

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH
HỌC KỲ I, NĂM HỌC 2025-2026

XÂY DỰNG WEBSITE CHO TRUNG TÂM
TRÍ TUỆ NHÂN TẠO

Giáo viên hướng dẫn:
ThS. Nguyễn Nhứt Lam

Sinh viên thực hiện:
Họ tên: Hoàng Tuấn Kiệt
MSSV: 110122099
Lớp: DA22TTD

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**

**XÂY DỰNG WEBSITE CHO TRUNG TÂM
TRÍ TUỆ NHÂN TẠO**

Giảng viên hướng dẫn:

ThS. Nguyễn Nhứt Lam

Sinh viên thực hiện:

Họ tên: Hoàng Tuấn Kiệt

MSSV:110122099

Lớp:DA22TTD

Vĩnh Long, tháng 12 năm 2025

[illegible]

(Ký và ghi rõ họ tên)

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.

(Ký và ghi rõ họ tên)

LỜI CẢM ƠN

Trước tiên, em xin cảm ơn sâu sắc đến trường Đại học Trà Vinh đã tạo điều kiện để em thực hiện dự án này, những người đã đưa ra những sự hỗ trợ, góp ý, cũng như là sự giúp đỡ thầy Nguyễn Nhứt Lam – người đóng vai trò quan trọng trong việc phát triển và làm cho dự án thành công tốt đẹp.

Em biết ơn tất cả những người đã dành thời công sức, kiến thức và thời gian quý báu của mình để hỗ trợ em. Điều đó sẽ là nguồn động lực giúp em phát triển bản thân, những ý kiến, lời khuyên đó sẽ góp phần tạo nên một môi trường làm việc tích cực.

Xin chân thành cảm ơn thầy Nguyễn Nhứt Lam và trường Đại học Trà Vinh đã giúp đỡ, do đây là dự án đầu tiên cũng như hạn chế về mặt kiến thức lẫn kinh nghiệm nên chắc chắn sẽ xảy ra những sai sót nhỏ, vậy nên em mong rằng sẽ nhận được những góp ý và lời khuyên, em sẽ đưa những lời khuyên, góp ý đó vào dự án này để tạo nên một thành công tốt đẹp.

Trân trọng

Sinh viên thực hiện

Hoàng Tuấn Kiệt

TÓM TẮT ĐỒ ĐỒ ÁN CHUYÊN NGÀNH

Đề tài "Xây dựng website cho Trung tâm Trí tuệ Nhân tạo" nhằm giải quyết hạn chế trong việc quản lý thủ công quy trình đăng ký khóa học, theo dõi học viên và quảng bá hoạt động của trung tâm. Trước đây, quy trình này được quản lý bằng bảng biểu rời rạc, file Excel hoặc trao đổi qua mạng xã hội, gây khó khăn trong việc tổng hợp thông tin, theo dõi tiến độ đăng ký và đảm bảo tính minh bạch. Vì vậy, hệ thống website được xây dựng giúp số hóa toàn bộ quy trình, tạo môi trường làm việc rõ ràng, đồng bộ và hiệu quả hơn cho học viên, giảng viên và quản trị viên.

Trong quá trình thực hiện đề tài, em tập trung tìm hiểu quy trình nghiệp vụ thực tế tại Trung tâm AI, xác định các nhóm người dùng chính (quản trị viên và học viên) và phân tích yêu cầu của từng đối tượng. Hướng tiếp cận của đề tài là xây dựng một hệ thống quản lý tập trung với giao diện web thân thiện, đồng thời đảm bảo tính phân quyền và bảo mật của từng loại tài khoản.

Để giải quyết bài toán, em thực hiện giải pháp gồm các chức năng chính: quản trị viên quản lý khóa học, đợt học, tin tức, đội ngũ giảng viên, đối tác và thông báo; học viên đăng ký tài khoản, xem thông tin khóa học, đăng ký học theo đợt, theo dõi trạng thái đăng ký và nhận thông báo; hệ thống gửi thông báo tự động khi có cập nhật về khóa học hoặc trạng thái đăng ký; người dùng có thể gửi phản hồi và liên hệ trực tiếp với trung tâm.

Kết quả đạt được của đề tài bao gồm: xây dựng thành công một website có giao diện trực quan với tông màu xanh dương và cam hiện đại, cho phép phân quyền rõ ràng giữa quản trị viên và học viên; triển khai được quy trình từ việc tạo khóa học, mở đợt học đến đăng ký và theo dõi trạng thái; mô-đun thông báo, tin tức và phản hồi hoạt động ổn định; hệ thống đảm bảo tính bảo mật thông tin thông qua mã hóa mật khẩu và xác thực JWT. Hệ thống góp phần giảm tải công việc quản lý, nâng cao hiệu quả quảng bá và tuyển sinh, đồng thời tạo ra môi trường làm việc khoa học, đồng bộ và hiện đại hơn cho Trung tâm Trí tuệ Nhân tạo.

MỞ ĐẦU

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ trong lĩnh vực giáo dục, các hoạt động quản lý đào tạo và tuyển sinh cần được tự động hóa nhằm nâng cao hiệu quả và tính minh bạch. Tại Trung tâm Trí tuệ Nhân tạo, quy trình quản lý khóa học và đăng ký học viên hiện nay vẫn còn được thực hiện chủ yếu theo phương thức thủ công như sử dụng bảng tính Excel, trao đổi qua mạng xã hội hoặc liên hệ trực tiếp. Điều này gây ra nhiều khó khăn như khó thống kê dữ liệu, khó theo dõi số lượng đăng ký, thông tin dễ thất lạc, đặc biệt là không có hệ thống tập trung để quảng bá hoạt động và hỗ trợ học viên đăng ký khóa học một cách thuận tiện. Chính những bất cập này là lý do quan trọng thúc đẩy em chọn đề tài "Xây dựng website cho Trung tâm Trí tuệ Nhân tạo".

Mục tiêu chính của đề tài là tạo ra một hệ thống website giúp giới thiệu thông tin về trung tâm, đội ngũ giảng viên, các khóa học AI và hỗ trợ học viên đăng ký học trực tuyến. Hệ thống giúp tạo môi trường làm việc chuyên nghiệp hơn, giảm gánh nặng quản lý cho quản trị viên và nâng cao trải nghiệm của học viên.

Đối tượng sử dụng hệ thống bao gồm: học viên muốn tìm hiểu và đăng ký khóa học, quản trị viên quản lý nội dung và hoạt động của trung tâm, cùng với khách truy cập muốn tìm hiểu thông tin.

Các tính năng chính của hệ thống gồm: quản trị viên quản lý khóa học, đợt học, tin tức, đội ngũ giảng viên, đối tác và thông báo; học viên đăng ký tài khoản, xem thông tin khóa học chi tiết, đăng ký học theo đợt và theo dõi trạng thái đăng ký; hệ thống gửi thông báo tự động và cho phép người dùng gửi phản hồi, liên hệ với trung tâm.

Phạm vi đề tài tập trung vào: quản lý tài khoản người dùng, quản lý khóa học và đợt học, đăng ký khóa học, quản lý tin tức và thông báo, giới thiệu đội ngũ giảng viên, quản lý đối tác và tiếp nhận phản hồi từ người dùng.

MỤC LỤC

CHƯƠNG 1. TỔNG QUAN NGHIÊN CỨU.....	1
1.1. Giới thiệu đề tài.....	1
1.2. Mục đích nghiên cứu.....	2
1.3. Đối tượng nghiên cứu.....	3
1.4. Phạm vi nghiên cứu.....	3
1.4.1 Công nghệ sử dụng.....	3
1.4.2 Chức năng ứng dụng.....	3
1.4.3 Khả năng nghiên cứu và áp dụng.....	5
1.5. Phương pháp nghiên cứu.....	6
CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT.....	7
2.1. Tổng quan thiết kế web.....	7
2.1.1 Kiến trúc ứng dụng web.....	8
2.1.2 Ngôn ngữ HTML, CSS, Javascript.....	9
2.2. Tìm hiểu về frontend framework.....	10
2.2.1 React.js.....	10
2.2.2 Node.js.....	10
2.2.3 Mongo DB.....	10
2.3. Tìm hiểu về mô hình Client-Server.....	11
2.4. Tìm hiểu về mô hình MVC.....	12
CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU.....	13
3.1. Mô tả bài toán.....	13
3.1.1 Các trang Hệ thống website bao gồm.....	13
3.1.2 Đối tượng sử dụng.....	14

3.1.3 Các nghiệp vụ cần quản lý	15
3.1.4 Các yêu cầu quan trọng	17
3.1.5 Kết quả mong đợi	18
3.2. Phân tích thiết kế hệ thống	19
3.2.1 Đặc tả yêu cầu hệ thống	19
3.2.2 Kiến trúc hệ thống	21
3.2.3 Công nghệ sử dụng	22
3.2.4 Thiết kế dữ liệu	23
3.2.5 Thiết kế xử lý	33
3.2.6 Thiết kế giao diện	35
CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU	37
4.1. Dữ liệu thử nghiệm	37
4.2. Kết quả thực nghiệm	40
4.2.1 Chức năng trang giao diện	40
4.2.2 Chức năng trang quản trị	47
CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	51
5.1. Kết luận	51
5.2. Hạn chế	51
5.3. Hướng phát triển	52
TÀI LIỆU THAM KHẢO	53

DANH MỤC HÌNH ẢNH

Hình 2.3 Mô hình Client-Server	11
Hình 2.4 Mô hình MVC	12
Hình 3.2.4 Mô hình ERD	24
Hình 3.2.6 Sơ đồ giao diện hệ thống	35
Hình 4.1 Collections coursebatches(Đợt học)	37
Hình 4.2 Collections courses(Khóa học)	37
Hình 4.3 Collections freedbacks(Phản hồi)	38
Hình 4.4 Collections news(Tin tức)	38
Hình 4.5 Collections notifications(Thông báo)	39
Hình 4.6 Collections partners(Đối tác)	39
Hình 4.7 Collections registrations(Đăng kí)	39
Hình 4.8 Collections teams (Đội ngũ giảng viên)	40
Hình 4.9 Collections users(Người dùng)	40
Hình 4.10 Giao diện trang chủ	40
Hình 4.11 Giao diện trang giới thiệu	41
Hình 4.12 Giao diện đội ngũ giảng viên	41
Hình 4.13 Giao diện trang khóa học	42
Hình 4.14 Giao diện trang chi tiết khóa học	42
Hình 4.15 Giao diện trang tin tức	43
Hình 4.16 Giao diện trang chi tiết tin tức	43
Hình 4.17 Giao diện trang liên hệ	44
Hình 4.18 Giao diện trang đăng nhập	44
Hình 4.19 Giao diện trang đăng ký	45
Hình 4.20 Giao diện trang cá nhân	45

Hình 4.21 Giao diện trang khóa học của tôi	46
Hình 4.22 Giao diện trang thông báo	46
Hình 4.23 Giao diện trang Dashboard	47
Hình 4.24 Giao diện trang Dashboard	47
Hình 4.25 Giao diện trang quản lý khóa học	48
Hình 4.26 Giao diện trang quản lý đăng ký khóa học	48
Hình 4.27 Giao diện trang quản lý tin tức	49
Hình 4.28 Giao diện trang quản lý phản hồi	49
Hình 4.29 Giao diện trang quản lý đối tác	50
Hình 4.30 Giao diện trang quản lý thông báo	50

DANH MỤC BẢNG BIỂU

Bảng 3.1 Collection Users	25
Bảng 3.2 Collection Courses	26
Bảng 3.3 Collection Registrations	28
Bảng 3.4 Collection News	28
Bảng 3.5 Collection Partners	29
Bảng 3.6 Collection Feedbacks	30
31Bảng 3.7 Collection Notifications	31

CHƯƠNG 1. TỔNG QUAN NGHIÊN CỨU

1.1. Giới thiệu đề tài

Trong bối cảnh công nghệ thông tin và trí tuệ nhân tạo (AI) đang phát triển mạnh mẽ, việc ứng dụng AI vào các lĩnh vực giáo dục, nghiên cứu và đào tạo đã trở thành xu hướng tất yếu. Trường Đại học Trà Vinh, với định hướng phát triển thành một trung tâm đào tạo và nghiên cứu hàng đầu về công nghệ, đã thành lập Trung tâm Trí tuệ Nhân tạo nhằm thúc đẩy nghiên cứu, ứng dụng và đào tạo nguồn nhân lực chất lượng cao trong lĩnh vực AI.

Để nâng cao hiệu quả hoạt động và tăng cường khả năng tiếp cận thông tin của cộng đồng, Trung tâm Trí tuệ Nhân tạo cần có một nền tảng website chuyên nghiệp. Website này sẽ đóng vai trò là cầu nối giữa trung tâm với sinh viên, học viên, các nhà nghiên cứu và doanh nghiệp, giúp giới thiệu các hoạt động, khóa học, dự án nghiên cứu, tin tức và thông tin về đội ngũ giảng viên. Đồng thời, website cũng cung cấp các tính năng tương tác như đăng ký khóa học trực tuyến, gửi phản hồi, liên hệ và tra cứu thông tin, từ đó tạo ra sự minh bạch và thuận tiện trong quá trình truyền thông và quản lý.

Website Trung tâm Trí tuệ Nhân tạo sẽ cung cấp nhiều tính năng quan trọng, bao gồm:

- Trang chủ: Hiển thị tổng quan về trung tâm, hình ảnh nổi bật, tin tức mới nhất và các hoạt động đang diễn ra.
- Giới thiệu: Cung cấp thông tin về mục tiêu, sứ mệnh, tầm nhìn và đội ngũ giảng viên, nghiên cứu viên của trung tâm.
- Khóa học: Danh sách các khóa học về AI, machine learning, deep learning với thông tin chi tiết và chức năng đăng ký trực tuyến.
- Tin tức & Sự kiện: Cập nhật tin tức, sự kiện, hội thảo, workshop liên quan đến AI.
- Liên hệ: Cung cấp thông tin liên hệ và form gửi phản hồi trực tuyến.

- Quản trị (Admin): Hệ thống quản lý nội dung cho phép thêm, sửa, xóa tin tức, khóa học và hình ảnh.

Trang web sẽ được thiết kế với giao diện hiện đại, thân thiện với người dùng và tối ưu hóa cho cả máy tính và thiết bị di động. Các tính năng sẽ được sắp xếp khoa học, rõ ràng và dễ tiếp cận, với menu điều hướng trực quan giúp người dùng dễ dàng tìm thấy thông tin cần thiết. Hệ thống cũng sẽ có các phần hướng dẫn sử dụng chi tiết để người dùng mới có thể làm quen nhanh chóng với các chức năng.

1.2. Mục đích nghiên cứu

Thiết kế và xây dựng website cho Trung tâm Trí tuệ Nhân tạo tại Trường Đại học Trà Vinh nhằm các mục đích sau:

- Tăng cường khả năng truyền thông và tiếp cận: Tạo ra một kênh thông tin chính thức, chuyên nghiệp để giới thiệu về trung tâm, các hoạt động nghiên cứu, đào tạo và dự án đến với cộng đồng sinh viên, học viên và doanh nghiệp.

- Số hóa quy trình đăng ký và quản lý: Thay thế quy trình thủ công trong việc đăng ký khóa học, gửi phản hồi và liên hệ bằng hệ thống trực tuyến, giúp tiết kiệm thời gian và giảm thiểu sai sót.

- Cung cấp thông tin minh bạch và kịp thời: Người dùng có thể dễ dàng tra cứu thông tin về khóa học, dự án, tin tức và sự kiện một cách nhanh chóng và chính xác.

- Hỗ trợ công tác quản lý nội dung: Xây dựng hệ thống quản trị (Admin) cho phép cán bộ quản lý dễ dàng cập nhật, chỉnh sửa nội dung website mà không cần kiến thức lập trình chuyên sâu.

- Nâng cao hình ảnh và uy tín: Một website chuyên nghiệp sẽ góp phần nâng cao hình ảnh của trung tâm, thể hiện sự chuyên nghiệp và hiện đại trong hoạt động đào tạo và nghiên cứu về AI.

- Tạo nền tảng tương tác: Xây dựng môi trường tương tác giữa trung tâm với người học, nhà nghiên cứu và đối tác thông qua các chức năng đăng ký, phản hồi và liên hệ trực tuyến.

1.3. Đối tượng nghiên cứu

- Tìm hiểu nhu cầu và yêu cầu của Trung tâm Trí tuệ Nhân tạo về một hệ thống website quản lý và truyền thông tin.
- Nghiên cứu các quy trình hiện tại trong việc giới thiệu khóa học, quản lý thông tin dự án, cập nhật tin tức và tương tác với người dùng.
- Phân tích các website tương tự của các trung tâm nghiên cứu và đào tạo về AI trong và ngoài nước để rút ra bài học kinh nghiệm.
- Nghiên cứu các công nghệ web hiện đại (ReactJS, Node.js, MongoDB) phù hợp để xây dựng hệ thống.
- Cơ sở dữ liệu lưu trữ thông tin về khóa học, tin tức, người dùng và đội ngũ giảng viên.

1.4. Phạm vi nghiên cứu

1.4.1 Công nghệ sử dụng

- Frontend: Sử dụng ReactJS để thiết kế giao diện người dùng động, hiện đại và tương tác cao. ReactJS giúp xây dựng các component tái sử dụng, tối ưu hiệu suất và trải nghiệm người dùng.
- Backend: Sử dụng Node.js với framework Express để xây dựng RESTful API, xử lý logic phía máy chủ và quản lý các chức năng động của ứng dụng.
- Cơ sở dữ liệu: Sử dụng MongoDB (NoSQL) để thiết kế và quản lý cơ sở dữ liệu, lưu trữ thông tin về khóa học, tin tức, người dùng và các dữ liệu liên quan.
- Công cụ hỗ trợ: Visual Studio Code (IDE), Postman (kiểm thử API), GitHub (quản lý mã nguồn), MongoDB Compass (quản lý database).

1.4.2 Chức năng ứng dụng

Phía người dùng (User):

Trang chủ: Hiển thị tổng quan về trung tâm, hình ảnh banner, tin tức nổi bật và các hoạt động mới nhất.

Trang giới thiệu: Cung cấp thông tin về mục tiêu, sứ mệnh, tầm nhìn, lĩnh vực nghiên cứu và đội ngũ giảng viên, nghiên cứu viên.

Trang khóa học: Hiển thị danh sách các khóa học về AI, machine learning, deep learning, xem thông tin chi tiết từng khóa học (mô tả, thời gian, học phí, đội ngũ giảng viên), Chức năng đăng ký khóa học trực tuyến, Tìm kiếm và lọc khóa học theo tiêu chí.

Trang tin tức & sự kiện: Hiển thị tin tức, bài viết, thông báo về hội thảo, workshop và các sự kiện liên quan đến AI.

Trang liên hệ: Form gửi phản hồi, câu hỏi và thông tin liên hệ của trung tâm.

Trang đội ngũ: Hiển thị thông tin về đội ngũ giảng viên, nghiên cứu viên của trung tâm với hình ảnh và mô tả chi tiết.

Trang khóa học của tôi: Cho phép người dùng đã đăng nhập xem danh sách các khóa học đã đăng ký và trạng thái đăng ký.

Chức năng đăng ký/đăng nhập: Hệ thống tài khoản cho phép người dùng đăng ký, đăng nhập để sử dụng các tính năng cá nhân hóa.

Chế độ sáng/tối (Dark/Light mode): Cho phép người dùng chuyển đổi giao diện theo sở thích cá nhân.

Phía quản trị (Admin):

Quản lý khóa học: Thêm, sửa, xóa thông tin khóa học.

Quản lý tin tức: Thêm, sửa, xóa bài viết tin tức và sự kiện.

Quản lý đăng ký: Xem danh sách người đăng ký khóa học và phản hồi từ người dùng.

Xác thực và phân quyền: Hệ thống đăng nhập, xác thực người dùng và phân quyền quản trị.

Quản lý người dùng: Xem danh sách, thêm, sửa, xóa tài khoản người dùng và phân quyền (user/admin).

Quản lý đăng ký khóa học: Xem danh sách đăng ký, duyệt/từ chối đăng ký, cập nhật trạng thái khóa học.

Quản lý phản hồi: Xem và quản lý các phản hồi, góp ý từ người dùng gửi qua form liên hệ.

Dashboard thống kê: Trang tổng quan hiển thị số liệu thống kê về khóa học, người dùng, đăng ký và phản hồi.

1.4.3 Khả năng nghiên cứu và áp dụng

- Phân tích công nghệ: Nghiên cứu cách các công nghệ ReactJS, Node.js, Express và MongoDB hoạt động và tương tác để tạo ra một ứng dụng web full-stack hoàn chỉnh.

- Áp dụng mô hình Agile: Chia dự án thành các giai đoạn (Sprint) để phát triển từng phần chức năng, dễ dàng kiểm soát tiến độ và điều chỉnh khi cần thiết.

- Tích hợp RESTful API: Xây dựng và tích hợp API để kết nối giữa frontend và backend, đảm bảo tính mở rộng và bảo trì trong tương lai.

- Tối ưu trải nghiệm người dùng: Thiết kế responsive, tương thích với nhiều thiết bị và trình duyệt khác nhau.

- Bảo mật cơ bản: Áp dụng các biện pháp bảo mật như mã hóa mật khẩu, xác thực JWT (JSON Web Token), và phân quyền người dùng.

Giới hạn nghiên cứu:

- Tập trung vào việc phát triển các chức năng cốt lõi của website, chưa mở rộng đến các tính năng nâng cao như chatbot AI, hệ thống gợi ý khóa học thông minh, hoặc tích hợp thanh toán trực tuyến.

- Chưa triển khai các công nghệ tiên tiến như microservices, containerization (Docker), hoặc CI/CD pipeline.

Yếu tố hỗ trợ thành công:

- Sự hỗ trợ của trung tâm và nhà trường: Sự quan tâm, cung cấp thông tin và phản hồi từ Trung tâm Trí tuệ Nhân tạo là rất quan trọng để dự án được triển khai thành công.

- Sự tham gia của người dùng: Việc thu thập ý kiến đóng góp từ sinh viên, giảng viên và người dùng tiềm năng sẽ giúp ứng dụng trở nên hữu ích và thực tế hơn.

- Khả năng mở rộng: Website được thiết kế với kiến trúc linh hoạt, dễ dàng mở rộng thêm chức năng trong tương lai khi có nhu cầu.

- Khả năng thích ứng: Ứng dụng cần có khả năng thích ứng với những thay đổi của môi trường công nghệ và yêu cầu của người dùng.

1.5. Phương pháp nghiên cứu

Phương pháp nghiên cứu lý thuyết: Tìm hiểu các công nghệ ReactJS, Node.js, Express và MongoDB để xây dựng ứng dụng web. Nghiên cứu về kiến trúc ứng dụng web, RESTful API, xác thực người dùng và các nguyên tắc thiết kế giao diện. Tham khảo tài liệu, sách và các website mẫu về hệ thống quản lý nội dung.

Phương pháp nghiên cứu thực nghiệm: Thiết kế, cài đặt và triển khai website cho Trung tâm Trí tuệ Nhân tạo tại Trường Đại học Trà Vinh. Quy trình thực hiện gồm các bước: phân tích yêu cầu của người dùng, thiết kế cơ sở dữ liệu và giao diện, phát triển backend và frontend, tích hợp hệ thống, kiểm thử trên nhiều thiết bị, và triển khai ứng dụng thực tế. Sử dụng Git/GitHub để quản lý mã nguồn và áp dụng mô hình Agile để phát triển từng giai đoạn.

CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT

2.1. Tổng quan thiết kế web

Thiết kế web ngày nay không chỉ đơn giản là xây dựng giao diện mà còn bao gồm nhiều yếu tố quan trọng như hiệu suất, bảo mật, khả năng mở rộng và tương thích với các thiết bị khác nhau. Mục tiêu là tạo ra một trải nghiệm người dùng tối ưu và một hệ thống hoạt động ổn định, dễ dàng bảo trì và nâng cấp.

Các công nghệ chính sử dụng trong thiết kế web hiện đại bao gồm:

Frontend: Đây là phần mà người dùng trực tiếp tương tác, bao gồm giao diện người dùng (UI), đồ họa và các chức năng động của trang web. Dự án sử dụng React.js - một thư viện JavaScript mạnh mẽ do Facebook phát triển, cho phép xây dựng giao diện người dùng theo hướng component-based, giúp tái sử dụng code và quản lý state hiệu quả. Vite được sử dụng làm build tool, mang lại tốc độ khởi động và hot reload nhanh chóng trong quá trình phát triển. React Router đảm nhận việc điều hướng giữa các trang, tạo trải nghiệm Single Page Application (SPA) mượt mà. Axios được sử dụng để gọi API và giao tiếp với backend. CSS modules được áp dụng để định dạng giao diện, đồng thời hỗ trợ thiết kế đáp ứng (responsive) và chế độ sáng/tối (dark mode) để đảm bảo trang web hoạt động tốt trên mọi thiết bị, từ máy tính để bàn đến điện thoại di động.

Backend: Chịu trách nhiệm xử lý các yêu cầu từ người dùng, quản lý cơ sở dữ liệu và kết nối với các hệ thống khác. Dự án sử dụng Node.js kết hợp với Express.js - một framework nhẹ và linh hoạt cho phép xây dựng RESTful API nhanh chóng. JWT (JSON Web Token) được sử dụng để xác thực và phân quyền người dùng, đảm bảo bảo mật cho hệ thống. Bcrypt được áp dụng để mã hóa mật khẩu trước khi lưu vào cơ sở dữ liệu. Multer kết hợp với Cloudinary xử lý việc upload và lưu trữ hình ảnh trên đám mây. Nodemailer hỗ trợ gửi email thông báo đến người dùng.

Cơ sở dữ liệu: Đây là nơi lưu trữ tất cả dữ liệu của hệ thống, bao gồm thông tin người dùng, khóa học, đăng ký, tin tức và các dữ liệu liên quan. MongoDB - một cơ sở dữ liệu NoSQL được sử dụng trong dự án này, mang lại sự linh hoạt trong việc lưu trữ dữ liệu dạng document với schema động. Mongoose ODM đóng vai trò là lớp trung gian giúp định nghĩa schema, validate dữ liệu và thực hiện các truy vấn một cách dễ dàng.

Sự kết hợp chặt chẽ giữa frontend và backend theo kiến trúc client-server tạo ra một hệ thống quản lý thông tin đồng bộ, hiệu quả và dễ sử dụng. Việc tách biệt frontend và backend cho phép phát triển song song, dễ dàng bảo trì và mở rộng. React.js đóng vai trò quan trọng trong việc giúp giao diện trở nên không chỉ đẹp mắt mà còn thân thiện với người dùng trên mọi thiết bị, trong khi Node.js và MongoDB đảm bảo hiệu suất và khả năng mở rộng của hệ thống.

2.1.1 Kiến trúc ứng dụng web

Client

Là giao diện người dùng, nơi người dùng sẽ tương tác trực tiếp với ứng dụng. Giao diện này được xây dựng bằng React.js kết hợp với CSS, có chức năng hiển thị dữ liệu từ phía máy chủ thông qua các API call sử dụng Axios và nhận yêu cầu tương tác từ người dùng. React.js cho phép xây dựng giao diện theo hướng component-based, giúp tái sử dụng code hiệu quả và quản lý state thông qua Context API. Vite được sử dụng làm build tool, mang lại tốc độ phát triển nhanh chóng với tính năng Hot Module Replacement (HMR). React Router đảm nhận việc điều hướng giữa các trang, tạo trải nghiệm Single Page Application (SPA) mượt mà mà không cần tải lại trang. Giao diện hỗ trợ thiết kế responsive để tương thích với nhiều kích thước màn hình và tích hợp chế độ sáng/tối (dark mode) để nâng cao trải nghiệm người dùng.

Server

Là máy chủ, nơi xử lý tính năng logic và quản lý dữ liệu. Node.js kết hợp với Express.js được sử dụng để xây dựng các API xử lý các yêu cầu từ người dùng, kết nối với cơ sở dữ liệu MongoDB và trả kết quả về giao diện người dùng thông qua RESTful API.

Cấu trúc:

Client (Frontend):

React.js: Được sử dụng để xây dựng giao diện người dùng theo hướng component-based, bao gồm các thành phần như danh sách khóa học, chi tiết khóa học, form đăng ký, bộ lọc tìm kiếm và trang quản trị.

CSS: Dùng để trang trí giao diện, tạo bố cục và thiết kế các phần tử sao cho hợp lý và đẹp mắt. Đảm bảo giao diện thân thiện với người dùng và hỗ trợ responsive để có thể sử dụng trên các thiết bị khác nhau (máy tính, điện thoại, tablet). Tích hợp chế độ sáng/tối (dark mode) để nâng cao trải nghiệm.

JavaScript (React): Tạo các tương tác động với người dùng thông qua Single Page Application (SPA), tự động hiển thị kết quả mà không phải tải lại trang, quản lý state với Context API và điều hướng mượt mà với React Router.

Server (Backend):

Node.js + Express.js: Xử lý các yêu cầu từ client thông qua RESTful API. Khi người dùng gửi yêu cầu (ví dụ: xem danh sách khóa học, đăng ký khóa học, đăng nhập), Express.js sẽ nhận yêu cầu đó, thực hiện logic xử lý, xác thực JWT Token và truy xuất dữ liệu từ cơ sở dữ liệu.

MongoDB + Mongoose: Hệ quản trị cơ sở dữ liệu NoSQL lưu trữ thông tin người dùng, khóa học, đăng ký, tin tức, phản hồi và các thông tin liên quan khác. Mongoose ODM đóng vai trò định nghĩa schema, validate dữ liệu và thực hiện các truy vấn để lấy dữ liệu và gửi trả kết quả về client dưới dạng JSON.

2.1.2 Ngôn ngữ HTML, CSS, Javascript

HTML

HTML (Hypertext Markup Language) là ngôn ngữ đánh dấu văn bản, giúp xây dựng cấu trúc cơ bản của trang web. Các thành phần cơ bản của HTML bao gồm tiêu đề, đoạn văn, liên kết, hình ảnh, danh sách và các biểu mẫu.

Cách hoạt động của HTML:

HTML hoạt động bằng cách sử dụng các thẻ để bao quanh các phần tử nội dung. Các thẻ này cho trình duyệt biết cách hiển thị nội dung. Trong dự án React, HTML được sử dụng thông qua cú pháp JSX và file index.html làm điểm khởi đầu cho ứng dụng Single Page Application.

CSS

CSS (Cascading Style Sheets) là ngôn ngữ giúp định kiểu và làm đẹp cho các phần tử HTML. CSS có thể thay đổi màu sắc, kiểu chữ, kích thước, bố cục, hiệu ứng động và tương tác với các thành phần khác của trang web.

Cách hoạt động của CSS:

CSS hoạt động bằng cách áp dụng các thuộc tính (properties) cho các phần tử HTML được chọn. CSS sử dụng các bộ chọn (selectors) để xác định phần tử nào cần định dạng và các thuộc tính để chỉ định kiểu định dạng đó.

JavaScript

JavaScript là ngôn ngữ lập trình kịch bản, giúp tạo các tương tác động và xử lý logic trên trang web. Các thành phần cơ bản của JavaScript bao gồm biến, hàm, điều kiện, vòng lặp, sự kiện và các thao tác với DOM.

Cách hoạt động của JavaScript:

JavaScript hoạt động bằng cách thực thi code trực tiếp trên trình duyệt, cho phép thay đổi nội dung trang web mà không cần tải lại. Trong dự án này, JavaScript được sử dụng cả phía client (React.js) và phía server (Node.js).

2.2. Tìm hiểu về frontend framework

2.2.1 React.js

React.js là thư viện JavaScript do Facebook phát triển, cho phép xây dựng giao diện người dùng theo hướng component-based. React sử dụng Virtual DOM để tối ưu hiệu suất render và JSX để viết cấu trúc HTML trong JavaScript.

Cách hoạt động của React.js:

React hoạt động bằng cách chia giao diện thành các component độc lập, có thể tái sử dụng. Mỗi component quản lý state riêng và render lại khi state thay đổi. React Router được sử dụng để điều hướng giữa các trang, tạo trải nghiệm Single Page Application (SPA) mượt mà.

2.2.2 Node.js

Node.js là môi trường runtime cho phép chạy JavaScript phía server, giúp xây dựng các ứng dụng backend nhanh và hiệu quả. Node.js sử dụng mô hình non-blocking I/O và event-driven, cho phép xử lý nhiều request đồng thời mà không bị chặn, từ đó nâng cao hiệu suất và khả năng mở rộng của hệ thống. Cơ chế này đặc biệt phù hợp với các ứng dụng thời gian thực, ứng dụng mạng và các hệ thống cần xử lý lượng lớn kết nối đồng thời. Cách hoạt động của Node.js:

Node.js hoạt động dựa trên cơ chế lắng nghe các request từ client, xử lý các logic nghiệp vụ và trả về response một cách nhanh chóng nhờ kiến trúc non-blocking, bất đồng bộ. Trong dự án này, Node.js được kết hợp với Express.js – một framework nhẹ, linh hoạt và mạnh mẽ – để xây dựng các RESTful API, xử lý xác thực bằng JWT, quản lý các route, middleware và kết nối hiệu quả với cơ sở dữ liệu. Sự kết hợp này giúp tăng tốc độ xử lý request, đơn giản hóa việc phát triển backend và đảm bảo tính mở rộng, bảo mật cho ứng dụng.

2.2.3 Mongo DB

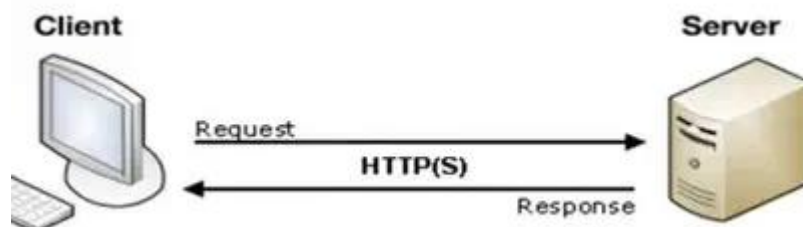
MongoDB là hệ quản trị cơ sở dữ liệu NoSQL, lưu trữ dữ liệu dưới dạng document JSON (BSON) linh hoạt. MongoDB cho phép mở rộng dễ dàng theo chiều ngang, hỗ trợ xử lý dữ liệu lớn, truy vấn dữ liệu nhanh chóng và hiệu quả. Ngoài ra, hệ thống không yêu cầu schema cứng như các cơ sở dữ liệu quan hệ truyền thống, giúp nhà phát triển dễ dàng thay đổi cấu trúc dữ liệu, tăng tính linh hoạt và rút ngắn thời gian phát triển ứng dụng.

Cách hoạt động của MongoDB:

MongoDB hoạt động bằng cách lưu trữ dữ liệu trong các collection, với mỗi document có cấu trúc linh hoạt và có thể khác nhau, cho phép ứng dụng dễ dàng mở rộng và thích ứng với các thay đổi về dữ liệu. Trong dự án này, Mongoose ODM được sử dụng để định nghĩa schema, giúp quy định rõ ràng cấu trúc dữ liệu, đồng thời thực hiện việc validate dữ liệu trước khi lưu vào cơ sở dữ liệu. Ngoài ra, Mongoose hỗ trợ các thao tác CRUD (Create, Read, Update, Delete) một cách thuận tiện, giúp tương tác với MongoDB dễ dàng hơn. Việc sử dụng Mongoose không chỉ tăng tính an toàn cho dữ liệu, hạn chế lỗi trong quá trình thao tác mà còn giúp việc quản lý và bảo trì cơ sở dữ liệu trở nên hiệu quả và đáng tin cậy hơn.

2.3. Tìm hiểu về mô hình Client-Server

Mô hình Client-Server là kiến trúc trong đó ứng dụng được phân tách thành hai phần riêng biệt: Client (Frontend) và Server (Backend). Client chịu trách nhiệm hiển thị giao diện, tiếp nhận và xử lý các tương tác từ người dùng, trong khi Server đảm nhiệm việc xử lý logic nghiệp vụ, xác thực, cũng như lưu trữ và quản lý dữ liệu trong cơ sở dữ liệu. Việc tách biệt này giúp hệ thống dễ bảo trì, mở rộng và phát triển độc lập từng phần.



Hình 2.3 Mô hình Client-Server

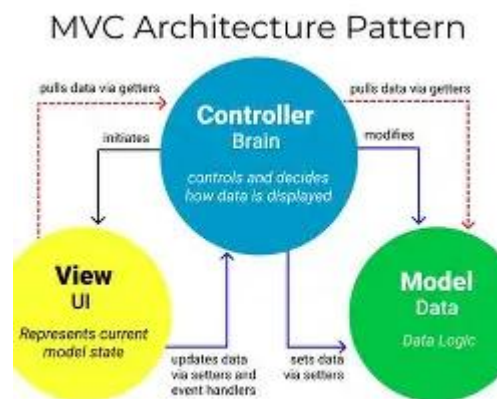
Cách hoạt động của mô hình Client-Server:

Client gửi các request đến Server thông qua giao thức HTTP/HTTPS. Server tiếp nhận request, xử lý logic cần thiết, truy vấn hoặc cập nhật dữ liệu trong cơ sở dữ liệu khi cần, sau đó trả về response cho Client, thường ở định dạng JSON. Trong dự án này, React.js đóng vai trò là Client và giao tiếp với Node.js ở phía Server thông qua các RESTful API bằng thư viện Axios. Cách tiếp cận này giúp đảm bảo sự tách biệt rõ ràng giữa giao diện người dùng và logic xử lý phía server, đồng thời tăng tính linh hoạt, bảo mật và khả năng mở rộng của hệ thống.

2.4. Tìm hiểu về mô hình MVC

MVC (Model-View-Controller) là mô hình kiến trúc phần mềm phân tách ứng dụng thành ba thành phần chính: Model chịu trách nhiệm quản lý dữ liệu và logic nghiệp vụ, View chịu trách nhiệm hiển thị giao diện người dùng, Controller đóng vai trò trung gian xử lý request và điều phối giữa Model và View.

MVC (Model–View–Controller) là mô hình kiến trúc phần mềm phân tách ứng dụng thành ba thành phần chính nhằm tăng tính rõ ràng và dễ bảo trì. Model chịu trách nhiệm quản lý dữ liệu, xử lý logic nghiệp vụ và tương tác với cơ sở dữ liệu. View đảm nhiệm việc hiển thị giao diện người dùng, trình bày dữ liệu một cách trực quan và tiếp nhận các thao tác từ người dùng. Controller đóng vai trò trung gian, tiếp nhận request từ View, xử lý yêu cầu, gọi các chức năng phù hợp trong Model và trả kết quả về cho View. Việc áp dụng mô hình MVC giúp tách biệt dữ liệu, giao diện và luồng xử lý, từ đó giảm sự phụ thuộc giữa các thành phần, hỗ trợ phát triển, mở rộng và bảo trì ứng dụng hiệu quả hơn.



Hình 2.4 Mô hình MVC

Cách hoạt động của mô hình MVC trong dự án:

Trong dự án này, mô hình MVC được triển khai một cách rõ ràng và nhất quán. Model được định nghĩa trong thư mục models/, sử dụng Mongoose schema để mô tả cấu trúc dữ liệu và tương tác trực tiếp với cơ sở dữ liệu MongoDB. Controller nằm trong thư mục controllers/, chịu trách nhiệm xử lý các logic nghiệp vụ như xác thực người dùng, thực hiện các thao tác CRUD với dữ liệu và trả về response cho client. View được xây dựng bằng React.js ở phía Frontend, đảm nhiệm việc hiển thị giao diện và dữ liệu nhận được từ các API. Bên cạnh đó, Routes trong thư mục routes/ có vai trò định tuyến các request HTTP đến các Controller tương ứng. Cách tổ chức theo mô hình MVC này giúp tách biệt rõ ràng giữa các thành phần, từ đó nâng cao tính dễ hiểu, dễ bảo trì và khả năng mở rộng của hệ thống.

CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU

3.1. Mô tả bài toán

Trong bối cảnh Trí tuệ Nhân tạo (AI) đang phát triển mạnh mẽ và được ứng dụng rộng rãi trong nhiều lĩnh vực của đời sống, nhu cầu tìm hiểu và học tập về AI ngày càng tăng cao. Các trung tâm đào tạo và nghiên cứu AI cần có một nền tảng trực tuyến chuyên nghiệp để giới thiệu thông tin về trung tâm, quảng bá các khóa học, cập nhật tin tức trong lĩnh vực AI, đồng thời quản lý việc đăng ký khóa học và tương tác với học viên một cách hiệu quả.

Xuất phát từ nhu cầu thực tế đó, đề tài "Xây dựng Website cho Trung tâm Trí tuệ Nhân tạo" được thực hiện nhằm phát triển một hệ thống web hoàn chỉnh phục vụ hoạt động của trung tâm. Website cho phép khách truy cập xem thông tin giới thiệu về trung tâm, đội ngũ giảng viên, các đối tác hợp tác, danh sách các khóa học AI đang mở cùng chi tiết nội dung từng khóa học, và các tin tức, bài viết liên quan đến lĩnh vực AI. Học viên có thể đăng ký tài khoản, đăng nhập hệ thống, đăng ký tham gia các khóa học và gửi phản hồi đánh giá. Về phía quản trị, hệ thống cung cấp trang Dashboard cho phép quản trị viên quản lý khóa học, tin tức, người dùng, đăng ký khóa học và phản hồi từ học viên.

Hệ thống được xây dựng theo mô hình Client-Server với Frontend sử dụng React.js kết hợp Vite, Backend sử dụng Node.js và Express.js, cơ sở dữ liệu MongoDB, xác thực người dùng bằng JWT, và hỗ trợ upload file thông qua Multer. Website được thiết kế responsive, tương thích trên nhiều thiết bị và hỗ trợ chế độ giao diện sáng/tối để nâng cao trải nghiệm người dùng.

3.1.1 Các trang Hệ thống website bao gồm

Trang công khai

Trang chủ: Hiển thị banner, giới thiệu tổng quan, khóa học nổi bật và tin tức mới nhất.

Giới thiệu: Thông tin về trung tâm, sứ mệnh, tầm nhìn.

Đội ngũ: Giới thiệu giảng viên, nhân sự của trung tâm.

Danh sách khóa học: Hiển thị tất cả khóa học với bộ lọc, tìm kiếm.

Chi tiết khóa học: Thông tin chi tiết khóa học, nội dung, đăng ký.

Danh sách tin tức: Hiển thị các bài viết, tin tức của trung tâm.

Chi tiết tin tức: Nội dung chi tiết bài viết.

Liên hệ: Form liên hệ, thông tin địa chỉ, bản đồ.

Chính sách bảo mật: Điều khoản về bảo mật thông tin.

Điều khoản dịch vụ: Điều khoản sử dụng dịch vụ.

Trang xác thực

Đăng nhập: Form đăng nhập tài khoản.

Đăng ký: Form đăng ký tài khoản mới.

Trang người dùng

Khóa học của tôi: Danh sách khóa học đã đăng ký.

Cảm ơn: Trang xác nhận sau khi đăng ký thành công.

Trang quản trị

Dashboard: Tổng quan thống kê, biểu đồ.

Quản lý khóa học: Thêm, sửa, xóa khóa học.

Quản lý tin tức: Thêm, sửa, xóa bài viết.

Quản lý người dùng: Quản lý tài khoản người dùng.

Quản lý đăng ký khóa học: Duyệt, quản lý đăng ký học viên.

Quản lý phản hồi: Xem và quản lý feedback.

Quản lý đối tác: Thêm, sửa, xóa thông tin đối tác.

Trang lỗi

Không tìm thấy (404): Hiện thị khi truy cập đường dẫn không tồn tại.

3.1.2 Đối tượng sử dụng

Khách truy cập (Guest)

Người dùng chưa đăng nhập vào hệ thống

Quyền hạn: (Xem trang chủ, giới thiệu, đội ngũ, danh sách và chi tiết khóa học, danh sách và chi tiết tin tức, thông tin liên hệ, đăng ký tài khoản mới, đăng nhập vào hệ thống).

Người dùng đã đăng ký (User)

Người dùng đã có tài khoản và đăng nhập vào hệ thống

Quyền hạn: (Tất cả quyền của khách truy cập, Đăng ký khóa học, Xem danh sách khóa học đã đăng ký (Khóa học của tôi), Gửi phản hồi và đánh giá, Quản lý thông tin cá nhân).

Quản trị viên (Admin)

Người quản lý hệ thống với quyền cao nhất

Quyền hạn:(Tất cả quyền của người dùng đã đăng ký,Truy cập trang quản trị (Dashboard),Quản lý khóa học (thêm, sửa, xóa),Quản lý tin tức (thêm, sửa, xóa),Quản lý người dùng (xem, sửa, xóa tài khoản),Quản lý đăng ký khóa học (duyet, từ chối, xóa),Quản lý phản hồi từ người dùng,Quản lý đối tác (thêm, sửa, xóa),Xem thống kê, báo cáo tổng quan).

3.1.3 Các nghiệp vụ cần quản lý

Quản lý khóa học

- Thêm mới khóa học
- Cập nhật thông tin khóa học
- Xóa khóa học
- Kích hoạt/Vô hiệu hóa khóa học

Thông tin quản lý:

- Tiêu đề, mô tả, nội dung chi tiết
- Hình ảnh khóa học
- Thời lượng, cấp độ (cơ bản, trung cấp, nâng cao)
- Giá gốc, giá khuyến mãi
- Giảng viên, danh mục
- Mục tiêu khóa học, đối tượng học viên
- Chương trình học (theo tuần)
- Lịch học (ngày bắt đầu, thời gian, địa điểm)
- Số lượng học viên tối đa
- Ưu đãi đi kèm

Quản lý tin tức

- Thêm mới bài viết
- Cập nhật bài viết
- Xóa bài viết
- Xuất bản/Ẩn bài viết

Thông tin quản lý:

- Tiêu đề, tóm tắt, nội dung
- Nội dung chi tiết (HTML)

- Hình ảnh đại diện
- Tác giả, danh mục
- Tags (từ khóa)
- Lượt xem
- Trạng thái xuất bản

Quản lý người dùng

- Xem danh sách người dùng
- Cập nhật thông tin người dùng
- Xóa tài khoản người dùng
- Phân quyền (user/admin)

Thông tin quản lý:

- Họ tên, email, số điện thoại
- Mật khẩu (mã hóa)
- Vai trò (user, admin)
- Ảnh đại diện

Quản lý đăng ký khóa học

- Xem danh sách đăng ký
- Duyệt đăng ký
- Từ chối đăng ký
- Xóa đăng ký
- Cập nhật trạng thái thanh toán

Thông tin quản lý:

- Người đăng ký (liên kết User)
- Khóa học đăng ký (liên kết Course)
- Trạng thái đăng ký (chờ duyệt, đã duyệt, từ chối)
- Thời gian đăng ký

Quản lý phản hồi

- Xem danh sách phản hồi
- Đánh dấu đã đọc

- Đánh dấu đã trả lời
- Xóa phản hồi

Thông tin quản lý:

- Họ tên, email, số điện thoại người gửi
- Chủ đề, nội dung phản hồi
- Trạng thái (chờ xử lý, đã đọc, đã trả lời)
- Thời gian gửi

Quản lý đối tác

- Thêm mới đối tác
- Cập nhật thông tin đối tác
- Xóa đối tác
- Kích hoạt/Vô hiệu hóa đối tác

Thông tin quản lý:

- Tên đối tác
- Logo
- Mô tả
- Trạng thái hoạt động

3.1.4 Các yêu cầu quan trọng

Hệ thống website cần đáp ứng các yêu cầu chức năng, phi chức năng và kỹ thuật để đảm bảo hoạt động hiệu quả và phục vụ tốt người dùng.

Về yêu cầu chức năng, đối với khách truy cập, hệ thống cho phép xem thông tin giới thiệu, đội ngũ giảng viên, danh sách khóa học với bộ lọc theo danh mục và cấp độ, cũng như chi tiết từng khóa học bao gồm nội dung, lịch học và học phí. Khách cũng có thể xem tin tức, gửi thông tin liên hệ, đăng ký tài khoản và đăng nhập hệ thống. Đối với người dùng đã đăng ký, họ có thể đăng ký tham gia khóa học, xem danh sách khóa học đã đăng ký, theo dõi trạng thái đăng ký và quản lý thông tin cá nhân. Đối với quản trị viên, hệ thống cung cấp đầy đủ chức năng quản lý CRUD cho khóa học, tin tức, người dùng và đối tác, cùng với khả năng duyệt đăng ký, cập nhật trạng thái thanh toán, xử lý phản hồi và xem thống kê tổng quan.

Về yêu cầu phi chức năng, hệ thống đảm bảo bảo mật thông qua mã hóa mật khẩu bằng bcrypt, xác thực JWT Token, phân quyền truy cập theo vai trò và bảo vệ các route admin khỏi truy cập trái phép. Về hiệu năng, website cần tải trang nhanh, tối ưu hình ảnh, hỗ trợ phân trang cho danh sách dữ liệu lớn và tìm kiếm, lọc dữ liệu

hiệu quả. Giao diện được thiết kế responsive, tương thích đa thiết bị, thân thiện với người dùng và hỗ trợ chế độ sáng/tối. Hệ thống cũng đảm bảo khả năng mở rộng với kiến trúc module hóa, tách biệt Frontend và Backend theo mô hình RESTful API, sử dụng cơ sở dữ liệu MongoDB linh hoạt.

Về yêu cầu kỹ thuật, Frontend được xây dựng bằng React.js với Vite, sử dụng React Router cho điều hướng, Context API cho quản lý state, Axios cho gọi API và CSS modules cho styling. Backend sử dụng Node.js với Express.js, MongoDB với Mongoose ODM, JWT cho xác thực, Bcrypt cho mã hóa mật khẩu và Multer cho upload file. Hệ thống hỗ trợ biến môi trường thông qua file .env, cấu hình CORS cho cross-origin requests và tuân theo cấu trúc thư mục rõ ràng theo mô hình MVC.

3.1.5 Kết quả mong đợi

Sau khi hoàn thành, hệ thống website sẽ mang lại những giá trị thiết thực cho các bên liên quan.

Đối với người dùng cuối, họ có thể dễ dàng tìm kiếm và xem thông tin các khóa học phù hợp với nhu cầu, đăng ký khóa học một cách nhanh chóng và thuận tiện, đồng thời theo dõi được trạng thái đăng ký cũng như thanh toán của mình. Người dùng cũng được cập nhật tin tức, sự kiện mới nhất của trung tâm và có thể liên hệ, gửi phản hồi dễ dàng thông qua giao diện thân thiện, hoạt động mượt mà trên mọi thiết bị.

Đối với quản trị viên, hệ thống cung cấp công cụ quản lý khóa học hiệu quả với đầy đủ thông tin chi tiết, cho phép duyệt và xử lý đăng ký học viên nhanh chóng. Việc đăng tải và quản lý tin tức, bài viết trở nên dễ dàng hơn, cùng với khả năng quản lý thông tin người dùng và phân quyền rõ ràng. Dashboard thống kê tổng quan giúp đưa ra quyết định kịp thời, đồng thời tiếp nhận và xử lý phản hồi từ khách hàng một cách hiệu quả.

Đối với trung tâm đào tạo, website giúp tăng khả năng tiếp cận học viên tiềm năng, tự động hóa quy trình đăng ký khóa học, từ đó giảm thời gian và công sức quản lý thủ công. Hệ thống góp phần xây dựng hình ảnh chuyên nghiệp cho trung tâm, quảng bá khóa học và tin tức hiệu quả, đồng thời hỗ trợ quản lý đối tác và mở rộng hợp tác.

Về mặt kỹ thuật, hệ thống đảm bảo hoạt động ổn định và bảo mật cao, tốc độ tải trang nhanh với hiệu năng được tối ưu. Kiến trúc code sạch, có cấu trúc rõ ràng theo chuẩn MVC giúp dễ dàng bảo trì và mở rộng tính năng trong tương lai. Dữ liệu được lưu trữ an toàn với khả năng backup, tương thích đa trình duyệt và thiết bị.

3.2. Phân tích thiết kế hệ thống

3.2.1 Đặc tả yêu cầu hệ thống

Yêu cầu chức năng

Quản lý người dùng

Hệ thống cho phép quản trị viên thêm mới, chỉnh sửa và xóa tài khoản người dùng với 2 vai trò: Admin và Người dùng. Mỗi người dùng phải đăng nhập bằng email và mật khẩu để truy cập vào các chức năng tương ứng với quyền hạn của mình. Hệ thống sử dụng JWT Token để xác thực và kiểm tra phân quyền trước mỗi thao tác quan trọng để đảm bảo bảo mật. Mật khẩu được mã hóa bằng bcrypt trước khi lưu vào cơ sở dữ liệu.

Quản lý khóa học

Admin có thể tạo mới các khóa học với đầy đủ thông tin bao gồm tiêu đề, mô tả, nội dung chi tiết, hình ảnh, thời lượng, cấp độ và học phí. Mỗi khóa học có thông tin về mục tiêu, đối tượng học viên, chương trình học theo tuần, lịch học và số lượng học viên tối đa. Admin có quyền chỉnh sửa, xóa hoặc kích hoạt/vô hiệu hóa các khóa học. Chỉ những khóa học đang hoạt động mới hiển thị cho người dùng trên trang công khai.

Quản lý đăng ký khóa học

Người dùng đã đăng nhập có thể xem danh sách các khóa học đang mở và thực hiện đăng ký. Mỗi đăng ký có trạng thái chờ duyệt, đã duyệt hoặc từ chối, cùng với trạng thái thanh toán chưa thanh toán hoặc đã thanh toán. Admin xem danh sách tất cả đăng ký và quyết định duyệt hoặc từ chối. Người dùng có thể theo dõi trạng thái đăng ký của mình trong mục "Khóa học của tôi". Hệ thống lưu trữ toàn bộ lịch sử đăng ký để dễ dàng truy xuất và quản lý.

Quản lý tin tức

Admin có thể tạo mới các bài viết, tin tức với tiêu đề, tóm tắt, nội dung chi tiết hỗ trợ định dạng HTML, hình ảnh đại diện, danh mục và tags. Hệ thống tự động đếm lượt xem cho mỗi bài viết. Admin có quyền chỉnh sửa, xóa hoặc xuất bản/ẩn bài viết. Chỉ những bài viết đã xuất bản mới hiển thị trên trang công khai. Người dùng có thể xem danh sách tin tức và đọc nội dung chi tiết từng bài.

Quản lý phản hồi

Khách truy cập và người dùng có thể gửi phản hồi, liên hệ thông qua form với thông tin họ tên, email, số điện thoại, chủ đề và nội dung. Mỗi phản hồi có trạng thái chờ xử lý, đã đọc hoặc đã trả lời. Admin xem danh sách tất cả phản hồi và cập nhật trạng thái xử lý. Hệ thống lưu trữ thời gian gửi để theo dõi và xử lý kịp thời.

Quản lý đối tác

Admin có thể thêm mới, chỉnh sửa và xóa thông tin đối tác bao gồm tên, logo, website và mô tả. Mỗi đối tác có trạng thái hoạt động hoặc không hoạt động. Chỉ những đối tác đang hoạt động mới hiển thị logo trên trang chủ. Hệ thống hỗ trợ upload logo lên Cloudinary để lưu trữ và hiển thị.

Thống kê và báo cáo

Admin xem thống kê tổng quan trên Dashboard bao gồm số lượng khóa học, người dùng, đăng ký và phản hồi trong hệ thống. Hệ thống cung cấp giao diện dashboard trực quan giúp quản trị viên nắm bắt thông tin nhanh chóng và đưa ra quyết định kịp thời.

Yêu cầu phi chức năng

Bảo mật

Hệ thống yêu cầu xác thực người dùng bằng tài khoản và mật khẩu trước khi cho phép truy cập các chức năng quản trị. Mật khẩu được mã hóa bằng thuật toán bcrypt trước khi lưu vào cơ sở dữ liệu MongoDB. Sử dụng JSON Web Token (JWT) để quản lý phiên đăng nhập và xác thực người dùng. Mỗi vai trò (Admin, User) chỉ được truy cập vào các chức năng và dữ liệu tương ứng với quyền hạn của mình. Hệ thống sử dụng Mongoose ODM để chống NoSQL Injection và validate dữ liệu đầu vào. Cấu hình CORS để kiểm soát các nguồn truy cập API.

Hiệu năng

Thời gian tải trang không vượt quá 3 giây trong điều kiện mạng bình thường. Hệ thống phải xử lý được ít nhất 50 người dùng truy cập đồng thời mà không bị giảm tốc độ đáng kể. Các truy vấn MongoDB được tối ưu hóa với indexing để giảm thời gian phản hồi. Hình ảnh được lưu trữ trên Cloudinary để giảm tải cho server và tăng tốc độ tải. Frontend sử dụng Vite để build và tối ưu hóa bundle size.

Tính khả dụng

Hệ thống hoạt động ổn định 24/7 với tỷ lệ uptime đạt tối thiểu 99%. Dữ liệu được sao lưu (backup) định kỳ dưới dạng file JSON để phòng tránh mất mát thông tin. Hệ thống có cơ chế xử lý lỗi và thông báo rõ ràng cho người dùng khi tính năng tạm thời không khả dụng. MongoDB Atlas cung cấp khả năng replica set để đảm bảo tính sẵn sàng cao của dữ liệu.

Tính dễ sử dụng

Giao diện người dùng thân thiện, trực quan với thiết kế hiện đại sử dụng React. Hệ thống hiển thị thông báo rõ ràng khi thao tác thành công hoặc thất bại để người dùng biết trạng thái hiện tại. Website có thiết kế responsive, tương thích với

nhiều trình duyệt phổ biến (Chrome, Firefox, Edge) và hiển thị tốt trên các thiết bị khác nhau (PC, tablet, mobile). Có trang hướng dẫn sử dụng chi tiết cho người dùng.

Khả năng bảo trì

Mã nguồn được tổ chức theo kiến trúc phân tầng rõ ràng với Frontend chia thành Components, Pages, Services, Context và Backend chia thành Routes, Controllers, Models, Middleware. Hệ thống ghi log các hành động quan trọng và lỗi để hỗ trợ việc debug và audit. Code được comment đầy đủ tại các phần phức tạp và tuân thủ coding convention. Sử dụng biến môi trường (.env) để quản lý cấu hình, dễ dàng thay đổi giữa các môi trường development và production.

Khả năng mở rộng

Kiến trúc RESTful API cho phép dễ dàng thêm chức năng mới trong tương lai như thống kê nâng cao, chatbot AI hoặc hệ thống đánh giá. Cơ sở dữ liệu MongoDB với schema linh hoạt, có thể mở rộng số lượng người dùng và dữ liệu mà không cần thay đổi cấu trúc lớn. Hệ thống hỗ trợ tích hợp với các dịch vụ bên ngoài như Cloudinary và Email service thông qua API.

Tính tương thích

Backend hoạt động trên môi trường Node.js phiên bản 16.x trở lên với framework Express.js. Cơ sở dữ liệu sử dụng MongoDB phiên bản 6.0 trở lên. Frontend build bằng Vite, tương thích với các trình duyệt hiện đại hỗ trợ ES6+. Giao diện không yêu cầu cài đặt plugin đặc biệt, chỉ cần trình duyệt web thông thường.

3.2.2 Kiến trúc hệ thống

Mô hình kiến trúc

Hệ thống được xây dựng dựa trên mô hình Client-Server kết hợp với kiến trúc RESTful API, trong đó Frontend và Backend được tách biệt hoàn toàn. Phía Frontend sử dụng React.js với mô hình Component-Based Architecture, giúp tái sử dụng code và quản lý giao diện hiệu quả. Phía Backend áp dụng mô hình MVC (Model-View-Controller) với Node.js và Express.js, trong đó Model đại diện cho cấu trúc dữ liệu và tương tác với MongoDB thông qua Mongoose ODM, Controller chứa logic xử lý nghiệp vụ và điều phối các request từ client, còn Routes đóng vai trò định tuyến các API endpoint. Việc áp dụng kiến trúc này giúp mã nguồn rõ ràng, dễ bảo trì, dễ mở rộng và cho phép phát triển song song giữa đội ngũ Frontend và Backend. Giao tiếp giữa hai phía được thực hiện thông qua HTTP Request/Response với định dạng JSON, đảm bảo tính nhất quán và dễ dàng tích hợp và mở rộng trong các dự án lớn. Ngoài ra, kiến trúc tách biệt còn hỗ trợ tối ưu hiệu suất, bảo mật dữ liệu và nâng cao khả năng quản lý toàn bộ hệ thống.

Các thành phần chính

Model (Mô hình dữ liệu): Chịu trách nhiệm tương tác trực tiếp với cơ sở dữ liệu MongoDB thông qua Mongoose ODM. Các lớp Model bao gồm User (quản lý người dùng), Course (quản lý khóa học), Registration (quản lý đăng ký), News (quản lý tin tức), Partner (quản lý đối tác) và Feedback (quản lý phản hồi). Mỗi Model định nghĩa schema dữ liệu, thực hiện các thao tác truy vấn, thêm, sửa, xóa và đảm bảo tính toàn vẹn của thông tin trước khi trả về cho Controller.

View (Giao diện): Được xây dựng bằng React.js với kiến trúc Component-Based, hiển thị dữ liệu cho người dùng dưới dạng Single Page Application (SPA). View nhận dữ liệu từ Backend thông qua API và trình bày thông qua các React Components kết hợp với CSS. Giao diện được thiết kế responsive để đảm bảo tương thích trên nhiều thiết bị khác nhau như PC, tablet và mobile.

Controller (Bộ điều khiển): Đóng vai trò trung gian, tiếp nhận yêu cầu (request) từ client, xử lý logic nghiệp vụ, gọi dữ liệu từ Model và trả về response dưới dạng JSON. Các Controller được chia theo chức năng bao gồm authController (xác thực đăng nhập), adminController (quản trị hệ thống), courseController (quản lý khóa học), newsController (quản lý tin tức), partnerController (quản lý đối tác), registrationController (quản lý đăng ký) và feedbackController (quản lý phản hồi). Middleware được sử dụng để kiểm tra quyền truy cập và xác thực JWT token trước khi cho phép thực hiện các thao tác.

3.2.3 Công nghệ sử dụng

Frontend(Giao diện người dùng)

Giao diện website được xây dựng bằng React.js, một thư viện JavaScript phổ biến, giúp phát triển Single Page Application (SPA) với hiệu suất cao và trải nghiệm người dùng mượt mà. Hệ thống sử dụng Vite làm công cụ build và development server, giúp tăng tốc độ khởi động dự án và hỗ trợ hot reload nhanh chóng trong quá trình phát triển. Giao diện được thiết kế theo kiến trúc Component-Based, chia thành các thành phần tái sử dụng bao gồm: Components (các thành phần UI dùng chung), Pages (các trang chức năng), Context (quản lý state toàn cục) và Services (xử lý các request gọi API). CSS được tổ chức theo từng module riêng biệt, giúp giao diện hiện đại, thống nhất và tự động responsive trên mọi thiết bị, từ máy tính để bàn đến điện thoại di động. Để điều hướng giữa các trang mà không cần tải lại toàn bộ ứng dụng, dự án sử dụng React Router DOM, giúp trải nghiệm người dùng liền mạch, đồng thời giữ cho mã nguồn rõ ràng và dễ quản lý. Thiết kế này không chỉ tối ưu hiệu suất mà còn hỗ trợ tái sử dụng code và mở rộng giao diện dễ dàng trong các dự án lớn.

Backend(Xử lý phía máy chủ)

Phía máy chủ của hệ thống được phát triển dựa trên Node.js kết hợp với Express.js, tuân theo mô hình kiến trúc MVC (Model–View–Controller), giúp tách biệt rõ ràng giữa dữ liệu, logic nghiệp vụ và luồng xử lý request/response. Express.js được lựa chọn nhờ ưu điểm là framework nhẹ, linh hoạt, dễ sử dụng và có hiệu suất cao, hỗ trợ xây dựng các RESTful API một cách nhanh chóng, đồng thời dễ dàng mở rộng khi hệ thống phát triển thêm các tính năng mới.

Hệ thống áp dụng các Middleware để thực hiện các chức năng quan trọng như xác thực JWT token, kiểm tra quyền truy cập của người dùng, validate dữ liệu đầu vào và xử lý lỗi, đảm bảo an toàn, bảo mật và tính nhất quán cho toàn bộ ứng dụng. Các Controller được tổ chức theo từng nhóm chức năng nghiệp vụ riêng biệt, bao gồm xác thực người dùng, quản lý khóa học, tin tức, đối tác, đăng ký khóa học và xử lý phản hồi từ người dùng. Cách phân chia này giúp quản lý logic nghiệp vụ rõ ràng, dễ dàng bảo trì và mở rộng khi cần thêm các tính năng mới.

Việc tách biệt hoàn toàn Frontend và Backend không chỉ giúp hai đội phát triển có thể làm việc song song mà còn tăng tính linh hoạt trong bảo trì và nâng cấp từng phần mà không ảnh hưởng đến toàn bộ hệ thống. Kiến trúc này cũng hỗ trợ tối ưu hiệu suất, dễ dàng kiểm thử và đảm bảo hệ thống có khả năng mở rộng, đáp ứng nhu cầu người dùng trong tương lai.

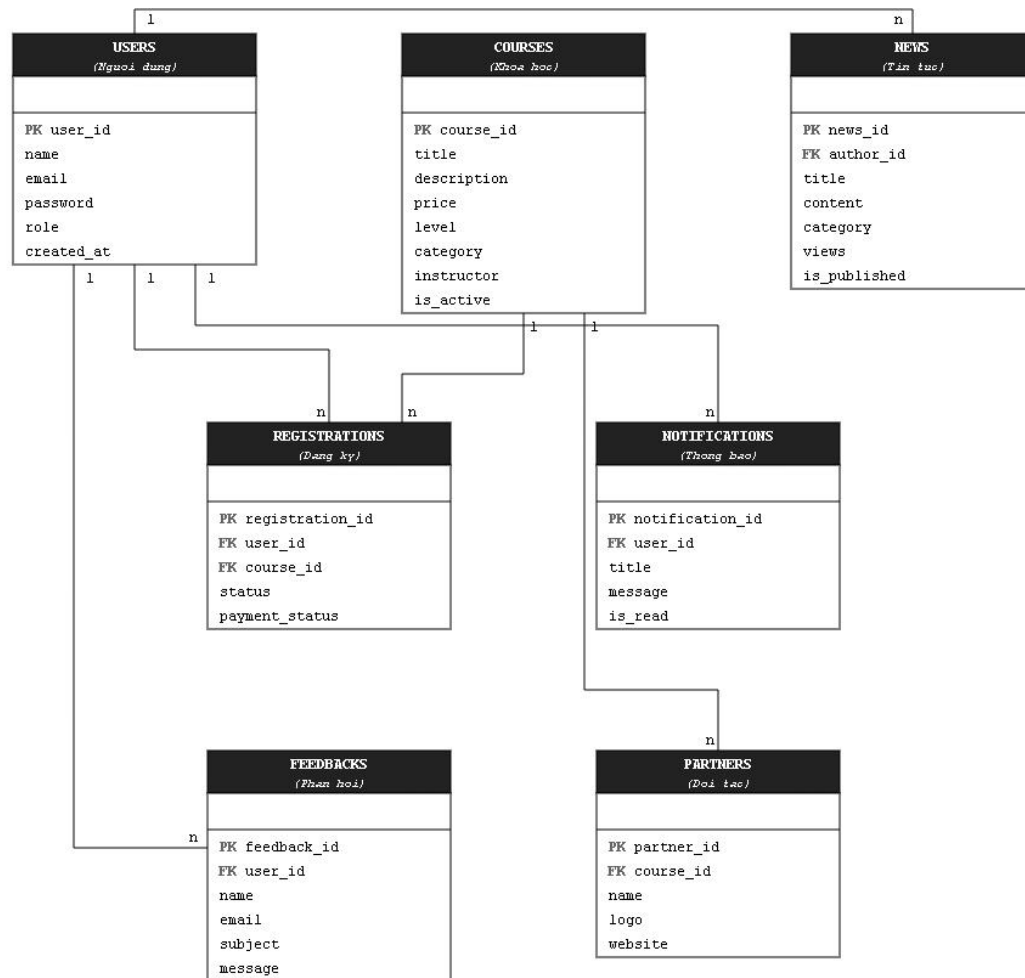
Database(Cơ sở dữ liệu)

Hệ thống sử dụng MongoDB, một hệ quản trị cơ sở dữ liệu NoSQL hướng document, để lưu trữ và quản lý toàn bộ dữ liệu của ứng dụng. Với đặc điểm lưu trữ dữ liệu dưới dạng JSON-like documents, MongoDB cho phép schema linh hoạt, dễ dàng mở rộng và thích ứng với các thay đổi hoặc yêu cầu mới trong tương lai.

Để tương tác hiệu quả với cơ sở dữ liệu, dự án sử dụng Mongoose ODM làm lớp trung gian, giúp định nghĩa schema rõ ràng, validate dữ liệu trước khi lưu trữ và thực hiện các thao tác CRUD một cách an toàn và đáng tin cậy. Cơ sở dữ liệu được thiết kế với các collection chính bao gồm: Users (người dùng), Courses (khóa học), Registrations (đăng ký khóa học), News (tin tức), Partners (đối tác) và Feedbacks (phản hồi), đảm bảo tổ chức dữ liệu khoa học, dễ quản lý và truy xuất.

Ngoài ra, các hình ảnh và file media được lưu trữ trên dịch vụ Cloudinary, giúp giảm tải cho server, tăng tốc độ truy xuất dữ liệu và đảm bảo khả năng mở rộng khi hệ thống cần lưu trữ lượng lớn file media. Kiến trúc này không chỉ tối ưu hiệu suất mà còn đảm bảo an toàn, nhất quán và linh hoạt cho toàn bộ hệ thống.

3.2.4 Thiết kế dữ liệu



Hình 3.2.4 Mô hình ERD

Hệ thống website Trung tâm AI được thiết kế với 6 collection chính trong MongoDB.

USERS lưu trữ thông tin người dùng bao gồm admin và học viên với các thuộc tính như name, email, password (đã mã hóa), role và created_at để phân quyền truy cập hệ thống.

COURSES quản lý các khóa học do trung tâm cung cấp với các thông tin title, description, price, level, category, instructor và trạng thái is_active, một người dùng (giảng viên) có thể phụ trách nhiều khóa học (quan hệ 1-n).

REGISTRATIONS ghi nhận việc học viên đăng ký khóa học với user_id, course_id, status và payment_status, một học viên có thể đăng ký nhiều khóa học và một khóa học có nhiều học viên đăng ký (quan hệ n-n qua collection trung gian).

NEWS quản lý tin tức và bài viết với author_id liên kết đến người đăng, bao gồm title, content, category, views và trạng thái is_published, một người dùng có thể đăng nhiều tin tức (quan hệ 1-n).

NOTIFICATIONS quản lý thông báo gửi đến người dùng với user_id, title, message và trạng thái is_read, một người dùng có thể nhận nhiều thông báo (quan hệ 1-n).

FEEDBACKS lưu trữ phản hồi từ người dùng hoặc khách truy cập với các thông tin name, email, subject và message, có thể liên kết với user_id nếu người gửi đã đăng nhập.

PARTNERS quản lý thông tin các đối tác hợp tác với trung tâm bao gồm name, logo và website, có thể liên kết với course_id nếu đối tác tài trợ cho khóa học cụ thể (quan hệ 1-n).

Danh sách các thực thể và mối kết hợp

Tên thực thể: Collection USERS (Người dùng)

Mô tả: Lưu trữ thông tin tài khoản người dùng

Chi tiết thực thể: Phân quyền và quản lý tài khoản admin, học viên

Bảng 3.1 Collection Users

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	_id	Mã định danh người dùng	ObjectId	PRIMARY KEY, AUTO GENERATE, NOT NULL
2	name	Họ và tên đầy đủ	String	NOT NULL
3	email	Địa chỉ email	String	UNIQUE, NOT NULL
4	password	Mật khẩu đã mã hóa bcrypt	String	NOT NULL
5	role	Vai trò (user/admin)	String (Enum)	NOT NULL, DEFAULT 'user'
6	phone	Số điện thoại liên hệ	String	NULL
7	avatar	Đường dẫn ảnh đại diện	String	NULL
8	createdAt	Thời gian tạo tài khoản	Date	AUTO GENERATE

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
9	updatedAt	Thời gian cập nhật	Date	AUTO GENERATE

Chi tiết các thực thể và mối kết hợp

Tên thực thể: Collection COURSES (Khóa học)

Mô tả: Lưu trữ thông tin các khóa học của trung tâm

Chi tiết thực thể: Quản lý nội dung, lịch học, học phí và thông tin chi tiết khóa học

Bảng 3.2 Collection Courses

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	_id	Mã định danh khóa học	ObjectId	PRIMARY KEY, AUTO GENERATE, NOT NULL
2	title	Tên khóa học	String	NOT NULL
3	description	Mô tả ngắn	String	NOT NULL
4	content	Nội dung chi tiết	String	NULL
5	image	Đường dẫn hình ảnh	String	NULL
6	duration	Thời lượng khóa học	String	NULL
7	level	Trình độ (beginner/intermediate/advanced)	String (Enum)	NULL
8	price	Học phí	Number	DEFAULT 0
9	originalPrice	Giá gốc	Number	DEFAULT 0

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
10	instructor	Tên giảng viên	String	NULL
11	category	Danh mục khóa học	String	NULL
12	isActive	Trạng thái hoạt động	Boolean	DEFAULT true
13	objectives	Mục tiêu khóa học	Array[String]	NULL
14	targetAudience	Đối tượng học viên	Array[String]	NULL
15	curriculum	Chương trình học theo tuần	Array[Object]	NULL
16	schedule	Lịch học (ngày, giờ, địa điểm)	Object	NULL
17	includes	Khóa học bao gồm	Array[String]	NULL
18	discounts	Ưu đãi giảm giá	Array[String]	NULL
19	maxStudents	Số học viên tối đa	Number	DEFAULT 30
20	createdAt	Thời gian tạo	Date	AUTO GENERATE
21	updatedAt	Thời gian cập nhật	Date	AUTO GENERATE

Tên thực thể: Collection REGISTRATIONS (Đăng ký)

Mô tả: Lưu trữ thông tin đăng ký khóa học của học viên

Chi tiết thực thể: Quản lý trạng thái đăng ký và thanh toán

Bảng 3.3 Collection Registrations

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	_id	Mã định danh đăng ký	ObjectId	PRIMARY KEY, AUTO GENERATE, NOT NULL
2	user	Mã người dùng đăng ký	ObjectId	FOREIGN KEY (Users), NOT NULL
3	course	Mã khóa học đăng ký	ObjectId	FOREIGN KEY (Courses), NOT NULL
4	status	Trạng thái (pending/approved /rejected)	String (Enum)	DEFAULT 'pending'
5	paymentStatus	Trạng thái thanh toán (unpaid/paid)	String (Enum)	DEFAULT 'unpaid'
6	createdAt	Thời gian đăng ký	Date	AUTO GENERATE
7	updatedAt	Thời gian cập nhật	Date	AUTO GENERATE

Tên thực thể: Collection NEWS (Tin tức)

Mô tả: Lưu trữ các bài viết, tin tức của trung tâm

Chi tiết thực thể: Quản lý nội dung tin tức, bài viết và lượt xem

Bảng 3.4 Collection News

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	_id	Mã định danh tin tức	ObjectId	PRIMARY KEY, AUTO GENERATE, NOT NULL

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
2	title	Tiêu đề bài viết	String	NOT NULL
3	content	Nội dung bài viết	String	NOT NULL
4	summary	Tóm tắt nội dung	String	NULL
5	detailedContent	Nội dung chi tiết (HTML)	String	NULL
6	image	Đường dẫn hình ảnh	String	NULL
7	author	Mã tác giả	ObjectId	FOREIGN KEY (Users), NULL
8	category	Danh mục tin tức	String	NULL
9	tags	Các thẻ tag	Array[String]	NULL
10	views	Số lượt xem	Number	DEFAULT 0
11	isPublished	Trạng thái xuất bản	Boolean	DEFAULT true
12	createdAt	Thời gian tạo	Date	AUTO GENERATE
13	updatedAt	Thời gian cập nhật	Date	AUTO GENERATE

Tên thực thể: Collection PARTNERS (Đối tác)

Mô tả: Lưu trữ thông tin các đối tác hợp tác với trung tâm

Chi tiết thực thể: Quản lý thông tin, logo và website của các đối tác

Bảng 3.5 Collection Partners

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	_id	Mã định danh đối tác	ObjectId	PRIMARY KEY, AUTO GENERATE, NOT NULL

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
2	name	Tên đối tác	String	NOT NULL
3	logo	Đường dẫn logo	String	NULL
4	website	Website đối tác	String	NULL
5	description	Mô tả đối tác	String	NULL
6	isActive	Trạng thái hoạt động	Boolean	DEFAULT true
7	createdAt	Thời gian tạo	Date	AUTO GENERATE
8	updatedAt	Thời gian cập nhật	Date	AUTO GENERATE

Tên thực thể: Collection FEEDBACKS (Phản hồi)

Mô tả: Lưu trữ phản hồi, góp ý từ người dùng

Chi tiết thực thể: Quản lý các liên hệ và phản hồi từ khách truy cập

Bảng 3.6 Collection Feedbacks

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	_id	Mã định danh phản hồi	ObjectId	PRIMARY KEY, AUTO GENERATE, NOT NULL
2	name	Họ tên người gửi	String	NOT NULL
3	email	Email người gửi	String	NOT NULL
4	phone	Số điện thoại	String	NULL
5	subject	Chủ đề phản hồi	String	NULL

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
6	message	Nội dung phản hồi	String	NOT NULL
7	status	Trạng thái (pending/read/replied)	String (Enum)	DEFAULT 'pending'
8	createdAt	Thời gian gửi	Date	AUTO GENERATE
9	updatedAt	Thời gian cập nhật	Date	AUTO GENERATE

Tên thực thể: Collection NOTIFICATIONS (Thông báo)

Mô tả: Lưu trữ thông báo gửi đến người dùng

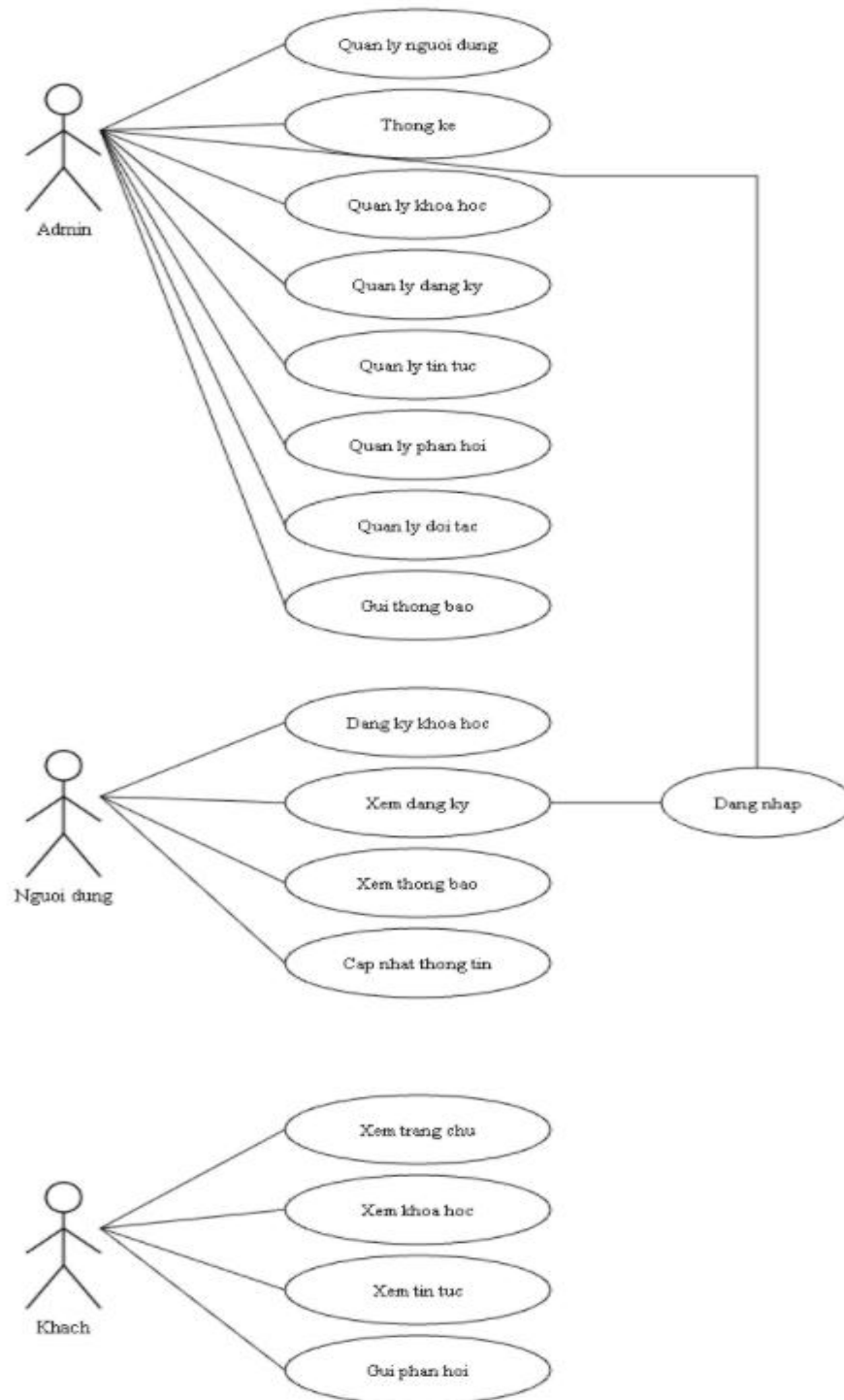
Chi tiết thực thể: Quản lý thông báo hệ thống, đăng ký, thanh toán, khóa học

Bảng 3.7 Collection Notifications

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
1	_id	Mã định danh thông báo	ObjectId	PRIMARY KEY, AUTO GENERATE, NOT NULL
2	user	Mã người nhận thông báo	ObjectId	FOREIGN KEY (Users), NOT NULL
3	title	Tiêu đề thông báo	String	NOT NULL
4	message	Nội dung thông báo	String	NOT NULL
5	type	Loại thông báo (registration/payment/course/news/system)	String (Enum)	DEFAULT 'system'
6	isRead	Trạng thái đã đọc	Boolean	DEFAULT false

STT	Tên thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc toàn vẹn
7	link	Link điều hướng khi click	String	NULL
8	createdAt	Thời gian tạo	Date	AUTO GENERATE
9	updatedAt	Thời gian cập nhật	Date	AUTO GENERATE

3.2.5 Thiết kế xử lý



Hình 3.2.5 Mô hình Use Case

Sơ đồ Use Case của hệ thống website trung tâm trí tuệ nhân tạo mô tả cách ba tác nhân chính tương tác với các chức năng của hệ thống, bao gồm Admin, Người dùng và Khách. Mỗi tác nhân có những quyền hạn và chức năng riêng biệt, được phân chia rõ ràng để đảm bảo tính bảo mật và hiệu quả trong quản lý hệ thống.

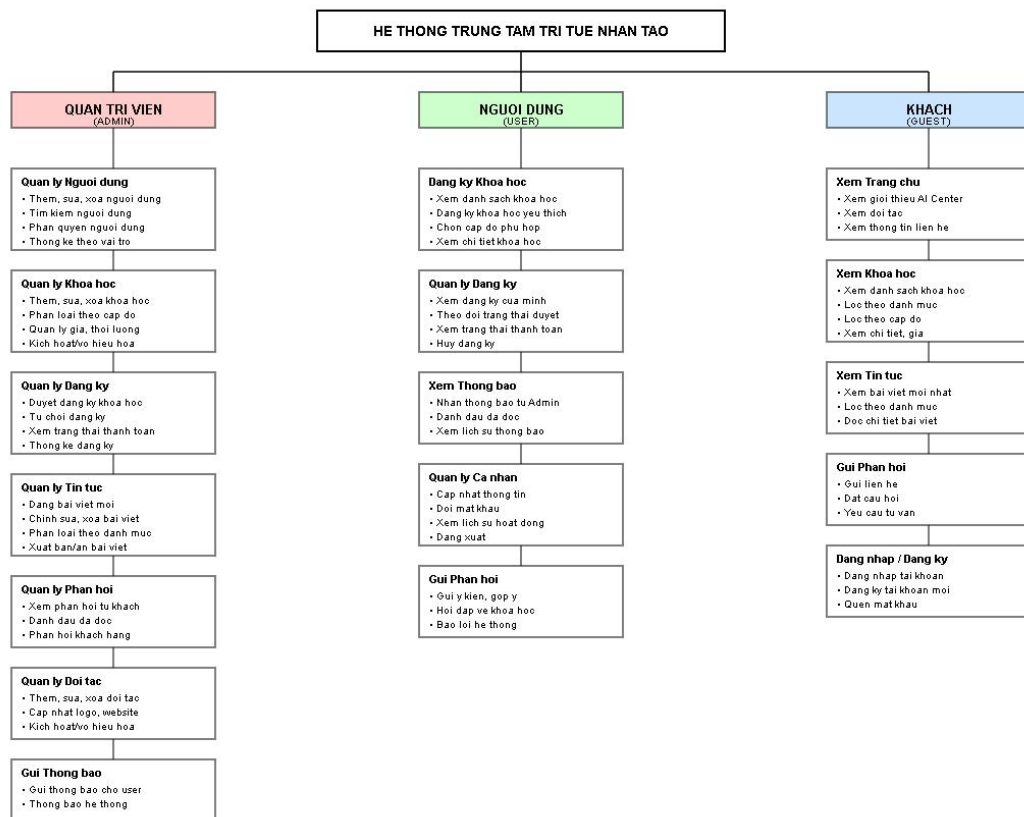
Admin là người quản trị toàn bộ hệ thống, có quyền cao nhất và chịu trách nhiệm vận hành, giám sát mọi hoạt động của website. Admin có thể quản lý tài khoản người dùng, bao gồm thêm mới tài khoản, chỉnh sửa thông tin, xóa tài khoản không còn hoạt động và phân quyền cho từng tài khoản. Bên cạnh đó, Admin có chức năng xem thống kê tổng hợp thông qua Dashboard trực quan, hiển thị số lượng khóa học đang hoạt động, tổng số người dùng đã đăng ký, số lượng đăng ký khóa học theo từng trạng thái và số phản hồi cần xử lý.

Admin thực hiện quản lý khóa học với đầy đủ các thao tác bao gồm thêm mới khóa học với thông tin chi tiết, chỉnh sửa thông tin khóa học khi cần cập nhật, xóa các khóa học không còn tổ chức và kích hoạt hoặc vô hiệu hóa khóa học để kiểm soát việc hiển thị trên trang công khai. Admin cũng quản lý đăng ký khóa học, xem danh sách tất cả đăng ký từ người dùng, duyệt hoặc từ chối các đăng ký dựa trên điều kiện của học viên và cập nhật trạng thái thanh toán khi nhận được học phí. Ngoài ra, Admin quản lý tin tức với các chức năng thêm bài viết mới, chỉnh sửa nội dung, xóa bài viết cũ và xuất bản hoặc ẩn bài viết để kiểm soát nội dung hiển thị. Admin quản lý phản hồi từ khách hàng bằng cách xem danh sách, đánh dấu đã đọc và đánh dấu đã trả lời. Admin quản lý thông tin đối tác và có thể gửi thông báo đến người dùng trong hệ thống.

Người dùng là người đã đăng ký tài khoản và đăng nhập vào hệ thống, có thể tận dụng đầy đủ các tính năng dành cho học viên. Sau khi đăng nhập, người dùng có thể xem danh sách tất cả khóa học đang mở, lọc khóa học theo danh mục và cấp độ, xem chi tiết từng khóa học và đăng ký tham gia các khóa học mong muốn. Người dùng có thể xem danh sách các khóa học đã đăng ký trong mục "Khóa học của tôi", theo dõi trạng thái đăng ký bao gồm chờ duyệt, đã duyệt hoặc từ chối, cũng như trạng thái thanh toán để biết mình đã hoàn thành học phí hay chưa. Người dùng có thể xem các thông báo từ hệ thống, đánh dấu đã đọc và cập nhật thông tin cá nhân của mình. Các chức năng này giúp người dùng quản lý quá trình học tập một cách hiệu quả.

Khách là người truy cập website mà chưa đăng nhập, có thể xem các thông tin công khai của trung tâm. Khách có thể xem trang chủ với banner giới thiệu về trung tâm, các khóa học nổi bật, tin tức mới nhất và danh sách đối tác. Khách có thể xem danh sách khóa học với khả năng lọc theo danh mục và cấp độ, xem chi tiết từng khóa học bao gồm nội dung chương trình, lịch học và học phí. Khách cũng có thể xem danh sách tin tức và nội dung chi tiết từng bài viết. Ngoài ra, khách có thể gửi phản hồi, liên hệ thông qua form liên hệ để được tư vấn hoặc giải đáp thắc mắc, đăng ký tài khoản mới để trở thành người dùng hoặc đăng nhập nếu đã có tài khoản.

3.2.6 Thiết kế giao diện



Hình 3.2.6 Sơ đồ giao diện hệ thống

Mô tả:

Hệ thống trung tâm trí tuệ nhân tạo vận hành theo một quy trình liên kết chặt chẽ giữa các vai trò tham gia, bao gồm Quản trị viên (Admin), Người dùng (User) và Khách (Guest).

Ở tầng quản trị, Admin giữ vai trò thiết lập dữ liệu nền tảng cho toàn bộ hệ thống thông qua các chức năng như Quản lý người dùng với khả năng thêm, sửa, xóa, tìm kiếm và phân quyền người dùng. Admin thực hiện Quản lý khóa học bao gồm thêm, sửa, xóa khóa học, phân loại theo cấp độ, quản lý giá và thời lượng, kích hoạt hoặc vô hiệu hóa khóa học. Chức năng Quản lý đăng ký cho phép Admin duyệt hoặc từ chối đăng ký, xem và cập nhật trạng thái thanh toán, thống kê đăng ký. Admin cũng thực hiện Quản lý tin tức với các thao tác đăng bài viết mới, chỉnh sửa, xóa, phân loại theo danh mục và xuất bản hoặc ẩn bài viết. Chức năng Quản lý phản hồi giúp Admin xem phản hồi từ khách, đánh dấu đã đọc và phản hồi khách hàng. Quản lý đối tác cho phép thêm, sửa, xóa đối tác, cập nhật logo và website, kích hoạt hoặc vô hiệu hóa hiển thị. Đồng thời, thông qua công cụ Gửi thông báo, Admin có thể truyền đạt thông tin quan trọng đến người dùng trong hệ thống.

Người dùng đã đăng ký được hỗ trợ bởi một chuỗi chức năng xuyên suốt và liền mạch. Chức năng Đăng ký khóa học cho phép xem danh sách khóa học, đăng ký khóa học yêu thích, chọn cấp độ phù hợp và xem chi tiết khóa học. Quản lý đăng ký giúp người dùng xem đăng ký của mình, theo dõi trạng thái duyệt, xem trạng thái thanh toán và hủy đăng ký khi cần. Chức năng Xem thông báo cho phép nhận thông báo từ Admin, đánh dấu đã đọc và xem lịch sử thông báo. Quản lý cá nhân hỗ trợ cập nhật thông tin, đổi mật khẩu, xem lịch sử hoạt động và đăng xuất. Người dùng cũng có thể Gửi phản hồi bao gồm gửi ý kiến, góp ý, hỏi đáp về khóa học và báo lỗi hệ thống.

Khách truy cập có thể Xem trang chủ với giới thiệu về trung tâm, xem đối tác và thông tin liên hệ. Chức năng Xem khóa học cho phép xem danh sách khóa học, lọc theo danh mục và cấp độ, xem chi tiết và giá. Xem tin tức giúp khách xem bài viết mới nhất, lọc theo danh mục và đọc chi tiết bài viết. Khách có thể Gửi phản hồi thông qua form liên hệ, đặt câu hỏi và yêu cầu tư vấn. Cuối cùng, chức năng Đăng nhập/Đăng ký cho phép khách đăng nhập tài khoản và đăng ký tài khoản mới.

CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU

4.1. Dữ liệu thử nghiệm

```
_id: ObjectId('694fa665a86c9100bdce02c2')
course : ObjectId('694fa664a86c9100bdce023e')
batchName : "Đợt 1 - Tháng 13/2025"
startDate : 2026-01-14T17:00:00.000+00:00
endDate : 2026-04-14T17:00:00.000+00:00
registrationDeadline : 2026-01-09T17:00:00.000+00:00
schedule : "Thứ 3, 5, 7 (19:00 - 21:00)"
location : "AI Center - 126 Nguyễn Huệ, TP. Vĩnh Long"
maxStudents : 30
currentStudents : 2
price : 5000000
status : "open"
isActive : true
createdAt : 2025-12-27T09:27:01.052+00:00
updatedAt : 2025-12-27T09:27:01.052+00:00
__v : 0
```

Hình 4.1 Collections coursebatches(Đợt học)

```
_id: ObjectId('694fa664a86c9100bdce023e')
title : "Machine Learning cơ bản"
description : "Khóa học giới thiệu về Machine Learning, các thuật toán cơ bản"
content : "Linear Regression, Logistic Regression, Decision Trees, Random Forest,..."
image : "https://images.unsplash.com/photo-1555949963-aa79dcee981c?w=800"
duration : "12 tuần"
level : "beginner"
price : 5000000
originalPrice : 0
instructor : "TS. Nguyễn Văn A"
category : "machine-learning"
isActive : true
▶ objectives : Array (empty)
▶ targetAudience : Array (empty)
▶ includes : Array (empty)
▶ discounts : Array (empty)
maxStudents : 30
createdAt : 2025-11-30T14:06:36.625+00:00
updatedAt : 2025-11-30T14:06:36.625+00:00
▶ curriculum : Array (empty)
__v : 0
```

Hình 4.2 Collections courses(Khóa học)


```
_id: ObjectId('694ff1de217918175734e236')
name: "Lê Công Vinh"
email: "lecongvinh@gmail.com"
phone: "0274734342"
subject: "Xin chào"
message: "Khó khăn trong việc đăng kí khóa học"
status: "pending"
createdAt: 2025-12-27T14:49:02.552+00:00
updatedAt: 2025-12-27T14:49:02.552+00:00
__v: 0
```

Hình 4.3 Collections freedbacks(Phản hồi)

```
_id: ObjectId('694fa664a86c9100bdce0270')
title: "AI Center khai giảng khóa học Machine Learning mới"
content: "AI Center vui mừng thông báo khai giảng khóa học Machine Learning cơ bản"
summary: "Khóa học Machine Learning cơ bản dành cho người mới bắt đầu sẽ khai gi..."
detailedContent: "Machine Learning - Nền tảng của AI hiện đại"

Machine Learning (ML)..."
image: "https://images.pexels.com/photos/3861969/pexels-photo-3861969.jpeg?aut..."
author: ObjectId('694fa663a86c9100bdce023a')
category: "Events"
tags: Array (3)
views: 214
isPublished: true
createdAt: 2025-11-30T14:06:36.670+00:00
updatedAt: 2025-11-30T14:06:36.670+00:00
__v: 0
```

Hình 4.4 Collections news(Tin tức)

```
_id: ObjectId('694fabb814fc98f521d84d83')
user: ObjectId('694fa663a86c9100bdce023c')
title: "hi"
message: "hãy đăng kí khóa học"
type: "registration"
isRead: true
link: "http://localhost:5173/courses"
createdAt: 2025-12-27T09:49:44.622+00:00
updatedAt: 2025-12-27T09:52:13.491+00:00
__v: 0
```

Hình 4.5 Collections notifications(Thông báo)

```
_id: ObjectId('694fa664a86c9100bdce02a6')
name: "Google"
logo: "https://www.google.com/images/branding/googlelogo/2x/googlelogo_color_..."
website: "https://www.google.com"
description: "Đối tác công nghệ hàng đầu thế giới"
isActive: true
createdAt: 2025-11-30T12:01:17.862+00:00
updatedAt: 2025-11-30T12:01:17.862+00:00
__v: 0
```

Hình 4.6 Collections partners(Đối tác)

```
_id: ObjectId('694faf9914fc98f521d84fe9')
user: ObjectId('694fa663a86c9100bdce023a')
course: ObjectId('694fa664a86c9100bdce0246')
batch: ObjectId('694fa665a86c9100bdce02ca')
status: "pending"
paymentStatus: "unpaid"
createdAt: 2025-12-27T10:06:17.253+00:00
updatedAt: 2025-12-27T10:06:17.253+00:00
__v: 0
```

Hình 4.7 Collections registrations(Đăng kí)

```
{
  "_id": "ObjectId('694fa664a86c9100bdce02bc')",
  "name": "TS. Nguyễn Văn An",
  "position": "Giám đốc Trung tâm",
  "bio": "Tiến sĩ Khoa học Máy tính tại Đại học Stanford. Hơn 15 năm kinh nghiệm...",
  "image": "https://images.unsplash.com/photo-1472099645785-5658abf4ff4e?w=400",
  "email": "an.nguyen@aicenter.edu.vn",
  "phone": "0901234567",
  "order": 1,
  "isActive": true,
  "__v": 0,
  "createdAt": "2025-12-27T09:27:00.922+00:00",
  "updatedAt": "2025-12-27T09:27:00.922+00:00"
}
```

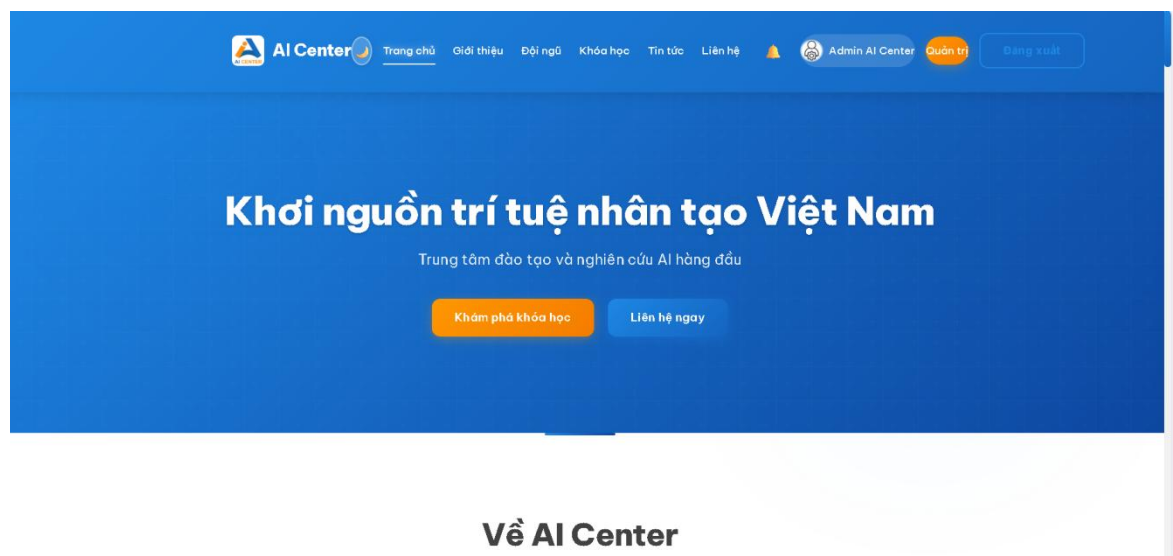
Hình 4.8 Collections teams (Đội ngũ giảng viên)

```
{
  "_id": "ObjectId('694fa663a86c9100bdce023c')",
  "name": "Nguyễn Văn A",
  "email": "user@gmail.com",
  "password": "$2a$10$HGom1aA5skbW6ARuLHn1S0Mmo1kfsEzQuUqD/T4.J4wRIi1llyR6y",
  "role": "user",
  "createdAt": "2025-12-27T09:26:59.841+00:00",
  "updatedAt": "2025-12-27T09:26:59.841+00:00",
  "__v": 0
}
```

Hình 4.9 Collections users(Người dùng)

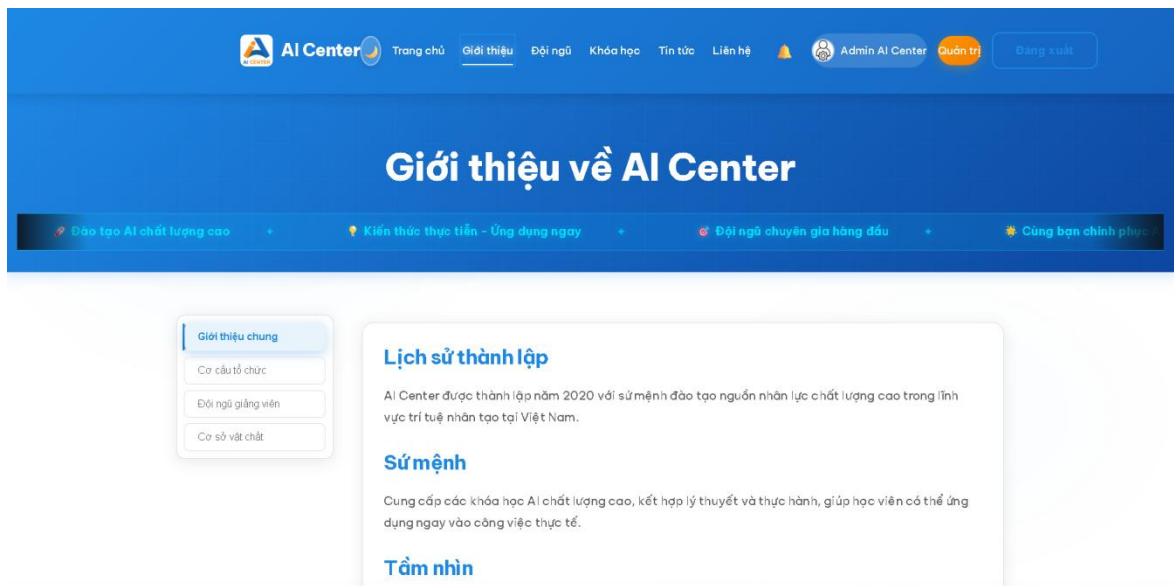
4.2. Kết quả thực nghiệm

4.2.1 Chức năng trang giao diện



Hình 4.10 Giao diện trang chủ

Giới thiệu tổng quan về trung tâm, hiển thị các khóa học nổi bật và thống kê số liệu. Người dùng có thể điều hướng đến các trang khác.



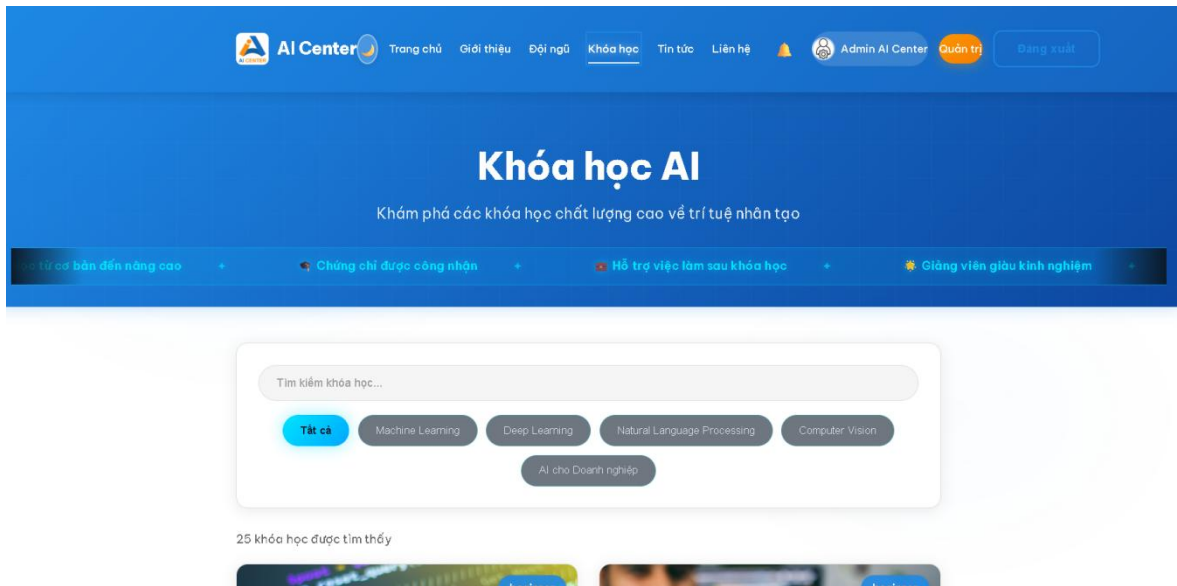
Hình 4.11 Giao diện trang giới thiệu

Cho phép người dùng xem thông tin về lịch sử, sứ mệnh, tầm nhìn của trung tâm, cơ cấu tổ chức, đội ngũ giảng viên, cơ sở vật chất.



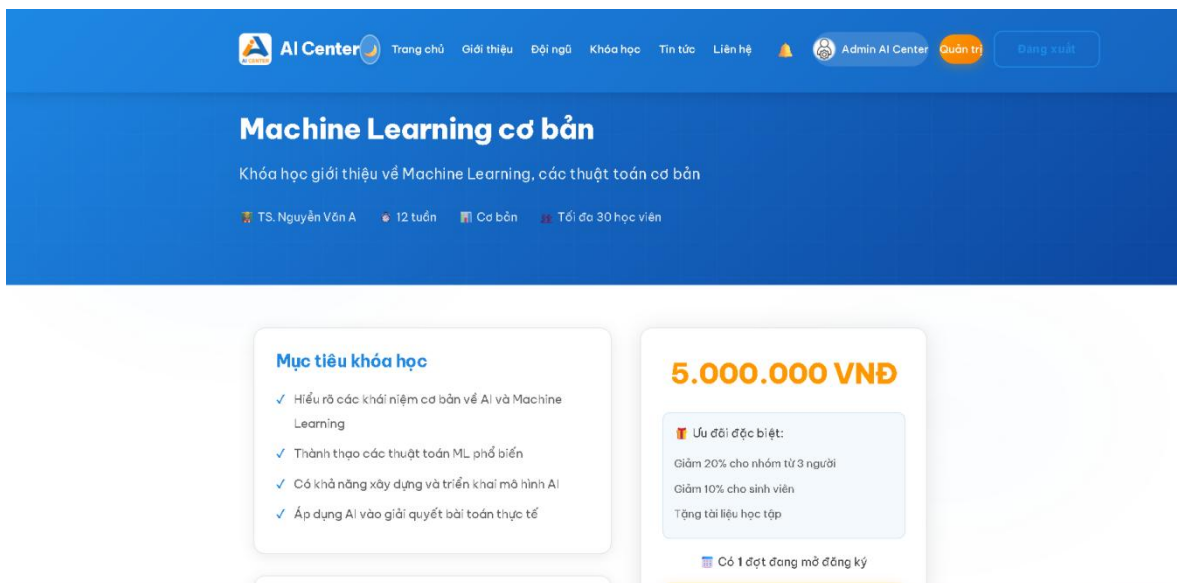
Hình 4.12 Giao diện đội ngũ giảng viên

Cho phép người dùng xem danh sách giảng viên của trung tâm với thông tin cơ bản gồm ảnh đại diện, họ tên và chức vụ.



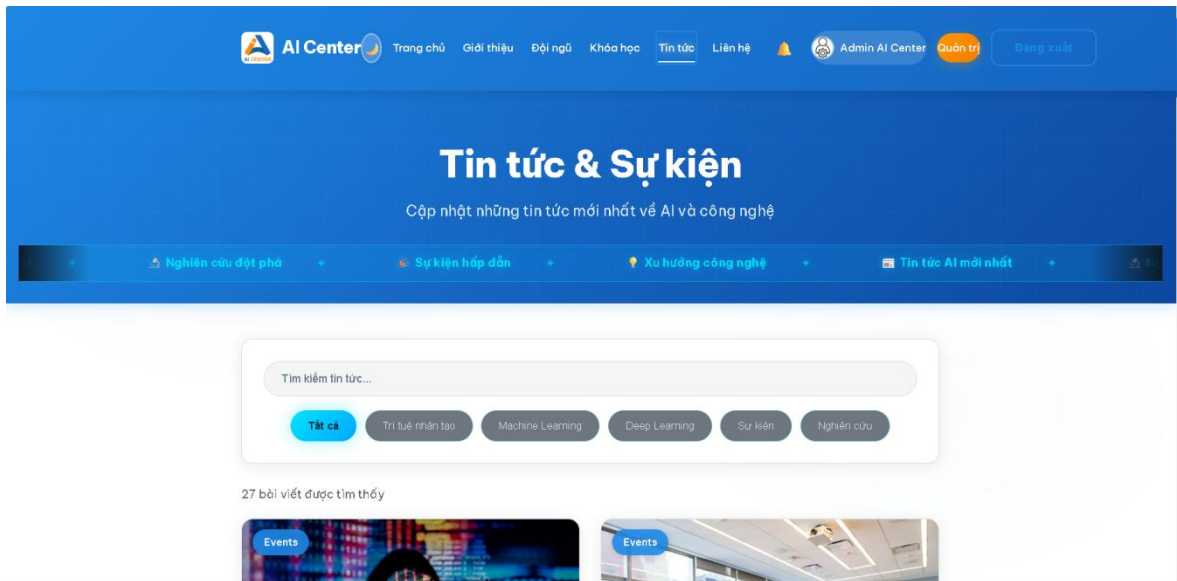
Hình 4.13 Giao diện trang khóa học

Cho phép người dùng xem, tìm kiếm khóa học theo tên, lọc theo danh mục và cấp độ.



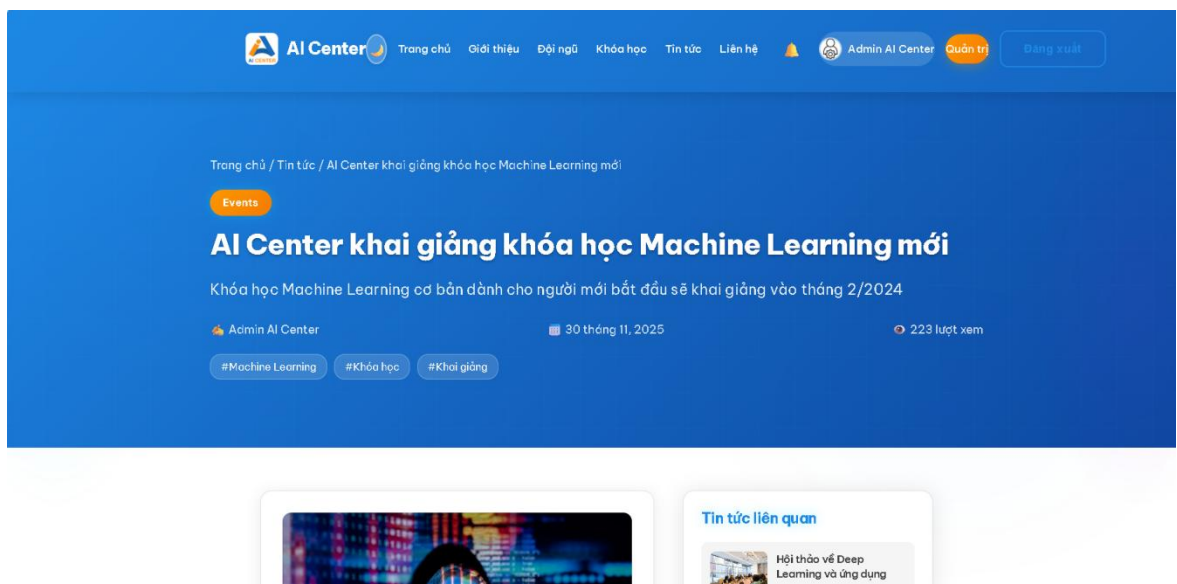
Hình 4.14 Giao diện trang chi tiết khóa học

Cho phép người dùng xem đầy đủ thông tin về khóa học bao gồm mục tiêu, đối tượng học viên, nội dung chương trình theo từng tuần, thông tin giảng viên và học phí. Trang hiển thị danh sách các đợt học sắp tới với thông tin ngày bắt đầu, lịch học, địa điểm và số lượng chỗ còn trống. Người dùng có thể chọn đợt học phù hợp và nhấn nút đăng ký để ghi danh vào khóa học theo ý muốn.



Hình 4.15 Giao diện trang tin tức

Cho phép người dùng xem danh sách các bài viết, tin tức và sự kiện của trung tâm. Hỗ trợ tìm kiếm bài viết theo tiêu đề và lọc theo danh mục như AI, Sự kiện, Machine Learning để dễ dàng tìm kiếm nội dung quan tâm.



Hình 4.16 Giao diện trang chi tiết tin tức

Cho phép người dùng đọc nội dung đầy đủ của bài viết với thông tin tác giả, ngày đăng và lượt xem. Trang còn hiển thị danh sách tin tức liên quan và hỗ trợ chia sẻ bài viết lên các mạng xã hội như Facebook, Twitter.



Thông tin liên hệ

Địa chỉ
126 Nguyễn Huệ, Phường 1, TP. Vinh Long

Số điện thoại
(0270) 123 4567

Gửi tin nhắn cho chúng tôi

Họ và tên *

Email *

Số điện thoại *

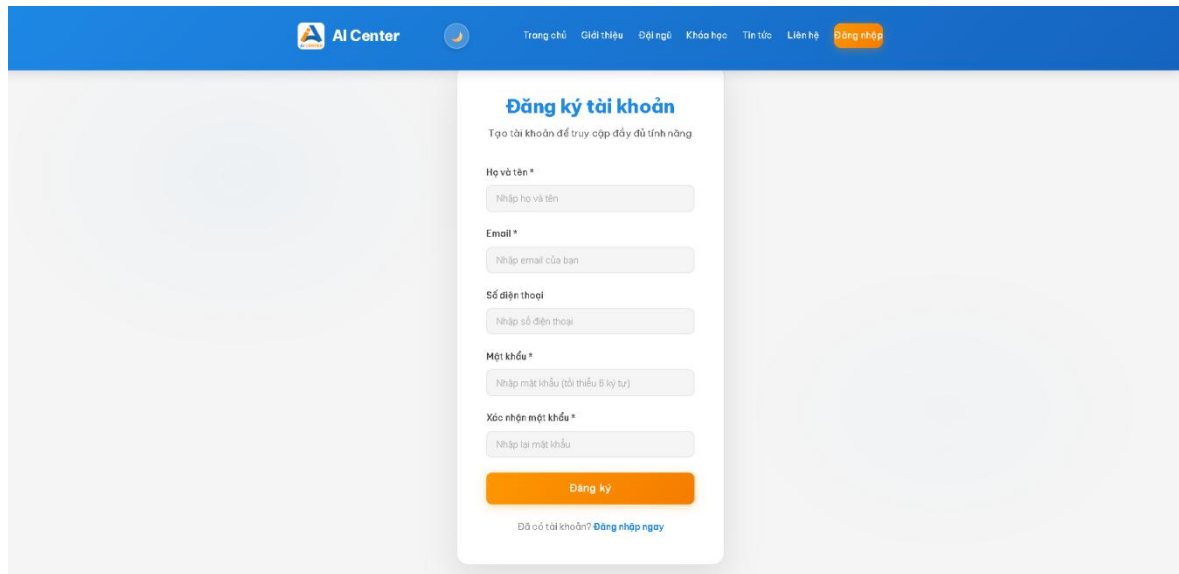
Tiêu đề

Hình 4.17 Giao diện trang liên hệ

Cho phép người dùng xem thông tin liên hệ, bản đồ và gửi phản hồi.

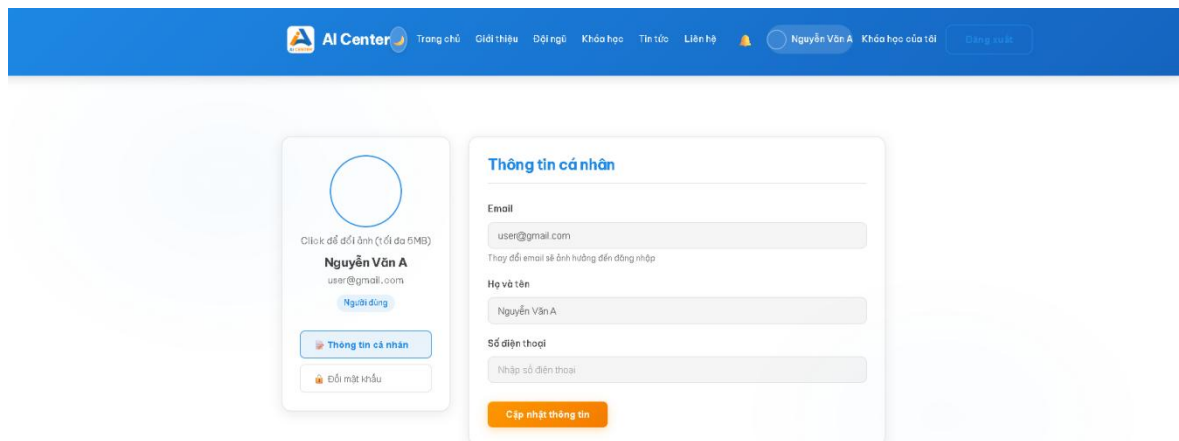
Hình 4.18 Giao diện trang đăng nhập

Cho phép người dùng đăng nhập vào hệ thống bằng địa chỉ email và mật khẩu đã đăng ký trước đó. Hệ thống sẽ xác thực thông tin và hiển thị thông báo lỗi nếu email không tồn tại hoặc mật khẩu không chính xác. Sau khi đăng nhập thành công, người dùng sẽ được chuyển hướng về trang chủ và có thể truy cập các chức năng dành riêng cho thành viên. Trang còn cung cấp liên kết đến trang quên mật khẩu và đăng ký tài khoản mới cho người dùng chưa có tài khoản.



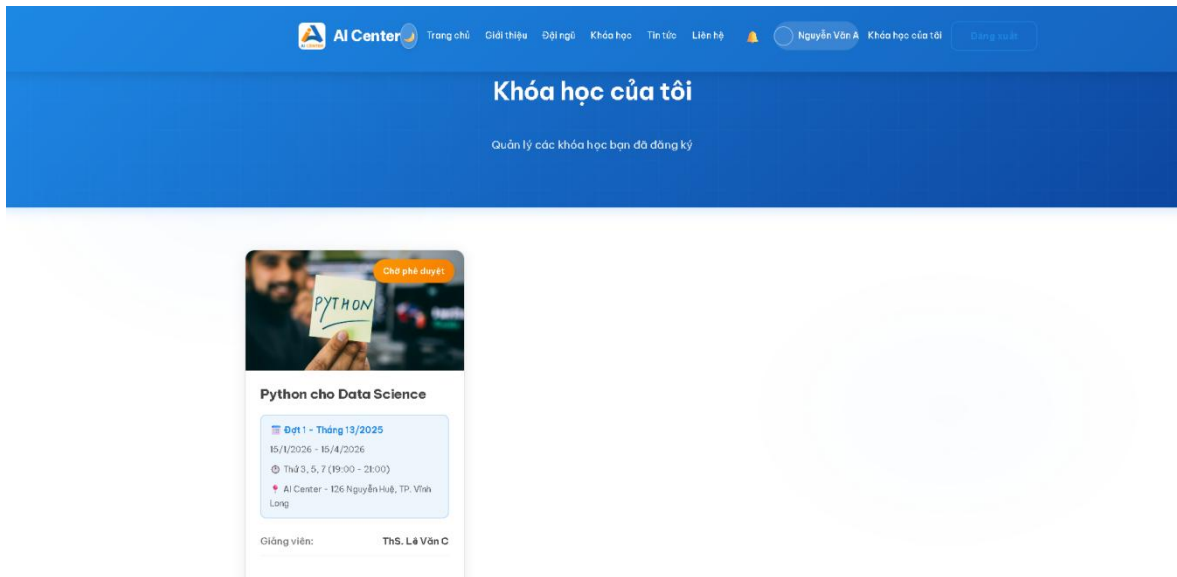
Hình 4.19 Giao diện trang đăng ký

Cho phép người dùng tạo tài khoản mới bằng cách điền các thông tin cơ bản bao gồm họ và tên, địa chỉ email, số điện thoại và mật khẩu. Hệ thống có xác thực dữ liệu đầu vào để đảm bảo email đúng định dạng, mật khẩu đủ độ dài và trường xác nhận mật khẩu khớp với mật khẩu đã nhập. Sau khi đăng ký thành công, người dùng sẽ được chuyển hướng đến trang đăng nhập để truy cập vào hệ thống.



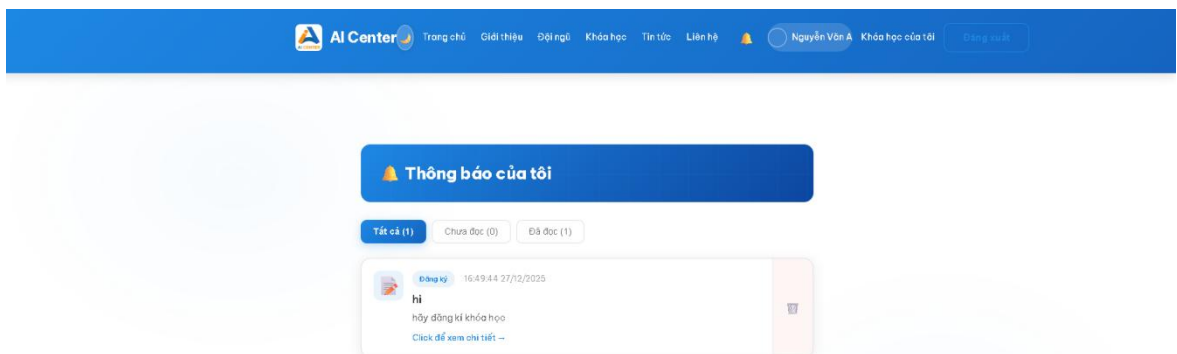
Hình 4.20 Giao diện trang cá nhân

Cho phép người dùng xem và cập nhật thông tin cá nhân bao gồm họ tên, email, số điện thoại. Người dùng có thể thay đổi ảnh đại diện bằng cách click vào avatar và chọn ảnh từ thiết bị. Ngoài ra, trang còn hỗ trợ chức năng đổi mật khẩu với yêu cầu nhập mật khẩu hiện tại và xác nhận mật khẩu mới.



Hình 4.21 Giao diện trang khóa học của tôi

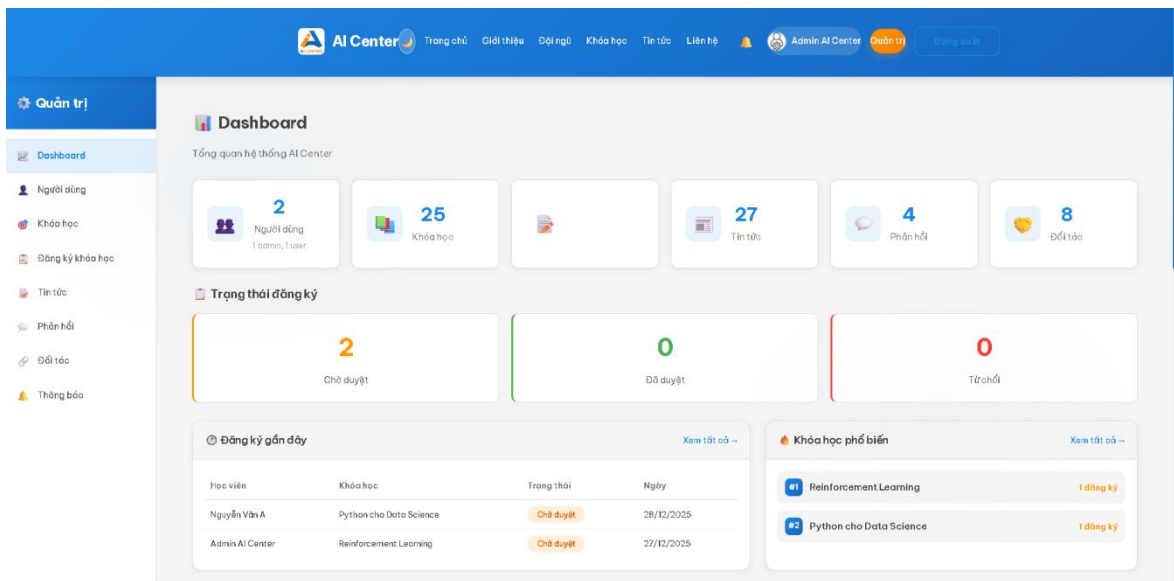
Cho phép người dùng xem danh sách tất cả các khóa học đã đăng ký với thông tin đợt học, giảng viên, học phí và trạng thái xử lý (chờ duyệt, đã duyệt, từ chối). Người dùng có thể theo dõi tiến trình duyệt đăng ký và thực hiện hủy đăng ký đối với các khóa học đang ở trạng thái chờ duyệt nếu thay đổi ý định.



Hình 4.22 Giao diện trang thông báo

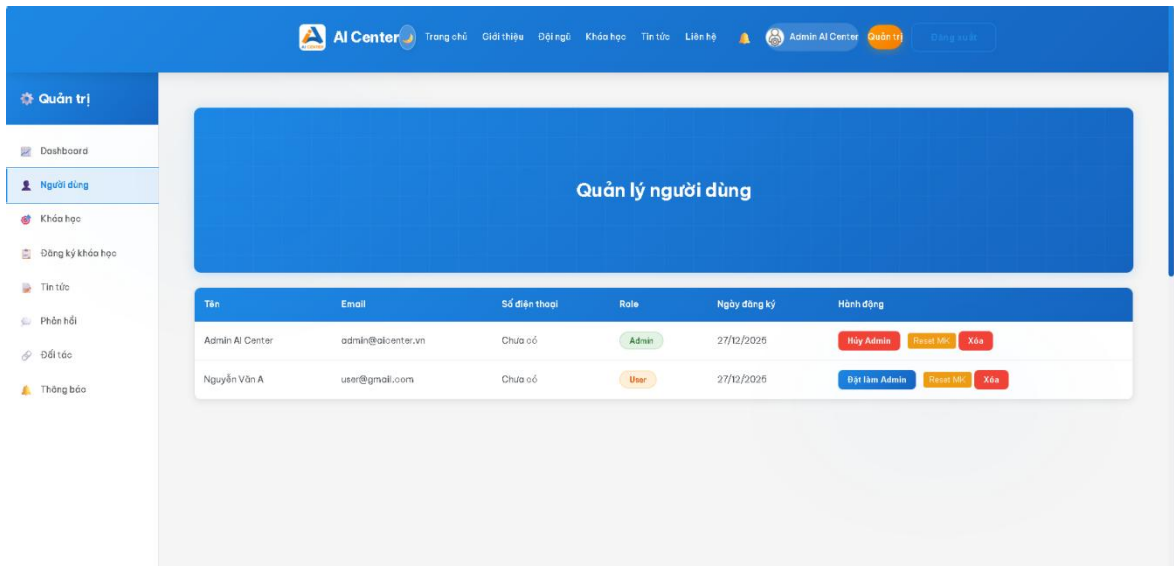
Cho phép người dùng xem danh sách các thông báo từ hệ thống bao gồm thông báo về đăng ký khóa học, cập nhật tin tức và thông báo chung. Hỗ trợ lọc theo trạng thái (tất cả, chưa đọc, đã đọc), đánh dấu tất cả đã đọc và xóa thông báo không cần thiết.

4.2.2 Chức năng trang quản trị



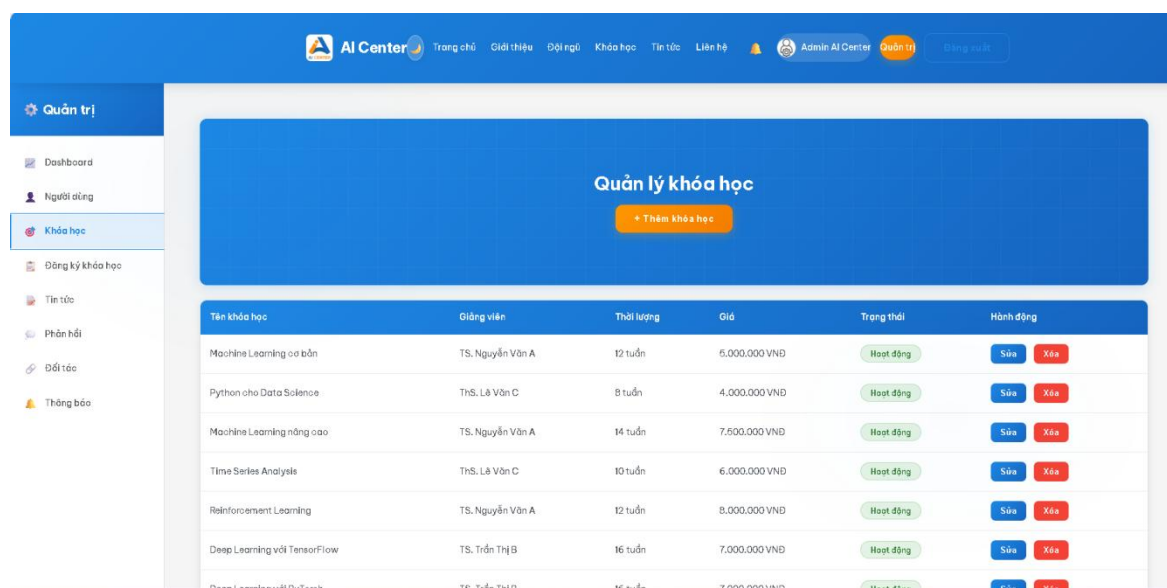
Hình 4.23 Giao diện trang Dashboard

Cho phép quản trị viên xem tổng quan về hoạt động của hệ thống với các số liệu thống kê như tổng số người dùng, khóa học, đăng ký và tin tức. Trang còn hiển thị danh sách các đăng ký gần đây và các khóa học phổ biến để quản trị viên nắm bắt tình hình nhanh chóng.



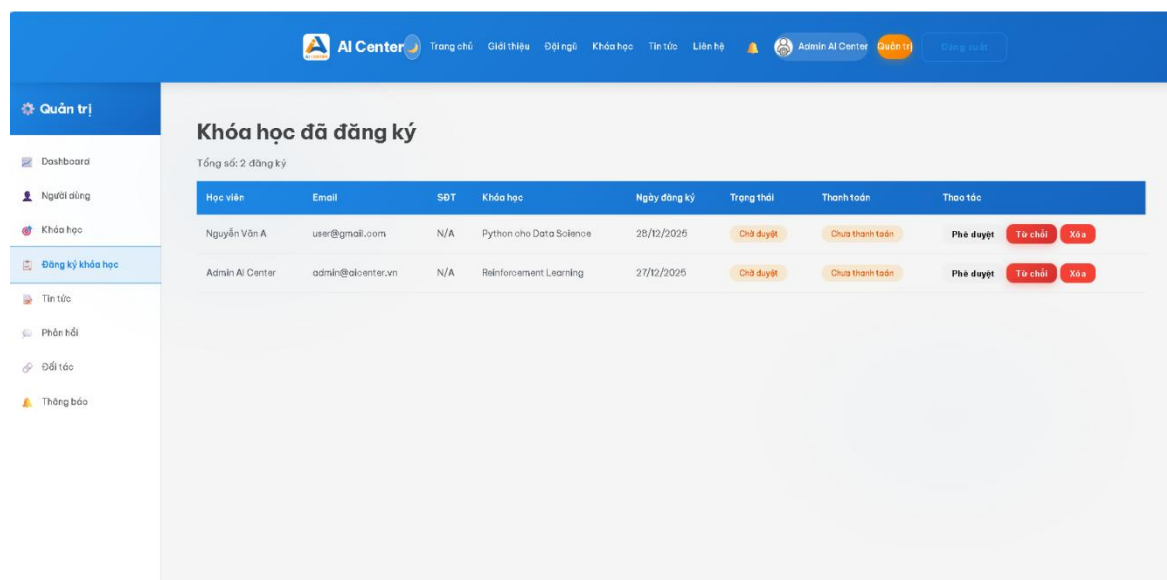
Hình 4.24 Giao diện trang Dashboard

Cho phép quản trị viên xem danh sách tất cả người dùng trong hệ thống với chức năng tìm kiếm và lọc theo vai trò. Hỗ trợ thêm người dùng mới, chỉnh sửa thông tin, xóa tài khoản và phân quyền (user/admin) cho từng người dùng.



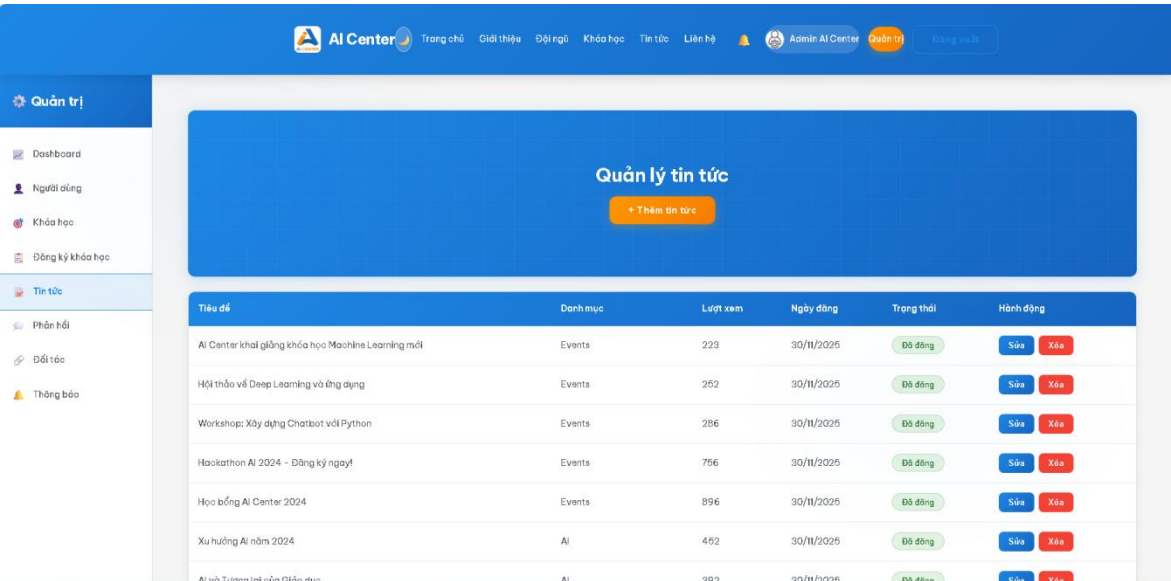
Hình 4.25 Giao diện trang quản lý khóa học

Cho phép quản trị viên xem danh sách khóa học với chức năng tìm kiếm và lọc theo trạng thái. Hỗ trợ thêm khóa học mới với đầy đủ thông tin (tên, mô tả, giảng viên, học phí, hình ảnh), chỉnh sửa và xóa khóa học. Ngoài ra còn quản lý các đợt học cho từng khóa với thông tin ngày bắt đầu, lịch học, địa điểm và sĩ số.



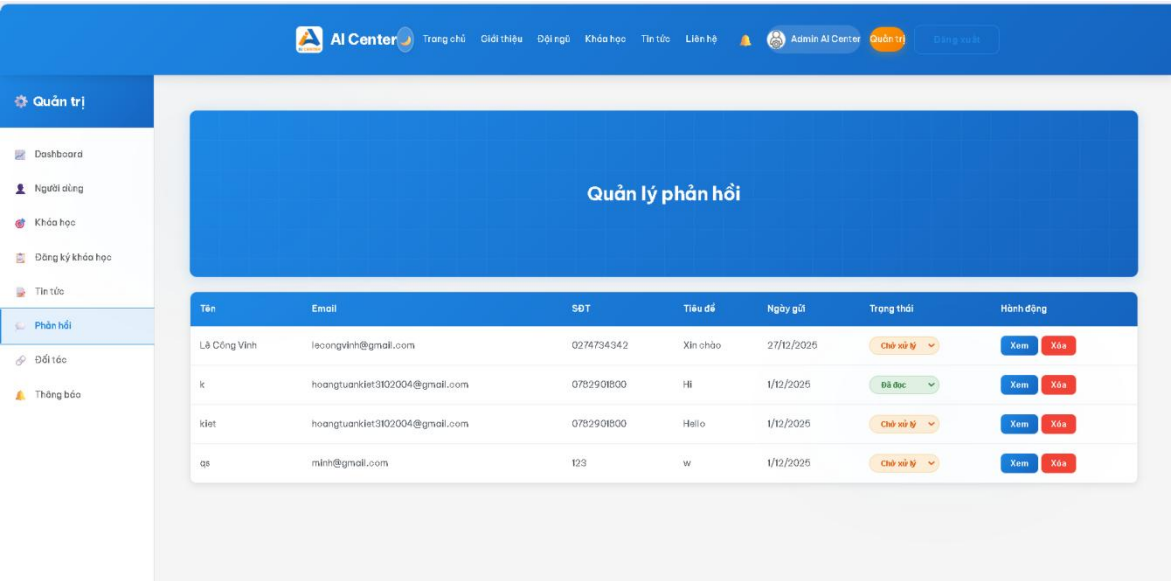
Hình 4.26 Giao diện trang quản lý đăng ký khóa học

Cho phép quản trị viên xem danh sách tất cả đăng ký khóa học từ người dùng với thông tin học viên, khóa học, đợt học và ngày đăng ký. Hỗ trợ lọc theo trạng thái (chờ duyệt, đã duyệt, từ chối) và thực hiện duyệt hoặc từ chối yêu cầu đăng ký.



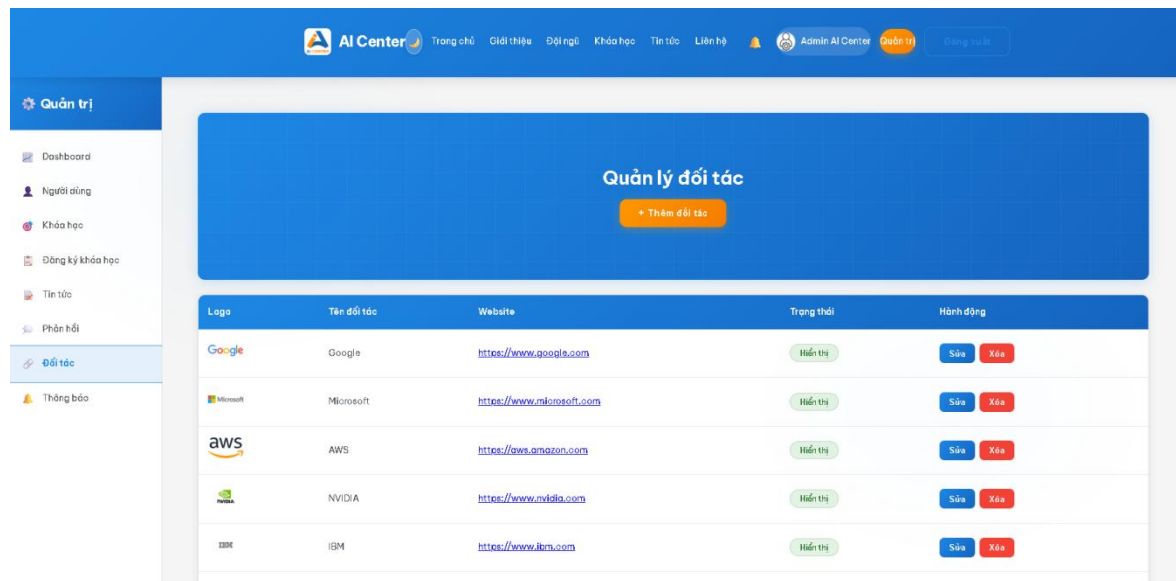
Hình 4.27 Giao diện trang quản lý tin tức

Cho phép quản trị viên xem danh sách bài viết với chức năng tìm kiếm và lọc theo danh mục. Hỗ trợ thêm bài viết mới với tiêu đề, nội dung, hình ảnh và danh mục, chỉnh sửa nội dung và xóa các bài viết không còn cần thiết.



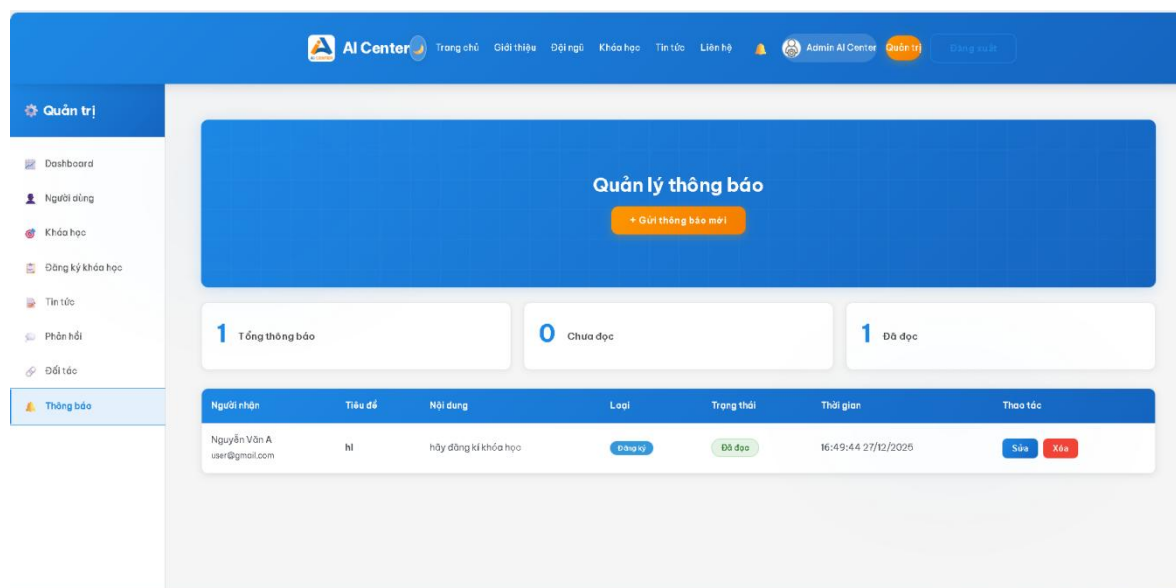
Hình 4.28 Giao diện trang quản lý phản hồi

Cho phép quản trị viên xem danh sách các tin nhắn, góp ý từ người dùng gửi qua trang liên hệ. Hỗ trợ lọc theo trạng thái (chưa đọc, đã đọc), xem chi tiết nội dung phản hồi và xóa các tin nhắn đã xử lý.



Hình 4.29 Giao diện trang quản lý đối tác

Cho phép quản trị viên xem danh sách các đối tác của trung tâm. Hỗ trợ thêm đối tác mới với thông tin tên, logo, website và mô tả, chỉnh sửa thông tin và xóa đối tác không còn hợp tác.



Hình 4.30 Giao diện trang quản lý thông báo

Cho phép quản trị viên xem danh sách các thông báo đã tạo và tạo thông báo mới gửi đến người dùng. Hỗ trợ phân loại thông báo theo loại (hệ thống, khóa học, đăng ký), chọn người nhận và chỉnh sửa hoặc xóa thông báo.

CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1. Kết luận

Đề tài "Xây dựng website cho Trung tâm Trí tuệ Nhân tạo" đã hoàn thành tốt các mục tiêu đề ra. Website được xây dựng như một hệ thống quản lý đào tạo toàn diện, cho phép người dùng dễ dàng tìm hiểu thông tin về các khóa học AI, đội ngũ giảng viên và thực hiện đăng ký học một cách thuận tiện. Giao diện được thiết kế trực quan với tông màu xanh dương và cam hiện đại, thân thiện với người dùng, cùng với logic xử lý rõ ràng và khả năng vận hành ổn định, đáp ứng được những nhu cầu thiết yếu của một website đào tạo chuyên nghiệp.

Thông qua quá trình thực hiện đề tài, nhóm không chỉ củng cố và vận dụng hiệu quả kiến thức về lập trình web với React.js và Node.js, cơ sở dữ liệu MongoDB và phân tích – thiết kế hệ thống, mà còn có cơ hội tiếp cận việc xây dựng một ứng dụng mang tính thực tiễn cao trong lĩnh vực giáo dục công nghệ. Hệ thống bao gồm các chức năng quản lý khóa học, đăng ký học viên, quản lý đợt học, tin tức, thông báo và phản hồi – tạo nên một nền tảng hoàn chỉnh phục vụ hoạt động của trung tâm.

Kết quả đạt được góp phần nâng cao trải nghiệm người dùng, hỗ trợ công tác quản lý đào tạo và tạo tiền đề cho việc phát triển, mở rộng hệ thống trong tương lai như tích hợp học trực tuyến, thanh toán online và chứng chỉ điện tử.

5.2. Hạn chế

Tuy đã đạt được những kết quả nhất định, hệ thống vẫn còn tồn tại một số hạn chế. Các chức năng hiện tại chủ yếu dừng lại ở mức cơ bản, chưa tích hợp các công nghệ hiện đại như hệ thống học trực tuyến (e-learning), thanh toán trực tuyến, cũng như chưa hỗ trợ đề xuất khóa học thông minh dựa trên trình độ và sở thích của học viên bằng trí tuệ nhân tạo. Bên cạnh đó, cơ chế phân quyền người dùng còn đơn giản, chưa đáp ứng tốt cho việc quản lý hệ thống ở quy mô lớn với nhiều vai trò khác nhau.

Trong thời gian tới, hệ thống có thể được cải tiến và mở rộng bằng cách bổ sung các tính năng như học trực tuyến với video bài giảng và bài tập tương tác, tích hợp cổng thanh toán online, cấp chứng chỉ điện tử tự động, đề xuất khóa học thông minh bằng AI và chatbot hỗ trợ tư vấn 24/7. Đồng thời, việc hoàn thiện mô hình phân quyền với các vai trò giảng viên, trợ giảng và phát triển ứng dụng di động sẽ giúp hệ thống đáp ứng tốt hơn nhu cầu thực tế và xu hướng phát triển của các nền tảng đào tạo công nghệ hiện đại.

5.3. Hướng phát triển

Trong tương lai, website có thể tiếp tục được phát triển theo hướng nâng cao trải nghiệm người dùng và ứng dụng các công nghệ hiện đại. Hệ thống có thể bổ sung các chức năng như học trực tuyến với video bài giảng tương tác, thanh toán học phí online qua các cổng thanh toán phổ biến, giúp học viên đăng ký và theo dõi khóa học một cách thuận tiện hơn. Bên cạnh đó, việc tích hợp các thuật toán gợi ý khóa học dựa trên trình độ, sở thích và lịch sử học tập của học viên sẽ góp phần cá nhân hóa lộ trình học tập và tăng tính hấp dẫn của website. Ngoài ra, hệ thống cũng có thể phát triển thêm chatbot AI hỗ trợ tư vấn 24/7, cấp chứng chỉ điện tử tự động và ứng dụng di động để học viên có thể học tập mọi lúc mọi nơi.

TÀI LIỆU THAM KHẢO

- [1] Ian Sommerville, *Software Engineering*, 10th Edition, Pearson Education, 2016.
- [2] Facebook Open Source, React – A JavaScript library for building user interfaces, <https://react.dev>, truy cập năm 2025.
- [3] Node.js Foundation, *Node.js Documentation*, <https://nodejs.org>, truy cập năm 2025.
- [4] MongoDB Inc., *MongoDB Manual*, <https://www.mongodb.com/docs>, truy cập năm 2025.
- [5] JSON Web Token, *Introduction to JSON Web Tokens (JWT)*, <https://jwt.io>, truy cập năm 2025.