ sở dữ liệu.

Nhờ PHP, hệ thống có thể xử lý dữ liệu một cách an toàn và nhất quán. Điều này giúp đảm bảo tính chính xác của thông tin học tập trong quá trình sử dụng hệ thống.

**Đánh giá ưu điểm và hạn chế của PHP**

PHP có ưu điểm là dễ học và dễ triển khai. PHP có cộng đồng sử dụng rộng lớn và được hỗ trợ tốt trên nhiều môi trường máy chủ. Việc tích hợp PHP với các công nghệ Web khác cũng tương đối thuận tiện.

Tuy nhiên, PHP chỉ đảm nhiệm vai trò xử lý phía máy chủ. PHP không xử lý trực tiếp các tương tác phía người dùng. Do đó, PHP cần được kết hợp với các công nghệ giao diện và cơ sở dữ liệu để xây dựng hệ thống Web hoàn chỉnh.

#### MySQL

Hệ quản trị cơ sở dữ liệu MySQL là hệ quản trị cơ sở dữ liệu quan hệ được sử dụng phổ biến trong các hệ thống Web. MySQL cho phép lưu trữ, tổ chức và truy xuất dữ liệu một cách có cấu trúc. Dữ liệu được quản lý theo các bảng và mối quan hệ giữa các bảng.[6]

Trong các hệ thống quản lý trực tuyến, MySQL đóng vai trò lưu trữ toàn bộ dữ liệu của hệ thống. Các thông tin như sinh viên, học phần, kế hoạch học tập và kết quả học tập đều được quản lý tập trung trong cơ sở dữ liệu. Việc sử dụng MySQL giúp đảm bảo tính nhất quán và toàn vẹn của dữ liệu.

MySQL thường được sử dụng kết hợp với ngôn ngữ xử lý phía máy chủ để xây dựng các hệ thống Web hoàn chỉnh. Sự kết hợp này giúp hệ thống xử lý hiệu quả cả phần nghiệp vụ và phần lưu trữ dữ liệu.

**Mô hình dữ liệu quan hệ trong MySQL**

MySQL quản lý dữ liệu theo mô hình quan hệ. Trong mô hình này, dữ liệu được tổ chức thành các bảng. Mỗi bảng gồm các cột và các dòng dữ liệu. Các bảng có thể liên kết với nhau thông qua khóa chính và khóa ngoại.

Việc tổ chức dữ liệu theo mô hình quan hệ giúp dữ liệu được lưu trữ có hệ thống. Các thao tác truy vấn và cập nhật dữ liệu được thực hiện một cách chính xác và hiệu quả.

**Cú pháp tạo bảng**

MySQL sử dụng ngôn ngữ truy vấn có cấu trúc để thao tác với dữ liệu. Các câu lệnh SQL cho phép tạo bảng, thêm dữ liệu, cập nhật dữ liệu và truy vấn thông tin.

Cú pháp tổng quan:

CREATE TABLE TenBang (

TenCot1 KieuDuLieu [Ràng buộc],

TenCot2 KieuDuLieu [Ràng buộc],

...

);

Ví dụ tạo bảng học phần:

CREATE TABLE SinhVien (

MaSinhVien VARCHAR(20) PRIMARY KEY,

HoTen VARCHAR(100) NOT NULL,

NgaySinh DATE,

Lop VARCHAR(20)

Trong ví dụ trên, trường MaSinhVien được khai báo là khóa chính nhằm đảm bảo mỗi sinh viên được định danh duy nhất, trong khi ràng buộc NOT NULL được áp dụng cho trường họ tên để tránh dữ liệu rỗng.

**Cú pháp thêm dữ liệu**

Sau khi bảng được tạo, dữ liệu được đưa vào hệ thống thông qua câu lệnh INSERT INTO. Câu lệnh này cho phép thêm các bản ghi mới vào bảng, tương ứng với các đối tượng thực tế trong hệ thống.

Cú pháp tổng quát:

INSERT INTO TenBang (TenCot1, TenCot2, ...)

VALUES (GiaTri1, GiaTri2, ...);

Ví dụ thêm thông tin sinh viên:

INSERT INTO SinhVien (MaSinhVien, HoTen, NgaySinh, Lop)

VALUES ('SV001', 'Nguyễn Văn A', '2004-03-18', 'DA22TTA');

Câu lệnh trên thực hiện việc lưu trữ một bản ghi sinh viên mới vào bảng SinhVien. Dữ liệu được thêm vào phải tuân thủ đúng cấu trúc bảng và các ràng buộc đã được định nghĩa trước đó.

**Cú pháp truy vấn dữ liệu**

Truy vấn dữ liệu là thao tác quan trọng nhất trong việc khai thác cơ sở dữ liệu. MySQL sử dụng câu lệnh SELECT để truy xuất dữ liệu từ một hoặc nhiều bảng, phục vụ cho việc hiển thị thông tin và xử lý nghiệp vụ.

Cú pháp cơ bản:

SELECT TenCot1, TenCot2

FROM TenBang;

Ví dụ truy vấn toàn bộ danh sách sinh viên:

SELECT \* FROM SinhVien;

Ví dụ truy vấn có điều kiện:

SELECT HoTen, Lop

FROM SinhVien

WHERE Lop = 'DA22TTA';

Thông qua mệnh đề WHERE, hệ thống chỉ truy xuất những bản ghi thỏa mãn điều kiện, giúp dữ liệu hiển thị chính xác theo yêu cầu của người dùng.

**Ràng buộc khóa chính**

Khóa chính là ràng buộc dùng để định danh duy nhất mỗi bản ghi trong một bảng. Mỗi bảng chỉ có một khóa chính và giá trị của khóa chính không được trùng lặp hoặc rỗng. Việc sử dụng khóa chính giúp đảm bảo tính toàn vẹn dữ liệu và tăng hiệu quả truy vấn.

Ví dụ khai báo khóa chính:

CREATE TABLE HocPhan (

MaHocPhan VARCHAR(20),

TenHocPhan VARCHAR(100),

SoTinChi INT,

PRIMARY KEY (MaHocPhan)

);

Trong bảng trên, trường MaHocPhan được sử dụng làm khóa chính nhằm đảm bảo mỗi học phần tồn tại duy nhất trong hệ thống.

Ràng buộc khóa ngoại

Khóa ngoại được sử dụng để thiết lập mối quan hệ giữa các bảng trong cơ sở dữ liệu. Một trường được xem là khóa ngoại khi nó tham chiếu đến khóa chính của bảng khác. Ràng buộc khóa ngoại giúp đảm bảo dữ liệu giữa các bảng có liên quan luôn nhất quán.

Cú pháp khai báo khóa ngoại:

FOREIGN KEY (TenCot)

REFERENCES BangThamChieu(TenCot);

Ví dụ bảng kế hoạch học tập liên kết với sinh viên và học phần:

CREATE TABLE KeHoachHocTap (

MaSinhVien VARCHAR(20),

MaHocPhan VARCHAR(20),

HocKy VARCHAR(20),

PRIMARY KEY (MaSinhVien, MaHocPhan),

FOREIGN KEY (MaSinhVien) REFERENCES SinhVien(MaSinhVien),

FOREIGN KEY (MaHocPhan) REFERENCES HocPhan(MaHocPhan)

);

Trong ví dụ trên, các ràng buộc khóa ngoại đảm bảo rằng mỗi kế hoạch học tập chỉ được tạo cho sinh viên và học phần đã tồn tại trong hệ thống, từ đó ngăn chặn dữ liệu không hợp lệ.

**Ý nghĩa của việc sử dụng các ràng buộc dữ liệu**

Việc áp dụng các ràng buộc như khóa chính và khóa ngoại giúp cơ sở dữ liệu được tổ chức chặt chẽ, hạn chế sai sót và đảm bảo tính nhất quán trong quá trình vận hành hệ thống. Đồng thời, các ràng buộc này tạo nền tảng cho việc truy vấn dữ liệu hiệu quả và mở rộng hệ thống trong tương lai.

Trong phạm vi đề tài, việc sử dụng MySQL cùng với các cú pháp SQL và ràng buộc dữ liệu phù hợp góp phần đảm bảo hệ thống quản lý kế hoạch học tập hoạt động ổn định, chính xác và đáp ứng tốt các yêu cầu nghiệp vụ đặt ra.

**Vai trò của MySQL trong đề tài**

Trong đề tài xây dựng hệ thống hỗ trợ lập kế hoạch học tập và đăng ký học phần, MySQL đóng vai trò lưu trữ và quản lý toàn bộ dữ liệu của hệ thống. MySQL giúp đảm bảo dữ liệu được lưu trữ an toàn và có tổ chức.

Việc sử dụng MySQL giúp hệ thống dễ dàng mở rộng và bảo trì. Các chức năng truy vấn và cập nhật dữ liệu phục vụ tốt cho việc theo dõi tiến độ học tập của sinh viên.

**Đánh giá ưu điểm và hạn chế của MySQL**

MySQL có ưu điểm là hiệu năng tốt và dễ sử dụng. MySQL hỗ trợ tốt cho các hệ thống Web có quy mô từ nhỏ đến trung bình. MySQL được hỗ trợ rộng rãi và có tài liệu phong phú.

Tuy nhiên, MySQL cần được thiết kế cơ sở dữ liệu hợp lý để đảm bảo hiệu quả sử dụng. Việc thiết kế không phù hợp có thể ảnh hưởng đến hiệu năng và khả năng mở rộng của hệ thống.

#### Tương tác giữa PHP và MySQL

**Tương tác giữa PHP và MySQL trong ứng dụng Web**

Trong các ứng dụng Web động, PHP được sử dụng để xử lý logic phía máy chủ, trong khi MySQL đảm nhiệm vai trò lưu trữ và quản lý dữ liệu. Sự tương tác giữa hai công nghệ này cho phép hệ thống tiếp nhận yêu cầu từ người dùng, thực hiện các thao tác truy xuất dữ liệu và trả kết quả về giao diện Web một cách linh hoạt và hiệu quả.

PHP cung cấp các thư viện hỗ trợ làm việc với MySQL, trong đó MySQLi là một trong những phương pháp phổ biến, cho phép thực hiện các thao tác kết nối và xử lý dữ liệu thông qua các câu lệnh SQL.

**Kết nối cơ sở dữ liệu MySQL bằng PHP**

Để PHP có thể làm việc với MySQL, trước hết cần thiết lập kết nối đến cơ sở dữ liệu. Việc kết nối thành công là điều kiện cần để thực hiện các thao tác thêm, sửa, xóa và truy vấn dữ liệu.

Ví dụ kết nối cơ sở dữ liệu:

<?php

$conn = new mysqli("localhost", "root", "", "ql\_hoc\_tap");

if ($conn->connect\_error) die("Kết nối thất bại");

?>

Đoạn mã trên thể hiện cách PHP thiết lập kết nối đến MySQL thông qua đối tượng mysqli.

**Thao tác thêm dữ liệu từ PHP vào MySQL**

PHP có thể gửi các câu lệnh SQL đến MySQL để thêm dữ liệu mới vào bảng. Thao tác này thường được sử dụng khi người dùng nhập dữ liệu thông qua các biểu mẫu trên giao diện Web.

Ví dụ thêm dữ liệu:

<?php

$conn->query(

"INSERT INTO SinhVien VALUES ('SV001','Nguyễn Văn A','2004-03-18','DA22TTA')"

);

?>

Câu lệnh trên minh họa cách PHP thực hiện thao tác thêm một bản ghi vào cơ sở dữ liệu MySQL.

**Truy vấn dữ liệu từ MySQL bằng PHP**

Để hiển thị dữ liệu cho người dùng, PHP sử dụng câu lệnh SELECT để truy vấn dữ liệu từ MySQL và xử lý kết quả trả về.

Ví dụ truy vấn dữ liệu:

<?php

$result = $conn->query("SELECT MaSinhVien, HoTen FROM SinhVien");

while ($row = $result->fetch\_assoc()) {

echo $row["HoTen"];

}

?>

Đoạn mã trên cho thấy cách PHP truy vấn và xử lý dữ liệu từ bảng trong MySQL.

**Sử dụng Prepared Statement nhằm đảm bảo an toàn dữ liệu**

Để hạn chế các rủi ro bảo mật như tấn công SQL Injection, PHP hỗ trợ cơ chế Prepared Statement, cho phép tách biệt câu lệnh SQL và dữ liệu đầu vào.

Ví dụ Prepared Statement:

<?php

$stmt = $conn->prepare(

"INSERT INTO SinhVien VALUES (?, ?, ?, ?)"

);

$stmt->bind\_param("ssss", $ma, $ten, $ngay, $lop);

$stmt->execute();

?>

Cơ chế này giúp nâng cao tính an toàn và ổn định cho hệ thống khi xử lý dữ liệu do người dùng cung cấp.

**Vai trò của tương tác PHP – MySQL trong hệ thống**

Sự kết hợp giữa PHP và MySQL cho phép xây dựng các hệ thống quản lý dữ liệu động, có khả năng xử lý linh hoạt và đáp ứng tốt các yêu cầu nghiệp vụ. PHP đảm nhiệm việc xử lý logic và giao tiếp với người dùng, trong khi MySQL chịu trách nhiệm lưu trữ dữ liệu một cách có cấu trúc và nhất quán.

Trong phạm vi đề tài, việc sử dụng PHP kết hợp với MySQL giúp hệ thống quản lý kế hoạch học tập hoạt động hiệu quả, ổn định và dễ dàng mở rộng trong tương lai.

### Các công cụ và môi trường hỗ trợ phát triển

Trong quá trình xây dựng hệ thống Web, bên cạnh các công nghệ lập trình, việc sử dụng các công cụ và môi trường hỗ trợ phát triển đóng vai trò quan trọng. Các công cụ này giúp quá trình phát triển, triển khai và kiểm thử hệ thống diễn ra thuận lợi và hiệu quả hơn. Việc lựa chọn công cụ phù hợp góp phần nâng cao chất lượng sản phẩm và giảm thiểu lỗi trong quá trình phát triển.

Các công cụ và môi trường hỗ trợ phát triển được sử dụng trong đề tài bao gồm môi trường máy chủ cục bộ, trình soạn thảo mã nguồn, nền tảng quản lý mã nguồn và trình duyệt Web. Mỗi công cụ đảm nhiệm một vai trò nhất định trong quá trình xây dựng hệ thống.

#### Môi trường máy chủ cục bộ XAMPP

XAMPP là môi trường máy chủ cục bộ được sử dụng để triển khai và kiểm thử các ứng dụng Web trong quá trình phát triển. XAMPP cung cấp đầy đủ các thành phần cần thiết để vận hành một hệ thống Web như máy chủ Web, ngôn ngữ xử lý phía máy chủ và hệ quản trị cơ sở dữ liệu.

Việc sử dụng XAMPP giúp người phát triển có thể xây dựng và kiểm thử hệ thống ngay trên máy cá nhân. Điều này giúp tiết kiệm thời gian và thuận tiện trong việc phát hiện, sửa lỗi trước khi triển khai hệ thống chính thức.

#### Trình soạn thảo mã nguồn Visual Studio Code

Visual Studio Code là trình soạn thảo mã nguồn được sử dụng phổ biến trong phát triển ứng dụng Web. Công cụ này hỗ trợ nhiều ngôn ngữ lập trình và cung cấp các tính năng như tô màu cú pháp, gợi ý mã nguồn và quản lý tập tin dự án.

Nhờ các tính năng hỗ trợ lập trình hiệu quả, Visual Studio Code giúp người phát triển viết mã nhanh hơn và hạn chế lỗi cú pháp. Công cụ này phù hợp cho việc phát triển cả giao diện người dùng và xử lý phía máy chủ trong hệ thống Web.

#### Nền tảng quản lý mã nguồn GitHub

GitHub là nền tảng hỗ trợ lưu trữ và quản lý mã nguồn của dự án. GitHub cho phép theo dõi các phiên bản mã nguồn và hỗ trợ làm việc nhóm hiệu quả. Mỗi thay đổi trong mã nguồn đều được ghi nhận, giúp dễ dàng kiểm soát quá trình phát triển hệ thống.

Trong đề tài, GitHub được sử dụng để lưu trữ mã nguồn của hệ thống và hỗ trợ quản lý quá trình phát triển. Việc sử dụng GitHub giúp đảm bảo an toàn mã nguồn và thuận tiện trong việc sao lưu và khôi phục dữ liệu khi cần thiết.

**Vai trò của GitHub trong dự án**

GitHub được sử dụng trong dự án nhằm đáp ứng các mục tiêu chính sau:

Thứ nhất, GitHub hỗ trợ **quản lý phiên bản mã nguồn**. Mỗi thay đổi trong mã nguồn đều được ghi nhận thông qua các phiên bản (commit), cho phép theo dõi lịch sử chỉnh sửa, xác định nội dung thay đổi và thời điểm thay đổi. Điều này giúp nhóm phát triển dễ dàng quay lại các phiên bản trước khi phát sinh lỗi, đồng thời nâng cao khả năng kiểm soát chất lượng mã nguồn.

Thứ hai, GitHub hỗ trợ **làm việc nhóm và phân công nhiệm vụ**. Thông qua cơ chế phân nhánh, mỗi thành viên có thể phát triển các chức năng riêng biệt mà không ảnh hưởng trực tiếp đến mã nguồn chính. Sau khi hoàn thành, các thay đổi được hợp nhất trở lại một cách có kiểm soát, giúp hạn chế xung đột và lỗi trong quá trình tích hợp.

Thứ ba, GitHub đóng vai trò là **nền tảng lưu trữ và chia sẻ tài liệu dự án**. Các tài liệu như mô tả hệ thống, hướng dẫn cài đặt và báo cáo kỹ thuật được lưu trữ cùng với mã nguồn, giúp đảm bảo tính đồng bộ và thuận tiện cho việc tham khảo, đánh giá và bảo trì hệ thống.

**Các thao tác GitHub được sử dụng trong dự án**

Trong quá trình thực hiện dự án, nhóm sử dụng các thao tác Git cơ bản để quản lý mã nguồn và đồng bộ dữ liệu giữa máy cá nhân và kho lưu trữ trực tuyến trên GitHub.

Ví dụ khởi tạo và đẩy mã nguồn lên GitHub:

git init

git add .

git commit -m "Khởi tạo dự án"

git push origin main

Các thao tác này giúp đảm bảo mọi thay đổi trong mã nguồn đều được lưu trữ an toàn và có thể khôi phục khi cần thiết.

**Lợi ích của GitHub đối với dự án**

Việc sử dụng GitHub trong dự án giúp nâng cao tính chuyên nghiệp trong quy trình phát triển phần mềm. Nhờ khả năng quản lý phiên bản và làm việc nhóm hiệu quả, GitHub giúp giảm thiểu rủi ro mất mát dữ liệu, hạn chế lỗi phát sinh và hỗ trợ tốt cho quá trình bảo trì hệ thống sau khi triển khai.

Bên cạnh đó, GitHub còn giúp minh bạch hóa quá trình phát triển dự án, cho phép giảng viên hoặc người đánh giá dễ dàng theo dõi tiến độ, lịch sử chỉnh sửa và mức độ đóng góp của từng thành viên trong nhóm.

**Đánh giá vai trò của GitHub trong phạm vi đề tài**

Trong phạm vi đề tài, GitHub không chỉ là công cụ lưu trữ mã nguồn mà còn là nền tảng hỗ trợ quản lý quá trình phát triển phần mềm một cách hiệu quả. Việc sử dụng GitHub góp phần đảm bảo hệ thống được phát triển có tổ chức, dễ kiểm soát và sẵn sàng cho việc mở rộng, nâng cấp trong tương lai.

#### Trình duyệt Web

GitHub là nền tảng hỗ trợ lưu trữ và quản lý mã nguồn của dự án. GitHub cho phép theo dõi các phiên bản mã nguồn và hỗ trợ làm việc nhóm hiệu quả. Mỗi thay đổi trong mã nguồn đều được ghi nhận, giúp dễ dàng kiểm soát quá trình phát triển hệ thống.

Trong đề tài, GitHub được sử dụng để lưu trữ mã nguồn của hệ thống và hỗ trợ quản lý quá trình phát triển. Việc sử dụng GitHub giúp đảm bảo an toàn mã nguồn và thuận tiện trong việc sao lưu và khôi phục dữ liệu khi cần thiết.

## Kết luận chương

Chương 2 đã trình bày các cơ sở lý thuyết và các công nghệ được sử dụng trong đề tài xây dựng hệ thống hỗ trợ lập kế hoạch học tập và đăng ký học phần. Nội dung chương tập trung giới thiệu tổng quan về Web, hệ thống Web và kiến trúc Client – Server, làm nền tảng cho việc xây dựng các hệ thống quản lý trực tuyến.

Bên cạnh đó, chương đã trình bày các công nghệ giao diện người dùng và công nghệ xử lý phía máy chủ được sử dụng trong đề tài. Các công nghệ này bao gồm ngôn ngữ đánh dấu siêu văn bản, ngôn ngữ định kiểu, ngôn ngữ kịch bản phía người dùng, ngôn ngữ xử lý phía máy chủ và hệ quản trị cơ sở dữ liệu. Việc giới thiệu các công nghệ giúp làm rõ vai trò và phạm vi ứng dụng của từng thành phần trong hệ thống.

Ngoài ra, chương cũng đề cập đến các công cụ và môi trường hỗ trợ phát triển, góp phần đảm bảo quá trình xây dựng và kiểm thử hệ thống được thực hiện hiệu quả. Những nội dung lý thuyết được trình bày trong chương này là cơ sở để triển khai các bước phân tích, thiết kế và xây dựng hệ thống ở các chương tiếp theo.

# HIỆN THỰC HÓA NGHIÊN CỨU

## Mô tả bài toán

### Mục tiêu hệ thống

Xuất phát từ nhu cầu thực tiễn trong công tác quản lý đào tạo theo học chế tín chỉ, đề tài tập trung nghiên cứu và xây dựng hệ thống nhằm giải quyết các bất cập của quy trình quản lý thủ công hiện tại. Mục tiêu cốt lõi của hệ thống là xây dựng một nền tảng quản lý tập trung trên môi trường Web, cho phép số hóa toàn diện quy trình lập và phê duyệt kế hoạch học tập. Hệ thống được thiết kế để hỗ trợ sinh viên chủ động xây dựng lộ trình học tập cá nhân phù hợp với khung chương trình đào tạo, đồng thời cung cấp công cụ trực quan để theo dõi tiến độ tích lũy tín chỉ và phát hiện sớm các học phần còn nợ hoặc chưa hoàn thành.

Song song với việc hỗ trợ sinh viên, hệ thống hướng đến mục tiêu nâng cao hiệu quả công tác cố vấn học tập thông qua việc cung cấp các công cụ giám sát và phê duyệt trực tuyến. Cụ thể, hệ thống hỗ trợ Cố vấn học tập trong việc tra cứu nhanh hồ sơ học tập, kiểm tra tính hợp lệ của kế hoạch và đưa ra các quyết định phê duyệt hoặc từ chối một cách chính xác dựa trên dữ liệu đã được chuẩn hóa. Ngoài ra, một mục tiêu quan trọng khác là đảm bảo tính chính xác của dữ liệu học vụ thông qua việc tự động hóa các quy tắc kiểm tra ràng buộc như điều kiện tiên quyết, giới hạn tín chỉ và trùng lịch học, từ đó giảm thiểu sai sót so với phương pháp quản lý truyền thống.

### Quy trình nghiệp vụ tổng quát

Quy trình nghiệp vụ của hệ thống được vận hành theo một chu trình khép kín, bắt đầu từ việc quản lý dữ liệu nền tảng và kết thúc bằng việc phê duyệt kế hoạch học tập. Trước tiên, Quản trị viên chịu trách nhiệm thiết lập và chuẩn hóa các dữ liệu đầu vào cần thiết cho hệ thống bao gồm danh mục khoa, lớp, danh sách học phần, cấu trúc chương trình đào tạo và các điều kiện tiên quyết tương ứng. Dựa trên nền tảng dữ liệu này, sinh viên truy cập hệ thống để xem xét tiến độ học tập hiện tại và thực hiện thao tác lập kế hoạch cho học kỳ mới. Tại bước này, hệ thống hỗ trợ sinh viên lựa chọn các học phần dự kiến dựa trên danh sách gợi ý hợp lệ, đồng thời thực hiện kiểm tra tự động các ràng buộc học vụ trước khi cho phép gửi yêu cầu đi.

Sau khi sinh viên hoàn tất việc gửi kế hoạch, quy trình chuyển sang giai đoạn kiểm tra và phê duyệt thuộc trách nhiệm của Cố vấn học tập. Cố vấn học tập tiếp nhận danh sách các kế hoạch chờ duyệt, xem xét chi tiết các học phần sinh viên đã đăng ký và đối chiếu với quy định đào tạo. Dựa trên kết quả kiểm tra chuyên môn, Cố vấn học tập thực hiện thao tác duyệt để xác nhận kế hoạch hoặc từ chối kèm theo lý do cụ thể để sinh viên điều chỉnh. Kết quả của quá trình này được lưu trữ vào cơ sở dữ liệu trung tâm, làm cơ sở cho việc đăng ký học phần chính thức và cập nhật trạng thái tiến độ học tập của sinh viên trong hệ thống.

### Đối tượng sử dụng

Hệ thống được thiết kế để phục vụ ba nhóm đối tượng người dùng chính, trong đó mỗi nhóm nắm giữ các vai trò và chức năng chuyên biệt nhằm đảm bảo quy trình vận hành được thông suốt. Nhóm đối tượng thứ nhất là Quản trị viên (Administrator), đóng vai trò là người quản lý và duy trì sự ổn định của hệ thống. Trách nhiệm chính của nhóm này bao gồm việc cấu hình dữ liệu hệ thống, quản lý các danh mục học vụ như chương trình đào tạo, học phần, lớp học và thực hiện phân quyền truy cập cho các nhóm người dùng khác.

Nhóm đối tượng thứ hai và cũng là đối tượng thụ hưởng chính của hệ thống là Sinh viên. Vai trò của sinh viên trong hệ thống là người trực tiếp thực hiện các nghiệp vụ lập kế hoạch học tập cá nhân. Thông qua hệ thống, sinh viên có thể tra cứu thông tin chương trình đào tạo, theo dõi biểu đồ tiến độ tích lũy tín chỉ, xem thời khóa biểu và gửi các yêu cầu đăng ký kế hoạch học tập đến Cố vấn học tập. Nhóm đối tượng thứ ba là Cố vấn học tập, giữ vai trò giám sát và định hướng lộ trình học tập cho sinh viên. Cố vấn học tập sử dụng hệ thống như một công cụ hỗ trợ chuyên môn để quản lý danh sách sinh viên thuộc lớp chủ nhiệm, xem xét chi tiết tình hình học tập và thực hiện quyền phê duyệt hoặc từ chối các kế hoạch học tập do sinh viên đề xuất.

## Phân tích yêu cầu hệ thống

Quá trình phân tích yêu cầu hệ thống đóng vai trò then chốt trong việc xác định phạm vi nghiệp vụ và các đặc tả kỹ thuật cần thiết để xây dựng ứng dụng. Dựa trên kết quả khảo sát thực tế quy trình quản lý đào tạo tín chỉ, hệ thống được phân tích và thiết kế nhằm đáp ứng nhu cầu của ba nhóm đối tượng chính, đảm bảo tính chặt chẽ về mặt dữ liệu và thuận tiện trong thao tác sử dụng.

### Tác nhân hệ thống

Hệ thống được thiết kế vận hành dựa trên sự tương tác của ba nhóm tác nhân chính, bao gồm Quản trị viên, Sinh viên và Cố vấn học tập, trong đó mỗi tác nhân đảm nhận các vai trò cụ thể nhằm đảm bảo quy trình quản lý được diễn ra thông suốt. Nhóm tác nhân thứ nhất là Quản trị viên (Admin), đóng vai trò là người quản lý cấp cao nhất, chịu trách nhiệm duy trì dữ liệu nền tảng và cấu hình hệ thống. Các hoạt động của quản trị viên tập trung vào việc quản lý các danh mục dữ liệu dùng chung và kiểm soát quyền truy cập của người dùng.

Nhóm tác nhân thứ hai là Sinh viên, đây là đối tượng sử dụng trung tâm và thường xuyên nhất của hệ thống. Vai trò của sinh viên là chủ động tương tác với phần mềm để thực hiện các nghiệp vụ học vụ cá nhân như lập kế hoạch, theo dõi tiến độ và tra cứu thông tin đào tạo. Nhóm tác nhân cuối cùng là Cố vấn học tập, đóng vai trò giám sát và hỗ trợ chuyên môn. Tác nhân này chịu trách nhiệm kiểm tra tính hợp lệ trong lộ trình học tập của sinh viên và đưa ra các quyết định phê duyệt nhằm đảm bảo sinh viên tuân thủ đúng quy chế đào tạo của nhà trường.

### Yêu cầu chức năng

Yêu cầu chức năng của hệ thống được xác định chi tiết dựa trên nhu cầu nghiệp vụ của từng nhóm tác nhân, đảm bảo hỗ trợ đầy đủ quy trình từ lập kế hoạch đến phê duyệt và quản lý. Đối với nhóm sinh viên, hệ thống cung cấp chức năng đăng nhập an toàn để xác thực danh tính người dùng trước khi truy cập dữ liệu cá nhân. Chức năng cốt lõi dành cho sinh viên là lập kế hoạch học tập, cho phép người dùng lựa chọn các học phần dự kiến đăng ký trong học kỳ tới dựa trên danh sách các môn học được mở và lộ trình mẫu. Bên cạnh đó, hệ thống hỗ trợ chức năng theo dõi tiến độ học tập thông qua các biểu đồ trực quan, giúp sinh viên dễ dàng nắm bắt số lượng tín chỉ đã tích lũy và số tín chỉ còn lại cần hoàn thành. Ngoài ra, hệ thống tích hợp các cơ chế cảnh báo tự động về các ràng buộc học vụ như điều kiện tiên quyết hay trùng lịch học để hỗ trợ sinh viên ra quyết định chính xác.

Đối với nhóm Cố vấn học tập, hệ thống cung cấp các chức năng hỗ trợ công tác quản lý lớp chủ nhiệm và phê duyệt kế hoạch. Cố vấn học tập có thể truy cập danh sách sinh viên thuộc quyền quản lý, xem chi tiết hồ sơ học tập và lịch sử đăng ký của từng sinh viên. Chức năng quan trọng nhất của nhóm tác nhân này là duyệt kế hoạch học tập, cho phép cố vấn chấp nhận hoặc từ chối kế hoạch học tập của sinh viên. Trong trường hợp từ chối, hệ thống yêu cầu cố vấn cung cấp lý do hoặc ghi chú hướng dẫn để sinh viên có cơ sở điều chỉnh lại kế hoạch cho phù hợp. Chức năng này giúp số hóa quy trình phê duyệt thủ công, giúp tiết kiệm thời gian và giảm thiểu sai sót trong công tác cố vấn.

Đối với nhóm Quản trị viên, các yêu cầu chức năng tập trung vào việc quản trị dữ liệu hệ thống (Master Data Management). Hệ thống cho phép quản trị viên thực hiện các thao tác thêm, xóa, sửa và cập nhật thông tin đối với các danh mục quan trọng như chương trình đào tạo, danh sách học phần, thông tin lớp học và các điều kiện tiên quyết của môn học. Đồng thời, quản trị viên có quyền quản lý tài khoản người dùng, bao gồm việc cấp mới tài khoản, đặt lại mật khẩu và phân quyền truy cập tương ứng cho sinh viên và cố vấn học tập, đảm bảo tính bảo mật và an toàn dữ liệu cho toàn bộ hệ thống.

### Yêu cầu phi chức năng

Bên cạnh các yêu cầu về mặt chức năng, hệ thống cần đáp ứng các tiêu chuẩn phi chức năng nghiêm ngặt để đảm bảo chất lượng, hiệu năng và trải nghiệm người dùng. Về mặt giao diện, hệ thống cần được thiết kế thân thiện, trực quan và nhất quán với bộ nhận diện thương hiệu của Trường Đại học Trà Vinh, sử dụng tông màu xanh dương chủ đạo để tạo cảm giác quen thuộc cho người sử dụng. Giao diện phải đáp ứng tiêu chuẩn Responsive, có khả năng hiển thị tốt và tương thích trên nhiều loại thiết bị khác nhau như máy tính cá nhân, máy tính bảng và điện thoại thông minh.

Về mặt hiệu năng và độ tin cậy, hệ thống phải đảm bảo tốc độ phản hồi nhanh đối với các thao tác tra cứu và xử lý dữ liệu, đặc biệt là trong các tác vụ tải biểu đồ thống kê hoặc truy vấn danh sách học phần. Tính bảo mật cũng là một yêu cầu phi chức năng quan trọng, đòi hỏi mật khẩu người dùng phải được mã hóa trước khi lưu trữ vào cơ sở dữ liệu và phiên làm việc phải được kiểm soát chặt chẽ để ngăn chặn các truy cập trái phép. Cuối cùng, hệ thống cần đảm bảo tính tương thích cao, hoạt động ổn định trên các trình duyệt web phổ biến hiện nay mà không yêu cầu người dùng phải cài đặt thêm bất kỳ phần mềm hỗ trợ nào khác.

## Phân tích và thiết kế hệ thống

### Sơ đồ Use Case

Mô tả sơ đồ: Sơ đồ Use Case tổng quát của hệ thống bao gồm ba tác nhân chính: Sinh viên, Cố vấn học tập và Quản trị viên. Trong đó, Sinh viên thực hiện các Use Case liên quan đến nghiệp vụ cá nhân như đăng nhập, lập kế hoạch học tập, xem tiến độ tích lũy và tra cứu thời khóa biểu. Tác nhân Cố vấn học tập tương tác với các Use Case quản lý như xem danh sách lớp chủ nhiệm, xem chi tiết kế hoạch của sinh viên và thực hiện phê duyệt hoặc từ chối kế hoạch. Quản trị viên chịu trách nhiệm với nhóm Use Case quản trị hệ thống bao gồm quản lý người dùng, cập nhật chương trình đào tạo và quản lý danh mục học phần.

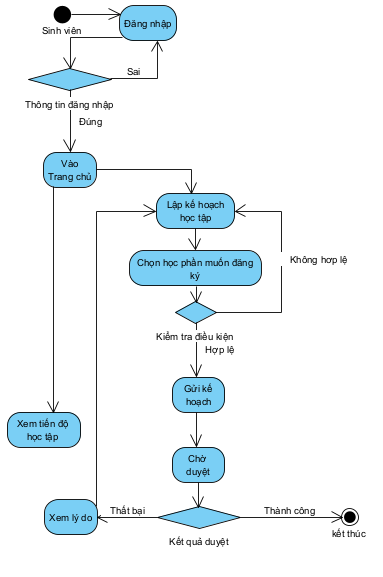


Hình 3.1 Sơ đồ Use Case

### Sơ đồ hoạt động

#### Sơ đồ hoạt động của Sinh Viên

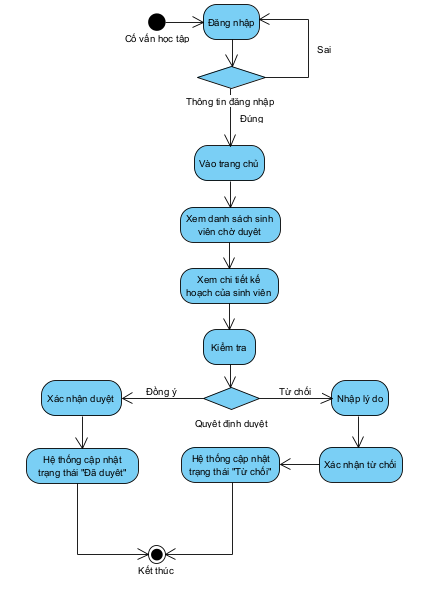
Quy trình bắt đầu khi sinh viên đăng nhập vào hệ thống. Sau khi xác thực thành công, sinh viên truy cập Trang chủ và chọn chức năng Lập kế hoạch học tập. Tại đây, sinh viên thực hiện chọn các học phần dự kiến, hệ thống sẽ tự động kiểm tra các điều kiện ràng buộc. Nếu dữ liệu không hợp lệ, sinh viên cần chọn lại; nếu hợp lệ, kế hoạch được gửi đi và chuyển sang trạng thái chờ duyệt. Cuối cùng, tùy thuộc vào kết quả phê duyệt của Cố vấn học tập mà quy trình sẽ kết thúc nếu thành công hoặc yêu cầu sinh viên xem lý do và thực hiện lập lại kế hoạch nếu thất bại.



Hình 3.2 Sơ đồ hoạt động của Sinh viên

#### Sơ đồ hoạt động của Cố vấn học tập

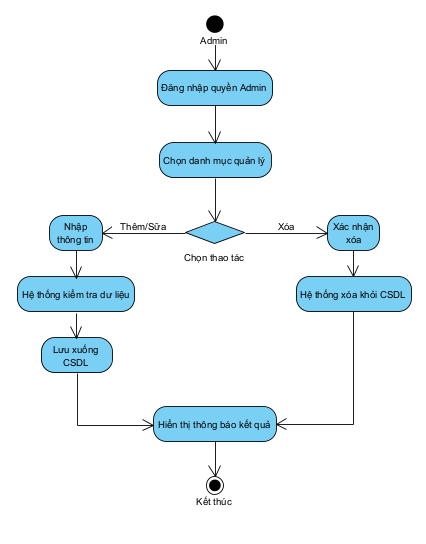
Quy trình bắt đầu khi sinh viên đăng nhập vào hệ thống và truy cập chức năng lập kế hoạch học tập. Tại đây, sinh viên thực hiện chọn các học phần dự kiến, hệ thống sẽ tự động kiểm tra các điều kiện ràng buộc như môn tiên quyết, số tín chỉ. Nếu dữ liệu hợp lệ, kế hoạch được gửi đi và chuyển sang trạng thái chờ duyệt. Cuối cùng, tùy thuộc vào kết quả phê duyệt của Cố vấn học tập mà quy trình sẽ kết thúc thành công hoặc yêu cầu sinh viên xem lý do từ chối để thực hiện điều chỉnh lại.



Hình 3.3 Sơ đồ hoạt động của Cố vấn học tập

#### Sơ đồ hoạt động của Admin

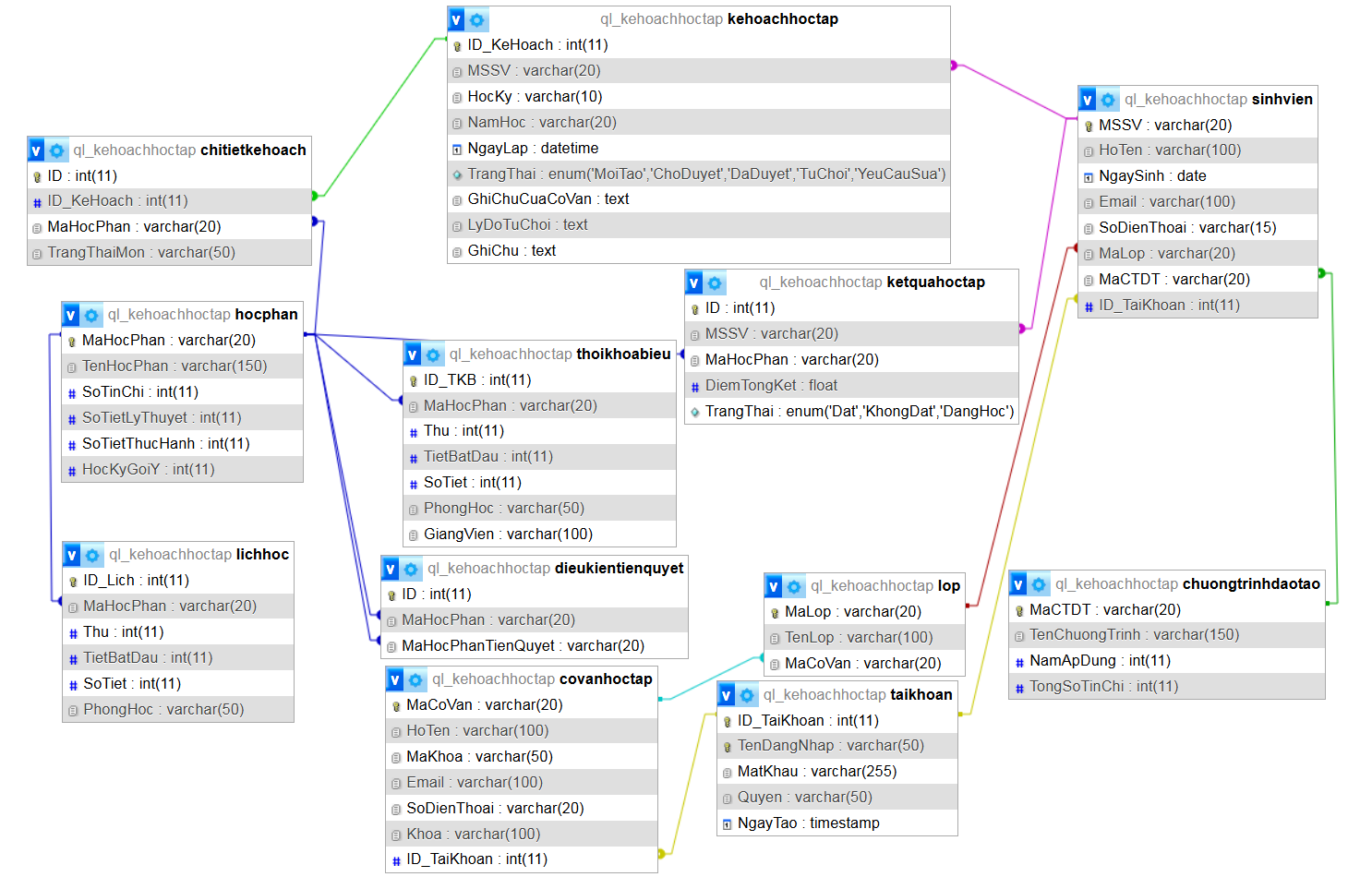
Quản trị viên đăng nhập và lựa chọn danh mục cần quản lý như Học phần, Lớp, Tài khoản. Tùy vào thao tác được chọn là Thêm/Sửa hay Xóa, hệ thống sẽ thực hiện quy trình nhập liệu, kiểm tra tính hợp lệ và lưu trữ vào cơ sở dữ liệu hoặc thực hiện xóa dữ liệu tương ứng, cuối cùng hiển thị thông báo kết quả cho người dùng.



Hình 3.4 Sơ đồ hoạt động của Admin

## Thiết kế Cơ sở dữ liệu

### Sơ đồ thực thể kết hợp



Hình 3.5 Sơ đồ ERD

### Thiết kế chi tiết các bảng

Bảng 3.1 Bảng SINHVIEN

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| MSSV | varchar(20) | Khóa chính, mã số sinh viên |
| HoTen | varchar(100) | Họ tên sinh viên |
| NgaySinh | date | Ngày sinh |
| Email | varchar(100) | Email |
| SoDienThoai | varchar(15) | Số điện thoại |
| MaLop | varchar(20) | Mã lớp |
| MaCTDT | varchar(20) | Mã chương trình đào tạo |
| ID\_TaiKhoan | int | Khóa ngoại tài khoản |

Bảng 3.2 Bảng KEHOACH

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| ID\_KeHoach | int | Khóa chính |
| MSSV | varchar(20) | Sinh viên lập kế hoạch |
| HocKy | varchar(10) | Học kỳ |
| NamHoc | varchar(20) | Năm học |
| NgayLap | datetime | Ngày lập kế hoạch |
| TrangThai | varchar(50) | Trạng thái xử lý |
| GhiChuCuaCoVan | text | Ghi chú cố vấn |
| LyDoTuChoi | text | Lý do từ chối |
| GhiChu | text | Ghi chú thêm |

Bảng 3.3 Bảng CHITIETKEHOACH

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| ID | int | Khóa chính |
| ID\_KeHoach | int | Khóa ngoại kế hoạch |
| MaHocPhan | varchar(20) | Mã học phần |
| TrangThaiMon | varchar(50) | Trạng thái môn học |

Bảng 3.4 Bảng HOCPHAN

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| MaHocPhan | varchar(20) | Khóa chính |
| TenHocPhan | varchar(150) | Tên học phần |
| SoTinChi | int | Số tín chỉ |
| SoTietLyThuyet | int | Số tiết lý thuyết |
| SoTietThucHanh | int | Số tiết thực hành |
| HocKyGoiY | int | Học kỳ gợi ý |

Bảng 3.5 Bảng THOIKHOABIEU

|  |  |  |
| --- | --- | --- |
| **Trường** | **Kiểu** | **Mô tả** |
| ID\_TKB | int | Khóa chính |
| MaHocPhan | varchar(20) | Mã học phần |
| Thu | int | Thứ trong tuần |
| TietBatDau | int | Tiết bắt đầu |
| SoTiet | int | Số tiết |
| PhongHoc | varchar(50) | Phòng học |
| GiangVien | varchar(100) | Giảng viên |

Bảng 3.6 Bảng KETQUAHOCTAP

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| ID | int | Khóa chính |
| MSSV | varchar(20) | Sinh viên |
| MaHocPhan | varchar(20) | Mã học phần |
| DiemTongKet | float | Điểm tổng kết |
| TrangThai | varchar(20) | Đạt / Không đạt |

Bảng 3.7 Bảng DIEUKIENTIENQUYET

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| ID | int | Khóa chính |
| MaHocPhan | varchar(20) | Mã học phần |
| MaHocPhanTienQuyet | varchar(20) | Mã học phần tiên quyết |

Bảng 3.8 Bảng LOP

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| MaLop | varchar(20) | Khóa chính |
| TenLop | varchar(100) | Tên lớp |
| MaCoVan | varchar(20) | Mã cố vấn học tập |

Bảng 3.9 Bảng CHUONGTRINHDAOTAO

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| MaCTDT | varchar(20) | Khóa chính |
| TenChuongTrinh | varchar(150) | Tên chương trình |
| NamApDung | int | Năm áp dụng |
| TongSoTinChi | int | Tổng số tín chỉ |

Bảng 3.10 Bảng COVANHOCTAP

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| MaCoVan | varchar(20) | Khóa chính |
| HoTen | varchar(100) | Họ tên cố vấn |
| MaKhoa | varchar(50) | Mã khoa |
| Email | varchar(100) | Email |
| SoDienThoai | varchar(20) | Số điện thoại |
| Khoa | varchar(100) | Khoa |
| ID\_TaiKhoan | int | Khóa ngoại tài khoản |

Bảng 3.11 Bảng TAIKHOAN

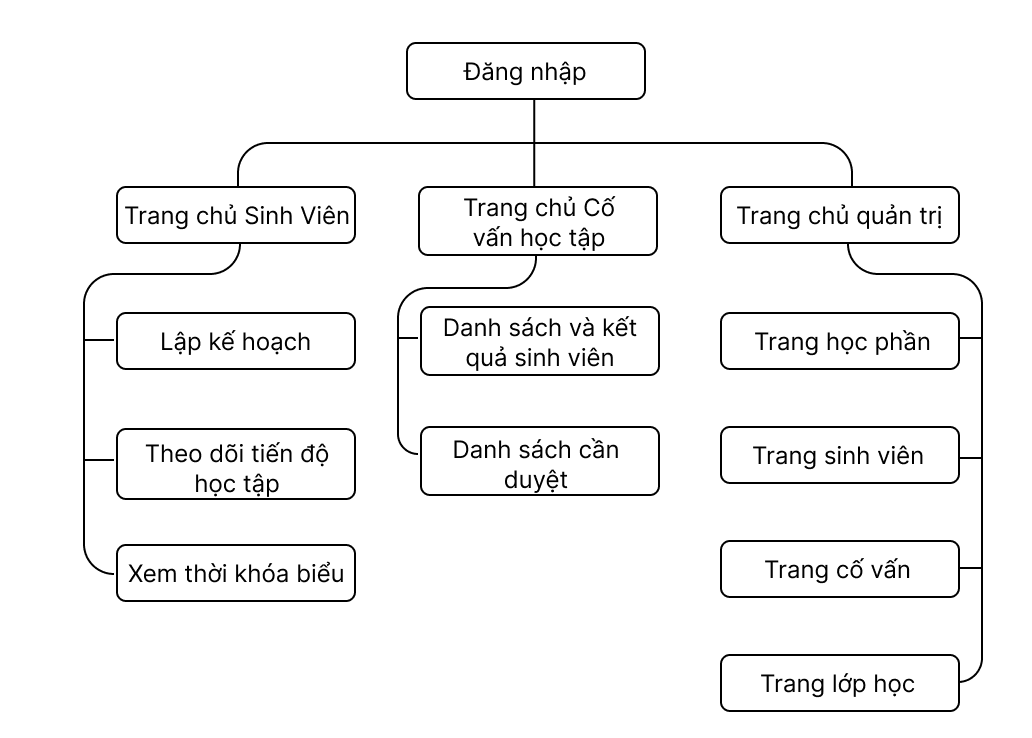
|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| ID\_TaiKhoan | int | Khóa chính |
| TenDangNhap | varchar(50) | Tên đăng nhập |
| MatKhau | varchar(255) | Mật khẩu |
| Quyen | varchar(50) | Quyền |
| NgayTao | datetime | Ngày tạo |

Bảng 3.12 Bảng LOCHHOC

|  |  |  |
| --- | --- | --- |
| Trường | Kiểu | Mô tả |
| ID\_Lich | int | Khóa chính |
| MaHocPhan | varchar(20) | Mã học phần |
| Thu | int | Thứ |
| TietBatDau | int | Tiết bắt đầu |
| SoTiet | int | Số tiết |
| PhongHoc | varchar(50) | Phòng học |

## Thiết kế giao diện

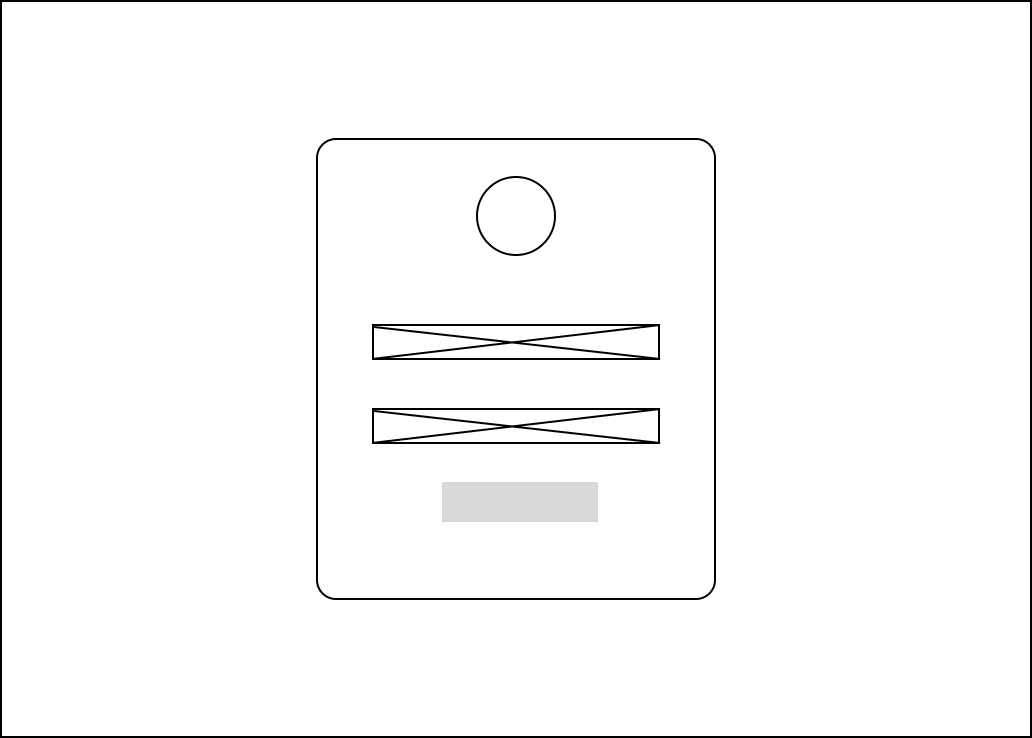
### Thiết kế bố cục chung



Hình 3.6 Sơ đồ bố cục chung

### Giao diện Đăng nhập

Màn hình đăng nhập sử dụng bố cục tập trung với một khung hình chữ nhật được bo góc đặt tại vị trí chính giữa màn hình trên nền trắng. Phần trên cùng của khung là vị trí hiển thị biểu tượng đại diện hoặc logo hệ thống. Ngay phía dưới là hai trường nhập liệu song song dành cho tên đăng nhập và mật khẩu truy cập. Phần cuối cùng của khung chứa nút chức năng đăng nhập được thiết kế trải dài theo chiều ngang nhằm tạo điểm nhấn thị giác và thuận tiện cho thao tác của người dùng.

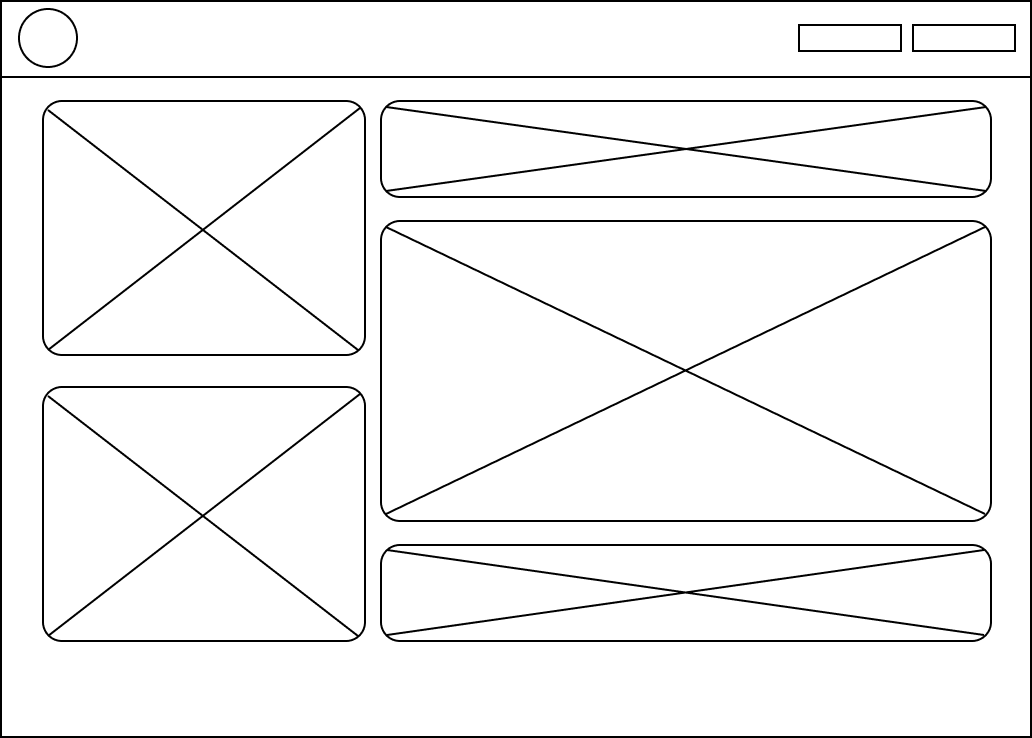


Hình 3.7 Giao diện đăng nhập

### Giao diện Sinh viên

Các giao diện thuộc phân hệ sinh viên tuân thủ quy chuẩn bố cục bảng điều khiển với thanh điều hướng dọc nằm bên trái và khu vực hiển thị nội dung chính nằm bên phải.

Giao diện Trang chủ Sinh viên: Bố cục màn hình được chia thành ba khu vực chính gồm thanh tiêu đề phía trên, thanh thực đơn dọc bên trái và vùng nội dung bên phải. Khu vực nội dung chính được phân chia thành các khối thông tin để tóm tắt trạng thái học tập. Cấu trúc này bao gồm hai khối thông tin nhỏ được sắp xếp theo chiều dọc ở phía bên trái và một khối thông tin lớn nằm bên phải dùng để hiển thị các thông báo quan trọng hoặc biểu đồ thống kê.



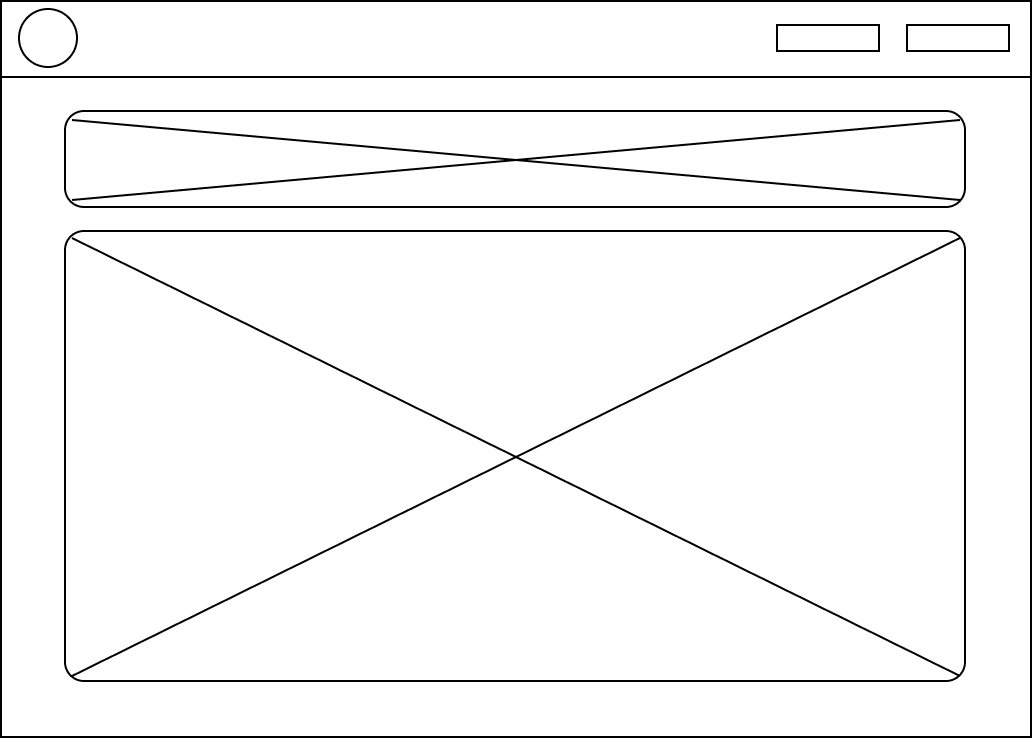
Hình 3.8 Giao diện trang chủ Sinh viên

Giao diện Lập kế hoạch học tập: Giao diện này giữ nguyên phần tiêu đề và thanh điều hướng. Khu vực nội dung chính sử dụng một khối lớn chiếm toàn bộ không gian để phục vụ thao tác chọn môn học. Góc dưới cùng bên phải của màn hình được bố trí hai nút chức năng để hỗ trợ người dùng lưu nháp hoặc gửi kế hoạch lên hệ thống.



Hình 3.9 Giao diện Lập kế hoạch học tập

Giao diện Xem thời khóa biểu: Phần trên của giao diện là khu vực dành cho các bộ lọc thời gian hoặc thông tin tuần học. Phần dưới tích hợp một bảng lưới dữ liệu lớn dùng để hiển thị chi tiết lịch học được sắp xếp theo các thứ trong tuần và tiết học tương ứng.

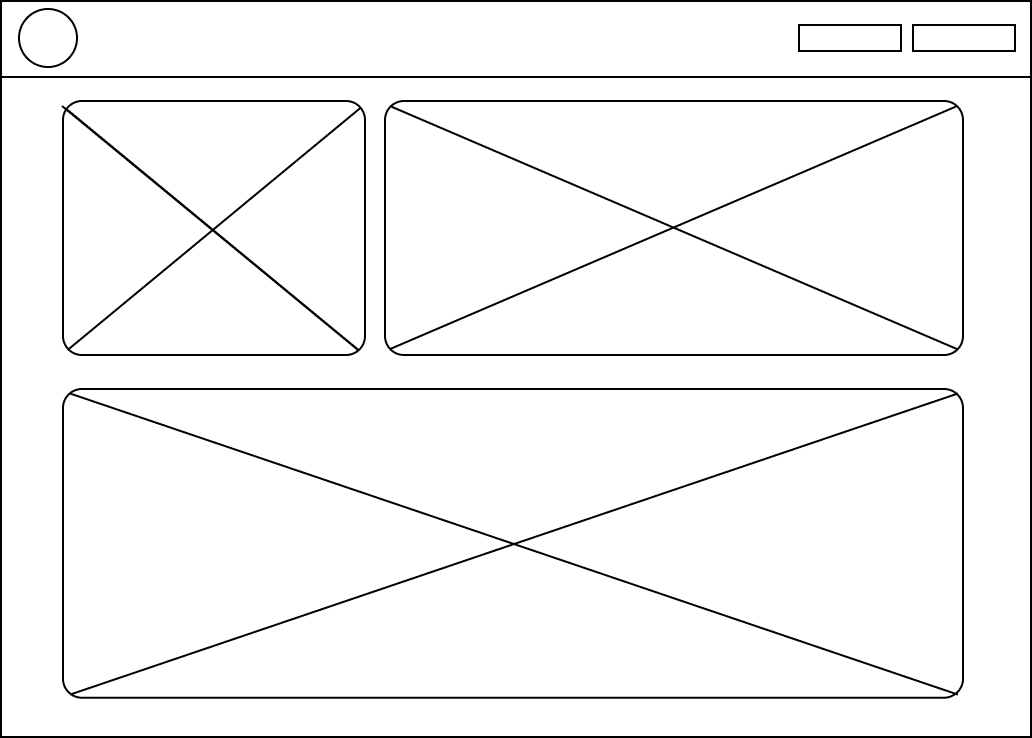


Hình 3.10 Giao diện Xem thời khóa biểu

### Giao diện Cố vấn học tập

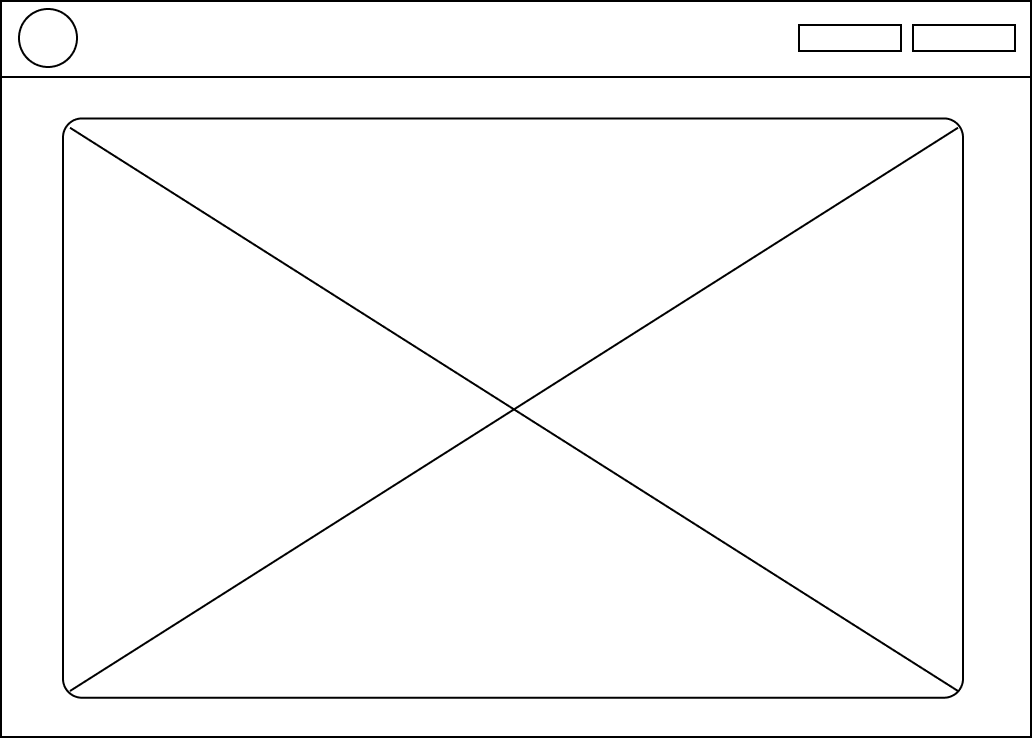
Để đảm bảo tính nhất quán của hệ thống, giao diện cố vấn sử dụng cấu trúc khung sườn tương đồng với sinh viên nhưng có sự điều chỉnh trong cách sắp xếp các thành phần nội dung.

Giao diện Trang chủ Cố vấn: Vùng nội dung chính được tái cấu trúc với hai khối thống kê nằm ngang ở phía trên và một khối nội dung lớn nằm phía dưới dành cho việc hiển thị danh sách các yêu cầu cần xử lý.



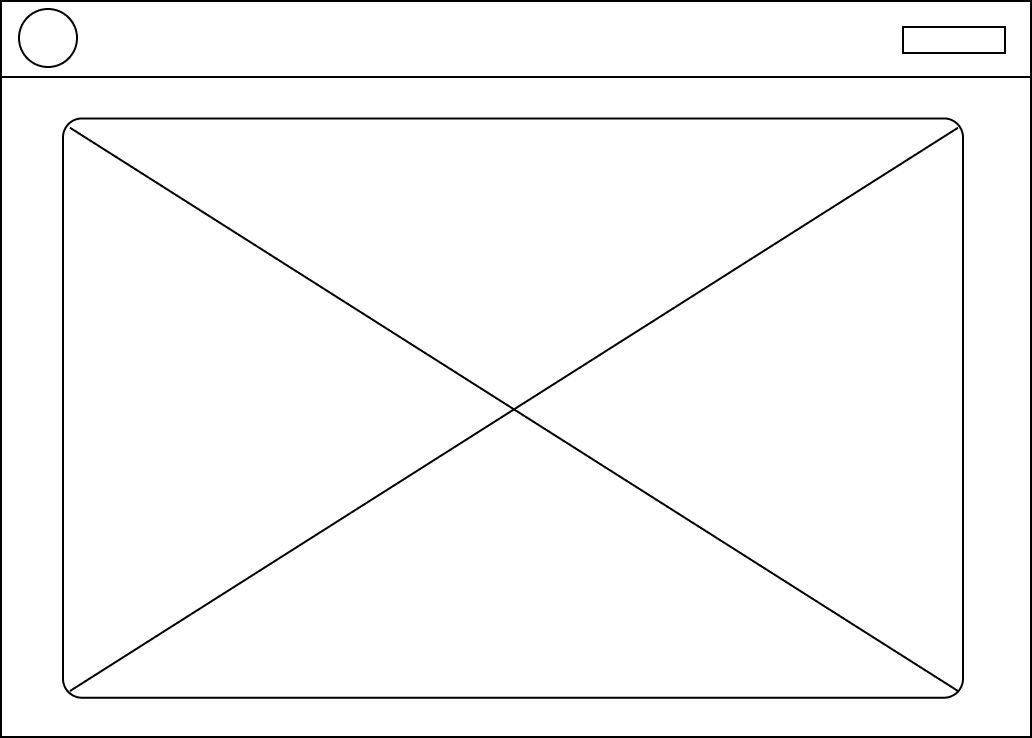
Hình 3.11 Giao diện trang chủ Cố vấn học tập

Giao diện Duyệt kế hoạch: Trọng tâm của màn hình là khu vực hiển thị chi tiết bản kế hoạch nằm ở giữa. Khác với giao diện của sinh viên, các nút hành động như duyệt hoặc từ chối được đặt tại góc trên cùng bên phải của khối nội dung nhằm hỗ trợ thao tác ra quyết định nhanh chóng.



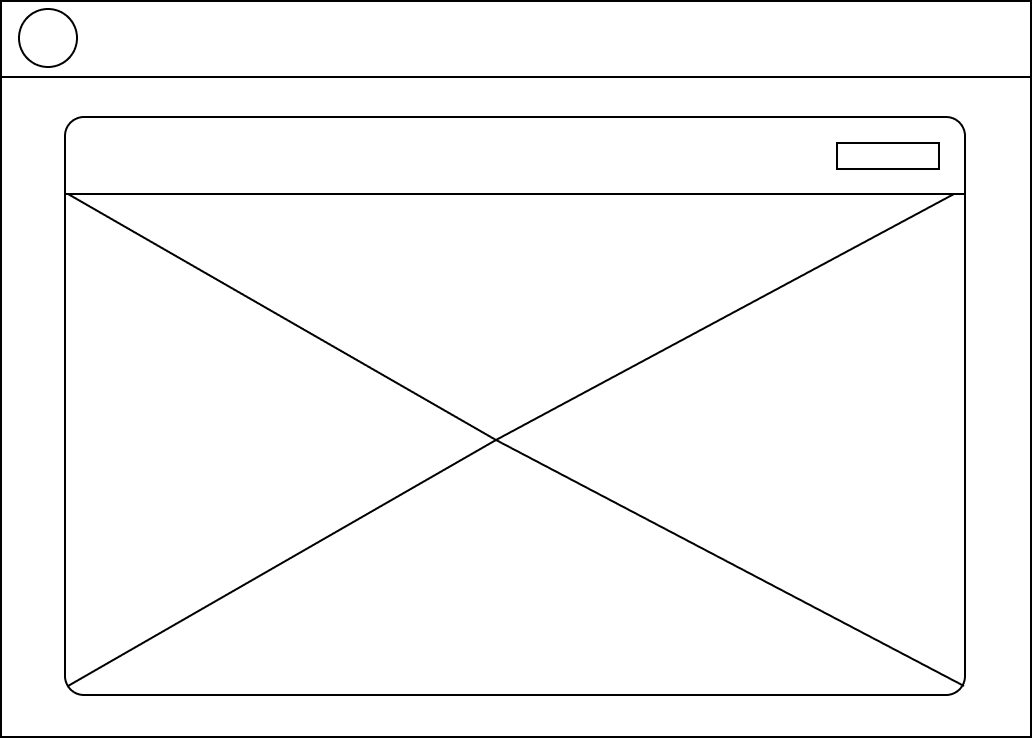
Hình 3.12 Giao diện duyệt Kế hoạch học tập

Giao diện Danh sách sinh viên: Giao diện này sử dụng một bảng dữ liệu kích thước lớn chiếm toàn bộ khu vực nội dung để liệt kê đầy đủ thông tin của các sinh viên thuộc lớp quản lý.



Hình 3.13 Giao diện Xem danh sách sinh viên

Giao diện Duyệt kết quả học tập: Thông tin chi tiết về bảng điểm được hiển thị trong một khung lớn thống nhất. Góc trên bên phải của khung có bố trí một thành phần điều khiển nhỏ phục vụ cho các tác vụ phụ trợ như in ấn hoặc xuất dữ liệu.

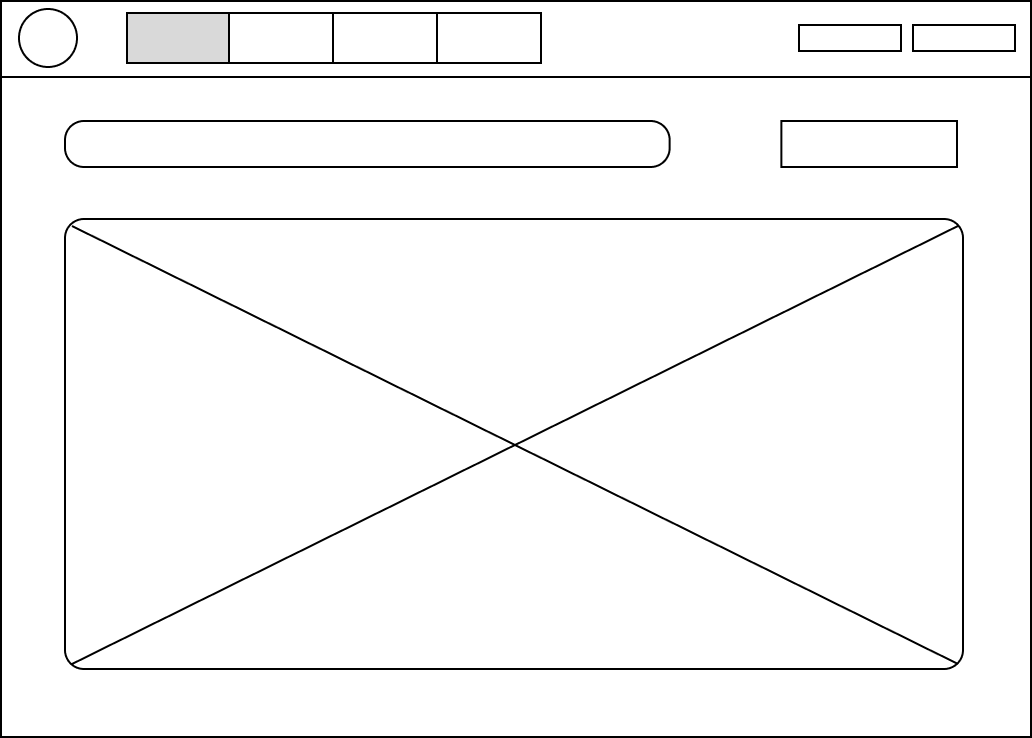


Hình 3.14 Giao diện Duyệt kết quả học tập

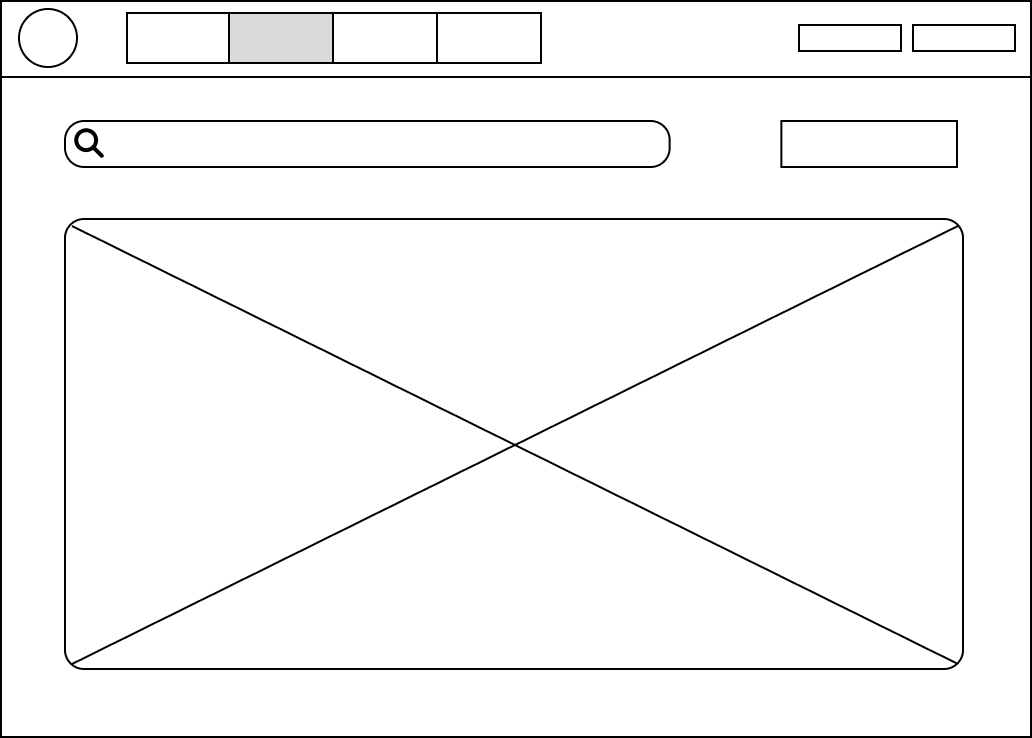
### Giao diện Admin

Nhóm giao diện cuối cùng thể hiện sự đồng bộ cao trong thiết kế các trang quản trị dữ liệu bao gồm Học phần, Sinh viên, Cố vấn và Lớp học.

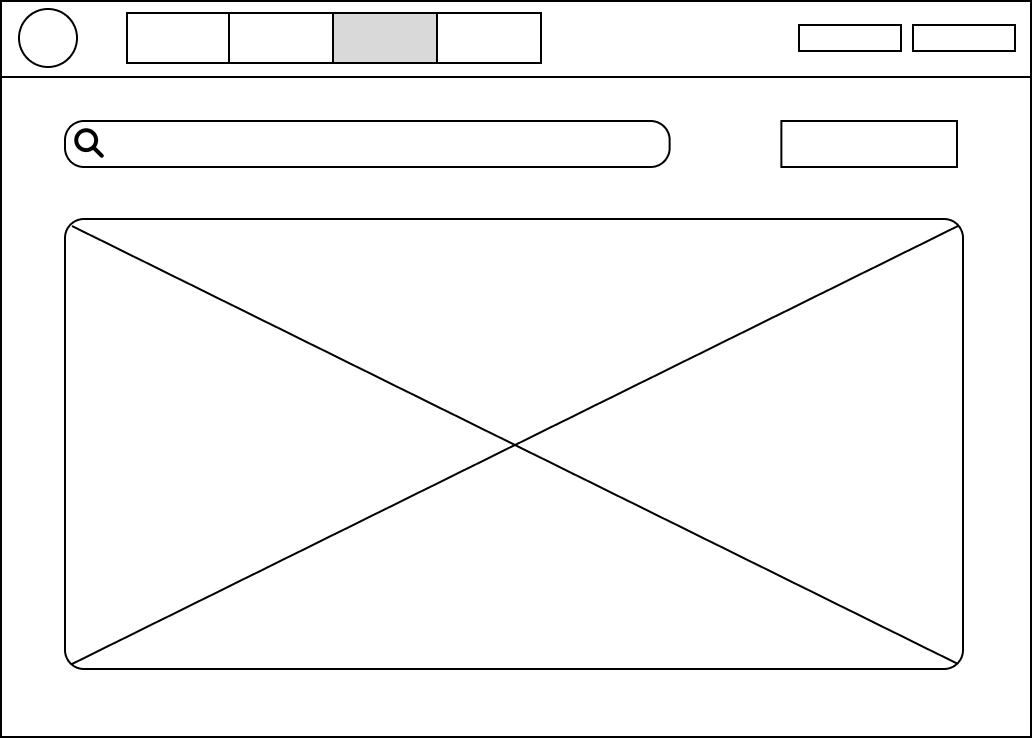
Tất cả các trang này đều tuân theo một bố cục tiêu chuẩn. Ngay phía dưới thanh tiêu đề là thanh công cụ tác nghiệp bao gồm một ô tìm kiếm dài nằm bên trái, các danh sách thả xuống dùng để lọc dữ liệu nằm ở giữa và nút chức năng thêm mới nằm bên phải. Khu vực hiển thị chính bên dưới là một bảng danh sách dữ liệu được chia dòng và cột rõ ràng. Cấu trúc khung sườn thống nhất này giúp người quản trị dễ dàng nắm bắt và thực hiện các thao tác quản lý dữ liệu trên toàn hệ thống.



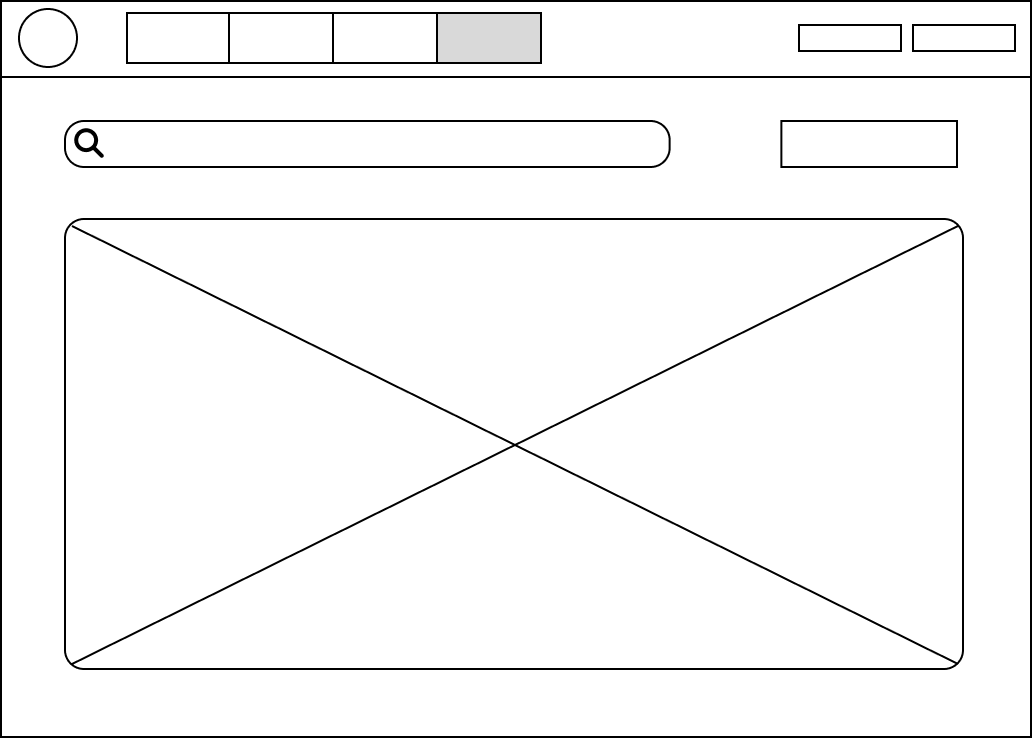
Hình 3.15 Giao diện Danh sách học phần



Hình 3.16 Giao diên Danh sách sinh viên



Hình 3.17 Giao diện Danh sách Cố vấn học tập



Hình 3.18 Giao diện Danh sách lớp

## Quy trình xây dựng ứng dụng

### Môi trường phát triển

Trong quá trình thưc hiện đề tài môi trường phát triển được thiết lập dựa trên kiến trúc Client-Server, sử dụng máy chủ cục bộ để mô phỏng quy trình vận hành thực tế. Hệ thống tích hợp các công cụ chuyên dụng nhằm đảm bảo tính đồng bộ giữa mã nguồn, cơ sở dữ liệu và giao diện người dùng.

#### Môi trường máy chủ cục bộ và bộ công cụ XAMPP

Hệ thống sử dụng bộ giải pháp phần mềm XAMPP để khởi tạo môi trường máy chủ web đa nền tảng. Đây là thành phần cốt lõi chịu trách nhiệm vận hành phần Backend của ứng dụng, bao gồm ba phân hệ chính:

Apache HTTP Server (Máy chủ Web): Đóng vai trò là cổng tiếp nhận các yêu cầu giao thức HTTP từ phía máy khách. Trong đề tài này, Apache được cấu hình module mod\_rewrite để hỗ trợ cơ chế định tuyến của mô hình MVC. Mọi yêu cầu truy cập truy cập Dashboard, gửi form đăng ký kế hoạch đều được Apache điều hướng thông qua tập tin index.php để gọi đến các Bộ điều khiển (Controller) tương ứng.

Hệ quản trị cơ sở dữ liệu MySQL: Đảm nhiệm chức năng lưu trữ dữ liệu bền vững và có cấu trúc. Cơ sở dữ liệu ql\_kehoachhoctap được thiết kế trên MySQL để quản lý các thực thể: Người dùng (users), Sinh viên (students), Cố vấn (advisors) và Kế hoạch học tập (plans). Hệ thống sử dụng bảng mã utf8mb4 để đảm bảo tính toàn vẹn dữ liệu đối với ngôn ngữ tiếng Việt.

Trình thông dịch PHP: Là ngôn ngữ lập trình phía máy chủ (Server-side) chịu trách nhiệm thực thi logic nghiệp vụ. PHP tương tác với MySQL thông qua thư viện PDO nhằm đảm bảo an toàn bảo mật dữ liệu. Trong hệ thống, PHP xử lý các thuật toán phức tạp như: kiểm tra điều kiện tiên quyết của học phần và tính toán tiến độ tích lũy tín chỉ.

#### Môi trường thực thi phía máy khách

Các trình duyệt web hiện đại (Google Chrome, Microsoft Edge) đóng vai trò là môi trường hiển thị và kiểm thử giao diện người dùng (Front-end).

Vai trò:

Hiển thị: Chuyển đổi mã HTML/CSS từ phía Server thành giao diện tương tác trực quan cho Sinh viên và Cố vấn (như các biểu đồ tiến độ, bảng danh sách sinh viên).

Kiểm thử và gỡ lỗi: Sử dụng bộ công cụ Developer Tools để kiểm tra cấu trúc DOM, giám sát các yêu cầu mạng và gỡ lỗi các kịch bản JavaScript, đảm bảo trải nghiệm người dùng mượt mà và đồng nhất trên các thiết bị khác nhau.

### Công cụ hỗ trợ

#### Môi trường phát triển tích hợp Visual Studio Code

Là công cụ soạn thảo mã nguồn chính, hỗ trợ đa ngôn ngữ (PHP, HTML, CSS, JS).

VS Code hỗ trợ quản lý cấu trúc thư mục dự án theo mô hình MVC một cách khoa học, đồng thời cung cấp các tiện ích (Extensions) giúp kiểm tra lỗi cú pháp và định dạng mã nguồn tự động.

#### Công cụ quản trị Cơ sở dữ liệu phpMyAdmin

Cung cấp giao diện đồ họa (GUI) trực quan trên nền web để tương tác với MySQL.

Công cụ này hỗ trợ thực hiện nhanh các tác vụ như: khởi tạo cơ sở dữ liệu, nhập liệu (Import) cấu trúc bảng từ tập tin database.sql và kiểm tra độ chính xác của các câu truy vấn SQL trước khi tích hợp vào mã nguồn.

#### Khung giao diện Bootstrap 5 Framework

Là bộ công cụ hỗ trợ thiết kế giao diện người dùng (Front-end).

Bootstrap 5 giúp xây dựng nhanh hệ thống bố cục đáp ứng, đảm bảo hiển thị tốt trên nhiều loại thiết bị. Nó cung cấp các thành phần giao diện sẵn có như: Thanh điều hướng, Thẻ thông tin, Bảng dữ liệu và các Cửa sổ bật lên dùng cho chức năng xét duyệt kế hoạch.

#### Hệ thống kiểm soát phiên bản GitHub

Đóng vai trò là kho lưu trữ mã nguồn trực tuyến, giúp sao lưu và quản lý các phiên bản của dự án qua từng giai đoạn phát triển.

GitHub hỗ trợ theo dõi lịch sử thay đổi của mã nguồn, giúp dễ dàng phát hiện lỗi và khôi phục lại trạng thái ổn định khi cần thiết.

#### Công cụ gỡ lỗi Browser Developer Tool

Bộ công cụ tích hợp sẵn trong trình duyệt giúp kiểm tra cấu trúc DOM, tinh chỉnh giao diện CSS và gỡ lỗi các kịch bản JavaScript phía máy khách (Client-side) trong quá trình kiểm thử hệ thống.

### Cấu trúc tổ chức mã nguồn

#### Nhóm điều khiển Controllers

Đây là lớp xử lý trung tâm, tiếp nhận yêu cầu từ người dùng và điều phối dữ liệu.

AdminController.php: Điều khiển toàn bộ hoạt động của phân hệ Quản trị viên. Tập tin này chứa các phương thức để xử lý việc thêm mới, cập nhật hoặc xóa dữ liệu đối với các danh mục dùng chung như: Học phần, Lớp hành chính, Tài khoản người dùng và thiết lập điều kiện tiên quyết.

AdvisorController.php: Điều khiển các hoạt động của Cố vấn học tập. Chức năng chính bao gồm: truy xuất danh sách sinh viên thuộc lớp quản lý, xem bảng điểm chi tiết để tham vấn và thực hiện thao tác phê duyệt hoặc từ chối kế hoạch học tập của sinh viên.

AuthController.php: Quản lý phiên làm việc của người dùng. Tập tin này chứa các phương thức kiểm tra thông tin đăng nhập, xác định vai trò người dùng (Phân quyền) để điều hướng về trang chủ tương ứng và xử lý đăng xuất an toàn.

PlanController.php: Chứa thuật toán cốt lõi của hệ thống. Tập tin này xử lý logic kiểm tra ràng buộc môn học (môn tiên quyết, số tín chỉ tối đa/tối thiểu) trước khi cho phép lưu kế hoạch vào cơ sở dữ liệu.

StudentController.php: Điều khiển giao diện làm việc của Sinh viên. Tập tin này cung cấp dữ liệu cho các trang như thống kê tiến độ học tập (Dashboard), biểu mẫu đăng ký kế hoạch và xem thời khóa biểu cá nhân.

#### Nhóm Mô hình Models

Lớp này chứa các phương thức tương tác trực tiếp với Cơ sở dữ liệu (CRUD).

AdminModel.php: Cung cấp các phương thức truy vấn tổng quát phục vụ cho trang quản trị, như thống kê tổng số lượng sinh viên, số lượng môn học và nhập liệu hàng loạt (Import) từ file Excel.

AdvisorModel.php: Truy vấn dữ liệu đặc thù cho Cố vấn, bao gồm: Lấy danh sách lớp được phân công, lấy danh sách kế hoạch đang ở trạng thái "Chờ duyệt" và cập nhật trạng thái duyệt vào bảng plans.

ClassModel.php: Quản lý thực thể "Lớp học". Chứa các hàm lấy danh sách lớp, thêm lớp mới và gán Cố vấn cho lớp.

CourseModel.php: Quản lý thực thể "Học phần". Chứa các hàm lấy thông tin chi tiết môn học, số tín chỉ và truy vấn danh sách môn tiên quyết của một học phần cụ thể.

PlanModel.php: Quản lý thực thể "Kế hoạch học tập". Chịu trách nhiệm lưu trữ chi tiết các môn học sinh viên đã chọn và lịch sử các bản kế hoạch đã lập.

StudentModel.php: Quản lý hồ sơ sinh viên. Chứa các hàm lấy thông tin cá nhân, chương trình đào tạo và lịch sử điểm số (dùng để tính toán tiến độ).

UserModel.php: Quản lý bảng tài khoản (users), thực hiện kiểm tra mật khẩu và cập nhật thông tin cá nhân người dùng.

#### Nhóm Giao diện Views

Chứa mã nguồn HTML/PHP để hiển thị giao diện người dùng.

a. Thư mục admin/ (Giao diện Quản trị):

class\_list.phpvàclass\_edit.php: Hiển thị bảng danh sách các lớp hành chính và biểu mẫu để thêm mới hoặc chỉnh sửa thông tin lớp.

course\_list.phpvàcourse\_edit.php: Giao diện quản lý danh mục môn học, cho phép Admin định nghĩa mã môn, tên môn và số tín chỉ.

course\_prerequisite.php: Giao diện đặc biệt cho phép thiết lập mối quan hệ giữa các môn học (Ví dụ: Cấu trúc dữ liệu là môn tiên quyết của Giải thuật).

course\_schedule.php: Giao diện sắp xếp lịch học (Thời khóa biểu) cho các lớp học phần.

student\_list.phpvàstudent\_edit.php: Giao diện quản lý hồ sơ sinh viên, hỗ trợ tìm kiếm và cập nhật thông tin sinh viên.

b. Thư mục advisor/ (Giao diện Cố vấn):

dashboard.php: Trang tổng quan của Cố vấn, hiển thị các thông báo quan trọng và số lượng yêu cầu cần xử lý.

student\_list.php: Hiển thị danh sách sinh viên thuộc các lớp do Cố vấn phụ trách, kèm nút chức năng để truy cập vào trang xét duyệt.

check\_progress.php: Giao diện chi tiết hiển thị bảng điểm và tiến độ tích lũy của một sinh viên cụ thể, hỗ trợ Cố vấn ra quyết định duyệt.

review\_plan.php: Giao diện xem lại lịch sử các kế hoạch đã duyệt hoặc từ chối.

class\_list.php: Hiển thị danh sách các lớp mà Cố vấn đang chủ nhiệm.

c. Thư mục student/ (Giao diện Sinh viên):

dashboard.php: Màn hình chính của sinh viên, hiển thị biểu đồ tiến độ học tập và tóm tắt trạng thái học kỳ hiện tại.

create\_plan.php: Biểu mẫu lập kế hoạch, bao gồm danh sách các môn học được phép đăng ký (đã lọc theo môn tiên quyết).

my\_plan.php: Trang quản lý các kế hoạch cá nhân, cho phép xem lại trạng thái (Đã duyệt/Từ chối) của các học kỳ trước.

schedule.php: Hiển thị thời khóa biểu chi tiết dạng lịch tuần.

plan\_print.php: Giao diện định dạng sẵn (tối giản CSS) để hỗ trợ sinh viên in ấn kế hoạch học tập ra giấy hoặc xuất file PDF.

d. Thư mục auth/ và layouts/ (Giao diện hệ thống):

login.php: Biểu mẫu đăng nhập vào hệ thống.

change\_password.php: Biểu mẫu đổi mật khẩu cá nhân.

header.php: Chứa thanh điều hướng ngang (Navbar), logo và thông tin người dùng đăng nhập.

sidebar.php: Chứa thanh menu dọc, có khả năng hiển thị động các chức năng tùy theo quyền hạn của người dùng (Admin/Cố vấn/Sinh viên).

footer.php: Chứa thông tin bản quyền và nạp các tập tin JavaScript chung.

#### Tập tin hệ thống và cấu hình

index.php (Tại thư mục gốc): Đóng vai trò là Bộ định tuyến (Router). Tập tin này tiếp nhận mọi yêu cầu HTTP, khởi tạo kết nối cơ sở dữ liệu và gọi đến Controller tương ứng dựa trên tham số truyền vào.

.gitignore: Cấu hình các thư mục hoặc tập tin cần bỏ qua khi đẩy mã nguồn lên GitHub (ví dụ: thư mục cấu hình môi trường cục bộ).

app/database/database\_schema.sql: Tập tin kịch bản SQL chứa câu lệnh khởi tạo cấu trúc bảng dữ liệu ban đầu cho hệ thống.

### Các giai đoạn thực hiện

#### Giai đoạn 1: Khởi tạo môi trường và Cơ sở dữ liệu nền tảng

Đây là giai đoạn thiết lập cho toàn bộ hệ thống:

* Thiết lập máy chủ: Sử dụng bộ công cụ XAMPP để khởi chạy Apache và MySQL. Cấu hình tập tin .htaccess để điều hướng đường dẫn thân thiện (URL Rewrite).
* Xây dựng CSDL **(**database\_schema.sql**):**
* Thiết kế các bảng dữ liệu cốt lõi: users (lưu tài khoản đăng nhập), roles (phân quyền).
* Thiết kế các bảng nghiệp vụ đào tạo: courses (học phần), classes (lớp học), prerequisites (tiên quyết), schedules (thời khóa biểu).
* Thiết kế các bảng nghiệp vụ người dùng: students (hồ sơ sinh viên), advisors (hồ sơ cố vấn), plans (kế hoạch học tập).
* Kết nối Dữ liệu **(**Database.php**):** Xây dựng lớp Database sử dụng thư viện PDO để tạo kết nối an toàn, ngăn chặn SQL Injection cho toàn bộ ứng dụng.

#### Giai đoạn 2: Xây dựng Phân hệ Quản trị

Phân hệ Admin được xây dựng đầu tiên để tạo dữ liệu nguồn cho hệ thống hoạt động.

* Xây dựng Model (AdminModel.php): Viết các hàm thực thi CRUD (Thêm, Xem, Sửa, Xóa) tác động trực tiếp lên toàn bộ CSDL. Ví dụ: hàm addCourse() để thêm môn học mới, assignAdvisor() để phân công cố vấn cho lớp.
* Xây dựng Controller (AdminController.php):
* Xử lý logic quản lý Học phần: Điều hướng đến trang danh sách môn học, xử lý form thêm mới môn học và kiểm tra ràng buộc dữ liệu (mã môn không trùng).
* Xử lý logic quản lý Tiên quyết: Xây dựng chức năng thiết lập mối quan hệ môn học (Môn A là tiên quyết của Môn B).
* Xử lý logic quản lý Lớp và Sinh viên: Import danh sách sinh viên, tạo lớp mới và cấp tài khoản tự động.
* Xây dựng Views (app/views/admin/): Thiết kế các giao diện quản trị chuyên biệt như:
* course\_list.php, course\_edit.php: Bảng quản lý danh mục môn học.
* class\_list.php, student\_list.php: Giao diện quản lý hồ sơ sinh viên và lớp học.
* course\_prerequisite.php**:** Giao diện cấu hình điều kiện tiên quyết trực quan.

#### Giai đoạn 3: Xây dựng Phân hệ Người dùng

Sau khi có dữ liệu từ Admin, tiến hành xây dựng chức năng cho người dùng cuối.

* Phân hệ Cố vấn **(**AdvisorController - AdvisorModel**):**
* Xây dựng chức năng Dashboard (dashboard.php) để hiển thị tổng quan số lượng yêu cầu cần duyệt.
* Phát triển logic "Xét duyệt": Viết hàm lấy danh sách sinh viên theo lớp (student\_list), xem chi tiết bảng điểm (check\_progress) và cập nhật trạng thái kế hoạch (Duyệt/Từ chối).
* Phân hệ Sinh viên **(**StudentController - PlanModel**):**
* Xây dựng thuật toán gợi ý môn học: Dựa trên dữ liệu tiên quyết và lịch sử học tập để lọc ra các môn sinh viên *được phép* đăng ký.
* Phát triển chức năng Lập kế hoạch (create\_plan): Cho phép sinh viên chọn môn, lưu kế hoạch nháp và gửi yêu cầu duyệt.
* Xây dựng chức năng Thời khóa biểu (schedule.php): Hiển thị lịch học dựa trên các môn đã đăng ký trong kế hoạch.

#### Giai đoạn 4: Hoàn thiện Giao diện và Kiểm thử luồng nghiệp vụ

* Tối ưu giao diện (UI/UX):
* Áp dụng Bootstrap 5 để xây dựng giao diện Responsive (tương thích di động).
* Thiết kế hệ thống Menu động (sidebar.php) hiển thị các chức năng khác nhau tùy theo vai trò người đăng nhập (Admin thấy menu quản lý, Sinh viên thấy menu học tập).
* Kiểm thử tích hợp (Integration Testing): Thực hiện kiểm thử trọn vẹn một quy trình khép kín:
* Admin thêm môn học mới và mở lớp.
* Sinh viên đăng nhập, thấy môn học mới và lập kế hoạch đăng ký.
* Cố vấn nhận thông báo, xem xét và bấm "Duyệt".
* Hệ thống cập nhật trạng thái thành công và hiển thị lên tiến độ của sinh viên.

## Kết luận chương

Chương 3 đã trình bày chi tiết quá trình hiện thực hóa nghiên cứu, chuyển đổi các yêu cầu nghiệp vụ và cơ sở lý thuyết thành các bản thiết kế hệ thống cụ thể. Các yêu cầu chức năng và phi chức năng đã được phân tích kỹ lưỡng để xác định phạm vi của hệ thống, bao gồm các tác nhân chính là Quản trị viên, Sinh viên và Cố vấn học tập . Dựa trên kết quả phân tích, hệ thống đã được mô hình hóa thông qua các biểu đồ UML như Use Case, Sơ đồ hoạt động và Sơ đồ tuần tự, giúp làm rõ luồng tương tác và quy trình xử lý nghiệp vụ .

Bên cạnh đó, cơ sở dữ liệu của hệ thống đã được thiết kế và chuẩn hóa trên hệ quản trị MySQL với các bảng dữ liệu chi tiết, đảm bảo khả năng lưu trữ và toàn vẹn dữ liệu cho các nghiệp vụ lập kế hoạch và đăng ký học phần . Giao diện người dùng cũng được thiết kế trực quan, hướng đến trải nghiệm người dùng tốt nhất cho cả ba nhóm đối tượng sử dụng . Cuối cùng, chương đã mô tả quy trình xây dựng ứng dụng, từ việc thiết lập môi trường phát triển đến tổ chức mã nguồn theo kiến trúc MVC . Những kết quả đạt được trong chương này đóng vai trò là nền tảng kỹ thuật vững chắc để tiến hành giai đoạn lập trình và kiểm thử, kết quả cụ thể của hệ thống sau khi hoàn thiện sẽ được trình bày chi tiết trong Chương 4.

# KẾT QUẢ NGHIÊN CỨU

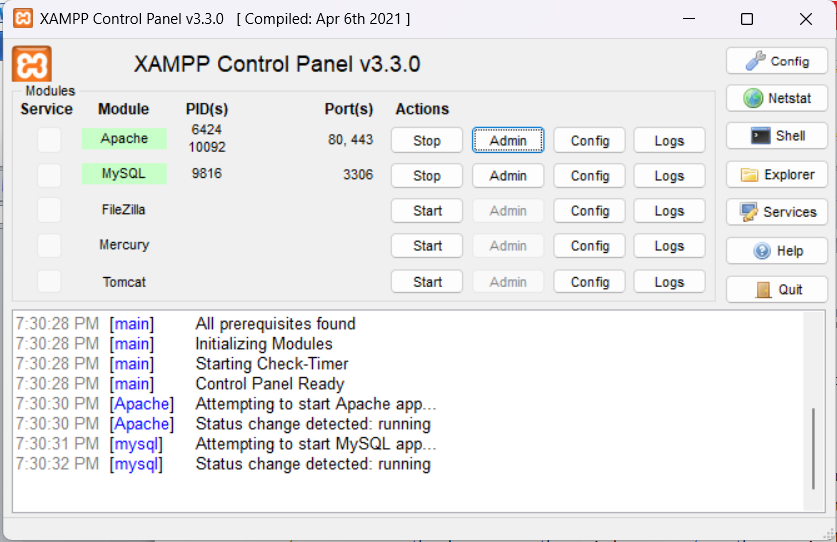
## Kết quả triển khai hệ thống

### Môi trường triển khai Xampp

Hệ thống được triển khai và vận hành thử nghiệm trên môi trường máy chủ cục bộ (Localhost) sử dụng bộ công cụ XAMPP Control Panel v3.3.0. Đây là môi trường tích hợp giúp giả lập máy chủ Web Apache và hệ quản trị cơ sở dữ liệu MySQL.

Apache Module: Dịch vụ Apache chạy ổn định trên cổng 80 (HTTP) và 443 (HTTPS), đóng vai trò tiếp nhận các yêu cầu từ trình duyệt và điều hướng đến mã nguồn PHP của ứng dụng.

MySQL Module: Dịch vụ cơ sở dữ liệu hoạt động trên cổng 3306, đảm bảo khả năng lưu trữ và truy xuất dữ liệu liên tục cho các chức năng đăng ký học phần và quản lý người dùng.



Hình 4.1 Kết quả chạy xampp

### Đánh giá hiệu quả sử dụng tài nguyên

Hệ thống "Phát triển Hệ thống Quản lý Kế hoạch Học tập và Đăng ký Học phần tại Trường Đại học Trà Vinh" được xây dựng dựa trên nền tảng các công nghệ mã nguồn mở như PHP, MySQL, Apache và kiến trúc MVC, mang lại hiệu quả cao trong việc tối ưu hóa tài nguyên phần cứng và hiệu suất xử lý. Kết quả đánh giá cụ thể như sau:

**Hiệu suất máy chủ và mức tiêu thụ tài nguyên**

Môi trường vận hành**:** Hệ thống hoạt động ổn định trên bộ công cụ XAMPP v3.3.0. Qua quá trình kiểm thử trên máy tính cá nhân, các dịch vụ Apache và MySQL chiếm dụng tài nguyên rất thấp.

Khả năng phản hồi**:** Thời gian phản hồi trung bình cho các tác vụ cơ bản như đăng nhập, chuyển trang, tải danh sách lớp là dưới 1 giây. Các tác vụ xử lý phức tạp hơn như "Kiểm tra điều kiện tiên quyết" khi sinh viên đăng ký môn học cũng được xử lý nhanh chóng nhờ thuật toán được tối ưu hóa ở phía Backend.

**Tối ưu hóa Cơ sở dữ liệu**

Cấu trúc dữ liệu: Cơ sở dữ liệu ql\_kehoachhoctap được thiết kế chuẩn hóa theo mô hình quan hệ. Các bảng dữ liệu như sinhvien, hocphan, kehoach được liên kết chặt chẽ thông qua các Khóa chính và Khóa ngoại.

Truy vấn: Việc sử dụng thư viện PDO trong PHP giúp tối ưu hóa các câu lệnh SQL, ngăn chặn việc truy vấn thừa và đảm bảo an toàn dữ liệu. Dung lượng cơ sở dữ liệu gọn nhẹ, không phát sinh dữ liệu rác trong quá trình vận hành.

**Hiệu quả của kiến trúc mã nguồn**

Tổ chức mã nguồn: Hệ thống áp dụng mô hình MVC giúp tách biệt rõ ràng giữa giao diện, logic xử lý và dữ liệu. Điều này không chỉ giúp mã nguồn gọn gàng, dễ bảo trì mà còn giảm tải cho bộ nhớ khi hệ thống chỉ nạp các Controller và Model cần thiết cho từng yêu cầu cụ thể thay vì nạp toàn bộ ứng dụng.

Khả năng tái sử dụng: Các thành phần giao diện chung như header.php, sidebar.php, footer.php được tái sử dụng triệt để, giúp giảm dung lượng mã nguồn và tăng tốc độ tải trang.

**Tối ưu hóa phía Máy khách**

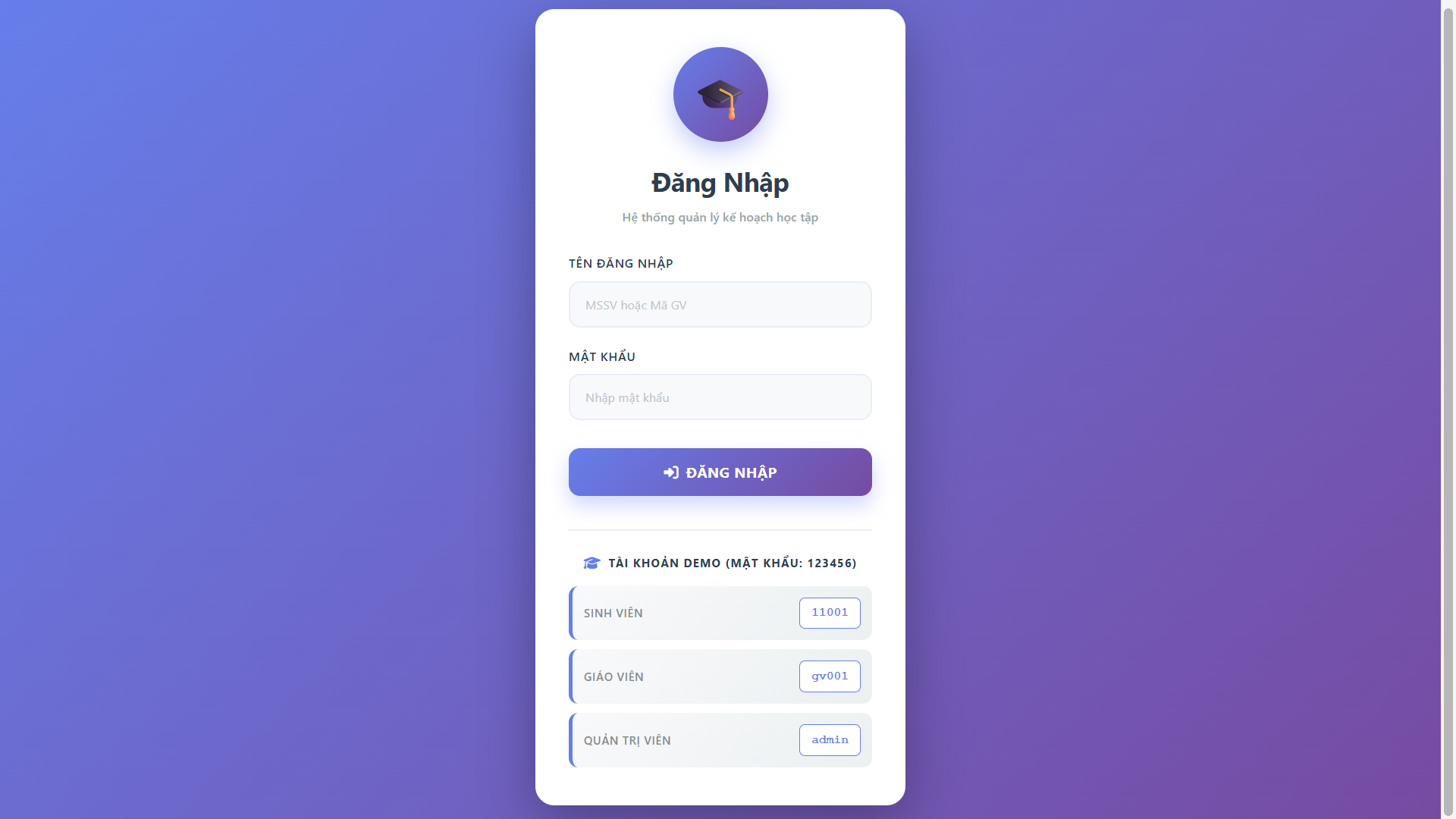
Giao diện: Sử dụng Framework Bootstrap 5 giúp giao diện nhẹ, tải nhanh và tự động điều chỉnh kích thước trên mọi thiết bị mà không cần tải thêm các tài nguyên hình ảnh nặng nề.

Xử lý bất đồng bộ: Việc sử dụng JavaScript và Chart.js để vẽ biểu đồ tiến độ học tập ngay tại trình duyệt giúp giảm bớt gánh nặng xử lý hình ảnh cho máy chủ, tận dụng sức mạnh xử lý của thiết bị người dùng.

## Kết quả giao diện chức năng người dùng

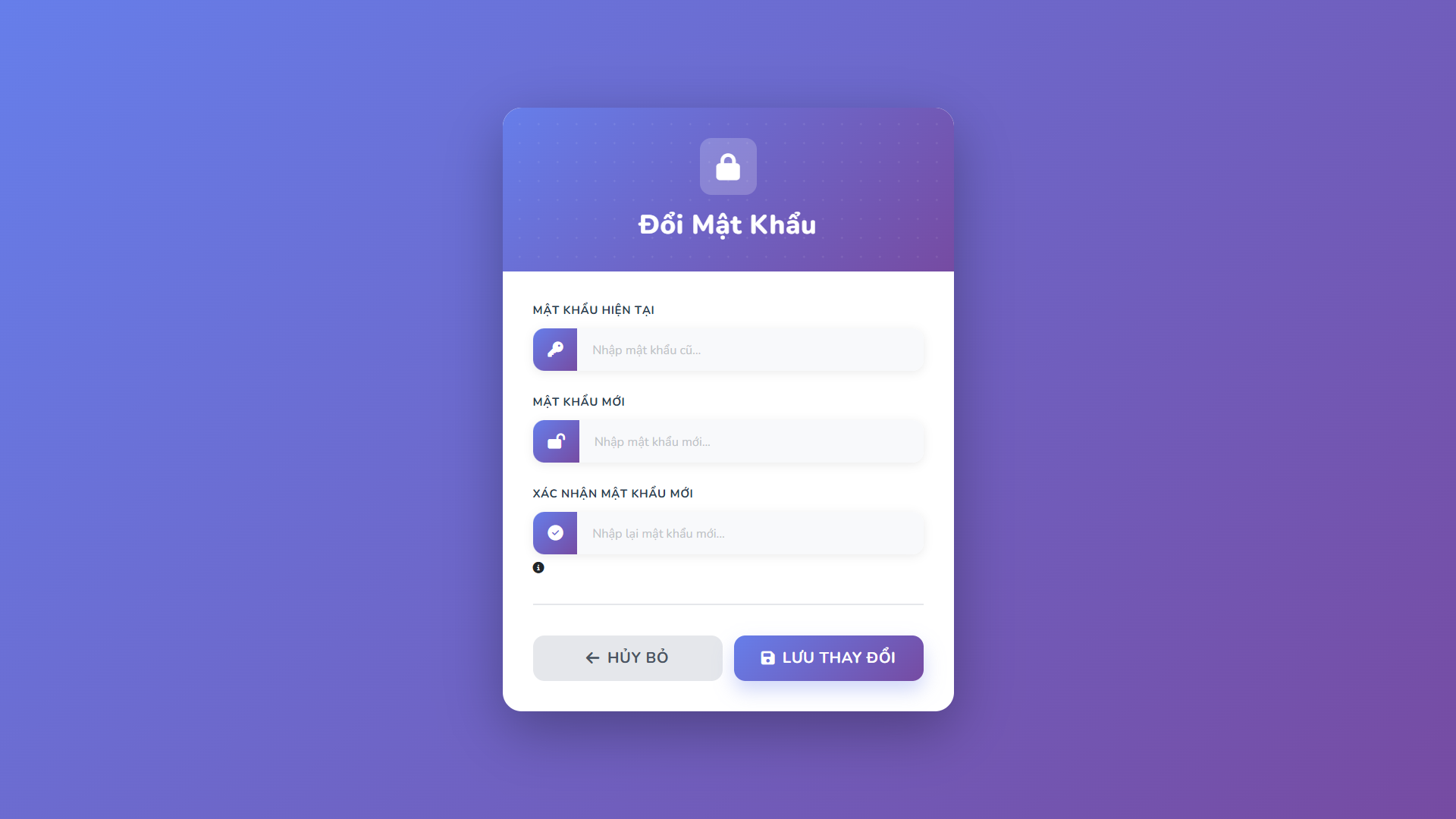
### Giao diện đăng nhập và đổi mật khẩu

Giao diện đăng nhập của hệ thống. Giao diện bao gồm các trường nhập tên đăng nhập mã số sinh viên hoặc mã giảng viên và mật khẩu, kèm theo nút đăng nhập để người dùng thực hiện xác thực. Ngoài ra, hệ thống cung cấp sẵn các tài khoản demo cho các vai trò như sinh viên, giáo viên và quản trị viên nhằm hỗ trợ quá trình kiểm thử và trình diễn chức năng. Giao diện được thiết kế trực quan, bố cục rõ ràng, giúp người dùng dễ dàng thao tác và sử dụng hệ thống.



Hình 4.2 Giao diện đăng nhập

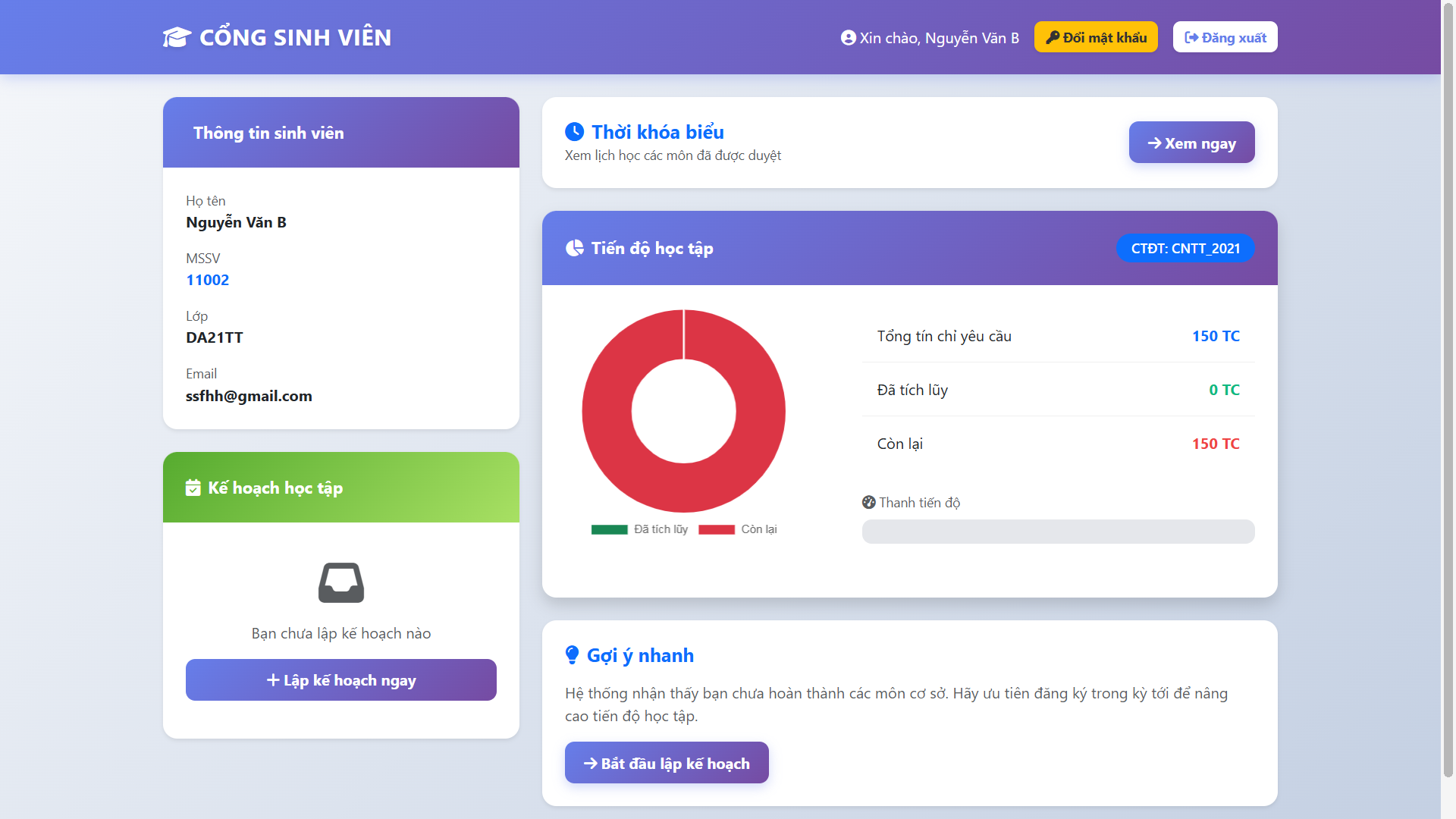
Giao diện đổi mật khẩu của hệ thống. Giao diện cho phép người dùng nhập mật khẩu hiện tại, mật khẩu mới và xác nhận mật khẩu mới nhằm đảm bảo tính chính xác trước khi cập nhật. Các nút chức năng như hủy bỏ và lưu thay đổi hỗ trợ người dùng thao tác linh hoạt. Giao diện được thiết kế trực quan, góp phần tăng cường bảo mật tài khoản và đảm bảo an toàn thông tin người dùng.



Hình 4.3 Giao diện đổi mật khẩu

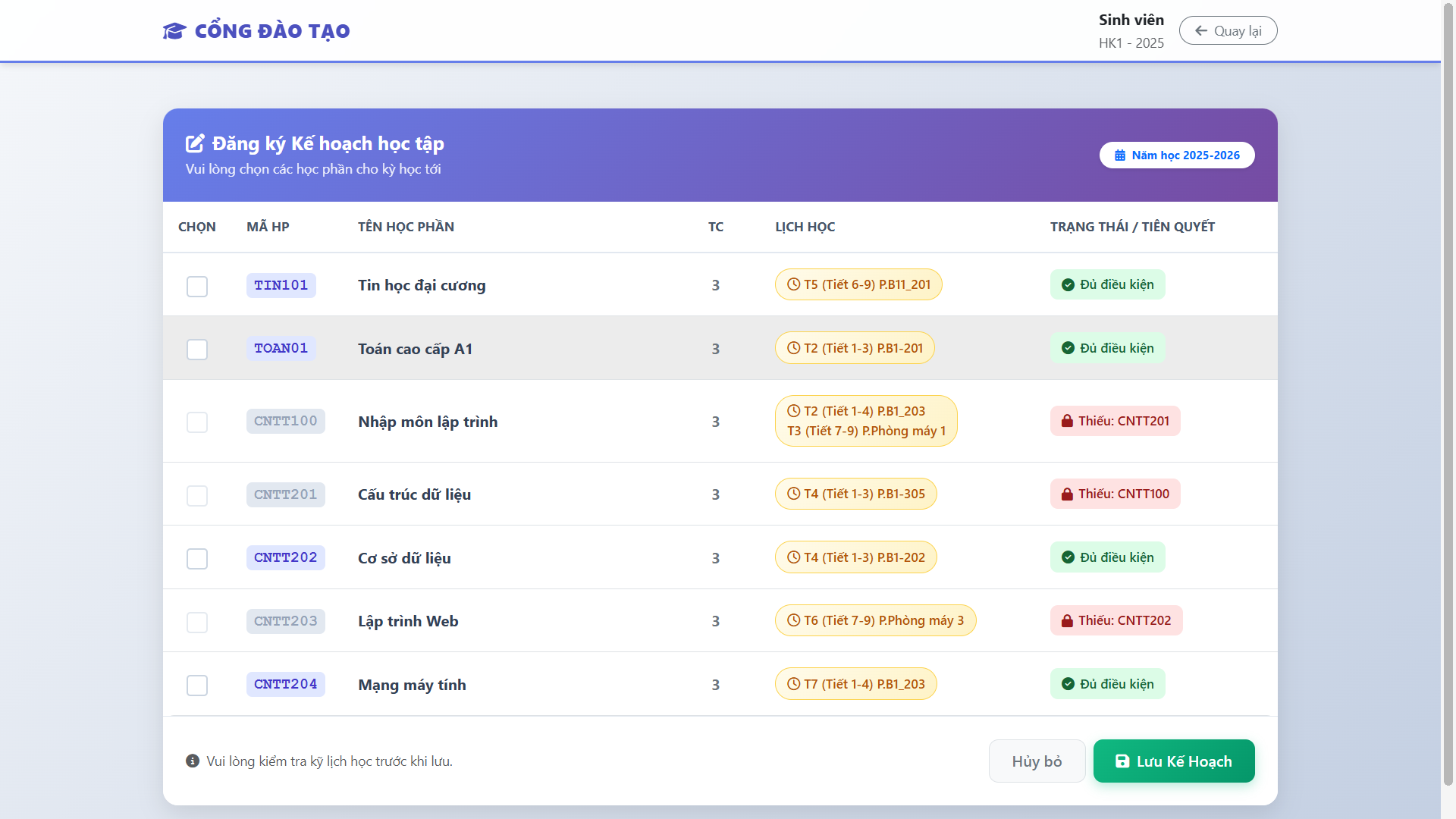
### Giao diện của Sinh viên

Giao diện trang chủ của sinh viên sau khi đăng nhập vào hệ thống. Giao diện hiển thị thông tin cá nhân của sinh viên, bao gồm họ tên, mã số sinh viên, lớp và email. Đồng thời, hệ thống cung cấp các chức năng chính như xem thời khóa biểu, theo dõi tiến độ học tập và lập kế hoạch học tập. Thông tin tiến độ học tập được thể hiện trực quan bằng biểu đồ, giúp sinh viên dễ dàng nắm bắt số tín chỉ đã tích lũy và còn lại. Giao diện được thiết kế rõ ràng, thân thiện, hỗ trợ sinh viên quản lý quá trình học tập một cách hiệu quả.



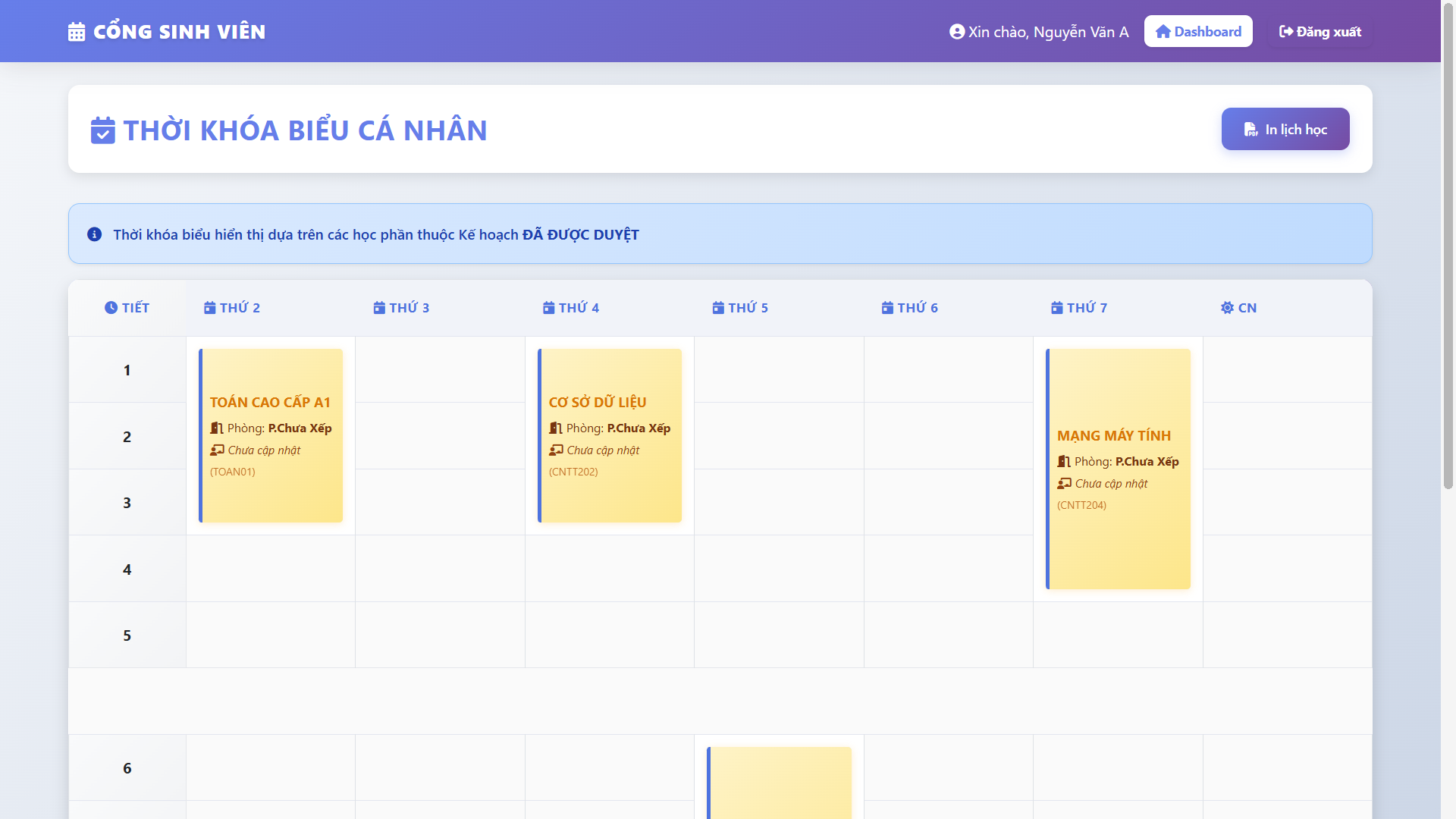
Hình 4.4 Giao diện trang chủ sinh viên

Giao diện đăng ký kế hoạch học tập của sinh viên. Giao diện hiển thị danh sách các học phần theo từng học kỳ, bao gồm mã học phần, tên học phần, số tín chỉ, lịch học và trạng thái điều kiện tiên quyết. Sinh viên có thể lựa chọn các học phần phù hợp để lập kế hoạch học tập cho học kỳ tiếp theo. Hệ thống tự động kiểm tra điều kiện tiên quyết và hiển thị trạng thái nhằm hỗ trợ sinh viên xây dựng kế hoạch học tập hợp lý và đúng quy định.



Hình 4.5 Giao diện đăng ký kế hoạch học tập

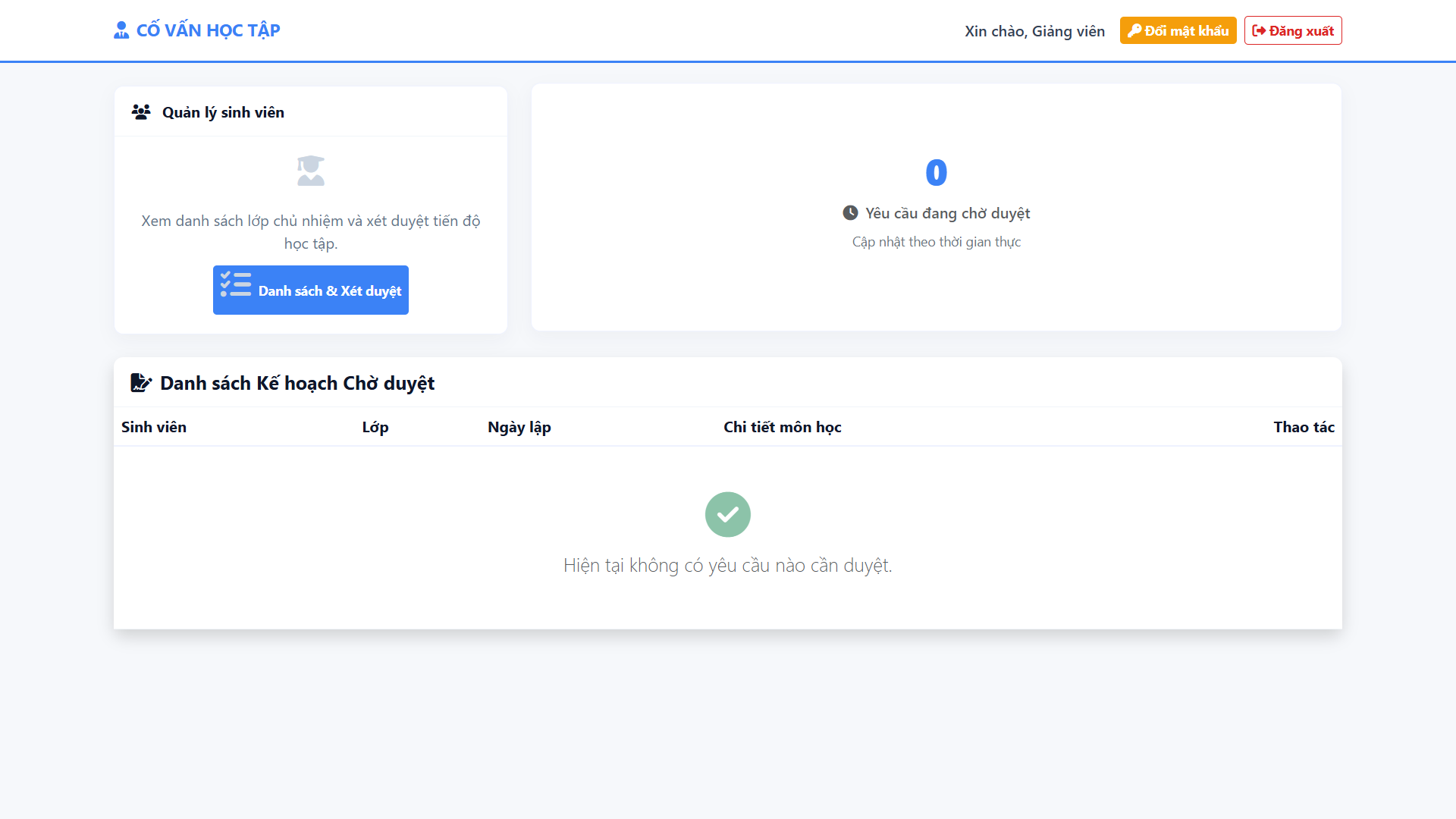
Giao diện xem thời khóa biểu của sinh viên trong hệ thống. Giao diện hiển thị danh sách các học phần đã đăng ký theo từng học kỳ, kèm theo thông tin về thời gian học, phòng học và số tín chỉ. Thông qua giao diện này, sinh viên có thể dễ dàng theo dõi lịch học của mình, từ đó chủ động sắp xếp thời gian học tập một cách hợp lý.



Hình 4.6 Giao diện xem thời khóa biểu

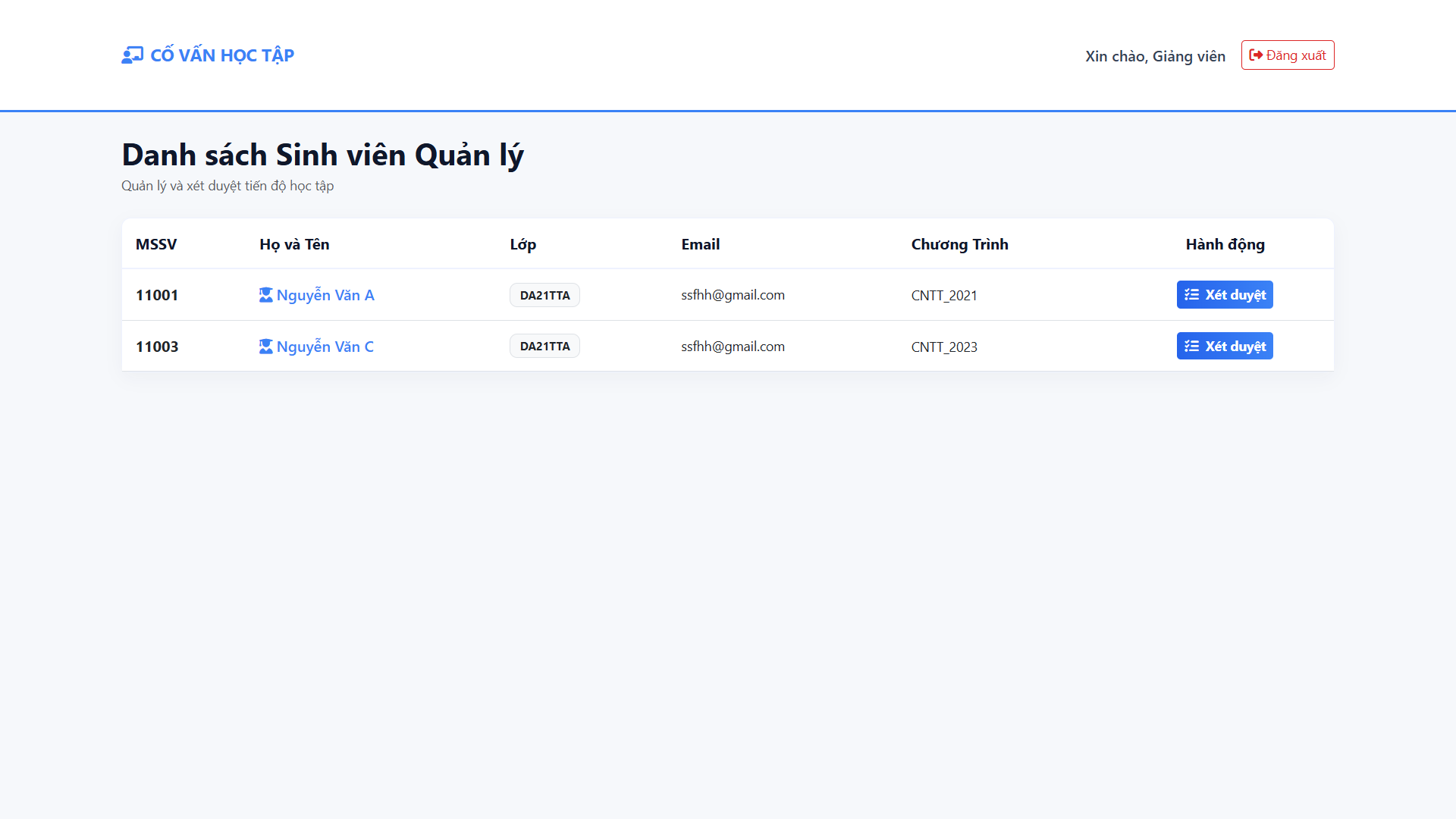
### Giao diện của Cố vấn học tập

Giao diện trang chủ của cố vấn học tập trong hệ thống. Giao diện hiển thị các thông tin tổng quan liên quan đến sinh viên phụ trách, tiến độ học tập và kế hoạch học tập của sinh viên. Thông qua các chức năng được cung cấp, cố vấn học tập có thể theo dõi, tư vấn và hỗ trợ sinh viên trong quá trình lập và điều chỉnh kế hoạch học tập.



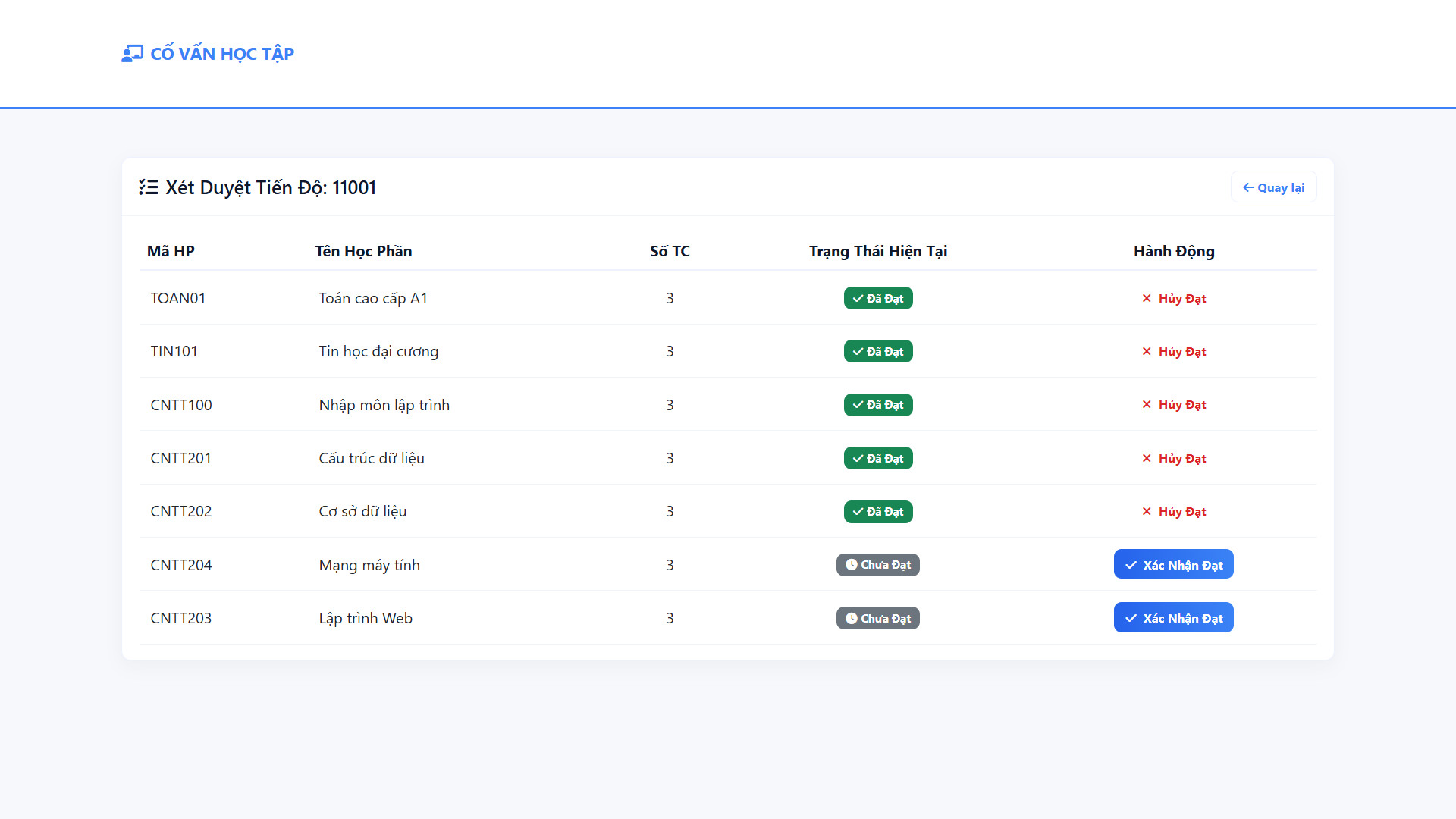
Hình 4.7 Giao diện trang chủ cố vấn học tập

Giao diện danh sách sinh viên trong hệ thống. Giao diện hiển thị thông tin cơ bản của sinh viên như mã sinh viên, họ tên, lớp và trạng thái học tập. Thông qua giao diện này, người dùng có thể dễ dàng theo dõi, tra cứu và quản lý danh sách sinh viên phục vụ cho công tác quản lý và cố vấn học tập.



Hình 4.8 Giao diện danh sách sinh viên

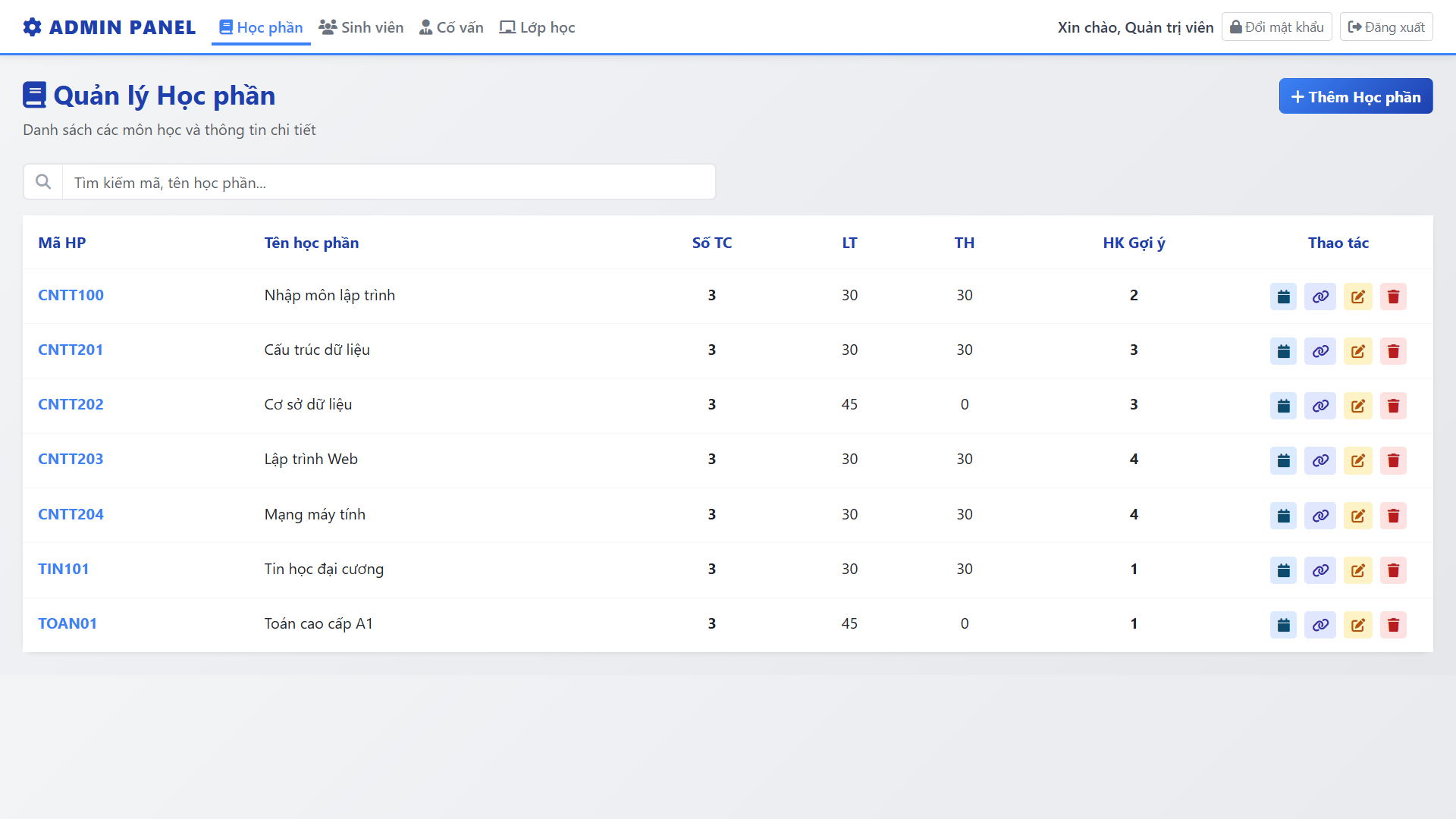
Giao diện xét kết quả học tập của sinh viên trong hệ thống. Giao diện hiển thị thông tin kết quả học tập, bao gồm các học phần đã hoàn thành, số tín chỉ tích lũy và tình trạng đạt hoặc chưa đạt. Thông qua giao diện này, cố vấn học tập hoặc quản trị viên có thể đánh giá tình hình học tập của sinh viên và đưa ra các nhận xét, hỗ trợ kịp thời khi cần thiết.



Hình 4.9 Giao diện xét kết quả học tập

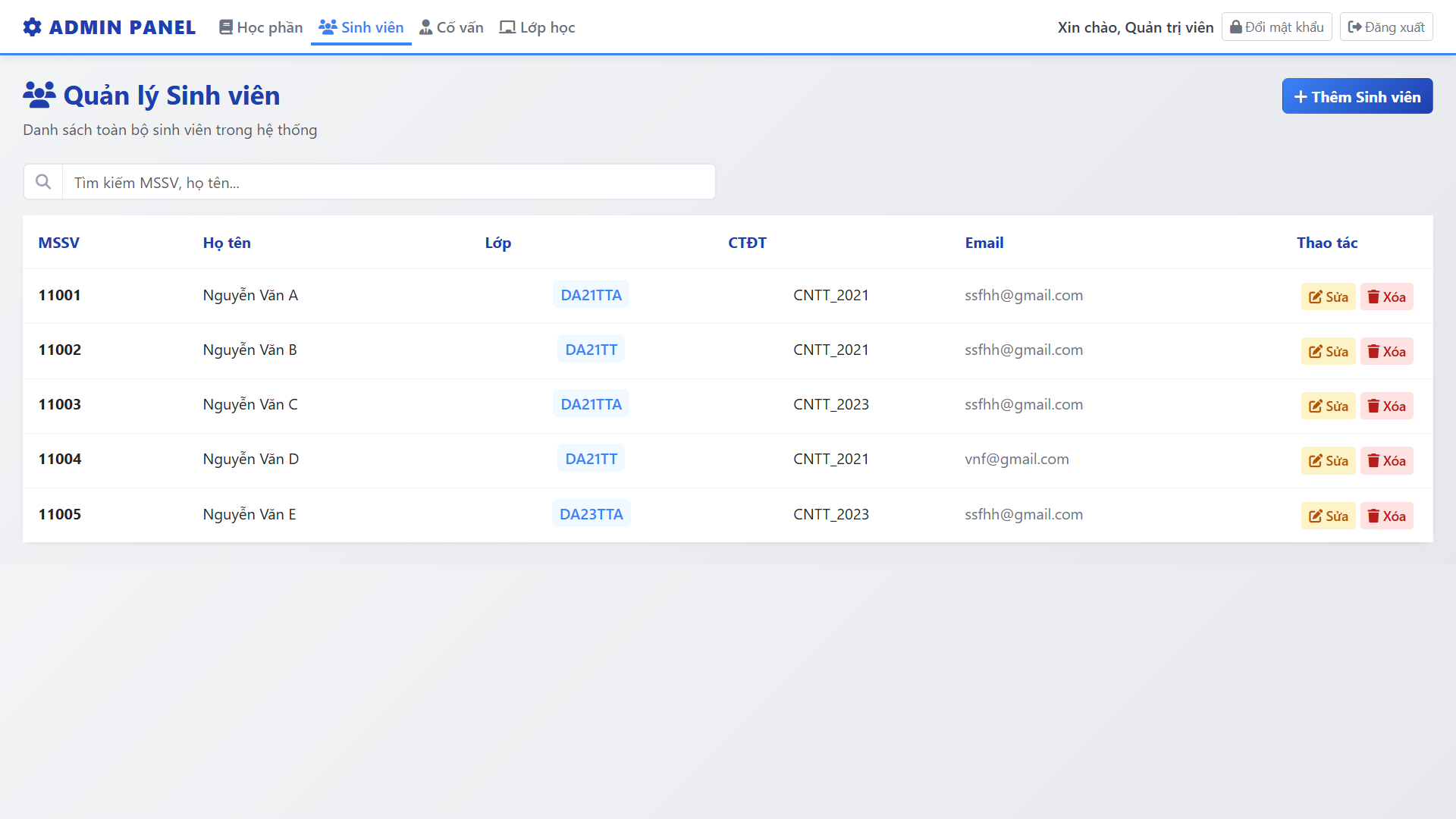
### Giao diện của Admin

Giao diện quản lý học phần trong hệ thống. Giao diện cho phép người quản trị xem, thêm, chỉnh sửa và xóa thông tin các học phần, bao gồm mã học phần, tên học phần, số tín chỉ và điều kiện tiên quyết. Thông qua giao diện này, việc quản lý và cập nhật danh sách học phần được thực hiện thuận tiện, đảm bảo dữ liệu phục vụ cho quá trình lập kế hoạch và đăng ký học tập của sinh viên.



Hình 4.10 Giao diện quản lý học phần

Giao diện quản lý sinh viên trong hệ thống. Giao diện cho phép người quản trị xem, thêm, chỉnh sửa và cập nhật thông tin sinh viên như mã sinh viên, họ tên, lớp, email và trạng thái học tập. Thông qua giao diện này, công tác quản lý thông tin sinh viên được thực hiện hiệu quả, đảm bảo dữ liệu chính xác và hỗ trợ tốt cho quá trình quản lý đào tạo.



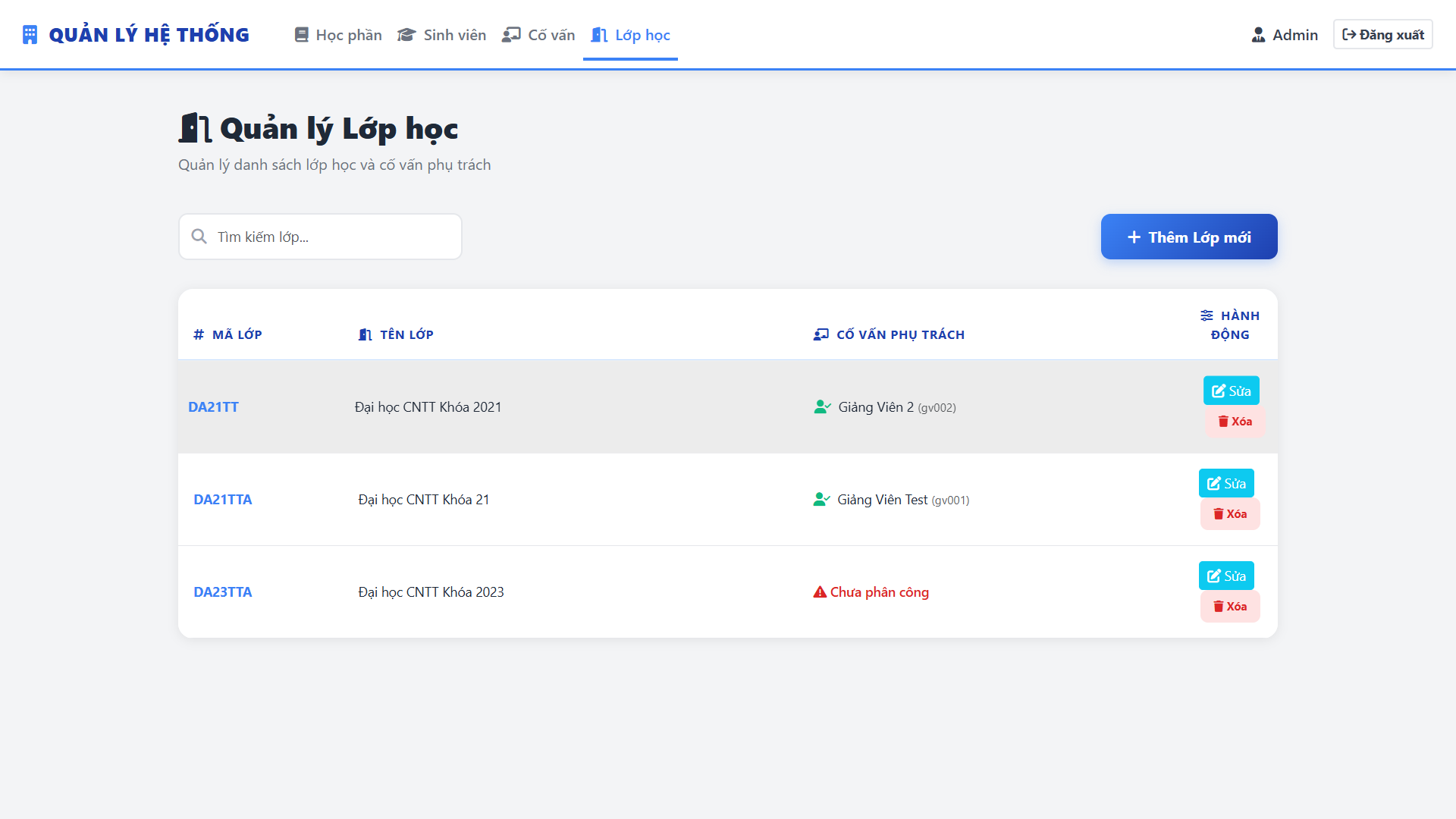
Hình 4.11 Giao diện quản lý sinh viên

Giao diện quản lý cố vấn học tập trong hệ thống. Giao diện cho phép người quản trị xem, thêm, chỉnh sửa và cập nhật thông tin cố vấn học tập, bao gồm mã cố vấn, họ tên, bộ môn và thông tin liên hệ. Thông qua giao diện này, hệ thống hỗ trợ quản lý và phân công cố vấn học tập cho sinh viên một cách hiệu quả, góp phần nâng cao chất lượng công tác tư vấn học tập.



Hình 4.12 Giao diện quản lý cố vấn học tập

Giao diện quản lý lớp học trong hệ thống. Giao diện cho phép người quản trị xem, thêm, chỉnh sửa và cập nhật thông tin lớp học như mã lớp, tên lớp, khóa học và cố vấn học tập phụ trách. Thông qua giao diện này, việc quản lý và tổ chức lớp học được thực hiện thuận tiện, đảm bảo dữ liệu phục vụ cho công tác quản lý đào tạo.



Hình 4.13 Giao diện quản lý lớp học

## Đánh giá hệ thống

### Đánh giá về mặt kỹ thuật

Hệ thống " Phát triển Hệ thống Quản lý Kế hoạch Học tập và Đăng ký Học phần tại Trường Đại học Trà Vinh" đã được xây dựng hoàn thiện và vận hành ổn định trên môi trường kiểm thử. Dựa trên các công nghệ và kiến trúc đã áp dụng, hệ thống được đánh giá cụ thể qua các tiêu chí sau:

**Kiến trúc phần mềm**

Mô hình MVC: Hệ thống tuân thủ chặt chẽ mô hình kiến trúc MVC. Việc tách biệt rõ ràng giữa giao diện hiển thị, logic xử lý nghiệp vụ và tương tác cơ sở dữ liệu giúp mã nguồn trong sáng, dễ dàng bảo trì và mở rộng tính năng sau này.

Tổ chức mã nguồn: Cấu trúc thư mục được tổ chức khoa học, phân chia rõ ràng các phân hệ dành cho Admin, Sinh viên và Cố vấn, giúp việc quản lý phiên bản và làm việc nhóm trên GitHub trở nên thuận lợi.

**Cơ sở dữ liệu**

Thiết kế chuẩn hóa: Cơ sở dữ liệu MySQL được thiết kế đạt chuẩn, đảm bảo tính toàn vẹn dữ liệu thông qua các ràng buộc khóa chính và khóa ngoại chặt chẽ. Sơ đồ thực thể kết hợp phản ánh đúng nghiệp vụ quản lý đào tạo tín chỉ.

Hiệu suất truy vấn: Sử dụng thư viện PDO để kết nối và truy vấn dữ liệu giúp tăng tốc độ xử lý, đồng thời ngăn chặn hiệu quả các nguy cơ tấn công SQL Injection.

**Giao diện và Trải nghiệm người dùng**

Tính tương thích**:** Nhờ sử dụng Framework Bootstrap 5, giao diện hệ thống hiển thị tốt và tự động tương thích trên nhiều loại thiết bị và độ phân giải màn hình khác nhau.

Tính trực quan: Việc tích hợp thư viện Chart.js để vẽ biểu đồ tiến độ học tập giúp sinh viên dễ dàng nắm bắt tình hình tích lũy tín chỉ một cách trực quan, thay vì phải xem các bảng số liệu khô khan.

**Hiệu năng hệ thống**

Hệ thống hoạt động mượt mà trên môi trường máy chủ cục bộ (XAMPP). Các chức năng kiểm tra điều kiện tiên quyết, tính toán tín chỉ và xử lý đăng ký học phần diễn ra với độ trễ thấp, đáp ứng tốt yêu cầu về tốc độ phản hồi.

### Tự so sánh với mục tiêu ban đầu

Sau quá trình thiết kế, cài đặt và kiểm thử, nhóm thực hiện đã tiến hành đối chiếu các chức năng thực tế của hệ thống với những mục tiêu nghiên cứu được đề ra trong đề cương chi tiết. Kết quả so sánh cho thấy hệ thống đã đáp ứng tốt các yêu cầu cốt lõi, cụ thể như sau:

Thứ nhất, về mục tiêu xây dựng hệ thống quản lý tập trung, đồ án đã hoàn thiện việc phát triển một nền tảng web thống nhất kết nối ba nhóm đối tượng người dùng chính là Quản trị viên, Sinh viên và Cố vấn học tập. Hệ thống đã thiết lập thành công cơ chế xác thực và phân quyền chặt chẽ, đảm bảo mỗi nhóm người dùng đều có giao diện và chức năng riêng biệt phù hợp với vai trò của mình, hoạt động ổn định trên cùng một cơ sở dữ liệu.

Thứ hai, đối với mục tiêu số hóa quy trình lập kế hoạch học tập, hệ thống đã thay thế hoàn toàn phương pháp lập kế hoạch thủ công bằng công cụ trực tuyến. Sinh viên hiện nay có thể chủ động truy cập hệ thống để tra cứu chương trình đào tạo, xem danh sách các học phần được gợi ý và thực hiện thao tác gửi yêu cầu đăng ký kế hoạch học tập trực tiếp đến cố vấn ngay trên giao diện web.

Thứ ba, một trong những mục tiêu kỹ thuật quan trọng là kiểm soát các ràng buộc học vụ đã được giải quyết triệt để. Hệ thống đã tích hợp thành công các thuật toán kiểm tra tự động đối với điều kiện tiên quyết, số tín chỉ tối đa/tối thiểu và trùng lịch học. Khi sinh viên thao tác chọn môn, hệ thống có khả năng phát hiện và hiển thị cảnh báo tức thời nếu học phần đó chưa đủ điều kiện đăng ký, giúp giảm thiểu sai sót so với quy trình cũ.

Thứ tư, mục tiêu hỗ trợ công tác quản lý của Cố vấn học tập cũng đã hoàn thành đúng kế hoạch. Hệ thống cung cấp cho giảng viên cố vấn các công cụ đắc lực để giám sát lớp chủ nhiệm, bao gồm khả năng xem chi tiết bảng điểm, theo dõi lịch sử học tập và thực hiện thao tác phê duyệt hoặc từ chối kế hoạch học tập kèm theo các ghi chú phản hồi cụ thể cho sinh viên.

Cuối cùng, về mục tiêu trực quan hóa dữ liệu và báo cáo, hệ thống đã tích hợp thành công thư viện Chart.js để hiển thị biểu đồ tiến độ tích lũy tín chỉ, giúp sinh viên và cố vấn dễ dàng nắm bắt tình hình học tập. Bên cạnh đó, các chức năng xuất báo cáo thống kê và in ấn kế hoạch học tập ra định dạng PDF/Excel cũng hoạt động ổn định, đáp ứng nhu cầu lưu trữ và quản lý hồ sơ.

## Kết luận chương

Chương 4 đã trình bày chi tiết các kết quả thực nghiệm đạt được sau quá trình phân tích, thiết kế và xây dựng hệ thống. Nhóm thực hiện đã triển khai thành công ứng dụng trên môi trường máy chủ cục bộ sử dụng bộ công cụ XAMPP, đảm bảo sự tương thích giữa mã nguồn PHP và cơ sở dữ liệu MySQL.

Các giao diện chức năng dành cho ba nhóm đối tượng chính là Sinh viên, Cố vấn học tập và Quản trị viên đã được hoàn thiện đúng theo thiết kế, đảm bảo tính trực quan và thuận tiện trong thao tác. Kết quả đánh giá hệ thống cho thấy các quy trình nghiệp vụ cốt lõi như lập kế hoạch học tập, kiểm tra điều kiện tiên quyết và xét duyệt trực tuyến đều hoạt động chính xác, đáp ứng tốt các mục tiêu kỹ thuật và yêu cầu thực tiễn đã đề ra ban đầu.

Những kết quả khả quan này là cơ sở quan trọng để khẳng định tính khả thi và hiệu quả của đề tài, tạo tiền đề cho việc tổng kết và đề xuất các hướng phát triển nâng cao trong chương cuối cùng của báo cáo.

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Đề tài **“**Phát triển Hệ thống Quản lý Kế hoạch Học tập và Đăng ký Học phần tại Trường Đại học Trà Vinh**”** đã được thực hiện và hoàn thành các mục tiêu nghiên cứu đề ra ban đầu. Thông qua quá trình nghiên cứu lý thuyết, phân tích nghiệp vụ và hiện thực hóa ứng dụng, đồ án đã đạt được những kết quả quan trọng sau:

**Về mặt lý thuyết và công nghệ:**

Đề tài đã vận dụng thành công quy trình phát triển phần mềm hiện đại, từ khâu khảo sát thực tế, phân tích yêu cầu đến thiết kế cơ sở dữ liệu và giao diện người dùng .

Hệ thống được xây dựng dựa trên kiến trúc MVC giúp mã nguồn rõ ràng, dễ bảo trì; kết hợp với các công nghệ phổ biến như ngôn ngữ PHP, hệ quản trị cơ sở dữ liệu MySQL và thư viện giao diện Bootstrap 5 .

**Về mặt thực tiễn và sản phẩm:**

Đã xây dựng hoàn thiện hệ thống website quản lý tập trung với ba phân hệ người dùng riêng biệt: Quản trị viên (Admin), Sinh viên và Cố vấn học tập, đảm bảo tính bảo mật và phân quyền chặt chẽ.

Giải quyết được bài toán số hóa quy trình lập kế hoạch học tập: Sinh viên có thể chủ động lựa chọn học phần, hệ thống tự động kiểm tra các điều kiện ràng buộc (môn tiên quyết, số tín chỉ) và cảnh báo kịp thời, giúp giảm thiểu sai sót so với phương pháp thủ công.

Cung cấp công cụ đắc lực cho Cố vấn học tập: Giảng viên dễ dàng theo dõi danh sách lớp, xem chi tiết tiến độ tích lũy tín chỉ của sinh viên trực quan hóa bằng biểu đồ Chart.js và thực hiện xét duyệt kế hoạch trực tuyến nhanh chóng.

Hỗ trợ công tác quản lý đào tạo: Hệ thống giúp chuẩn hóa dữ liệu học phần, chương trình đào tạo và cung cấp các báo cáo thống kê cần thiết cho nhà trường .

Nhìn chung, hệ thống đã đáp ứng tốt các yêu cầu chức năng cốt lõi, hoạt động ổn định trên môi trường thử nghiệm và có tính ứng dụng cao trong bối cảnh chuyển đổi số giáo dục.

## Hướng phát triển

Mặc dù hệ thống đã đạt được những kết quả khả quan, tuy nhiên trong phạm vi thời gian và nguồn lực của đồ án thực tập, sản phẩm vẫn còn một số hạn chế nhất định. Để hệ thống có thể triển khai rộng rãi và mang lại hiệu quả cao hơn, đề tài đề xuất các hướng phát triển trong tương lai như sau:

**Triển khai trên môi trường Internet:** Hiện tại, hệ thống đang được vận hành trên môi trường máy chủ cục bộ. Hướng phát triển tiếp theo là đưa hệ thống lên các nền tảng Hosting hoặc Cloud Server để sinh viên và giảng viên có thể truy cập mọi lúc, mọi nơi thông qua Internet, đảm bảo tính tiện lợi tối đa.

**Tích hợp hệ thống thông báo đa kênh:** Bổ sung tính năng gửi thông báo tự động qua Email hoặc SMS/Zalo cho sinh viên khi kế hoạch học tập được duyệt hoặc từ chối, cũng như nhắc nhở lịch đăng ký học phần, giúp tăng cường khả năng tương tác giữa người dùng và hệ thống.

**Kết nối dữ liệu thời gian thực:** Nghiên cứu xây dựng API để đồng bộ dữ liệu tự động với hệ thống Quản lý đào tạo chính thức của Trường Đại học Trà Vinh. Việc này sẽ giúp cập nhật điểm số, lịch học và danh sách sinh viên theo thời gian thực (Real-time) thay vì phải nhập liệu thủ công hoặc Import từ Excel như hiện tại.

**Ứng dụng trí tuệ nhân tạo gợi ý lộ trình:** Tích hợp các thuật toán gợi ý thông minh để đề xuất lộ trình học tập cá nhân hóa cho từng sinh viên dựa trên kết quả học tập quá khứ và định hướng chuyên ngành, giúp sinh viên tối ưu hóa thời gian ra trường.

**Phát triển ứng dụng di động:** Xây dựng phiên bản ứng dụng trên nền tảng Android/iOS, cho phép sinh viên xem thời khóa biểu, theo dõi tiến độ và nhận thông báo ngay trên thiết bị di động.

TÀI LIỆU THAM KHẢO

[1] Mozilla Developer Network, “Web Technology for Developers.” 2024. [Online]. Available: https://developer.mozilla.org/en-US/docs/Web

[2] World Wide Web Consortium, “Architecture of the World Wide Web.” 2004. [Online]. Available: https://www.w3.org/TR/webarch/

[3] IBM, “Client–Server Architecture.” 2023. [Online]. Available: https://www.ibm.com/docs/en

[4] R. W. Sebesta, *Programming the World Wide Web*, 8th ed. Pearson, 2018.

[5] PHP Group, “PHP Manual.” 2024. [Online]. Available: https://www.php.net/manual/en/

[6] Oracle, “MySQL Documentation.” 2024. [Online]. Available: https://dev.mysql.com/doc/