

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH
HỌC KỲ 1, NĂM HỌC 2025 - 2026

THIẾT KẾ HỆ THỐNG CHIA SẺ TÀI LIỆU HỌC TẬP VÀ HỎI ĐÁP GIỮA SINH VIÊN

Giáo viên hướng dẫn:

ThS. Hà Thị Thúy Vi

Sinh viên thực hiện:

Họ tên: Hà Gia Lộc

MSSV: 110122103

Lớp: DA22TTD

Vĩnh Long, tháng 12 năm 2025

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ
KHOA CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH
HỌC KỲ 1, NĂM HỌC 2024 - 2025

THIẾT KẾ HỆ THỐNG CHIA SẺ TÀI LIỆU HỌC TẬP VÀ HỎI ĐÁP GIỮA SINH VIÊN

Giáo viên hướng dẫn:

ThS. Hà Thị Thúy Vi

Sinh viên thực hiện:

Họ tên: Hà Gia Lộc

MSSV: 110122103

Lớp: DA22TTD

Vĩnh Long, tháng 12 năm 2025

This image shows a full page of a document template designed for handwritten notes or answers. It features approximately 28 evenly spaced horizontal dotted lines across the entire width of the page, providing a guide for letter height and placement. The background is plain white, and there are no margins, headers, or footers visible.

Giáo viên hướng dẫn
(Ký tên và ghi rõ họ tên)

This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the entire width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the paper.

Giáo viên hướng dẫn
(Ký tên và ghi rõ họ tên)

LỜI CẢM ƠN

Trước khi đi sâu vào dự án, em xin gửi lời cảm ơn chân thành đến trường Đại học Trà Vinh đã tạo điều kiện cho em thực hiện dự án này, những cá nhân đã hỗ trợ và giúp đỡ em một cách tận tình. Cũng như là sự hỗ trợ và giúp đỡ của cô Hà Thị Thúy Vi , người đã đóng vai trò quan trọng trong việc phát triển và hoàn thành dự án này.

Em rất cảm kích vì sự giúp đỡ của những người quanh em, những người đã dành thời gian, công sức và kiến thức của họ để giúp đỡ em. Các ý kiến đóng góp và sự hợp tác của mọi người là nguồn động lực giúp em phát triển bản thân.

Em rất quý trọng những người đã hỗ trợ và giúp đỡ em trong suốt thời gian qua, đặt biệt là sự giúp đỡ của cô Hà Thị Thúy Vi, nhờ có sự giúp đỡ của cô mà em mới có thể thực hiện và hoàn thành dự án.

Một lần nữa, em xin cảm ơn trường Đại học Trà Vinh và cô Hà Thị Thúy Vi đã giúp đỡ và em mong rằng sẽ nhận được sự ủng hộ của mọi người trong những dự án sắp tới.

Trân trọng

Sinh viên thực hiện

Hà Gia Lộc

MỤC LỤC

CHƯƠNG 1. TỔNG QUAN	14
1.1. Giới thiệu đề tài	14
1.2. Mục đích nghiên cứu	14
1.3. Đối tượng nghiên cứu	15
1.4. Phạm vi nghiên cứu	16
1.5. Phương pháp nghiên cứu	17
CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT	18
2.1. Tổng quan thiết kế web	18
2.1.1. Kiến trúc ứng dụng web	18
2.1.2. Lớp giao diện người dùng (Presentation Layer)	19
2.1.3. Lớp xử lý nghiệp vụ (Business Logic Layer)	20
2.1.4. Lớp dữ liệu (Data Layer)	20
2.2. ReactJS	21
2.2.1. Đặc điểm	21
2.2.2. Kiến thức cơ bản về ReactJS	22
2.2.3. Chú thích trong ReactJS	23
2.2.4. Các thành phần trong ReactJS	23
2.2.4.1. Thành phần (Component)	23
2.2.4.2. Props và State	24
2.2.4.3. Xử lý biểu mẫu (Form) trong ReactJS	25
2.2.5. Ứng dụng của ReactJS trong xây dựng website	25
2.2.6. Kết luận	26
2.3. NodeJS	26
2.3.1. Đặc điểm	26

2.3.2. Chú thích trong NodeJS	27
2.3.3. Cấu trúc cơ bản	27
2.3.4. Khai báo biến	28
2.3.5. Câu lệnh điều kiện	28
2.3.6. Kết nối với MySQL	28
2.3.7. Truy vấn dữ liệu.....	29
2.3.8. Xử lý form.....	29
2.3.9. Quản lý lỗi	30
2.3.10. Kết luận.....	31
2.4. Ngôn ngữ MySQL.....	31
2.4.1. Đặc điểm	31
2.4.2. Tạo cơ sở dữ liệu	32
2.4.3. Tạo cấu trúc bảng.....	32
2.4.4. Các câu lệnh truy vấn.....	32
2.4.5. Kết luận.....	34
CHƯƠNG 3. HIỆN THỰC HÓA NGHIÊN CỨU	35
3.1. Mô tả vấn đề	35
3.2. Xác định yêu cầu người dùng.....	35
3.3. Phân loại người dùng	36
3.4. Phân tích thiết kế hệ thống	36
3.4.1. Đặc tả yêu cầu hệ thống	36
3.4.1.1. Yêu cầu chức năng	36
3.4.1.2. Yêu cầu phi chức năng	37
3.4.2. Kiến trúc hệ thống.....	37
3.4.2.1. Lớp giao diện người dùng:	37

3.4.2.2. Lớp xử lý nghiệp vụ:	38
3.4.2.3. Lớp dữ liệu:	38
3.4.3. Thiết kế dữ liệu	39
3.4.3.1. Sơ đồ cơ sở dữ liệu	39
3.4.3.2. Danh sách các thực thể và mối kết hợp	40
3.4.3.3. Chi tiết các thực thể	41
3.4.4. Thiết kế giao diện	48
3.4.4.1. Sơ đồ Use Case	48
3.4.4.2. Sơ đồ website	52
CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU	55
4.1. Dữ liệu thử nghiệm	55
4.2. Kết quả thực nghiệm	60
4.2.1. Trang đăng nhập	60
4.2.2. Trang đăng ký	61
4.2.3. Trang quên mật khẩu	62
4.2.4. Trang danh sách tài liệu	64
4.2.5. Trang hỏi đáp	65
4.2.6. Trang chi tiết câu hỏi	66
4.2.7. Trang chia sẻ tài liệu	67
4.2.8. Trang hồ sơ cá nhân	68
4.2.9. Trang chỉnh sửa hồ sơ	69
4.2.10. Trang đổi mật khẩu	70
4.2.11. Chức năng báo cáo vi phạm	71
4.2.12. Trang dashboard quản trị	71
4.2.13. Trang quản lý tài liệu	72

4.2.14. Trang quản lý câu hỏi	74
4.2.15. Trang quản lý câu trả lời.....	75
4.2.16. Trang quản lý báo cáo vi phạm.....	76
4.2.17. Trang quản lý người dùng.....	76
CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	78
5.1. Kết luận	78
5.2. Hạn chế.....	78
5.3. Hướng phát triển.....	79
DANH MỤC TÀI LIỆU THAM KHẢO	81

DANH MỤC HÌNH ẢNH

Hình 3.1 Sơ đồ cơ sở dữ liệu	39
Hình 3.2 Sơ đồ Use Case về trang người dùng	48
Hình 3.3 Sơ đồ Use Case về trang quản trị.....	50
Hình 3.4 Sơ đồ website.....	52
Hình 4.1 Dữ liệu của bảng sinhvien	58
Hình 4.2 Dữ liệu của bảng tailieu	58
Hình 4.3 Dữ liệu của bảng cautraloi	59
Hình 4.4 Dữ liệu của bảng baocaovipham.....	59
Hình 4.5 Dữ liệu của bảng cauhoi	60
Hình 4.6 Giao diện đăng nhập	60
Hình 4.7 Giao diện đăng ký	61
Hình 4.8 Giao diện nhập email	62
Hình 4.9 Giao diện đặt lại mật khẩu	63
Hình 4.10 Giao diện trang danh sách tài liệu	64
Hình 4.11 Giao diện trang hỏi đáp	65
Hình 4.12 Giao diện chi tiết câu hỏi và các câu trả lời.....	66
Hình 4.13 Giao diện nhập câu trả lời.....	66
Hình 4.14 Giao diện trang chia sẻ tài liệu	67
Hình 4.15 Giao diện trang hồ sơ cá nhân	68
Hình 4.16 Giao diện trang chỉnh sửa hồ sơ	69
Hình 4.17 Giao diện trang đổi mật khẩu	70
Hình 4.18 Giao diện các form báo cáo vi phạm	71
Hình 4.19 Giao diện dashboard quản trị.....	72
Hình 4.20 Giao diện quản lý tài liệu.....	72

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Hình 4.21 Giao diện sửa tài liệu	73
Hình 4.22 Giao diện trang quản lý câu hỏi.....	74
Hình 4.23 Giao diện chức năng sửa tài liệu.....	74
Hình 4.24 Giao diện trang quản lý câu trả lời	75
Hình 4.25 Giao diện chức năng sửa câu trả lời	75
Hình 4.26 Giao diện quản lý báo cáo vi phạm	76
Hình 4.27 Giao diện trang quản lý người dùng	77
Hình 4.28 Giao diện sửa thông tin sinh viên	77

DANH MỤC BẢNG BIỂU

Bảng 3.1 Bảng thực thể	40
Bảng 3.2 Bảng thuộc tính của thực thể SinhVien.....	41
Bảng 3.3 Bảng thuộc tính của thực thể QuanTriVien	41
Bảng 3.4 Bảng thuộc tính của thực thể Ngành	42
Bảng 3.5 Bảng thuộc tính của thực thể Môn	42
Bảng 3.6 Bảng thuộc tính của thực thể LoaiTaiLieu.....	42
Bảng 3.7 Bảng thuộc tính của thực thể TaiLieu	43
Bảng 3.8 Bảng thuộc tính của thực thể DinhDang	43
Bảng 3.9 Bảng thuộc tính của thực thể DanhSachTaiLieu.....	44
Bảng 3.10 Bảng thuộc tính của thực thể CauHoi	44
Bảng 3.11 Bảng thuộc tính của thực thể CauTraLoi	45
Bảng 3.12 Bảng thuộc tính của thực thể DanhGiaCauHoi.....	45
Bảng 3.13 Bảng thuộc tính của thực thể DanhGiaCauTraLoi.....	46
Bảng 3.14 Bảng thuộc tính của thực thể Tag	46
Bảng 3.15 Bảng thuộc tính của thực thể DanhSachTags	46
Bảng 3.16 Bảng thuộc tính của thực thể BaoCaoViPham.....	47
Bảng 3.17 Bảng mô tả Use Case trang người dùng.....	49
Bảng 3.18 Bảng mô tả Use Case trang quản trị.....	51
Bảng 4.1 Dữ liệu của bảng dinh dang.....	55
Bảng 4.2 Dữ liệu của bảng luutailieu	55
Bảng 4.3 Dữ liệu của bảng ngành.....	55
Bảng 4.4 Dữ liệu của bảng quantri vien	55
Bảng 4.5 Dữ liệu của bảng tag.....	55
Bảng 4.6 Dữ liệu của bảng loaitailieu	56

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Bảng 4.7 Dữ liệu của bảng mon	56
Bảng 4.8 Dữ liệu của bảng danhgiacauhoi	56
Bảng 4.9 Dữ liệu của bảng danhgiacautraloi.....	57
Bảng 4.10 Dữ liệu của bảng danhsachtag.....	57
Bảng 4.11 Dữ liệu của bảng danhsachtailieu	57

Vấn đề cần nghiên cứu

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên.

Các hướng tiếp cận

Tiến hành tìm hiểu một số trang web hoặc ứng dụng web về dịch vụ, tìm hiểu xem cách họ xây dựng một trang web hoặc ứng dụng web, các thành phần phải có, các nền tảng để thiết kế và đưa nó vào thực tiễn.

Cách giải quyết vấn đề

Tiến hành tìm hiểu các ứng dụng, phần mềm và ngôn ngữ lập trình để thiết kế ứng dụng web, những phần cần có của một ứng dụng web và sử dụng NodeJS, ReactJS và MySQL để tạo nền tảng cơ bản cho ứng dụng web, tiến hành sửa lỗi có thể có trong quá trình nghiên cứu, kiểm nghiệm và nâng cấp theo thời gian.

Kết quả đạt được

Tạo ra một hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên với đầy đủ chức năng cơ bản như hiển thị danh sách tài liệu, xem chi tiết tài liệu, tìm kiếm tài liệu, tải lên tài liệu, hiển thị câu hỏi, đặt câu hỏi, trả lời câu hỏi.

MỞ ĐẦU

Lý do chọn đề tài

Trong quá trình học tập, sinh viên thường gặp khó khăn trong việc tìm kiếm tài liệu phù hợp và thiếu một nền tảng tập trung để trao đổi học thuật. Tài liệu học tập hiện nay còn rời rạc, khó kiểm chứng và việc hỏi đáp thường diễn ra trên các nền tảng không chuyên biệt. Vì vậy, đề tài “Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên” được lựa chọn nhằm xây dựng một môi trường hỗ trợ sinh viên chia sẻ tài liệu, trao đổi kiến thức và nâng cao hiệu quả học tập.

Mục đích nghiên cứu

Mục đích của đề tài là xây dựng một hệ thống web cho phép sinh viên chia sẻ tài liệu, tìm kiếm nội dung học tập và hỏi đáp theo chủ đề. Hệ thống hướng đến việc tạo ra một nền tảng trực quan, dễ sử dụng và đáp ứng đầy đủ các chức năng cơ bản như quản lý tài liệu, hỏi đáp, tìm kiếm và đánh giá, đồng thời ứng dụng các công nghệ hiện đại như ReactJS, NodeJS và MySQL.

Đối tượng nghiên cứu

Đối tượng nghiên cứu bao gồm hệ thống chia sẻ tài liệu và hỏi đáp dành cho sinh viên, hướng đến việc quản lý thông tin tài liệu, câu hỏi, câu trả lời và người dùng. Bên cạnh đó, đề tài tập trung vào hoạt động của các nhóm người dùng như sinh viên, khách truy cập và quản trị viên nhằm bảo đảm hệ thống vận hành đúng mục tiêu.

Phạm vi nghiên cứu

Phạm vi của đề tài giới hạn trong việc xây dựng hệ thống web với các chức năng cơ bản như chia sẻ tài liệu, tìm kiếm, hỏi đáp và quản trị. Hệ thống được triển khai bằng Docker, sử dụng ReactJS cho frontend, NodeJS cho backend và MySQL cho quản lý dữ liệu. Đề tài không mở rộng sang phát triển ứng dụng di động hoặc tích hợp các tính năng nâng cao ngoài phạm vi đồ án.

CHƯƠNG 1. TỔNG QUAN

1.1. Giới thiệu đề tài

Trong bối cảnh giáo dục đại học ngày càng chú trọng vào việc học tập chủ động và hợp tác, nhu cầu tìm kiếm tài liệu học tập đáng tin cậy cũng như trao đổi kiến thức giữa sinh viên ngày càng trở nên cấp thiết. Tuy nhiên, việc chia sẻ tài liệu hiện nay vẫn còn rời rạc, thiếu tính hệ thống và khó kiểm chứng chất lượng. Điều này gây khó khăn cho sinh viên trong quá trình tự học và ôn tập. Việc xây dựng một hệ thống web hỗ trợ chia sẻ tài liệu học tập và đặt câu hỏi – trả lời giữa sinh viên sẽ góp phần giải quyết những hạn chế trên, đồng thời tạo môi trường học tập cộng tác, hiệu quả và linh hoạt hơn.

1.2. Mục đích nghiên cứu

Mục đích của đề tài này là tập trung vào việc xây dựng và phát triển một hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên. Hệ thống cần đáp ứng hai tiêu chí quan trọng: khả năng cập nhật nhanh chóng và độ chính xác của thông tin. Ứng dụng không chỉ giúp sinh viên dễ dàng trao đổi tài liệu học tập mà còn hỗ trợ đặt câu hỏi, thảo luận và nhận phản hồi từ cộng đồng sinh viên một cách thuận tiện và hiệu quả. Cụ thể:

- **Tạo công cụ quản lý tài liệu hiệu quả:** Phát triển cơ sở dữ liệu hỗ trợ lưu trữ, truy xuất và quản lý các loại tài liệu học tập như bài giảng, bài tập, đề thi, tài liệu tham khảo, đồng thời tích hợp chức năng phân loại theo môn học, ngành học và đánh giá từ người dùng.

- **Tạo công cụ quản lý tài liệu hiệu quả:** Phát triển cơ sở dữ liệu hỗ trợ lưu trữ, truy xuất và quản lý các loại tài liệu học tập như bài giảng, bài tập, đề thi, tài liệu tham khảo, đồng thời tích hợp chức năng phân loại theo môn học, ngành học và đánh giá từ người dùng.

- **Cải thiện trải nghiệm người dùng:** Thiết kế giao diện thân thiện, rõ ràng, giúp sinh viên nhanh chóng tìm kiếm tài liệu hoặc câu hỏi liên quan thông qua các công cụ lọc, gợi ý và tìm kiếm nâng cao.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Nghiên cứu này sử dụng ReactJS để xây dựng frontend, NodeJS (ExpressJS) để xây dựng backend, MySQL để thiết kế và quản lý cơ sở dữ liệu. Sự kết hợp này góp phần tạo nên một ứng dụng web trực quan, linh hoạt và đáp ứng tốt nhu cầu học tập của sinh viên.

1.3. Đối tượng nghiên cứu

Dự án hướng đến việc xây dựng một hệ thống web hỗ trợ sinh viên chia sẻ tài liệu học tập và đặt câu hỏi – trả lời trong môi trường học thuật. Hệ thống cung cấp các chức năng quản lý tài liệu, tìm kiếm tài liệu theo môn học hoặc chủ đề, đặt câu hỏi, trả lời câu hỏi và đánh giá mức độ hữu ích của các câu trả lời. Bên cạnh đó, hệ thống được thiết kế với giao diện hiện đại, thân thiện và dễ sử dụng nhằm hỗ trợ sinh viên truy cập, trao đổi và tiếp cận tri thức một cách thuận tiện nhất. Cụ thể:

Ngôn ngữ thiết kế và phát triển web: Sử dụng ReactJS để xây dựng giao diện người dùng, xử lý hiển thị, định tuyến và các tương tác động. NodeJS và ExpressJS để xây dựng API backend, xử lý logic nghiệp vụ và kết nối với cơ sở dữ liệu. MySQL để thiết kế, lưu trữ và quản lý dữ liệu người dùng, tài liệu, câu hỏi và các tương tác đánh giá. API được tổ chức theo mô hình RESTful, đảm bảo tính mở rộng và dễ dàng bảo trì.

Quy trình quản lý dữ liệu và xử lý thông tin: Hệ thống hỗ trợ các thao tác tạo, sửa, xóa và truy vấn tài liệu hoặc câu hỏi trong cơ sở dữ liệu. Backend xử lý yêu cầu từ giao diện ReactJS thông qua API, đồng thời đảm bảo tính toàn vẹn và tốc độ truy vấn. Quy trình truyền tải dữ liệu giữa frontend và backend được tối ưu nhằm mang đến trải nghiệm nhanh chóng và mượt mà.

Việc nghiên cứu và triển khai các thành phần công nghệ trên giúp đảm bảo hệ thống hoạt động ổn định, đáp ứng đầy đủ nhu cầu chia sẻ tài liệu và hỏi đáp học thuật của sinh viên, đồng thời mang lại trải nghiệm sử dụng hiệu quả và linh hoạt.

1.4. Phạm vi nghiên cứu

Các ngôn ngữ được dùng:

- Sử dụng ReactJS (kết hợp HTML, CSS và JavaScript) để xây dựng giao diện người dùng hiện đại, thân thiện và hỗ trợ tương tác động, cập nhật nội dung theo thời gian thực.
- Sử dụng NodeJS (ExpressJS) để xây dựng backend, xử lý logic nghiệp vụ, kiểm tra dữ liệu và quản lý yêu cầu từ phía người dùng.
- Sử dụng MySQL để thiết kế và quản trị cơ sở dữ liệu lưu trữ thông tin người dùng, tài liệu, câu hỏi, câu trả lời và các đánh giá.

Chức năng ứng dụng:

- Chức năng chia sẻ tài liệu học tập cho phép sinh viên đăng tải, xem và tải về tài liệu.
- Chức năng tìm kiếm tài liệu theo môn học, ngành tiêu đề hoặc từ khóa.
- Chức năng đặt câu hỏi học thuật hỗ trợ sinh viên thảo luận, đặt thắc mắc và trao đổi kiến thức.
- Chức năng trả lời câu hỏi giúp sinh viên hỗ trợ lẫn nhau.
- Chức năng đánh giá tài liệu và đánh giá câu trả lời giúp ưu tiên nội dung chất lượng.
- Quản lý tài khoản người dùng bao gồm đăng ký, đăng nhập và chỉnh sửa thông tin cá nhân.

Quy mô dự án:

- Ứng dụng tập trung vào việc hỗ trợ sinh viên chia sẻ tài liệu và hỏi đáp học thuật.
- Hệ thống được thiết kế để phục vụ sinh viên, giảng viên hoặc người dùng nội bộ trong môi trường học tập.
- Các chức năng được xây dựng ở mức phù hợp cho một hệ thống web đơn giản nhưng đủ hoàn chỉnh để sử dụng thực tế.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Phạm vi nghiên cứu tập trung vào giao diện người dùng, xử lý logic phía máy chủ, thiết kế cơ sở dữ liệu và sự tương tác giữa các thành phần. Việc giới hạn phạm vi giúp tối ưu thời gian và nguồn lực, đảm bảo hệ thống hoạt động hiệu quả và đáp ứng nhu cầu chia sẻ tài liệu và hỏi đáp của sinh viên.

1.5. Phương pháp nghiên cứu

- **Phương pháp nghiên cứu:** Tìm hiểu tài liệu, giáo trình và các nguồn trực tuyến về Node.js, MySQL và các công nghệ giao diện như HTML, CSS, JavaScript, ReactJS để hiểu rõ và nắm vững các công nghệ sử dụng trong hệ thống.

- **Phương pháp phân tích và thiết kế:** Sử dụng sơ đồ Use Case, mô hình phân tích hướng chức năng và mô hình ERD để xác định các chức năng, luồng xử lý và mối quan hệ giữa các bảng dữ liệu.

- **Phương pháp xây dựng hệ thống:** Phát triển backend bằng Node.js để xây dựng logic nghiệp vụ theo kiến trúc mô-đun, frontend bằng HTML, CSS, JavaScript, React. Cơ sở dữ liệu được quản lý bằng MySQL.

- **Phương pháp kiểm thử:** Tiến hành kiểm thử chức năng, tích hợp và giao diện nhằm đảm bảo hệ thống hoạt động đúng yêu cầu và ổn định.

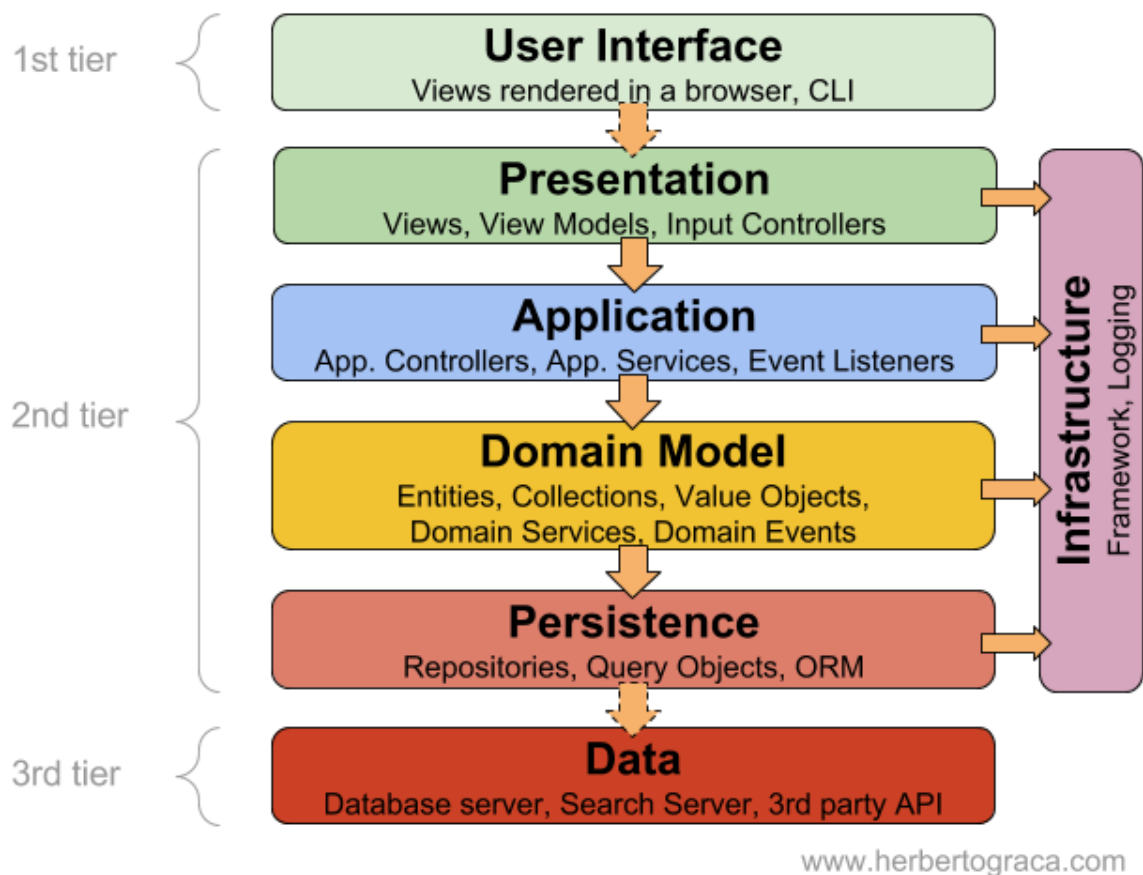
- **Phương pháp triển khai:** Cài đặt và chạy thử hệ thống trên môi trường localhost, sau đó triển khai bằng Docker.

CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT

2.1. Tổng quan thiết kế web

2.1.1. Kiến trúc ứng dụng web

Ứng dụng web trong đề tài được thiết kế dựa trên kiến trúc phân lớp (Layered Architecture). Đây là mô hình kiến trúc phổ biến trong phát triển ứng dụng web hiện đại, cho phép tách biệt rõ ràng các thành phần theo chức năng, giúp hệ thống dễ phát triển, mở rộng và bảo trì. [1]



Hình 2.1 Kiến trúc phân lớp (Layered Architecture)

Kiến trúc phân lớp chia hệ thống thành ba lớp chính:

Lớp giao diện người dùng (Presentation Layer) chịu trách nhiệm hiển thị dữ liệu và xử lý các tương tác trực tiếp với người dùng; mọi thao tác như nhập thông tin, tìm kiếm hay xem phòng trọ đều được lớp này tiếp nhận và gửi về backend thông qua API.

Lớp xử lý nghiệp vụ (Business Logic Layer) đóng vai trò trung tâm, nơi thực hiện các quy tắc nghiệp vụ, xử lý yêu cầu từ frontend, kiểm tra dữ liệu, xác thực người dùng và điều phối giao tiếp với cơ sở dữ liệu.

Lớp dữ liệu (Data Layer) đảm nhận việc lưu trữ, truy vấn và quản lý các bảng dữ liệu trong hệ quản trị MySQL, bảo đảm tính toàn vẹn và nhất quán của thông tin. Ba lớp kết hợp với nhau giúp hệ thống hoạt động ổn định, dễ mở rộng và dễ bảo trì.

2.1.2. Lớp giao diện người dùng (Presentation Layer)

Lớp giao diện người dùng (Presentation Layer) là tầng chịu trách nhiệm hiển thị dữ liệu và xử lý toàn bộ các tương tác trực tiếp giữa người dùng và hệ thống. Trong đề tài này, lớp giao diện được phát triển bằng ReactJS, một thư viện JavaScript hiện đại hỗ trợ xây dựng giao diện theo hướng thành phần (component-based), giúp tăng khả năng tái sử dụng, cải thiện hiệu suất và tối ưu trải nghiệm người dùng.

Tầng này đảm nhiệm các chức năng chính như: hiển thị thông tin phòng trọ, xử lý thao tác tìm kiếm, tiếp nhận dữ liệu từ biểu mẫu (đăng nhập, đăng ký...) và gửi yêu cầu đến backend thông qua API. Các công nghệ được sử dụng bao gồm:

- **ReactJS:** Dùng để xây dựng cấu trúc giao diện và quản lý trạng thái. ReactJS hỗ trợ cập nhật dữ liệu theo thời gian thực, xử lý điều hướng và mang lại trải nghiệm tương tác mượt mà.

- **HTML & CSS:** Kết hợp trong các thành phần React để tạo bố cục, định dạng nội dung và đảm bảo tính thẩm mỹ của giao diện. CSS giúp giao diện thích ứng tốt trên nhiều thiết bị như máy tính, điện thoại và máy tính bảng.

- **JavaScript:** Được sử dụng để triển khai các logic phía client, xử lý sự kiện, gọi API, cập nhật trạng thái và tạo ra những tương tác động trong ứng dụng.

Với sự kết hợp của các công nghệ web hiện đại, lớp giao diện người dùng bảo đảm khả năng phản hồi nhanh, hiển thị trực quan và mang lại trải nghiệm liền mạch khi người dùng thực hiện tra cứu, xem thông tin phòng trọ hoặc tương tác với các chức năng khác của hệ thống.

2.1.3. Lớp xử lý nghiệp vụ (Business Logic Layer)

Lớp xử lý nghiệp vụ (Business Logic Layer) là tầng trung tâm của hệ thống, nơi thực thi các quy tắc nghiệp vụ, xử lý yêu cầu từ giao diện người dùng và điều phối quá trình truy xuất dữ liệu. Trong đề tài này, lớp nghiệp vụ được phát triển bằng NodeJS kết hợp với ExpressJS, cho phép xây dựng các API nhẹ, linh hoạt và dễ mở rộng.

Backend đảm nhiệm việc tiếp nhận yêu cầu từ frontend, kiểm tra và xử lý dữ liệu, thực hiện xác thực người dùng, áp dụng các quy tắc nghiệp vụ và tương tác với cơ sở dữ liệu. Các công nghệ chính được sử dụng gồm:

- **NodeJS:** Nền tảng chạy JavaScript phía server, hỗ trợ xử lý bất đồng bộ, phù hợp với các ứng dụng web cần tốc độ phản hồi nhanh và khả năng mở rộng. NodeJS thực hiện các tác vụ như xử lý logic nghiệp vụ, quản lý phiên, kiểm tra dữ liệu và điều hướng yêu cầu.

- **ExpressJS:** Framework nhẹ cho NodeJS, cung cấp cấu trúc rõ ràng để xây dựng API theo mô hình RESTful. ExpressJS hỗ trợ định tuyến (routing), xử lý middleware, quản lý request/response và kết nối với cơ sở dữ liệu.

Thông qua sự kết hợp của NodeJS và ExpressJS, lớp nghiệp vụ đảm bảo xử lý yêu cầu hiệu quả, duy trì tính nhất quán dữ liệu và tạo cầu nối giữa frontend và cơ sở dữ liệu một cách nhanh chóng, ổn định và dễ bảo trì.

2.1.4. Lớp dữ liệu (Data Layer)

Lớp dữ liệu (Data Layer) là tầng chịu trách nhiệm lưu trữ, quản lý và truy vấn toàn bộ thông tin của hệ thống. Trong đề tài này, tầng dữ liệu được xây dựng bằng MySQL, một hệ quản trị cơ sở dữ liệu quan hệ mạnh mẽ, hỗ trợ tốt cho các hệ thống web cần quản lý dữ liệu có cấu trúc và đảm bảo tính toàn vẹn.

Tầng dữ liệu thực hiện các chức năng chính như: lưu trữ thông tin người dùng, tài liệu học tập, câu hỏi, câu trả lời, đánh giá, phân loại theo môn học và ngành cũng như quản lý các tương tác của người dùng đối với tài liệu và nội dung hỏi đáp. Backend (NodeJS – ExpressJS) kết nối với MySQL thông qua các truy vấn SQL để xử lý dữ liệu theo yêu cầu của người dùng.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Một số đặc điểm nổi bật của lớp dữ liệu gồm:

MySQL hỗ trợ thiết kế dữ liệu theo mô hình quan hệ, cho phép xây dựng các bảng có khóa chính, khóa ngoại để đảm bảo tính liên kết giữa người dùng – tài liệu – câu hỏi – câu trả lời – đánh giá.

Cơ sở dữ liệu được thiết kế dựa trên mô hình ERD của hệ thống, giúp tối ưu hóa việc truy vấn và tránh trùng lặp dữ liệu.

Tầng backend sử dụng các thư viện như mysql2 hoặc ORM như Sequelize để giao tiếp với MySQL, đảm bảo quá trình thêm, sửa, xóa và truy vấn dữ liệu được thực hiện nhanh chóng và an toàn.

Cấu trúc dữ liệu hỗ trợ đầy đủ các chức năng của đề tài, bao gồm chia sẻ tài liệu, bình luận, đánh giá, đặt câu hỏi, trả lời và phân loại nội dung theo môn học hoặc chủ đề.

Nhờ sử dụng MySQL làm nền tảng lưu trữ, lớp dữ liệu đảm bảo hệ thống hoạt động ổn định, dữ liệu nhất quán và đáp ứng tốt nhu cầu truy cập, chia sẻ thông tin học tập và hỏi đáp giữa sinh viên.

2.2. ReactJS

ReactJS là một thư viện JavaScript mã nguồn mở do Facebook phát triển, được sử dụng để xây dựng giao diện người dùng cho các ứng dụng web hiện đại. ReactJS cho phép phát triển UI theo hướng thành phần (component-based), giúp tối ưu khả năng tái sử dụng mã nguồn, tăng tính linh hoạt và cải thiện hiệu suất hiển thị. Trong hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên, ReactJS đóng vai trò quan trọng trong việc tạo ra giao diện trực quan, phản hồi nhanh và dễ mở rộng.

2.2.1. Đặc điểm

ReactJS được đánh giá là một trong những thư viện frontend phổ biến nhất nhờ sở hữu nhiều ưu điểm nổi bật. Trước hết, ReactJS phát triển giao diện theo hướng thành phần (component-based), cho phép chia nhỏ giao diện thành các đơn vị độc lập, dễ tái sử dụng và đơn giản hóa quá trình bảo trì. Bên cạnh đó, ReactJS sử dụng cơ chế Virtual DOM giúp tối ưu hóa việc cập nhật giao diện: chỉ những thành phần thực sự thay đổi mới được render lại, từ đó cải thiện đáng kể hiệu suất so với

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

cách thao tác trực tiếp trên DOM truyền thống. Thư viện này cũng áp dụng mô hình truyền dữ liệu một chiều (one-way data binding), giúp luồng dữ liệu trở nên rõ ràng, dễ kiểm soát và hạn chế xung đột trạng thái. Ngoài ra, ReactJS hỗ trợ JSX – một phần mở rộng cú pháp cho phép viết giao diện bằng mã giống HTML ngay trong JavaScript, giúp mã nguồn trực quan, dễ đọc và dễ quản lý hơn. Nhờ sự kết hợp của các đặc điểm trên, ReactJS mang lại tốc độ phản hồi nhanh và hiệu quả cao, đặc biệt phù hợp với các ứng dụng web có yêu cầu cập nhật dữ liệu liên tục. [2]

2.2.2. Kiến thức cơ bản về ReactJS

Trong ReactJS, ứng dụng được xây dựng dựa trên các thành phần (components). Một component đơn giản có thể được định nghĩa như sau: [3]

```
import { createRoot } from 'react-dom/client';

function Hello() {
  return (
    <h1>Hello World!</h1>
  );
}

createRoot(document.getElementById('root')).render(
  <Hello />
);
```

Cấu trúc cơ bản của một ứng dụng React trong ví dụ này gồm:

- index.html: Tập HTML gốc chứa phần tử <div id="root"></div>, đây là nơi React sẽ gắn nội dung giao diện được render từ JavaScript.
- index.js: Tập khởi chạy ứng dụng, nơi sử dụng createRoot() để lấy phần tử có id "root" và render component <Hello /> lên giao diện.
- Hello.js (hoặc được khai báo trực tiếp trong index.js như ví dụ): Thành phần (component) React đơn giản, trả về nội dung <h1>Hello World!</h1> để hiển thị lên trình duyệt.
- Các component khác (nếu có): Có thể được mở rộng thêm như Header, Footer, giao diện câu hỏi, danh sách tài liệu hoặc các thành phần chức năng khác tùy nhu cầu ứng dụng.

2.2.3. Chú thích trong ReactJS

Trong ReactJS, chú thích được viết trong JavaScript bằng các cú pháp: [3]

- Chú thích một dòng: `// nội dung`
- Chú thích nhiều dòng: `/* nội dung */`
- Chú thích trong JSX: `{/* Đây là chú thích trong JSX */}`

2.2.4. Các thành phần trong ReactJS

2.2.4.1. Thành phần (Component)

Component trong ReactJS dùng để chia giao diện thành các phần độc lập, mỗi phần tự quản lý hiển thị và logic của mình. Chúng giúp ứng dụng dễ bảo trì và tái sử dụng. Hiện nay, ReactJS hỗ trợ hai kiểu component chính: [2] [3]

- **Functional Component (phổ biến nhất):** Là dạng component được xây dựng dưới dạng hàm, có cú pháp ngắn gọn và hiệu suất tốt. Nhờ sự hỗ trợ của React Hooks, Functional Component có thể quản lý trạng thái và xử lý vòng đời, trở thành lựa chọn tiêu chuẩn trong các ứng dụng React hiện đại. Ví dụ Functional Component:

```
function Car(props) {  
  return (  
    <h2>I am a {props.color} Car!</h2>  
  );  
}  
createRoot(document.getElementById('root')).render(  
  <Car color="red"/>  
)
```

- **Class Component (ít dùng hơn):** Được xây dựng dựa trên cú pháp lớp của JavaScript, hỗ trợ state và các phương thức vòng đời truyền thống. Mặc dù ít được sử dụng trong các dự án mới do cú pháp phức tạp hơn, Class Component vẫn quan trọng trong việc duy trì và hiểu các ứng dụng React phiên bản cũ. Ví dụ Class Component:

```
class Car extends React.Component {
```

```
render() {  
  return <h2>Hi, I am a Car!</h2>;  
}  
}
```

2.2.4.2. Props và State

Trong ReactJS, *props* và *state* là hai khái niệm quan trọng dùng để quản lý dữ liệu và điều khiển cách component hoạt động. [2] [3]

- **Props** (viết tắt của “properties”) là dữ liệu được truyền từ component cha sang component con, giúp component con có thể hiển thị nội dung hoặc hành vi theo giá trị được cung cấp. Props có tính chất *bất biến* (immutable), nghĩa là component con không được phép thay đổi giá trị props nhận vào. Ví dụ một component nhận props và hiển thị lời chào như sau:

```
function Car(props) {  
  return (  
    <h2>I am a {props.color} Car!</h2>  
  );  
}  
createRoot(document.getElementById('root')).render(  
  <Car color="red"/>  
)
```

- **State** là dữ liệu nội tại của component, có thể thay đổi theo thời gian thông qua các thao tác của người dùng hoặc các sự kiện trong hệ thống. State giúp giao diện có khả năng phản hồi và cập nhật động. Khi state thay đổi, component sẽ được render lại để hiển thị dữ liệu mới. Dưới đây là ví dụ quản lý state trong Functional Component bằng React Hook `useState`:

```
class Car extends React.Component {  
  constructor(props) {  
    super(props);  
    this.state = {brand: "Ford"};  
  }  
  render() {  
    return (  

```

```
<div>
  <h1>My Car</h1>
</div>

);
}
```

Props và state đóng vai trò trọng tâm trong cơ chế hoạt động của ReactJS, giúp component vừa có khả năng nhận dữ liệu từ bên ngoài, vừa tự quản lý và cập nhật thông tin theo các tương tác của người dùng.

2.2.4.3. Xử lý biểu mẫu (Form) trong ReactJS

Trong React, dữ liệu của form được quản lý bằng state. Code ví dụ: [2] [3]

```
function MyForm() {
  return (
    <form>
      <label>Enter your name:
        <input type="text" />
      </label>
    </form>
  )
}

createRoot(document.getElementById('root')).render(
  <MyForm />
);
```

2.2.5. Ứng dụng của ReactJS trong xây dựng website

ReactJS được sử dụng để xây dựng giao diện người dùng một cách linh hoạt và hiệu quả trong các ứng dụng web hiện đại. Một số ứng dụng quan trọng của ReactJS bao gồm:

- **Xây dựng giao diện người dùng linh hoạt:** ReactJS cho phép chia giao diện thành các component độc lập, dễ tái sử dụng và dễ bảo trì.

- **Cập nhật giao diện hiệu quả:** Nhờ cơ chế Virtual DOM, ReactJS giúp website phản hồi nhanh và cập nhật dữ liệu động mà không cần tải lại toàn bộ trang.

- **Hỗ trợ điều hướng trong ứng dụng:** React Router cho phép xây dựng hệ thống điều hướng mượt mà trong ứng dụng đơn trang (SPA).

- **Kết hợp dễ dàng với các công nghệ khác:** ReactJS hoạt động tốt cùng CSS, thư viện UI, API backend và các công cụ JavaScript, giúp mở rộng tính năng và tối ưu thiết kế giao diện.

- **Tăng hiệu suất và trải nghiệm người dùng:** Nhờ cơ chế render tối ưu và mô hình quản lý trạng thái rõ ràng, ReactJS giúp website vận hành ổn định và mang lại trải nghiệm liền mạch cho người dùng.

2.2.6. Kết luận

ReactJS là một công nghệ mạnh mẽ và hiện đại, đóng vai trò quan trọng trong việc xây dựng giao diện ứng dụng web. Với mô hình phát triển hướng thành phần, cơ chế Virtual DOM và khả năng mở rộng linh hoạt, ReactJS giúp tạo ra giao diện trực quan, dễ sử dụng và hiệu suất cao. Việc ứng dụng ReactJS trong hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên giúp nâng cao trải nghiệm người dùng và đảm bảo tính ổn định của giao diện trong quá trình vận hành.

2.3. NodeJS

2.3.1. Đặc điểm

NodeJS là nền tảng lập trình phổ biến nhờ vào hiệu năng cao, khả năng mở rộng tốt và sử dụng chung một ngôn ngữ JavaScript cho cả frontend và backend. Được xây dựng trên V8 Engine của Google Chrome, NodeJS có tốc độ xử lý nhanh và tối ưu cho các tác vụ I/O. Các thư viện phong phú thông qua npm (Node Package Manager) cho phép mở rộng tính năng dễ dàng, đồng thời cộng đồng phát triển rộng lớn giúp NodeJS trở thành một lựa chọn mạnh mẽ cho các ứng dụng web hiện đại.

[4]

Các đặc điểm nổi bật của NodeJS gồm:

- **Bất đồng bộ và Non-blocking I/O:** Giúp xử lý nhiều yêu cầu cùng lúc mà không làm nghẽn hệ thống.
- **Một ngôn ngữ thống nhất:** Sử dụng JavaScript ở cả frontend và backend.
- **Hiệu năng cao:** Dựa trên V8 Engine tối ưu hóa tốc độ xử lý.
- **Hệ sinh thái phong phú:** npm cung cấp hàng trăm nghìn thư viện hỗ trợ phát triển nhanh chóng.

2.3.2. Chú thích trong NodeJS

NodeJS sử dụng cú pháp chú thích của JavaScript: [3]

- Chú thích một dòng: `// nội dung`
- Chú thích nhiều dòng: `/* nội dung */`

2.3.3. Cấu trúc cơ bản

Trong NodeJS, mã nguồn được lưu dưới dạng tệp .js và được thực thi bằng lệnh node. Một ứng dụng backend thường sử dụng ExpressJS để xây dựng API. [3]

Ví dụ:

```
let http = require('http');

http.createServer(function (req, res) {
  res.writeHead(200, {'Content-Type': 'text/plain'});
  res.end('Hello World!');
}).listen(8080);
```

Trong đó:

- `require('express')`: Tải module Express vào chương trình.
- `app.get(...)`: Xử lý yêu cầu GET tại đường dẫn chỉ định.
- `res.redirect()`: Chuyển hướng người dùng sang URL khác.
- `app.listen()`: Khởi động server.

2.3.4. Khai báo biến

NodeJS hỗ trợ khai báo biến bằng ba từ khóa: var, let, và const. Trong thực tế, let và const được ưu tiên do an toàn và rõ ràng hơn.

Ví dụ:

```
const dbName = "shared_docs";  
  
let totalDocuments = 100;
```

2.3.5. Câu lệnh điều kiện

NodeJS sử dụng các câu lệnh điều kiện giống JavaScript như: if, else if, else và switch.

Ví dụ:

```
if (!req.session.user) {  
    return res.redirect('/login');  
}
```

2.3.6. Kết nối với MySQL

NodeJS kết nối MySQL thông qua các thư viện như mysql2 hoặc ORM như Sequelize.

Ví dụ kết nối bằng mysql2:

```
const mysql = require('mysql2');  
  
const connection = mysql.createConnection({  
    host: 'localhost',  
    user: 'root',  
    password: '',  
    database: 'shared_docs'  
});  
  
connection.connect((err) => {  
    if (err) {  
        console.error('Kết nối thất bại:', err);  
    }  
});
```

```
        return;

    }

    console.log('Kết nối MySQL thành công');

});
```

2.3.7. Truy vấn dữ liệu

NodeJS cho phép thực hiện truy vấn SQL linh hoạt thông qua hàm `query()` của `mysql2`.

Ví dụ:

```
const mysql = require('mysql2');

const connection = mysql.createConnection({

    host: 'localhost',

    user: 'root',

    password: '',

    database: 'shared_docs'

});

connection.connect((err) => {

    if (err) {

        console.error('Kết nối thất bại:', err);

        return;

    }

    console.log('Kết nối MySQL thành công');

});
```

2.3.8. Xử lý form

Trong ExpressJS, dữ liệu gửi từ form được xử lý thông qua `req.body` sau khi cấu hình middleware `express.urlencoded()` hoặc `express.json()`.

Ví dụ:

```
app.post('/login', (req, res) => {
```

```
const { username, password } = req.body;

const sql = "SELECT * FROM users WHERE username = ? AND password = ?";
connection.query(sql, [username, password], (err, result) => {
    if (result.length > 0) {
        res.send("Đăng nhập thành công");
    } else {
        res.send("Sai thông tin đăng nhập");
    }
});
});
```

2.3.9. Quản lý lỗi

NodeJS hỗ trợ quản lý lỗi bằng khối try-catch, đặc biệt khi xử lý promise hoặc async/await.

Ví dụ:

```
async function deleteFile(fileId) {
    try {
        const result = await fetch('/delete-file', {
            method: 'POST',
            headers: { 'Content-Type': 'application/json' },
            body: JSON.stringify({ id: fileId })
        });

        const response = await result.json();

        if (response.success) {
            console.log("Xóa thành công");
        } else {
            console.log("Lỗi khi xóa");
        }
    } catch (error) {
        console.error("Đã xảy ra lỗi:", error);
    }
}
```


2.3.10. Kết luận

NodeJS là một nền tảng mạnh mẽ cho phát triển backend hiện đại nhờ khả năng xử lý bất đồng bộ, hiệu năng cao và sử dụng chung ngôn ngữ JavaScript trên toàn bộ hệ thống. Việc ứng dụng NodeJS giúp đơn giản hóa quá trình xây dựng API, tối ưu hóa tốc độ phản hồi và tăng tính linh hoạt trong việc mở rộng ứng dụng web. Đây là lựa chọn phù hợp cho các hệ thống yêu cầu khả năng xử lý đồng thời lớn và tốc độ xử lý nhanh.

2.4. Ngôn ngữ MySQL

MySQL là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) hữu hiệu, được thiết kế để quản lý và truy xuất dữ liệu có cấu trúc một cách nhanh chóng và tin cậy. MySQL thường được tích hợp trong các ứng dụng web lớn như WordPress, Drupal, và Magento.

2.4.1. Đặc điểm

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (“RDBMS”) được sử dụng rộng rãi trong nhiều lĩnh vực như quản lý doanh nghiệp, phát triển web, và ứng dụng di động. Tính phổ biến của nó bắt nguồn từ hiệu quả vận hành, độ tin cậy và tính linh hoạt cao trong việc tích hợp với nhiều ngôn ngữ lập trình. [5]

- **Hiệu suất cao:** MySQL đã chứng minh khả năng xử lý khối lượng dữ liệu cực lớn với độ trễ thấp, đáp ứng tính toàn vẹn cho các hệ thống giao dịch trực tuyến và ứng dụng doanh nghiệp có khối lượng truy cập lớn.
- **Tính bền vững:** MySQL đảm bảo bảo toàn dữ liệu thông qua các cơ chế sao lưu (backup), khôi phục (recovery), và hỗ trợ giao dịch (transaction support). Khả năng này rất hữu ích cho các doanh nghiệp cần bảo đảm sự liên tục của hệ thống.
- **Khả năng tích hợp:** MySQL hỗ trợ nhiều giao thức lập trình ứng dụng (API) và khả năng tích hợp dễ dàng với các ngôn ngữ như PHP, Python, Java, Ruby và Node.js. Tính linh hoạt này giúp nó phù hợp cho nhiều loại hệ thống và ứng dụng từ web đơn giản đến các hệ thống quy mô lớn.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Nhờ vào những đặc điểm nêu trên, MySQL không chỉ là sự lựa chọn hàng đầu cho những nhà phát triển mà còn là một nền tảng quan trọng trong hầu hết các hệ sinh thái phần mềm hiện đại.

2.4.2. Tạo cơ sở dữ liệu

MySQL sử dụng ngôn ngữ SQL (Structured Query Language) để thực hiện các thao tác trên cơ sở dữ liệu. Một tập hợp có tổ chức của dữ liệu, thường được lưu trữ trong các bảng.

Cú pháp tạo cơ sở dữ liệu:

```
create database kef_db
```

Cú pháp chọn cơ sở dữ liệu:

```
use kef_db
```

2.4.3. Tạo cấu trúc bảng

Cấu trúc dữ liệu chính trong MySQL bao gồm các bản ghi (rows) và các trường (columns).

Cú pháp tạo bảng:

```
CREATE TABLE `quantrivien` (  
  `maQuanTriVien` int(11) NOT NULL,  
  `hoTenQTV` varchar(255) NOT NULL,  
  `emailQTV` varchar(255) NOT NULL,  
  `matKhauQTV` varchar(255) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_vietnamese_ci;
```

2.4.4. Các câu lệnh truy vấn

Câu lệnh SELECT: Truy xuất dữ liệu từ bảng.

```
SELECT maQuanTriVien, hoTenQTV, emailQTV FROM quantrivien  
ORDER BY maQuanTriVien;
```

Câu lệnh INSERT: Thêm dữ liệu mới vào bảng.

```
INSERT INTO `nganh` (`maNganh`, `tenNganh`) VALUES  
(1, 'Công nghệ thông tin');
```

Câu lệnh UPDATE: Cập nhật dữ liệu trong bảng.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

```
UPDATE quantrivien SET hoTenQTV = adminTest WHERE  
maQuanTriVien = 3;
```

Câu lệnh DELETE: Xóa dữ liệu khỏi bảng.

```
DELETE FROM quantrivien WHERE maQuanTriVien = 3;
```

Khóa chính (Primary Key): Một cột hoặc nhóm cột xác định duy nhất mỗi hàng trong bảng. Có 2 cách tạo khóa chính:

Tạo khóa chính ngay khi tạo bảng:

```
CREATE TABLE `quantrivien` (  
  `maQuanTriVien` int(11) NOT NULL PRIMARY KEY,  
  `hoTenQTV` varchar(255) NOT NULL,  
  `emailQTV` varchar(255) NOT NULL,  
  `matKhanhQTV` varchar(255) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_vietnamese_ci;
```

Tạo khóa chính sau khi tạo bảng:

```
ALTER TABLE `quantrivien`  
  ADD PRIMARY KEY (`maQuanTriVien`);
```

Khóa ngoại (Foreign Key): Một cột hoặc nhóm cột liên kết với khóa chính của một bảng khác, dùng để duy trì tính toàn vẹn dữ liệu.

Tạo khóa ngoại ngay khi tạo bảng:

```
CREATE TABLE `cauhoi` (  
  `maCauHoi` int(11) NOT NULL,  
  `maSinhVien` int(11) NOT NULL,  
  `maMon` int(11) NOT NULL,  
  `tieuDeCH` varchar(255) NOT NULL,  
  `noiDungCH` text DEFAULT NULL,  
  `ngayDatCH` datetime NOT NULL DEFAULT  
current_timestamp(),  
  `trangThaiCH` enum('open','answered','closed') NOT NULL,  
  `luotTraLoi` int(11) NOT NULL  
  FOREIGN KEY (`maMon`) REFERENCES `mon` (`maMon`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_vietnamese_ci;
```

Tạo khóa ngoại sau khi tạo bảng:

```
ALTER TABLE `cauhoi`  
  ADD CONSTRAINT `fk_cauhoi_mon` FOREIGN KEY (`maMon`)  
  REFERENCES `mon` (`maMon`) ON UPDATE CASCADE;
```

Nhờ vào những đặc điểm và khả năng trên, MySQL trở thành công cụ không thể thiếu trong việc quản lý và xử lý dữ liệu trong các hệ thống phần mềm hiện đại.

2.4.5. Kết luận

MySQL là hệ quản trị cơ sở dữ liệu quan hệ ổn định và đáng tin cậy, hỗ trợ lưu trữ và truy vấn dữ liệu hiệu quả cho các ứng dụng web. Nhờ cấu trúc dữ liệu chặt chẽ, khả năng xử lý truy vấn mạnh mẽ và tính toàn vẹn cao, MySQL giúp đảm bảo dữ liệu được quản lý an toàn và nhất quán. Đây là nền tảng quan trọng giúp hệ thống vận hành bền vững và đáp ứng tốt các nhu cầu quản lý dữ liệu quy mô vừa và lớn.

3.1. Mô tả vấn đề

Trong quá trình học tập, sinh viên thường gặp khó khăn khi tìm kiếm tài liệu phù hợp và thiếu không gian chung để trao đổi, giải đáp các thắc mắc học thuật. Tài liệu học tập bị phân tán trên nhiều nền tảng khiến việc tiếp cận thông tin trở nên mất thời gian và kém hiệu quả.

Vì vậy, đề tài “Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên” được xây dựng nhằm cung cấp một nền tảng tập trung, hỗ trợ chia sẻ tài liệu, tra cứu nội dung học tập và đặt câu hỏi theo từng môn học, góp phần nâng cao tính kết nối và hiệu quả tự học của sinh viên.

3.2. Xác định yêu cầu người dùng

Trước khi xác định yêu cầu người dùng, cần làm rõ các chức năng cốt lõi mà hệ thống chia sẻ tài liệu và hỏi đáp phải đáp ứng. Mục tiêu của hệ thống là tạo ra một nền tảng tập trung giúp sinh viên dễ dàng chia sẻ tài liệu học tập, tra cứu thông tin môn học và đặt câu hỏi về các chủ đề liên quan đến học tập. Các yêu cầu cơ bản của hệ thống gồm:

- Giao diện đơn giản, trực quan, dễ sử dụng; hỗ trợ truy cập và quản lý tài liệu học tập theo môn học hoặc chủ đề.
- Có tính năng tìm kiếm tài liệu theo từ khóa, môn học, ngành, tác giả, thời gian đăng tải hoặc mức độ phổ biến.
- Cung cấp khu vực đặt câu hỏi và trả lời, giúp sinh viên trao đổi học thuật và giải đáp thắc mắc.
- Hỗ trợ đăng tải tài liệu học tập (PDF, hình ảnh, slide, ghi chú), kèm mô tả và phân loại rõ ràng.
- Có cơ chế đánh giá tài liệu hoặc câu trả lời để giúp người dùng nhận diện nội dung chất lượng.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Để xác định yêu cầu người dùng một cách chính xác, cần xem xét nhu cầu thực tế của đối tượng sử dụng chính là sinh viên. Yêu cầu tối thiểu mà hệ thống phải đáp ứng gồm:

- Hệ thống phải có chức năng lọc tài liệu theo môn học, ngành, từ khóa, tệp định dạng, độ phổ biến, hoặc người đăng.
- Cung cấp thông tin chi tiết về tài liệu như mô tả, thời gian đăng tải, tác giả, đánh giá và số lượt tải.
- Cho phép người dùng đặt câu hỏi, trả lời, đánh giá và theo dõi các chủ đề họ quan tâm.

3.3. Phân loại người dùng

Những người sử dụng hệ thống này gồm ba nhóm chính:

- **Sinh viên:** Sinh viên có chức năng tìm kiếm và tải xuống tài liệu học tập, đặt câu hỏi, tham gia thảo luận và chia sẻ tài liệu cho cộng đồng, đóng góp nội dung cho hệ thống.
- **Người dùng khách:** Chỉ có chức năng tìm kiếm, tải xuống tài liệu và xem các phiên hỏi đáp, ngoài ra người dùng khách còn có chức năng đăng ký và đăng nhập.
- **Quản trị viên:** Giám sát hoạt động hệ thống, quản lý các chức năng, phê duyệt hoặc gỡ bỏ tài liệu, câu hỏi, câu trả lời vi phạm, đảm bảo môi trường học thuật an toàn và đúng quy định.

3.4. Phân tích thiết kế hệ thống

3.4.1. Đặc tả yêu cầu hệ thống

3.4.1.1. Yêu cầu chức năng

Để dự án này có thể hoạt động đúng với mục đích ban đầu thì cần phải có các chức năng:

- Đăng nhập và đăng ký.
- Chức năng lọc và tìm kiếm.

- Chức năng quản lý hồ sơ cá nhân.
- Chức năng chia sẻ tài liệu.
- Chức năng hỏi đáp.
- Chức năng quản trị dành cho quản trị viên.

3.4.1.2. Yêu cầu phi chức năng

Để dự án này có thể hoạt động tốt và hiệu quả thì cần phải đảm bảo các yêu cầu sau:

- Ứng dụng hoạt động mượt mà, hiệu quả cao.
- Phải bảo mật dữ liệu của người dùng.
- Giao diện thân thiện và dễ sử dụng.
- Đảm bảo tính mở rộng và bảo trì.

3.4.2. Kiến trúc hệ thống

Để hệ thống web chia sẻ tài liệu và hỏi đáp hoạt động hiệu quả, dễ mở rộng và dễ bảo trì, đề tài áp dụng kiến trúc phân tầng. Kiến trúc này giúp tách biệt rõ ràng giữa giao diện, xử lý nghiệp vụ và dữ liệu, đảm bảo sự linh hoạt khi phát triển và triển khai hệ thống.

3.4.2.1. Lớp giao diện người dùng:

Lớp giao diện người dùng chịu trách nhiệm hiển thị dữ liệu và tương tác trực tiếp với người dùng. Ở đề tài này, giao diện được xây dựng bằng ReactJS, kết hợp HTML, CSS và JavaScript. Lớp này đảm nhiệm các chức năng:

- Nhận yêu cầu từ người dùng như đăng nhập, đăng ký, tra cứu, đánh giá hoặc tải tài liệu.
- Gửi yêu cầu đến backend thông qua API.
- Hiển thị dữ liệu nhận được từ backend theo bố cục dễ hiểu.
- Đảm bảo trải nghiệm người dùng mượt mà, trực quan.

3.4.2.2. Lớp xử lý nghiệp vụ:

Lớp xử lý nghiệp vụ được xây dựng bằng NodeJS (ExpressJS) theo mô hình mô-đun. Đây là thành phần trung tâm thực hiện:

- Xử lý các yêu cầu từ frontend và điều phối tới các module tương ứng.
- Triển khai các chức năng chính như xác thực người dùng, xử lý đánh giá, tìm kiếm dữ liệu, quản lý tài liệu và câu hỏi.
- Kiểm tra dữ liệu đầu vào, đảm bảo tính hợp lệ và bảo mật.
- Tương tác với cơ sở dữ liệu thông qua các truy vấn MySQL.

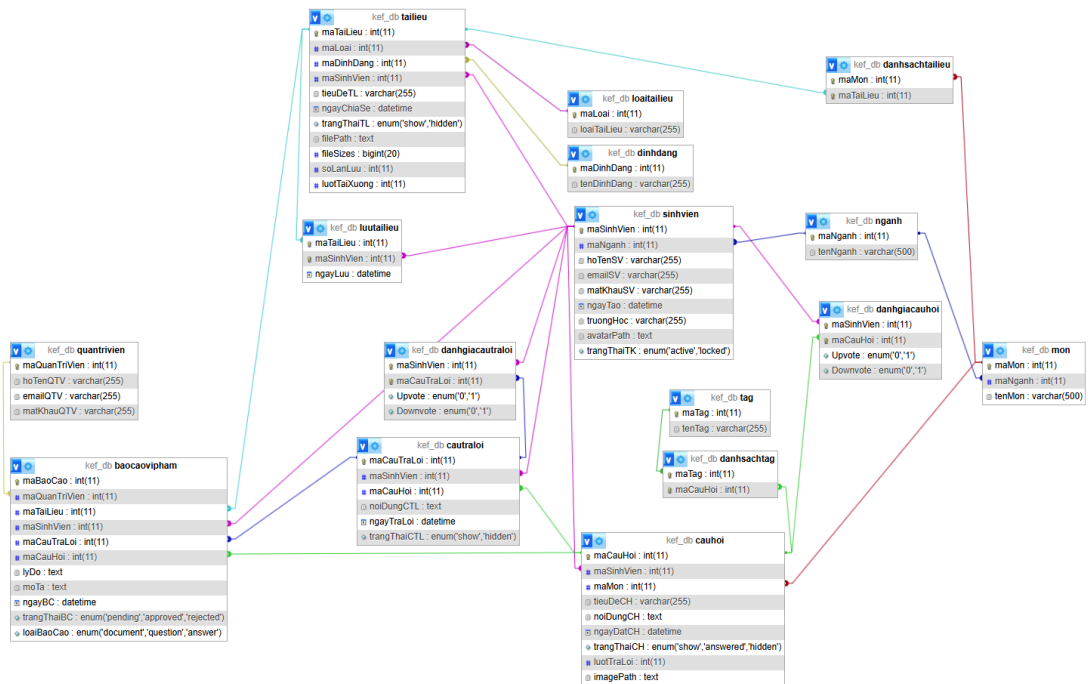
3.4.2.3. Lớp dữ liệu:

Lớp dữ liệu sử dụng MySQL để lưu trữ và quản lý toàn bộ thông tin của hệ thống, bao gồm người dùng, tài liệu, câu hỏi, câu trả lời và các hoạt động đánh giá. Tầng này cung cấp:

- Lưu trữ dữ liệu bền vững và có cấu trúc.
- Các thao tác CRUD (Create, Read, Update, Delete).
- Đảm bảo tính toàn vẹn dữ liệu.

3.4.3. Thiết kế dữ liệu

3.4.3.1. Sơ đồ cơ sở dữ liệu



Hình 3.1 Sơ đồ cơ sở dữ liệu

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

3.4.3.2. Danh sách các thực thể và mối kết hợp

Bảng 3.1 Bảng thực thể

STT	Tên thực thể	Diễn giải
1	SinhVien	Lưu trữ thông tin người dùng là sinh viên, bao gồm tài khoản, thông tin cá nhân và trạng thái hoạt động.
2	QuanTriVien	Lưu trữ thông tin tài khoản và thông tin cá nhân của quản trị viên hệ thống.
3	Nganh	Quản lý danh sách các ngành học mà sinh viên theo học.
4	Mon	Lưu trữ thông tin môn học, thuộc một ngành cụ thể.
5	TaiLieu	Chứa thông tin chi tiết của tài liệu học tập mà sinh viên tải lên như tiêu đề, mô tả, tệp đính kèm, loại tài liệu và định dạng.
6	LoaiTaiLieu	Lưu trữ các nhóm loại tài liệu như bài giảng, bài tập, đề thi...
7	DinhDang	Lưu trữ thông tin về định dạng tệp (PDF, DOCX, PNG...).
8	DanhSachTaiLieu	Mối kết hợp giữa SinhVien và TaiLieu, lưu lịch sử tài liệu mà sinh viên đã lưu hoặc đánh dấu.
9	CauHoi	Lưu thông tin câu hỏi học thuật do sinh viên đăng, bao gồm tiêu đề, nội dung và môn học liên quan.
10	CauTraLoi	Chứa danh sách các câu trả lời cho từng câu hỏi, do sinh viên đăng.
11	DanhGiaCauHoi	Mối kết hợp giữa SinhVien và CauHoi, lưu thông tin Upvote/Downvote cho câu hỏi.
12	DanhGiaCauTraLoi	Mối kết hợp giữa SinhVien và CauTraLoi, lưu thông tin Upvote/Downvote cho câu trả lời.
13	Tag	Lưu danh sách các thẻ dùng để phân loại câu hỏi theo chủ đề.
14	DanhSachTags	Mối kết hợp giữa Tag và CauHoi (một câu hỏi có nhiều tag).
15	BaoCaoViPham	Lưu thông tin báo cáo vi phạm đối với tài liệu, câu hỏi hoặc câu trả lời.
16	LuuTaiLieu	Ghi lại sinh viên nào đã lưu tài liệu nào và thời điểm lưu.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

3.4.3.3. Chi tiết các thực thể

Tên thực thể: **SinhVien**

Mô tả: Lưu trữ thông tin người dùng là sinh viên.

Chi tiết thực thể:

Bảng 3.2 Bảng thuộc tính của thực thể SinhVien

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maSinhVien	Mã sinh viên	int	PK
2	maNganh	Mã ngành	int	FK
3	hoTenSV	Họ tên sinh viên	varchar(255)	NOT NULL
4	emailSV	Email sinh viên	varchar(255)	NOT NULL
5	matKhauSV	Mật khẩu đăng nhập	varchar(255)	NOT NULL
6	ngayTao	Ngày tạo tài khoản	datetime	NOT NULL
7	truongHoc	Trường học	varchar(255)	NULL
8	trangThaiTK	Trạng thái tài khoản	enum('active', 'locked')	NOT NULL
9	avatarPath	Đường dẫn ảnh đại diện	text	NULL

Tên thực thể: **QuanTriVien**

Mô tả: Lưu trữ thông tin đăng nhập của tài khoản quản trị viên

Chi tiết thực thể:

Bảng 3.3 Bảng thuộc tính của thực thể QuanTriVien

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maQuanTriVien	Mã quản trị viên	int	PK
2	hoTenQTV	Họ tên	varchar(255)	NOT NULL
3	emailQTV	Email	varchar(255)	NOT NULL
4	matKhauQTV	Mật khẩu	varchar(255)	NOT NULL

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Tên thực thể: **Nganh**

Mô tả: Lưu trữ danh sách các ngành học.

Chi tiết thực thể:

Bảng 3.4 Bảng thuộc tính của thực thể Nganh

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maNganh	Mã ngành	int	PK
2	tenNganh	Tên ngành	varchar(255)	NOT NULL

Tên thực thể: **Mon**

Mô tả: Lưu trữ danh sách các môn học thuộc từng ngành.

Chi tiết thực thể:

Bảng 3.5 Bảng thuộc tính của thực thể Mon

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maMon	Mã môn học	int	PK
2	maNganh	Mã ngành	int	FK
3	tenMon	Tên môn học	varchar(255)	NOT NULL

Tên thực thể: **LoaiTaiLieu**

Mô tả: Lưu trữ danh sách loại tài liệu học tập.

Chi tiết thực thể:

Bảng 3.6 Bảng thuộc tính của thực thể LoaiTaiLieu

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maLoai	Mã loại tài liệu	int	PK
2	loaiTaiLieu	Tên loại tài liệu	varchar(255)	NOT NULL

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Tên thực thể: **TaiLieu**

Mô tả: Lưu trữ thông tin chi tiết của tài liệu học tập do sinh viên chia sẻ.

Chi tiết thực thể:

Bảng 3.7 Bảng thuộc tính của thực thể TaiLieu

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maTaiLieu	Mã tài liệu	int	PK
2	maSinhVien	Người đăng tài liệu	int	FK
3	maLoai	Loại tài liệu	int	FK
4	maDinhDang	Định dạng tệp	int	FK
5	tieuDeTL	Tiêu đề tài liệu	varchar(255)	NOT NULL
6	ngayChiaSe	Ngày chia sẻ	datetime	NOT NULL
7	trangThaiTL	Trạng thái hiển thị	enum('show', 'hidden')	NOT NULL
8	filePath	Đường dẫn tệp	text	NOT NULL
9	fileSize	Kích thước tệp	bigint	NOT NULL
10	soLuotLuu	Số lượt lưu tài liệu	int	NOT NULL
11	luotTaiXuong	Số lượt tải xuống	int	NOT NULL

Tên thực thể: **DinhDang**

Mô tả: Lưu trữ danh sách các định dạng tệp tài liệu.

Chi tiết thực thể:

Bảng 3.8 Bảng thuộc tính của thực thể DinhDang

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maDinhDang	Mã định dạng	int	PK
2	tenDinhDang	Tên định dạng (PDF, DOCX, JPG...)	varchar(50)	NOT NULL

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Tên thực thể: **DanhSachTaiLieu**

Mô tả: Lưu quan hệ N–N giữa SinhVien và TaiLieu.

Chi tiết thực thể:

Bảng 3.9 Bảng thuộc tính của thực thể DanhSachTaiLieu

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maMon	Mã môn	int	PK, FK
2	maTaiLieu	Mã tài liệu	int	PK, FK

Tên thực thể: **CauHoi**

Mô tả: Lưu trữ thông tin câu hỏi học thuật do sinh viên đăng lên hệ thống.

Chi tiết thực thể:

Bảng 3.10 Bảng thuộc tính của thực thể CauHoi

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maCauHoi	Mã câu hỏi	int	PK
2	maSinhVien	Người đăng câu hỏi	int	FK
3	maMon	Mã môn học liên quan	int	FK
4	tieuDeCH	Tiêu đề câu hỏi	varchar(255)	NOT NULL
5	noiDungCH	Nội dung câu hỏi	text	NULL
6	ngayDatCH	Ngày đăng câu hỏi	datetime	NOT NULL
7	trangThaiCH	Trạng thái câu hỏi	enum('show', 'answered', 'hidden')	NOT NULL
8	luotTraLoi	Số lượng câu trả lời	int	NOT NULL
9	imagePath	Đường dẫn hình ảnh	text	NULL

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Tên thực thể: **CauTraLoi**

Mô tả: Lưu trữ các câu trả lời cho từng câu hỏi.

Chi tiết thực thể:

Bảng 3.11 Bảng thuộc tính của thực thể CauTraLoi

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maCauTraLoi	Mã câu trả lời	int	PK
2	maSinhVien	Người trả lời	int	FK
3	maCauHoi	Câu hỏi liên quan	int	FK
4	noiDungCTL	Nội dung câu trả lời	text	NOT NULL
5	ngayTraLoi	Ngày trả lời	datetime	NOT NULL
6	trangThaiCTL	Trạng thái câu trả lời	enum('show', 'hidden')	NOT NULL

Tên thực thể: **DanhGiaCauHoi**

Mô tả: Lưu thông tin đánh giá của sinh viên đối với câu trả lời.

Chi tiết thực thể:

Bảng 3.12 Bảng thuộc tính của thực thể DanhGiaCauHoi

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maSinhVien	Người đánh giá câu hỏi	int	PK, FK
2	maCauHoi	Mã câu hỏi được đánh giá	int	PK, FK
3	Upvote	Đánh giá tích cực (1 hoặc 0)	enum('0', '1')	NOT NULL
4	Downvote	Đánh giá tiêu cực (1 hoặc 0)	enum('0', '1')	NOT NULL

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Tên thực thể: **DanhGiaCauTraLoi**

Mô tả: Lưu thông tin đánh giá của sinh viên đối với câu trả lời.

Chi tiết thực thể:

Bảng 3.13 Bảng thuộc tính của thực thể DanhGiaCauTraLoi

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maSinhVien	Người đánh giá câu trả lời	int	PK, FK
2	maCauTraLoi	Mã câu trả lời được đánh giá	int	PK, FK
3	Upvote	Đánh giá tích cực (1 hoặc 0)	enum('0', '1')	NOT NULL
4	Downvote	Đánh giá tiêu cực (1 hoặc 0)	enum('0', '1')	NOT NULL

Tên thực thể: **Tag**

Mô tả: Lưu trữ danh sách các thẻ (tags) dùng để phân loại chủ đề câu hỏi.

Chi tiết thực thể:

Bảng 3.14 Bảng thuộc tính của thực thể Tag

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maTag	Mã tag	int	PK
2	tenTag	Tên tag (chủ đề)	varchar(255)	NOT NULL

Tên thực thể: **DanhSachTag**

Mô tả: Lưu thông tin tag nào được gán cho câu hỏi nào.

Chi tiết thực thể:

Bảng 3.15 Bảng thuộc tính của thực thể DanhSachTags

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maTag	Mã tag	int	PK, FK
2	maCauHoi	Mã câu hỏi	int	PK, FK

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

Tên thực thể: **BaoCaoViPham**

Mô tả: Lưu trữ thông tin các báo cáo vi phạm do sinh viên gửi lên và do quản trị viên xử lý. Một báo cáo có thể liên quan đến tài liệu, câu hỏi, hoặc câu trả lời.

Chi tiết thực thể:

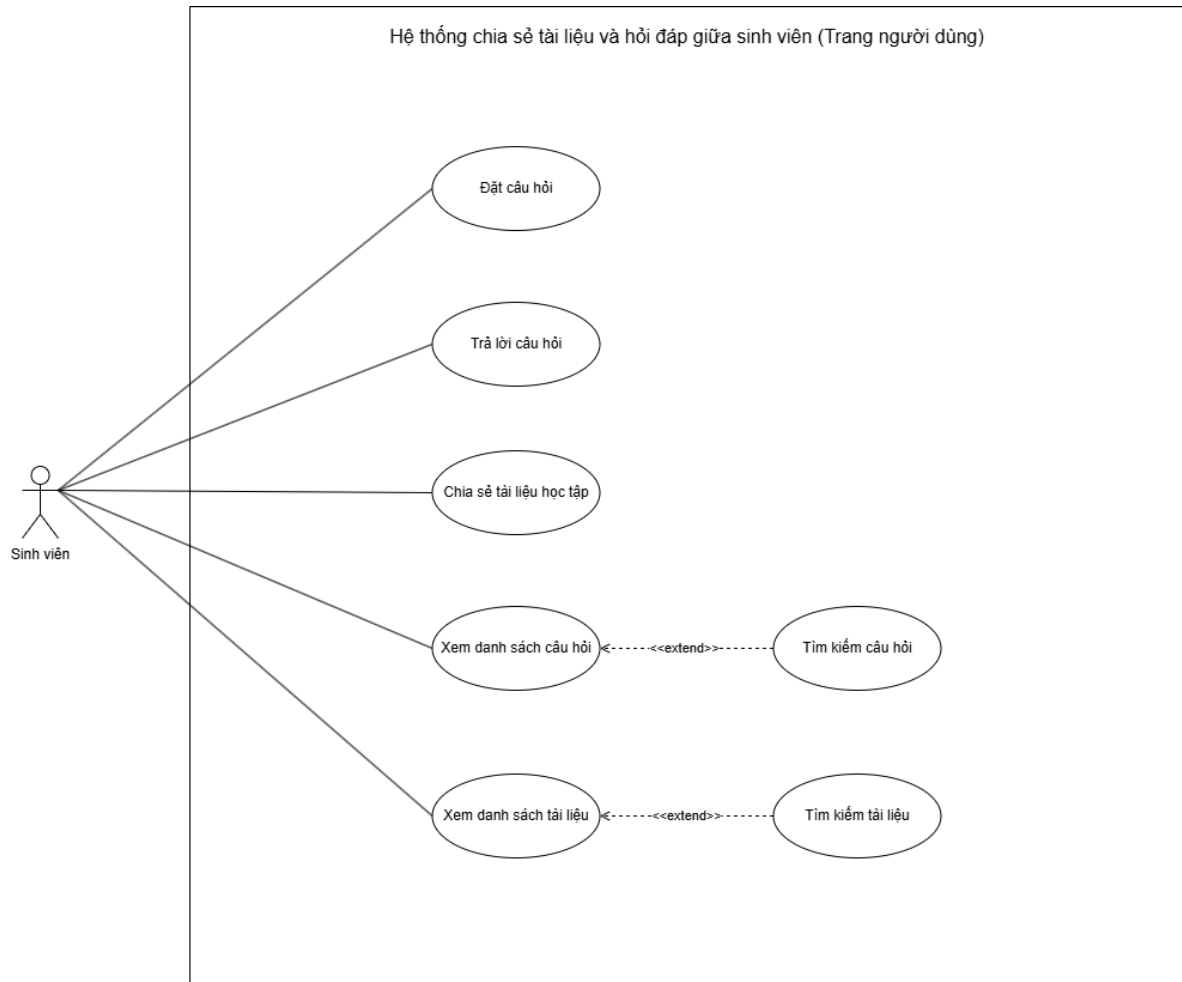
Bảng 3.16 Bảng thuộc tính của thực thể BaoCaoViPham

STT	Thuộc tính	Diễn giải	Kiểu dữ liệu	Ràng buộc
1	maBaoCao	Mã báo cáo	int	PK
2	maQuanTriVien	Quản trị viên xử lý	int	FK
3	maSinhVien	Người gửi báo cáo	int	FK
4	maTaiLieu	Liên quan tài liệu	int	FK, NULL
5	maCauHoi	Liên quan câu hỏi	int	FK, NULL
6	maCauTraLoi	Liên quan câu trả lời	int	FK, NULL
7	lyDo	Lý do báo cáo	enum('copyright', 'misinformation', 'inappropriate_content', 'spam', 'other')	NOT NULL
8	moTa	Nội dung báo cáo nếu lý do là khác	text	NULL
9	ngayBC	Ngày gửi báo cáo	datetime	NOT NULL
10	trangThaiBC	Trạng thái xử lý	enum('pending', 'approved', 'rejected')	NOT NULL
11	loaiBaoCao	Loại vi phạm	enum('document', 'question', 'asnsver')	NOT NULL

3.4.4. Thiết kế giao diện

3.4.4.1. Sơ đồ Use Case

3.4.4.1.1. Trang người dùng

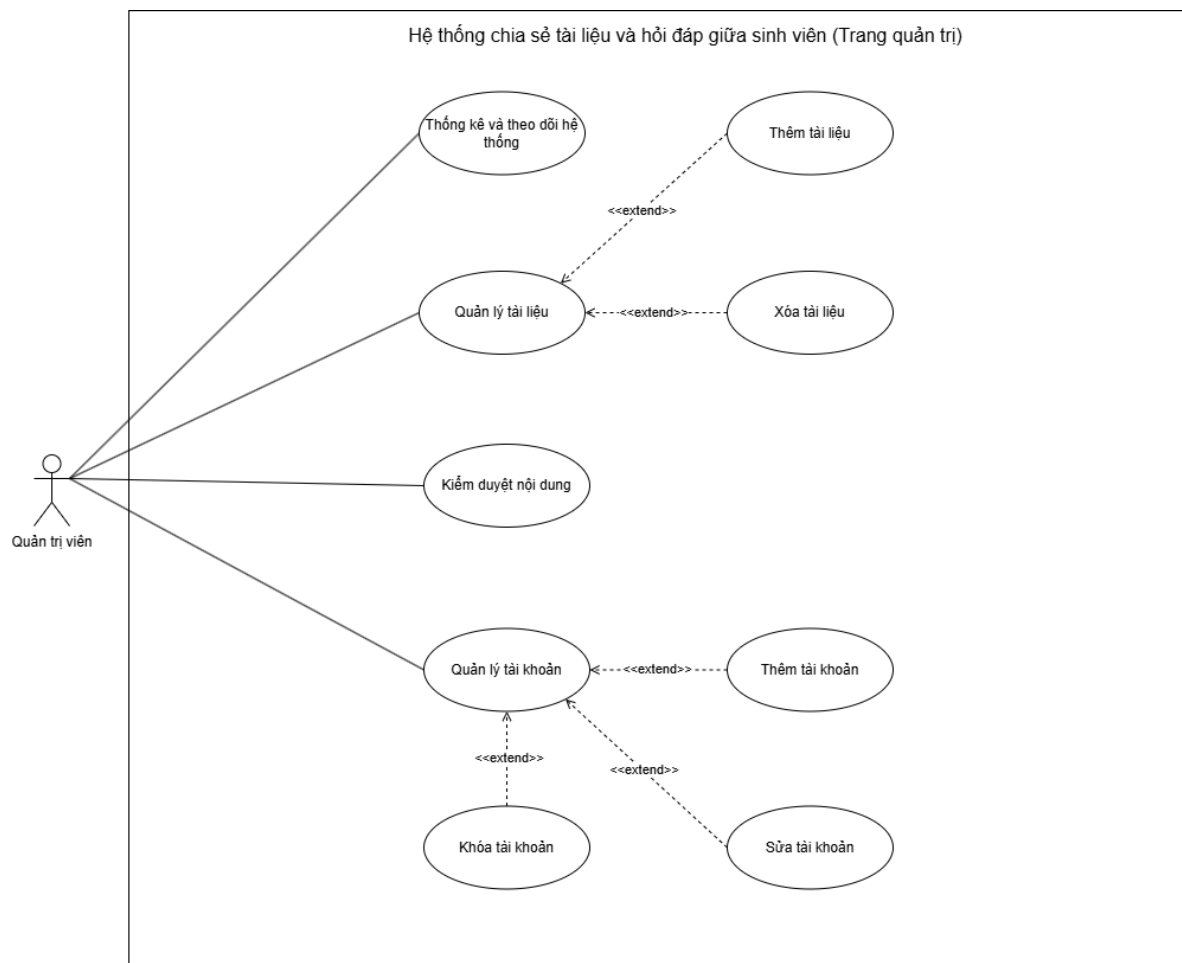


Hình 3.2 Sơ đồ Use Case về trang người dùng

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên*Bảng 3.17 Bảng mô tả Use Case trang người dùng*

Tên Use Case	Tác nhân	Mối quan hệ	Mô tả
Đặt câu hỏi	Sinh viên	Không	Sinh viên tạo câu hỏi mới để nhờ hỗ trợ.
Trả lời câu hỏi		Không	Sinh viên trả lời câu hỏi của người khác.
Chia sẻ tài liệu học tập		Không	Tải tài liệu học tập lên để chia sẻ.
Xem danh sách câu hỏi		Chức năng này bao gồm chức năng tìm kiếm câu hỏi	Hiển thị toàn bộ câu hỏi trong hệ thống.
Tìm kiếm câu hỏi		Đây là chức năng mở rộng của “Xem danh sách câu hỏi”	Tìm câu hỏi theo từ khóa.
Xem danh sách tài liệu		Chức năng này bao gồm chức năng tìm kiếm tài liệu	Hiển thị danh sách các tài liệu.
Tìm kiếm tài liệu		Đây là chức năng mở rộng của “Xem danh sách tài liệu”	Tìm tài liệu theo tiêu đề hoặc từ khóa.

3.4.4.1.2. Trang quản trị

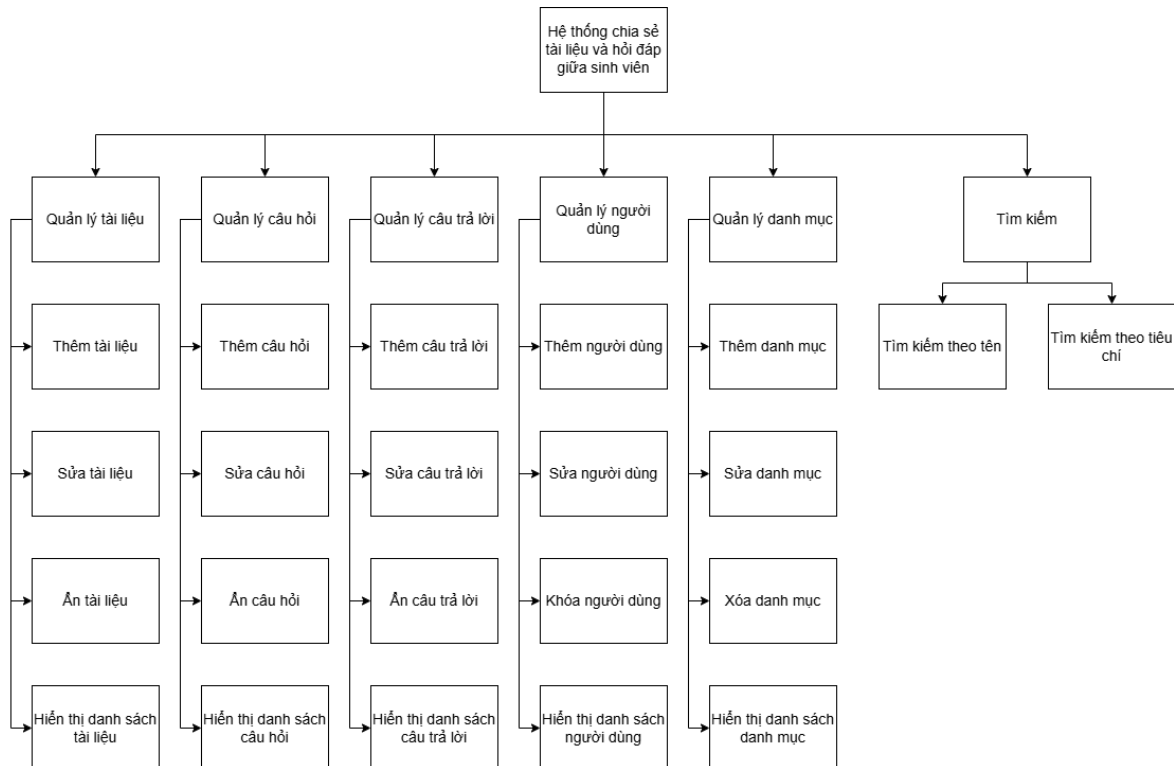


Hình 3.3 Sơ đồ Use Case về trang quản trị

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên*Bảng 3.18 Bảng mô tả Use Case trang quản trị*

Tên Use Case	Tác nhân	Mối quan hệ	Mô tả
Thông kê và theo dõi hệ thống	Quản trị viên	Không	Xem các số liệu thống kê của hệ thống.
Quản lý tài liệu		Chức năng này bao gồm thêm và xóa tài liệu	Quản lý toàn bộ tài liệu trong hệ thống.
Thêm tài liệu		Đây là chức năng mở rộng của “Quản lý tài liệu”	Thêm tài liệu mới vào hệ thống.
Xóa tài liệu		Đây là chức năng mở rộng của “Quản lý tài liệu”	Xóa tài liệu không phù hợp.
Kiểm duyệt nội dung		Không	Kiểm tra và duyệt các bài viết, câu hỏi, tài liệu.
Quản lý tài khoản		Chức năng này bao gồm thêm, chỉnh sửa và khóa tài khoản người dùng	Quản lý thông tin tài khoản trong hệ thống.
Thêm tài khoản		Đây là chức năng mở rộng của “Quản lý tài khoản”	Thêm người dùng mới.
Sửa tài khoản		Đây là chức năng mở rộng của “Quản lý tài khoản”	Chỉnh sửa thông tin tài khoản.
Khóa tài khoản		Đây là chức năng mở rộng của “Quản lý tài khoản”	Khóa tài khoản vi phạm.

3.4.4.2. Sơ đồ website



Hình 3.4 Sơ đồ website

3.4.4.2.1. Quản lý tài liệu

Chức năng này dành cho người quản trị hoặc người dùng có quyền đăng tài liệu. Hệ thống cho phép thao tác với các tài liệu học tập được chia sẻ trên website. Nhóm chức năng này bao gồm các chức năng con với vai trò cụ thể nhằm hỗ trợ hệ thống hoạt động hiệu quả và đầy đủ hơn:

- **Thêm tài liệu:** Người dùng có thể tải lên tài liệu mới như PDF, Word, slide bài giảng,... kèm theo thông tin mô tả, tên tài liệu và danh mục liên quan.
- **Sửa tài liệu:** Cho phép chỉnh sửa thông tin của tài liệu đã đăng, ví dụ cập nhật phiên bản mới hoặc sửa lỗi mô tả.
- **Ẩn tài liệu:** Dùng trong trường hợp tài liệu không phù hợp, cần kiểm duyệt hoặc cần tạm ẩn khỏi người dùng.
- **Danh sách tài liệu:** Hiện thị toàn bộ tài liệu trong hệ thống để quản trị viên xem, lọc và quản lý.

3.4.4.2.2. Quản lý câu hỏi

Hệ thống hỗ trợ sinh viên đặt câu hỏi để nhờ cộng đồng giải đáp. Chức năng này giúp quản trị hoặc người có quyền chỉnh sửa đảm bảo câu hỏi đúng chủ đề và chất lượng. Nhóm chức năng này bao gồm các chức năng con với vai trò cụ thể nhằm hỗ trợ hệ thống hoạt động hiệu quả và đầy đủ hơn:

- **Thêm câu hỏi:** Sinh viên có thể tạo câu hỏi mới, chọn danh mục liên quan để những người khác dễ tìm và trả lời.
- **Sửa câu hỏi:** Người đặt câu hỏi hoặc quản trị viên có thể cập nhật lại nội dung khi cần.
- **Ấn câu hỏi:** Ấn các câu hỏi không phù hợp, trùng lặp hoặc vi phạm quy định.
- **Danh sách câu hỏi:** Hiển thị danh sách toàn bộ câu hỏi để quản trị viên theo dõi hoạt động của hệ thống.

3.4.4.2.3. Quản lý câu trả lời

Chức năng này giúp quản lý các câu trả lời do sinh viên gửi lên để giải đáp câu hỏi. Nhóm chức năng này bao gồm các chức năng con với vai trò cụ thể nhằm hỗ trợ hệ thống hoạt động hiệu quả và đầy đủ hơn:

- **Thêm câu trả lời:** Người dùng có thể phản hồi câu hỏi của sinh viên khác bằng cách gửi câu trả lời.
- **Sửa câu trả lời:** Chỉnh sửa nội dung câu trả lời khi cần bổ sung thông tin hoặc update chính xác hơn.
- **Ấn câu trả lời:** Ấn các câu trả lời không đúng nội dung, spam hoặc vi phạm.
- **Danh sách câu trả lời:** Hiển thị danh sách toàn bộ câu trả lời để quản trị viên theo dõi hoạt động của hệ thống.

3.4.4.2.4. Quản lý người dùng

Giúp quản trị viên kiểm soát hoạt động và thông tin của người dùng trên hệ thống. Nhóm chức năng này bao gồm các chức năng con với vai trò cụ thể nhằm hỗ trợ hệ thống hoạt động hiệu quả và đầy đủ hơn:

- **Thêm người dùng:** Tạo tài khoản mới cho sinh viên hoặc thêm tài khoản quản trị.
- **Sửa người dùng:** Thay đổi thông tin như họ tên, email, mật khẩu, trường học, ngành.
- **Khóa người dùng:** Khóa tài khoản người dùng vi phạm hoặc có hoạt động bất thường.
- **Danh sách người dùng:** Danh sách toàn bộ người dùng và trạng thái tài khoản.

3.4.4.2.5. Quản lý danh mục

Danh mục giúp phân loại tài liệu, câu hỏi theo môn học hoặc chủ đề. Nhóm chức năng này bao gồm các chức năng con với vai trò cụ thể nhằm hỗ trợ hệ thống hoạt động hiệu quả và đầy đủ hơn:

- **Thêm danh mục:** Thêm danh mục mới (Xử lý ảnh, Toán rời rạc,...).
- **Sửa danh mục:** Thay đổi tên danh mục cho phù hợp.
- **Xóa danh mục:** Xóa danh mục không còn sử dụng.
- **Danh sách danh mục:** Danh sách tất cả danh mục của hệ thống.

3.4.4.2.6. Tìm kiếm

Hỗ trợ sinh viên tìm nội dung nhanh và chính xác. Nhóm chức năng này bao gồm các chức năng con với vai trò cụ thể nhằm hỗ trợ hệ thống hoạt động hiệu quả và đầy đủ hơn:

- **Tìm kiếm theo tên:** Nhập tên tài liệu hoặc câu hỏi để tìm nhanh.
- **Tìm kiếm theo tiêu chí:** Bộ lọc theo danh mục, môn học, loại nội dung, ngày đăng,... Giúp tìm chính xác tài liệu/câu hỏi phù hợp.

CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU

4.1. Dữ liệu thử nghiệm

Bảng 4.1 Dữ liệu của bảng dinhDang

maDinhDang	tenDinhDang
1	.pdf
2	.docx
3	.pptx
4	.ppt
5	.txt

Bảng 4.2 Dữ liệu của bảng luutailieu

maTaiLieu	maSinhVien	ngayLuu
1	1	2025-12-04 18:05:41
8	4	2025-12-04 18:05:41

Bảng 4.3 Dữ liệu của bảng ngành

maNganh	tenNganh
1	Công nghệ thông tin
2	Kế toán
3	Ngôn ngữ Anh
4	Quản trị kinh doanh
5	Nông nghiệp

Bảng 4.4 Dữ liệu của bảng quantri vien

maQuanTriVien	hoTenQTV	emailQTV	matKhauQTV
1	Admin 1	admin1@gmail.com	1admin123
2	Admin 2	admin2@gmail.com	2admin123
3	Admin 3	admin3@gmail.com	3admin123

Bảng 4.5 Dữ liệu của bảng tag

maTag	tenTag
1	C
2	SQL
3	Algorithm
4	Accounting
5	Tax
6	English
7	Translate
8	Marketing
9	Management
10	Agriculture
11	Test

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên*Bảng 4.6 Dữ liệu của bảng loaitailieu*

maLoai	loaiTaiLieu
1	Giáo trình
2	Slide bài giảng
3	Đề thi
4	Tài liệu tham khảo
5	Bài tập lớn
6	Bài báo khoa học
7	Sổ tay môn học

Bảng 4.7 Dữ liệu của bảng mon

maMon	maNganh	tenMon
1	1	Lập trình C
2	1	Cơ sở dữ liệu
3	1	Thuật toán
4	2	Nguyên lý kế toán
5	2	Thuế doanh nghiệp
6	2	Kế toán tài chính
7	3	Ngữ âm học
8	3	Dịch thuật cơ bản
9	3	Ngữ pháp nâng cao
10	4	Marketing căn bản
11	4	Quản trị học
12	4	Tài chính doanh nghiệp
13	5	Khoa học cây trồng
14	5	Chăn nuôi đại cương
15	5	Bảo vệ thực vật

Bảng 4.8 Dữ liệu của bảng danhgiacauhoi

maSinhVien	maCauHoi	Upvote	Downvote
1	1	1	0
1	2	1	0
1	3	0	0
2	1	1	0
2	2	0	1
2	3	0	0
3	1	0	1
3	2	1	0
3	3	1	0

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên*Bảng 4.9 Dữ liệu của bảng danhgiacautraloi*

maSinhVien	maCauTraLoi	Upvote	Downvote
1	1	1	0
1	4	1	0
2	1	0	0
2	4	1	0
3	1	0	1
3	4	0	0

Bảng 4.10 Dữ liệu của bảng danhstactag

maTag	maCauHoi
1	2
2	11
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11

Bảng 4.11 Dữ liệu của bảng danhstactailieu

maMon	maTaiLieu
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
10	9
12	10

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

maSinhVien	maNganh	hoTenSV	emailSV	matKhauSV	ngayTao	truongHoc	avatarPath	trangThaiTK
1	1	Nguyễn Văn A	sva@gmail.com	\$2a\$10\$2eieYsa8q3oQanI/dKJOS99UIKgZIWrLyEwPzh5T...	2025-11-18 22:20:04	NULL	NULL	active
2	2	Tran Thi B	svb@gmail.com	\$2a\$10\$X.Mu3TpgSQq4GSvHjIdO1eGmJFzdBGHTXzqaW2vQgk...	2025-11-18 22:21:45	NULL	NULL	active
3	1	Le Van C	svc@gmail.com	\$2a\$10\$bLjmW7Pm/KwTmCH9LCWIBu1iIKsKCh964kQfCv/CjWV...	2025-11-18 22:22:13	NULL	NULL	active
4	1	Hà Gia Lộc	hagialoc@gmail.com	\$2a\$10\$46SbHE19S13/OI.GJmUVieETIuzX5O9TmAh3iRPwcf...	2025-11-19 00:27:50	NULL	NULL	active

Hình 4.1 Dữ liệu của bảng sinhvien

maTaiLieu	maLoai	maDinhDang	maSinhVien	tiêuDeTL	ngayChiaSe	trangThaiTL	filePath	fileSizes	soLanLuu	luotTaiXuong
1	1	1	1	Giáo trình C căn bản	2025-11-18 22:20:04	show	ti1.pdf	50000	10	5
2	2	3	2	Slide CSDL chương 1	2025-11-18 22:21:45	show	ti2.pptx	30000	4	3
3	3	1	3	Đề thi lập trình C	2025-11-18 22:22:13	show	ti3.pdf	20000	2	1
4	4	1	1	Tài liệu thuật toán nâng cao	2025-11-18 22:30:00	show	ti4.pdf	44000	5	2
5	5	2	2	BTL cơ sở dữ liệu	2025-11-18 22:40:00	show	ti5.docx	35000	7	4
6	6	1	3	Bài báo về AI	2025-11-18 22:50:00	show	ti6.pdf	60000	8	6
7	7	1	1	Sổ tay sinh viên CNTT	2025-11-18 23:00:00	show	ti7.pdf	25000	3	2
8	1	1	2	Giáo trình kế toán	2025-11-18 23:10:00	show	ti8.pdf	45000	1	1
9	2	4	3	Slide Marketing	2025-11-18 23:20:00	show	ti9.ppt	30000	0	0
10	4	1	1	Tài liệu dịch thuật	2025-11-18 23:30:00	show	ti10.pdf	38000	9	7

Hình 4.2 Dữ liệu của bảng tailieu

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

maCauTraLoi	maSinhVien	maCauHoi	noiDungCTL	ngayTraLoi	trangThaiCTL
4	1	2	Thiết kế CSDL nên bắt đầu bằng xác định yêu cầu, s...	2025-11-18 23:04:00	show
5	2	2	Bạn nên chuẩn hóa dữ liệu đến 3NF để giảm dư thừa....	2025-11-18 23:05:00	show
6	3	2	Dùng MySQL Workbench để mô hình hóa rất tiện. Bạn ...	2025-11-18 23:06:00	show
7	1	3	QuickSort chia mảng thành 2 phần dựa vào pivot, sa...	2025-11-18 23:07:00	show
8	2	3	Pivot nên chọn phần tử giữa mảng để hạn chế trường...	2025-11-18 23:08:00	show
9	3	3	QuickSort trung bình có độ phức tạp $O(n \log n)$, nh...	2025-11-18 23:09:00	show
10	1	4	Nguyên lý kế toán cung cấp khái niệm tài sản, nguồ...	2025-11-18 23:10:00	show
11	2	4	Công thức kế toán cơ bản: Tài sản = Nợ phải trả + ...	2025-11-18 23:11:00	show
12	3	4	Bạn nên đọc giáo trình chương 1 và 2 để nắm rõ các...	2025-11-18 23:12:00	show
13	1	5	Thuế GTGT đầu ra = Giá tính thuế * thuế suất. Thuế...	2025-11-18 23:13:00	show
14	2	5	Tùy loại hàng hóa mà thuế suất có thể 5% hoặc 10%....	2025-11-18 23:14:00	show
15	3	5	Bạn nên xem bảng thuế GTGT do Bộ Tài chính ban hàn...	2025-11-18 23:15:00	show
16	1	6	Âm hữu thanh khi phát âm sẽ rung dây thanh quản. B...	2025-11-18 23:16:00	show
17	2	6	Ví dụ âm hữu thanh: /b/, /d/, /g/. Âm vô thanh: /p...	2025-11-18 23:17:00	show
18	3	6	Bạn xem video trên YouTube về "voiced vs voiceless...	2025-11-18 23:18:00	show
19	1	7	Dịch câu phải xét ngữ cảnh, người nói, mục đích, t...	2025-11-18 23:19:00	show
20	2	7	Bạn nên dùng từ điển Oxford hoặc Cambridge để tra ...	2025-11-18 23:20:00	show
21	3	7	Hãy đọc nhiều câu mẫu để quen ngữ pháp và cách điể...	2025-11-18 23:21:00	show
22	1	8	4P gồm: Product (sản phẩm), Price (giá), Place (kê...	2025-11-18 23:22:00	show
23	2	8	Ngoài 3P trên, yếu tố cuối là Promotion (xúc tiến)...	2025-11-18 23:23:00	show
24	3	8	Bạn có thể tìm mô hình 4P trong bất kỳ giáo trình ...	2025-11-18 23:24:00	show
25	1	9	Quản trị là quá trình hoạch định, tổ chức, lãnh đạ...	2025-11-18 23:25:00	show
26	2	9	Nhà quản trị có vai trò điều hành nhân sự, đưa ra ...	2025-11-18 23:26:00	show
27	3	9	Quản trị giúp sử dụng hiệu quả nguồn lực như con n...	2025-11-18 23:27:00	show
28	1	10	Quy trình gồm: làm đất, gieo sạ hoặc cấy, chăm sóc...	2025-11-18 23:28:00	show
29	2	10	Giai đoạn chăm sóc cần theo dõi nước và dinh dưỡng...	2025-11-18 23:29:00	show
30	3	10	Thu hoạch khi lúa đạt độ chín vàng 85–90%.	2025-11-18 23:30:00	show
31	4	11	Test	2025-12-04 03:46:46	show

Hình 4.3 Dữ liệu của bảng cautraloi

maBaoCao	maQuanTriVien	maTaiLieu	maSinhVien	maCauHoi	maCauTraLoi	lyDo	moTa	ngayBC	trangThaiBC	loaiBaoCao
1	1	7	1	NULL	NULL	spam	NULL	2025-11-18 23:59:00	pending	document
2	1	3	2	NULL	NULL	inappropriate_content	NULL	2025-11-18 23:59:00	approved	document
3	1	2	3	NULL	NULL	other	Test	2025-11-18 23:59:00	pending	document
4	1	NULL	1	8	NULL	spam	NULL	2025-11-18 23:59:00	pending	question
5	1	NULL	2	3	NULL	misinformation	NULL	2025-11-18 23:59:00	approved	question
6	1	NULL	3	2	NULL	copyright	NULL	2025-11-18 23:59:00	rejected	question
7	1	10	4	NULL	NULL	other	Test 2	2025-12-04 03:27:35	pending	document
8	1	NULL	4	NULL	31	misinformation	NULL	2025-12-04 03:51:18	pending	answer
9	1	NULL	4	11	NULL	spam	NULL	2025-12-04 03:51:27	pending	question

Hình 4.4 Dữ liệu của bảng baocaovipham

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên


maCauHoi	maSinhVien	maMon	tiêuDeCH	noiDungCH	ngayDatCH	trangThaiCH	luotTraLoi	imagePath
1	1	1	Khai báo biến trong C như thế nào?	Mình mới học lập trình C và còn khá mơ hồ về cách ...	2025-11-18 22:11:00	show	3	NULL
2	2	2	Hướng dẫn thiết kế cơ sở dữ liệu từ đầu?	Mình đang muốn tự thiết kế một cơ sở dữ liệu đơn g...	2025-11-18 22:12:00	show	3	NULL
3	3	3	Thuật toán QuickSort hoạt động như thế nào?	Mình biết đây là thuật toán sắp xếp nhanh nhưng vấ...	2025-11-18 22:13:00	show	3	NULL
4	1	4	Khái niệm cơ bản trong Nguyên lý kế toán?	Môn Nguyên lý kế toán có nhiều khái niệm như tài s...	2025-11-18 22:14:00	show	3	NULL
5	2	5	Cách tính thuế GTGT đầu ra và đầu vào?	Mình đang làm bài tập kế toán và bị rối phần tính ...	2025-11-18 22:15:00	show	3	NULL
6	3	7	Phân biệt âm hữu thanh và vô thanh trong tiếng Anh...	Mình học môn Ngữ âm học và gặp phần âm hữu thanh (...)	2025-11-18 22:16:00	show	3	NULL
7	1	8	Hỏi về cách dịch câu tiếng Anh đúng ngữ cảnh?	Mình đang tập dịch các câu trong bài tập nhưng khô...	2025-11-18 22:17:00	show	3	NULL
8	2	10	Marketing Mix (4P) gồm những nội dung gì?	Trong môn Marketing căn bản, nhóm mình được giao t...	2025-11-18 22:18:00	show	3	NULL
9	3	11	Khái niệm quản trị là gì?	Mình học Quản trị học và muốn hiểu đúng khái niệm ...	2025-11-18 22:19:00	show	3	NULL
10	1	13	Quy trình trồng lúa từ lúc làm đất đến thu hoạch?	Mình làm bài tập môn Khoa học cây trồng nhưng chưa...	2025-11-18 22:20:00	show	3	NULL
11	4	2	Test	Test	2025-12-04 03:46:34	show	0	NULL

Hình 4.5 Dữ liệu của bảng cauhoi

4.2. Kết quả thực nghiệm

4.2.1. Trang đăng nhập

Chức năng đăng nhập cho phép người dùng truy cập vào hệ thống bằng tài khoản đã đăng ký. Giao diện hỗ trợ nhập email và mật khẩu, đồng thời hệ thống tự động phân quyền để đảm bảo mỗi loại người dùng được truy cập đúng chức năng phù hợp.



Đăng nhập

Chưa có tài khoản? [Đăng ký ngay](#)

Email

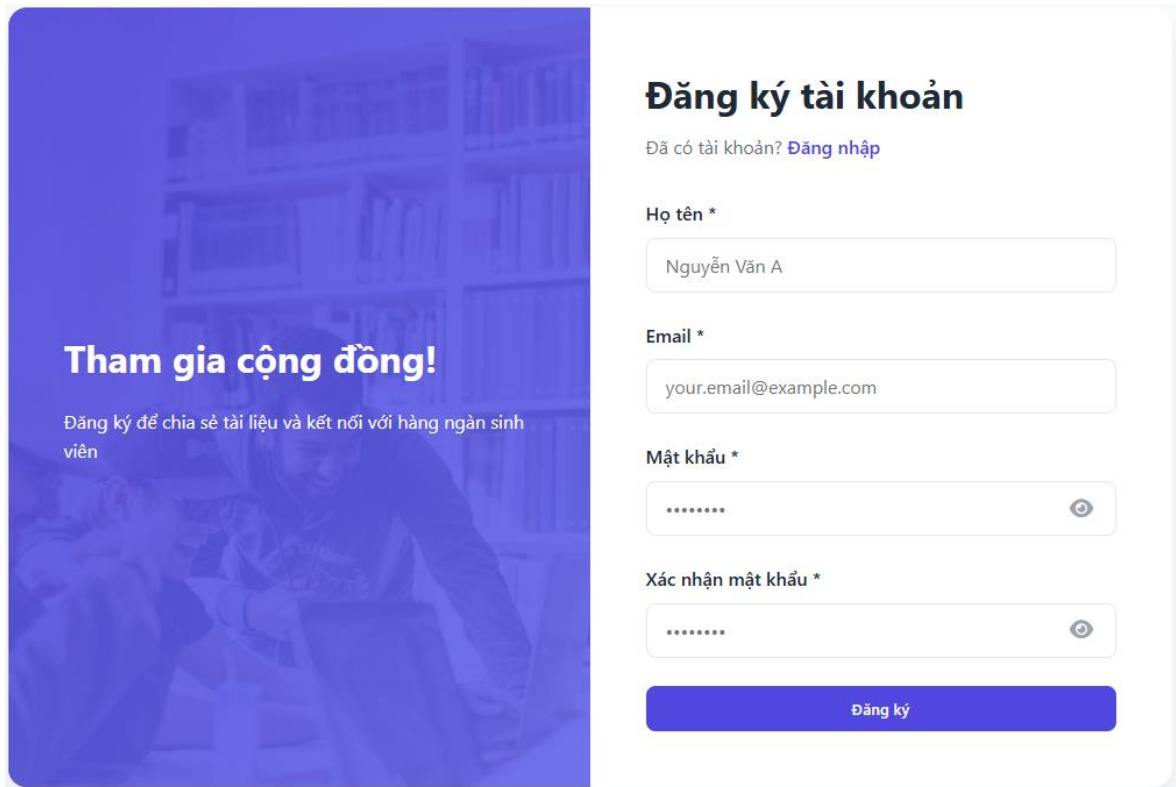
Mật khẩu

☐ Ghi nhớ đăng nhập [Quên mật khẩu?](#)

Hình 4.6 Giao diện đăng nhập

4.2.2. Trang đăng ký

Chức năng đăng ký cho phép sinh viên tạo tài khoản mới bằng cách cung cấp các thông tin cá nhân cần thiết. Giao diện được thiết kế rõ ràng, giúp người dùng dễ dàng hoàn tất quá trình đăng ký trước khi truy cập và sử dụng các chức năng của hệ thống.

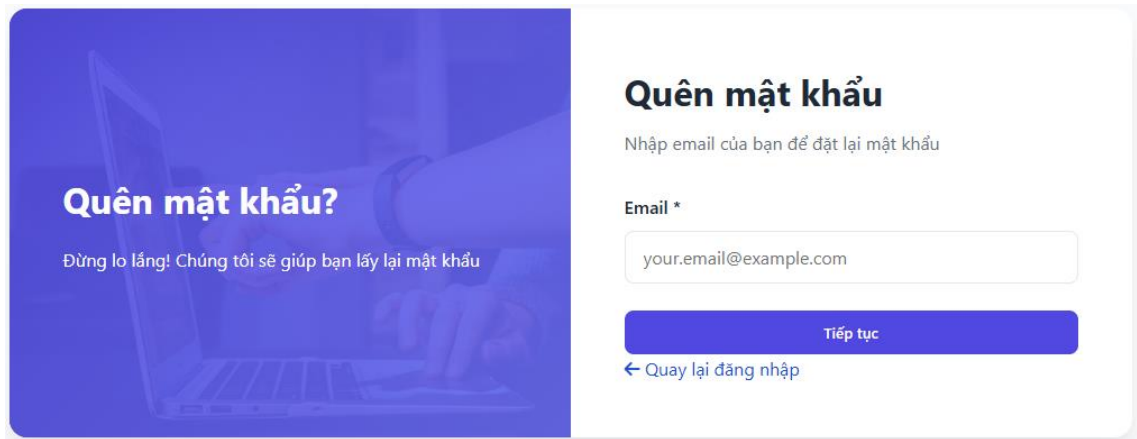


The image shows a web interface for user registration. On the left, there is a blue banner with the text "Tham gia cộng đồng!" (Join the community!) and a subtext "Đăng ký để chia sẻ tài liệu và kết nối với hàng ngàn sinh viên" (Register to share documents and connect with thousands of students). On the right, there is a white registration form titled "Đăng ký tài khoản" (Register account). The form includes a link "Đã có tài khoản? Đăng nhập" (Already have an account? Log in). The form fields are: "Họ tên *" (Full name) with the value "Nguyễn Văn A", "Email *" with the value "your.email@example.com", "Mật khẩu *" (Password) with masked characters "*****", and "Xác nhận mật khẩu *" (Confirm password) with masked characters "*****". A blue "Đăng ký" (Register) button is at the bottom of the form.

Hình 4.7 Giao diện đăng ký

4.2.3. Trang quên mật khẩu

Chức năng quên mật khẩu hỗ trợ người dùng khôi phục quyền truy cập tài khoản thông qua việc nhập email đã đăng ký để nhận liên kết đặt lại mật khẩu. Sau đó, hệ thống cung cấp giao diện thiết lập mật khẩu mới theo các tiêu chí bảo mật. Hai hình dưới đây minh họa màn hình nhập email và màn hình đặt lại mật khẩu trong hệ thống.



Quên mật khẩu?

Đừng lo lắng! Chúng tôi sẽ giúp bạn lấy lại mật khẩu

Quên mật khẩu

Nhập email của bạn để đặt lại mật khẩu

Email *

your.email@example.com

Tiếp tục

[← Quay lại đăng nhập](#)

Hình 4.8 Giao diện nhập email

Đặt lại mật khẩu

Tạo mật khẩu mới an toàn cho tài khoản của bạn

Email: **hagialoc@gmail.com**

Mật khẩu mới *

.....

Mật khẩu phải có ít nhất 6 ký tự

Xác nhận mật khẩu mới *

.....

Đặt lại mật khẩu

i Mẹo tạo mật khẩu mạnh

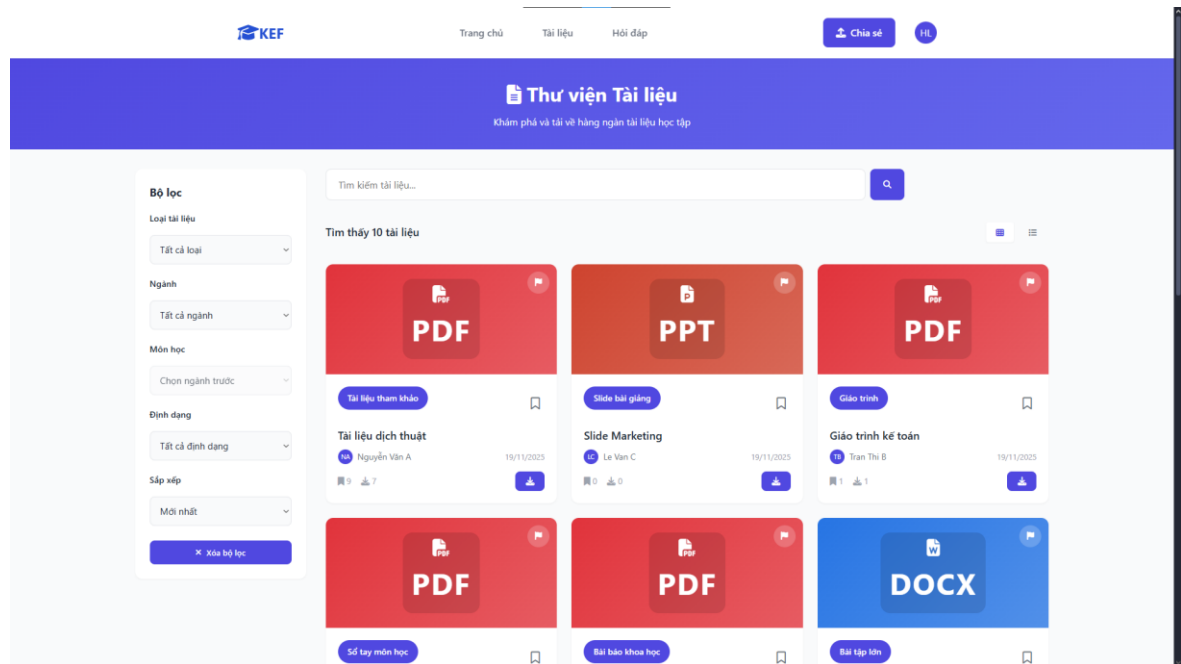
- ✓ Sử dụng ít nhất 8 ký tự
- ✓ Kết hợp chữ hoa, chữ thường, số và ký tự đặc biệt
- ✓ Không sử dụng thông tin cá nhân dễ đoán
- ✓ Không dùng lại mật khẩu của các tài khoản khác

[← Quay lại đăng nhập](#)

Hình 4.9 Giao diện đặt lại mật khẩu

4.2.4. Trang danh sách tài liệu

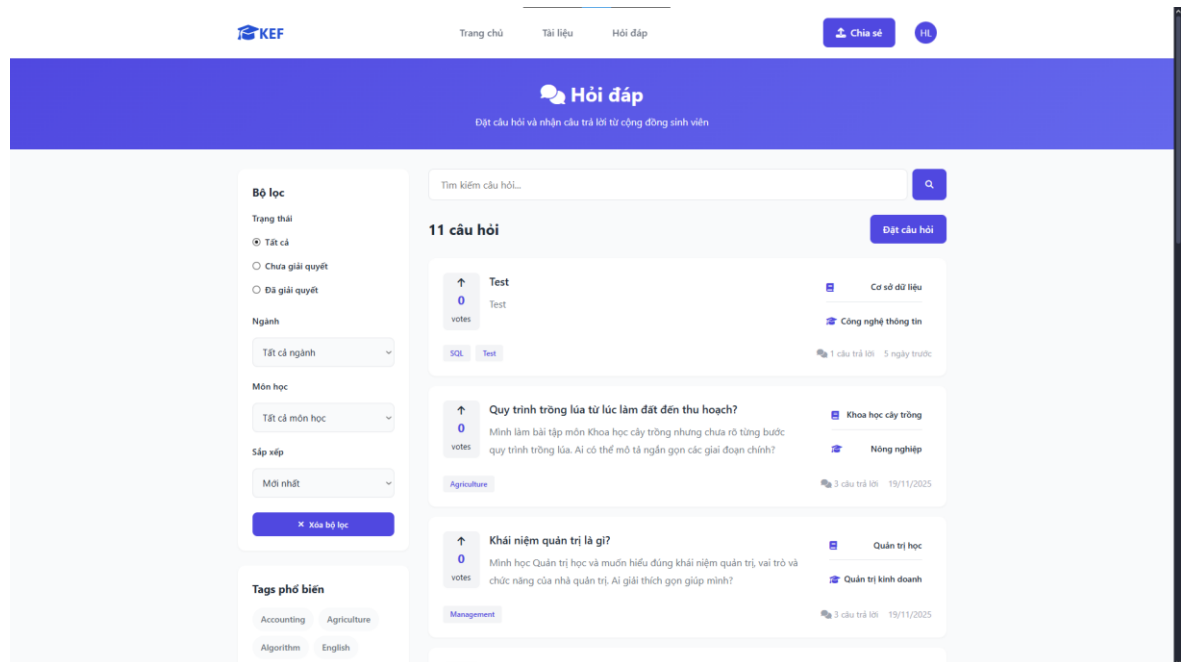
Trang danh sách tài liệu hiển thị đầy đủ các tài liệu hiện có trong hệ thống và hỗ trợ người dùng tìm kiếm nhanh bằng từ khóa hoặc lọc theo môn học, ngành học và loại tài liệu. Giao diện được sắp xếp rõ ràng, giúp người dùng dễ dàng truy cập và lựa chọn tài liệu phù hợp với nhu cầu học tập.



Hình 4.10 Giao diện trang danh sách tài liệu

4.2.5. Trang hỏi đáp

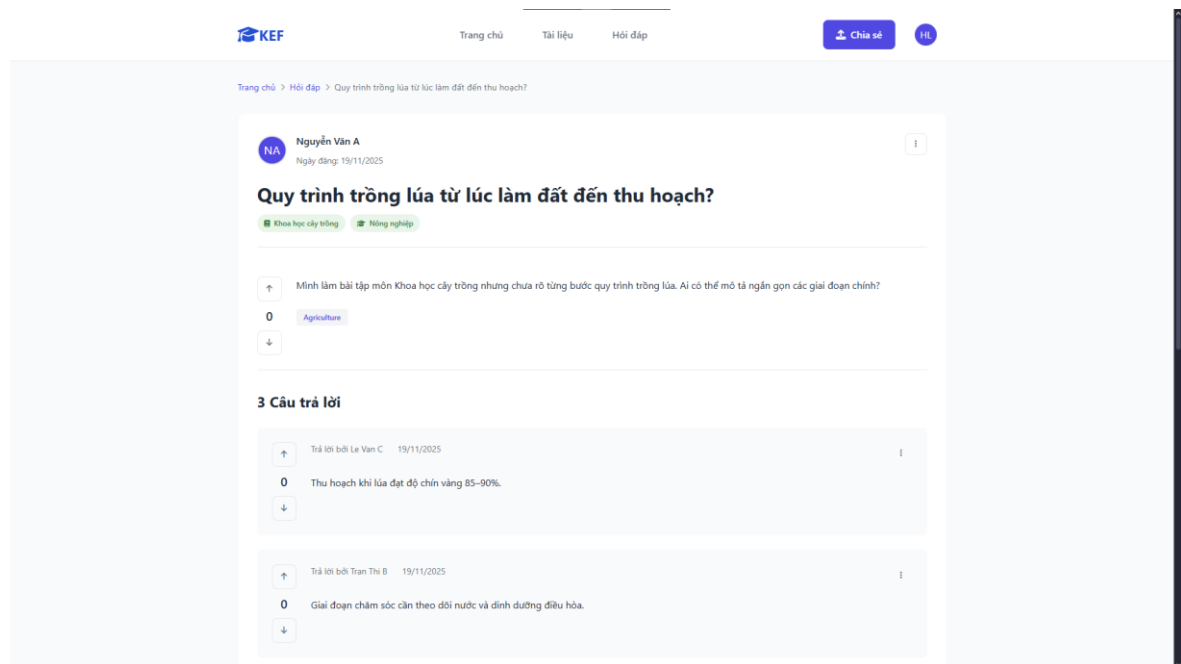
Trang hỏi đáp tổng hợp các câu hỏi do sinh viên đăng tải và cung cấp chức năng tìm kiếm theo từ khóa hoặc lọc theo môn học. Giao diện được bố trí rõ ràng, giúp người dùng nhanh chóng tìm thấy những nội dung phù hợp với nhu cầu học tập của mình.



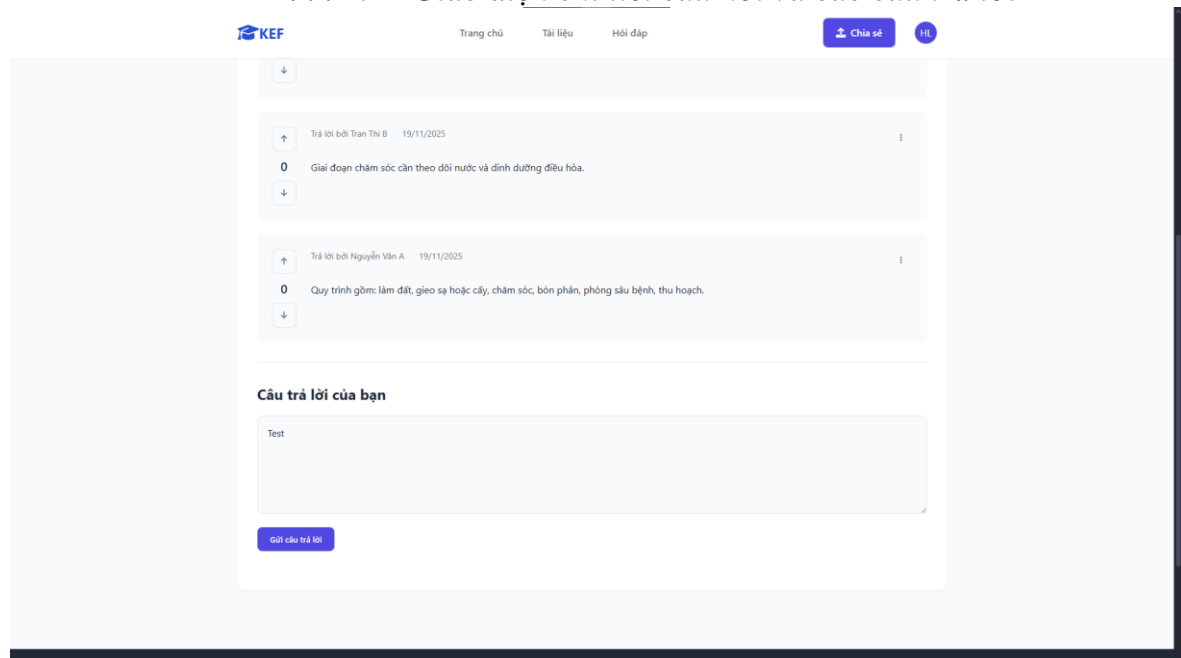
Hình 4.11 Giao diện trang hỏi đáp

4.2.6. Trang chi tiết câu hỏi

Trang chi tiết câu hỏi cho phép người dùng xem đầy đủ nội dung câu hỏi và danh sách các câu trả lời liên quan. Giao diện hỗ trợ đánh giá câu trả lời, đồng thời tạo điều kiện để sinh viên đóng góp lời giải, chia sẻ kiến thức và hỗ trợ cộng đồng học tập một cách trực tiếp.



Hình 4.12 Giao diện chi tiết câu hỏi và các câu trả lời



Hình 4.13 Giao diện nhập câu trả lời

4.2.7. Trang chia sẻ tài liệu

Trang chia sẻ cho phép sinh viên đăng tải các tài liệu học tập mới lên hệ thống, kèm theo tiêu đề, mô tả chi tiết và lựa chọn môn học hoặc loại tài liệu phù hợp. Giao diện được thiết kế trực quan, giúp người dùng dễ dàng nhập thông tin và đóng góp tài liệu cho cộng đồng học tập.

KEF

Trang chủ Tài liệu Hỏi đáp

Chia sẻ HL

Upload Tài liệu

Chia sẻ tài liệu của bạn với cộng đồng sinh viên

Kéo thả file vào đây hoặc click để chọn

Hỗ trợ: PDF, DOC, DOCX, PPT, PPTX, XLS, XLSX (Tối đa 50MB)

Thông tin tài liệu

Tiêu đề *

Nhập tiêu đề tài liệu

Loại tài liệu * Môn học * Ngành *

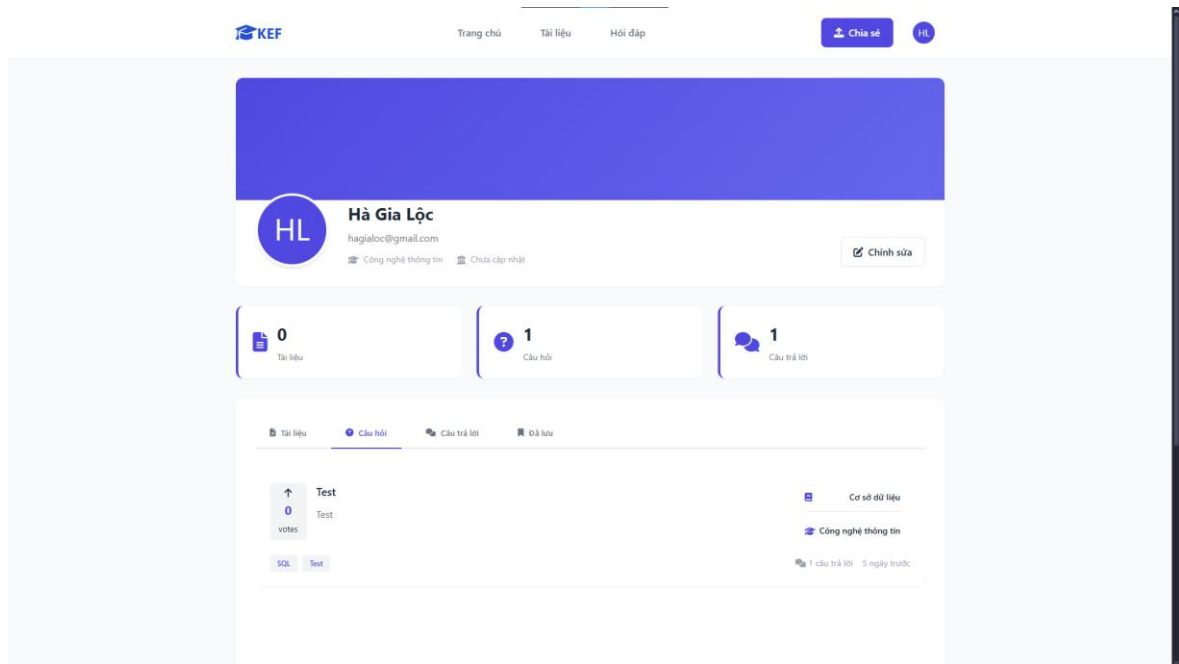
Chọn loại Chọn ngành trước Chọn ngành

Hủy Upload tài liệu

Hình 4.14 Giao diện trang chia sẻ tài liệu

4.2.8. Trang hồ sơ cá nhân

Trang hồ sơ cá nhân hiển thị thông tin cơ bản của người dùng cùng các thống kê hoạt động như số tài liệu đã đăng, số câu hỏi đã tạo và số câu trả lời đã đóng góp. Giao diện cũng cho phép người dùng chỉnh sửa thông tin cá nhân và theo dõi danh sách các bài đăng, câu hỏi và câu trả lời của mình một cách thuận tiện.



Hình 4.15 Giao diện trang hồ sơ cá nhân

4.2.9. Trang chỉnh sửa hồ sơ

Trang chỉnh sửa hồ sơ cho phép người dùng cập nhật các thông tin cá nhân như họ tên, email, trường học và ngành học. Giao diện hiển thị ảnh đại diện cùng tùy chọn thay đổi hình ảnh, đồng thời cung cấp các trường nhập liệu rõ ràng và nút lưu thay đổi giúp người dùng quản lý thông tin của mình một cách dễ dàng và thuận tiện.

The screenshot shows the 'Chỉnh sửa hồ sơ' (Edit Profile) page. The header is blue with the title 'Chỉnh sửa hồ sơ' and the subtitle 'Cập nhật thông tin cá nhân của bạn'. The main content area contains a profile card with a circular profile picture placeholder labeled 'HL' and buttons for 'Chọn ảnh' and 'Ảnh mặc định'. Below this is a form titled 'Thông tin cá nhân' with fields for 'Họ và tên' (filled with 'Hà Gia Lộc'), 'Email' (filled with 'hagialoc@gmail.com'), 'Trường học' (with a 'Chưa cập nhật' button), and 'Ngành' (a dropdown menu showing 'Công nghệ thông tin'). At the bottom right of the form are 'Hủy' and 'Lưu thay đổi' buttons.

Hình 4.16 Giao diện trang chỉnh sửa hồ sơ

4.2.10. Trang đổi mật khẩu

Trang đổi mật khẩu cho phép người dùng thay đổi mật khẩu hiện tại sang mật khẩu mới một cách an toàn. Giao diện bao gồm các trường nhập mật khẩu cũ, mật khẩu mới và xác nhận mật khẩu mới, kèm theo nút lưu thay đổi rõ ràng. Bên dưới còn có các gợi ý tạo mật khẩu mạnh nhằm giúp người dùng nâng cao mức độ bảo mật cho tài khoản của mình.

KEF

Trang chủ Tài liệu Hỏi đáp

Chia sẻ

Đổi mật khẩu

Thay đổi mật khẩu để bảo vệ tài khoản của bạn

Mật khẩu cũ *

.....

Mật khẩu mới *

.....

Mật khẩu phải có ít nhất 6 ký tự

Xác nhận mật khẩu mới *

.....

Hủy Đổi mật khẩu

Mẹo tạo mật khẩu mạnh

- ✓ Sử dụng ít nhất 8 ký tự
- ✓ Kết hợp chữ hoa, chữ thường, số và ký tự đặc biệt
- ✓ Không sử dụng thông tin cá nhân dễ đoán
- ✓ Không dùng lại mật khẩu của các tài khoản khác

Hình 4.17 Giao diện trang đổi mật khẩu

4.2.11. Chức năng báo cáo vi phạm

Chức năng báo cáo vi phạm cho phép người dùng gửi phản hồi khi phát hiện tài liệu, câu hỏi hoặc câu trả lời có nội dung không phù hợp. Giao diện hiển thị rõ nội dung cần báo cáo cùng danh mục lý do như vi phạm bản quyền, sai lệch thông tin, nội dung không phù hợp, spam hoặc các vấn đề khác. Người dùng chỉ cần chọn lý do phù hợp và nhấn nút gửi báo cáo, giúp hệ thống kịp thời xử lý và đảm bảo chất lượng nội dung trên nền tảng.

The image displays three side-by-side screenshots of the 'Báo cáo vi phạm' (Report Violation) form, each for a different content type. Each form has a title bar with a close button (X) and a dropdown menu to select the content type: 'Tài liệu' (Document), 'Câu hỏi' (Question), and 'Câu trả lời' (Answer). Below the title bar, the 'Nội dung báo cáo' (Report Content) field is pre-filled with the content type. The 'Lý do báo cáo' (Reason for report) section contains a list of radio button options. At the bottom, there are 'Hủy' (Cancel) and 'Gửi báo cáo' (Submit report) buttons.

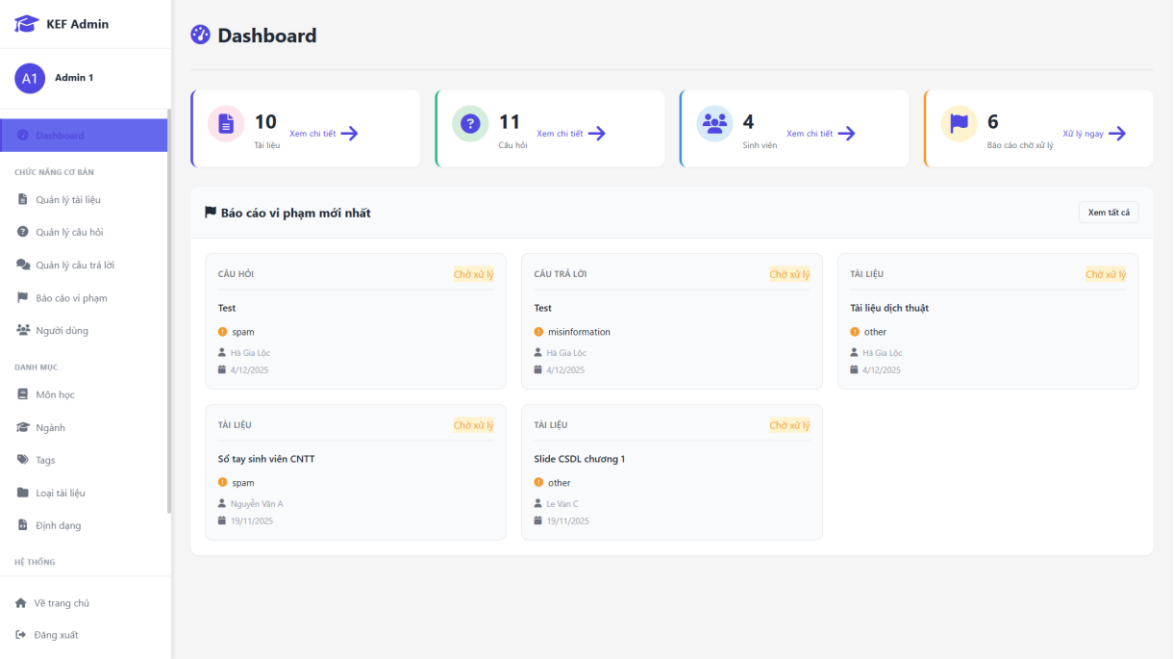
- Form 1 (Tài liệu):** Nội dung báo cáo: Tài liệu dịch thuật. Lý do báo cáo: ☐ Nội dung vi phạm bản quyền, ☐ Tài liệu sai thông tin, ☐ Nội dung không phù hợp, ☐ Spam hoặc quảng cáo, ☐ Lý do khác.
- Form 2 (Câu hỏi):** Nội dung báo cáo: Quy trình trồng lúa từ lúc làm đất đến thu hoạch? Lý do báo cáo: ☐ Câu hỏi trùng lặp, ☐ Nội dung không phù hợp, ☐ Spam, ☐ Câu hỏi không rõ ràng, ☐ Lý do khác.
- Form 3 (Câu trả lời):** Nội dung báo cáo: Thu hoạch khi lúa đạt độ chín vàng 85-90%. Lý do báo cáo: ☐ Thông tin sai lệch, ☐ Nội dung không phù hợp, ☐ Spam hoặc quảng cáo, ☐ Vi phạm quy định, ☐ Lý do khác.

Hình 4.18 Giao diện các form báo cáo vi phạm

4.2.12. Trang dashboard quản trị

Trang dashboard quản trị cung cấp cái nhìn tổng quan về các hoạt động trên hệ thống, bao gồm số lượng tài liệu, câu hỏi, câu trả lời và các báo cáo vi phạm mới nhất. Giao diện được bố cục rõ ràng với các thẻ thống kê giúp quản trị viên nhanh chóng nắm bắt tình hình chung. Bên dưới là danh sách báo cáo vi phạm được phân loại theo loại nội dung, hỗ trợ quản trị viên theo dõi, xử lý và đảm bảo chất lượng nội dung trên nền tảng một cách hiệu quả.

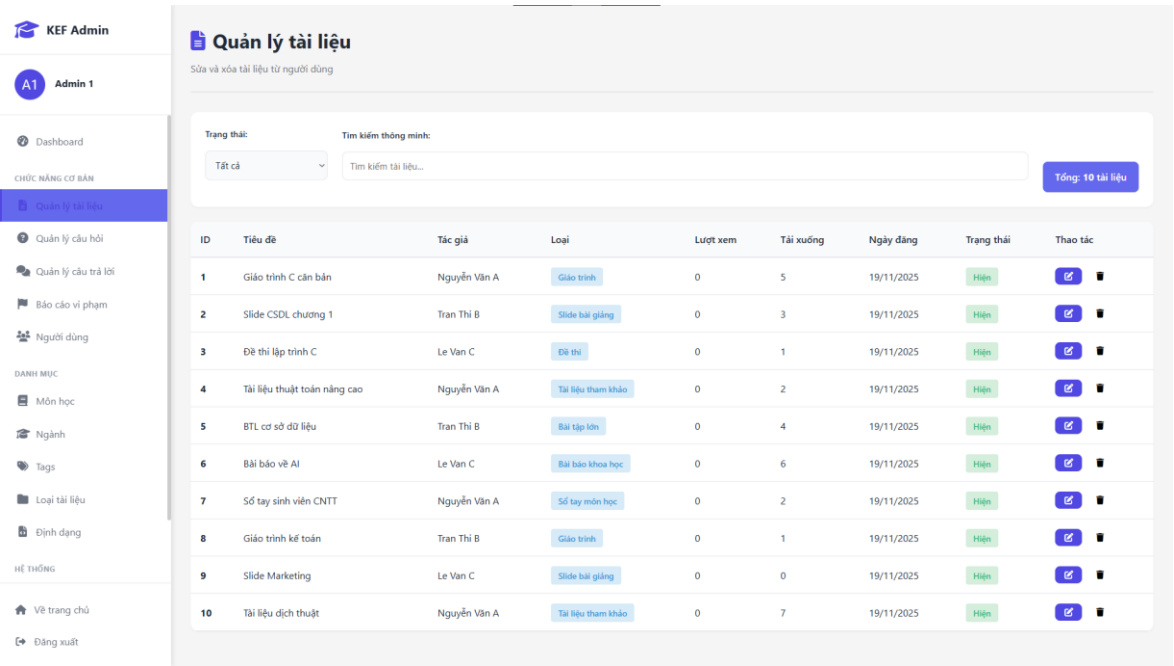
Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên



Hình 4.19 Giao diện dashboard quản trị

4.2.13. Trang quản lý tài liệu

Chức năng quản trị tài liệu cho phép quản trị viên theo dõi, kiểm duyệt và xử lý các tài liệu do người dùng đăng tải. Giao diện hiển thị rõ thông tin tài liệu, giúp quản trị viên dễ dàng phê duyệt, ẩn hoặc xóa những nội dung vi phạm nhằm duy trì môi trường chia sẻ thông tin an toàn và lành mạnh.



Hình 4.20 Giao diện quản lý tài liệu

Chỉnh sửa tài liệu

Mã tài liệu

1

Tiêu đề

Giáo trình C căn bản

Loại tài liệu

Định dạng

.pdf

Ngành

Công nghệ thông tin

Môn học

Lập trình C

Trạng thái

Hiện

Hủy

Lưu thay đổi

Hình 4.21 Giao diện sửa tài liệu

4.2.14. Trang quản lý câu hỏi

Trang quản trị câu hỏi cho phép quản trị viên theo dõi và kiểm duyệt các câu hỏi do người dùng đăng tải. Giao diện cung cấp đầy đủ thông tin cần thiết, hỗ trợ quản trị viên nhanh chóng phát hiện nội dung không phù hợp và thực hiện các thao tác xử lý một cách hiệu quả.

ID	Tiêu đề	Tác giả	Trả lời	Đánh giá	Lượt xem	Ngày đăng	Trạng thái	Thao tác
11	Test	Hà Gia Lộc	1	0	0	5 ngày trước	Hiện	
10	Quy trình trồng lúa từ lúc làm đất đến thu hoạch?	Nguyễn Văn A	3	0	0	19/11/2025	Hiện	
9	Khái niệm quản trị là gì?	Le Van C	3	0	0	19/11/2025	Hiện	
8	Marketing Mix (4P) gồm những nội dung gì?	Tran Thi B	3	0	0	19/11/2025	Hiện	
7	Hỏi về cách dịch câu tiếng Anh đúng ngữ cảnh?	Nguyễn Văn A	3	0	0	19/11/2025	Hiện	
6	Phân biệt âm hữu thanh và vô thanh trong tiếng Anh?	Le Van C	3	0	0	19/11/2025	Hiện	
5	Cách tính thuế GTGT đầu ra và đầu vào?	Tran Thi B	3	0	0	19/11/2025	Hiện	
4	Khái niệm cơ bản trong Nguyên lý kế toán?	Nguyễn Văn A	3	0	0	19/11/2025	Hiện	
3	Thuật toán QuickSort hoạt động như thế nào?	Le Van C	3	+1	0	19/11/2025	Hiện	
2	Hướng dẫn thiết kế cơ sở dữ liệu từ đầu?	Tran Thi B	3	-1	0	19/11/2025	Hiện	
1	Khai báo biến trong C như thế nào?	Nguyễn Văn A	0	0	0	19/11/2025	Hiện	

Hình 4.22 Giao diện trang quản lý câu hỏi

Chỉnh sửa câu hỏi

Mã câu hỏi

11

Đánh giá

0

Tiêu đề

Test

Ngành

Công nghệ thông tin

Môn học

Cơ sở dữ liệu

Nội dung

Test

Trạng thái

Hiện

Hình 4.23 Giao diện chức năng sửa tài liệu

4.2.15. Trang quản lý câu trả lời

Trang quản trị câu trả lời cho phép quản trị viên theo dõi, kiểm duyệt và xử lý các nội dung trả lời do người dùng đăng tải. Giao diện hiển thị đầy đủ thông tin cần thiết, giúp quản trị viên dễ dàng phát hiện các nội dung không phù hợp và thực hiện các thao tác quản lý một cách hiệu quả.

ID	Câu hỏi	Nội dung trả lời	Tác giả	Đánh giá	Ngày trả lời	Trạng thái	Thao tác
31	Test	Test	Hà Gia Lộc	0	5 ngày trước	Hiện	Chỉnh sửa Xóa
30	Quy trình trồng lúa từ lúc làm đất đến thu hoạch?	Thu hoạch khi lúa đạt độ chín vàng 85-9...	Le Van C	0	19/11/2025	Hiện	Chỉnh sửa Xóa
29	Quy trình trồng lúa từ lúc làm đất đến thu hoạch?	Giai đoạn chăm sóc cần theo dõi nước v...	Tran Thi B	0	19/11/2025	Hiện	Chỉnh sửa Xóa
28	Quy trình trồng lúa từ lúc làm đất đến thu hoạch?	Quy trình gồm: làm đất, gieo sạ hoặc cấy...	Nguyễn Văn A	0	19/11/2025	Hiện	Chỉnh sửa Xóa
27	Khái niệm quản trị là gì?	Quản trị giúp sử dụng hiệu quả nguồn L...	Le Van C	0	19/11/2025	Hiện	Chỉnh sửa Xóa
26	Khái niệm quản trị là gì?	Nhà quản trị có vai trò điều hành nhân s...	Tran Thi B	0	19/11/2025	Hiện	Chỉnh sửa Xóa
25	Khái niệm quản trị là gì?	Quản trị là quá trình hoạch định, tổ chức...	Nguyễn Văn A	0	19/11/2025	Hiện	Chỉnh sửa Xóa
24	Marketing Mix (4P) gồm những nội dung gì?	Bạn có thể tìm mô hình 4P trong bất kỳ ...	Le Van C	0	19/11/2025	Hiện	Chỉnh sửa Xóa
23	Marketing Mix (4P) gồm những nội dung gì?	Ngoài 3P trên, yếu tố cuối là Promotion ...	Tran Thi B	0	19/11/2025	Hiện	Chỉnh sửa Xóa
22	Marketing Mix (4P) gồm những nội dung gì?	4P gồm: Product (sản phẩm), Price (giá), ...	Nguyễn Văn A	0	19/11/2025	Hiện	Chỉnh sửa Xóa
21	Hỏi về cách dịch câu tiếng Anh đúng ngữ cảnh?	Hãy đọc nhiều câu mẫu để quen ngữ ph...	Le Van C	0	19/11/2025	Hiện	Chỉnh sửa Xóa

Hình 4.24 Giao diện trang quản lý câu trả lời

Chỉnh sửa câu trả lời

Mã câu trả lời

31

Đánh giá

0

Nội dung

Test

Trạng thái

Hiện

Hủy

Lưu thay đổi

Hình 4.25 Giao diện chức năng sửa câu trả lời

4.2.16. Trang quản lý báo cáo vi phạm

Trang quản lý báo cáo vi phạm cho phép quản trị viên theo dõi và xử lý các báo cáo do người dùng gửi lên. Giao diện hiển thị danh sách đầy đủ gồm nội dung bị báo cáo, loại vi phạm, lý do, người gửi báo cáo, thời gian và trạng thái xử lý. Các bộ lọc theo trạng thái và loại báo cáo giúp quản trị viên dễ dàng tìm kiếm và phân loại thông tin. Hệ thống cũng cung cấp các tùy chọn thao tác trực tiếp trên từng báo cáo, hỗ trợ quá trình kiểm duyệt diễn ra nhanh chóng và hiệu quả.

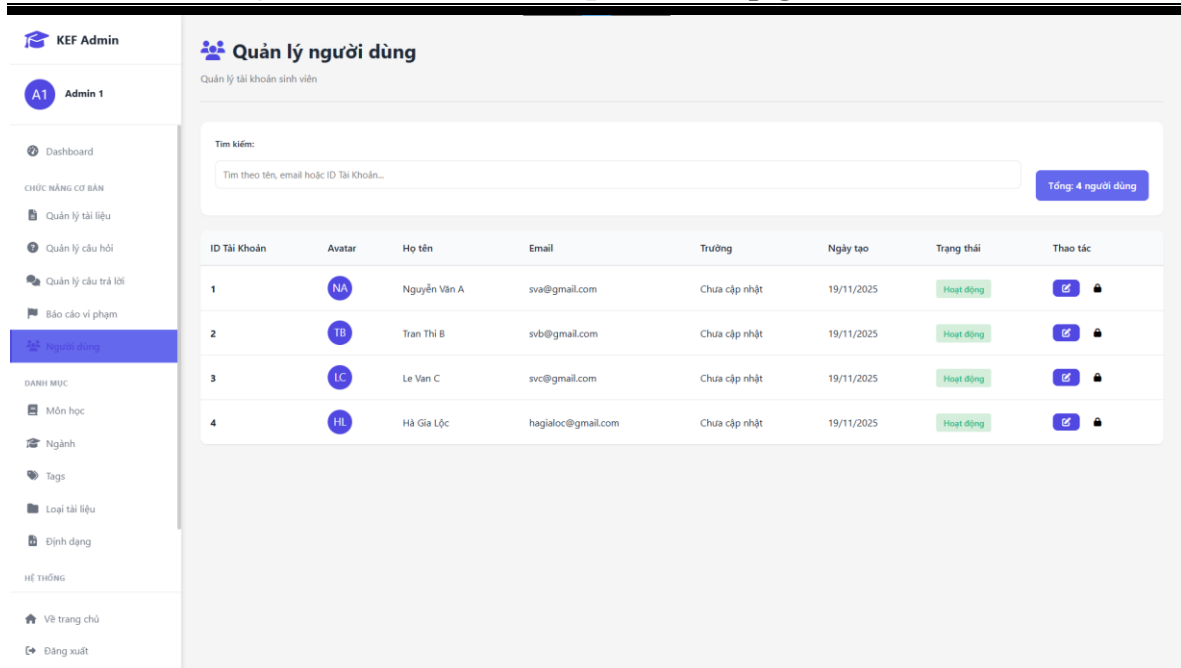
Quản lý báo cáo vi phạm						
Xử lý các báo cáo từ người dùng						
Trạng thái:		Loại báo cáo:		Tổng: 9 báo cáo		
Tất cả		Tất cả				
Đối tượng báo cáo	Loại	Lý do	Người báo cáo	Thời gian báo cáo	Trạng thái	Thao tác
Câu hỏi #11 - Test	Câu hỏi	Spam	Hà Gia Lộc	5 ngày trước	Chờ xử lý	✓ ✕ 🗑
Câu trả lời #31 - Test	Câu trả lời	Thông tin sai lệch	Hà Gia Lộc	5 ngày trước	Chờ xử lý	✓ ✕ 🗑
Tài liệu #10 - Tài liệu dịch thuật	Tài liệu	Khắc	Hà Gia Lộc	5 ngày trước	Chờ xử lý	✓ ✕ 🗑
Tài liệu #7 - Sổ tay sinh viên CNTT	Tài liệu	Spam	Nguyễn Văn A	19/11/2025	Chờ xử lý	✓ ✕ 🗑
Tài liệu #3 - Đề thi lập trình C	Tài liệu	Nội dung không phù hợp	Trần Thị B	19/11/2025	Đã duyệt	🗑
Tài liệu #2 - Slide CSDL chương 1	Tài liệu	Khắc	Le Van C	19/11/2025	Chờ xử lý	✓ ✕ 🗑
Câu hỏi #0 - Marketing Mix (4P) gồm những nội dung gì?	Câu hỏi	Spam	Nguyễn Văn A	19/11/2025	Chờ xử lý	✓ ✕ 🗑
Câu hỏi #3 - Thuật toán QuickSort hoạt động như thế nào?	Câu hỏi	Thông tin sai lệch	Trần Thị B	19/11/2025	Đã duyệt	🗑

Hình 4.26 Giao diện quản lý báo cáo vi phạm

4.2.17. Trang quản lý người dùng

Chức năng quản lý người dùng cho phép quản trị viên theo dõi và điều chỉnh thông tin tài khoản của người dùng, bao gồm thêm mới, chỉnh sửa hoặc khóa tài khoản khi cần thiết. Giao diện được tổ chức rõ ràng, hỗ trợ quản trị viên kiểm soát hoạt động của người dùng và đảm bảo môi trường hệ thống luôn an toàn, ổn định.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên



Hình 4.27 Giao diện trang quản lý người dùng

Chỉnh sửa thông tin sinh viên

ID Tài Khoản

1

Họ tên *

Nguyễn Văn A

Email *

sva@gmail.com

Mật khẩu

Để trống nếu không muốn thay đổi

Trường học

Ngành

Công nghệ thông tin

Trạng thái tài khoản

Hoạt động

Hủy

Lưu thay đổi

Hình 4.28 Giao diện sửa thông tin sinh viên

5.1. Kết luận

Dự án “Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên” đã hoàn thành các mục tiêu đặt ra, xây dựng được một hệ thống hỗ trợ sinh viên tra cứu tài liệu, đặt câu hỏi và trao đổi học thuật một cách thuận tiện và hiệu quả. Ứng dụng tích hợp các chức năng cốt lõi như tìm kiếm tài liệu theo môn học hoặc từ khóa, chia sẻ tài liệu, đặt và trả lời câu hỏi, cùng với hệ thống quản trị cho phép giám sát, kiểm duyệt và quản lý người dùng. Giao diện đơn giản, hiện đại và dễ sử dụng giúp mang lại trải nghiệm tốt cho sinh viên khi truy cập hệ thống.

Bên cạnh đó, việc ứng dụng các công nghệ như ReactJS, NodeJS và MySQL giúp hệ thống vận hành ổn định, bảo mật và dễ mở rộng. Những kết quả đạt được không chỉ thể hiện sự thành công của dự án mà còn tạo nền tảng vững chắc để phát triển thêm nhiều tính năng nâng cao trong tương lai.

5.2. Hạn chế

- Mặc dù dự án đã hoàn thiện các chức năng cốt lõi, hệ thống vẫn còn tồn tại một số hạn chế như sau:
 - Hệ thống chưa sử dụng dịch vụ lưu trữ đám mây: Tập hiện được lưu trực tiếp trong thư mục backend, nằm chung với mã nguồn, thay vì sử dụng dịch vụ cloud storage (Amazon S3, Cloudflare R2). Điều này gây khó mở rộng và tăng rủi ro quá tải dung lượng khi số lượng tệp ngày càng lớn.
 - Không có chức năng thông báo: Hệ thống chưa tích hợp tính năng thông báo và cũng chưa hỗ trợ thông báo theo thời gian thực (WebSocket hoặc SSE). Do đó, người dùng không nhận được cảnh báo ngay khi có câu trả lời mới hoặc khi báo cáo được xử lý.
 - Chỉ hỗ trợ một ngôn ngữ: Hệ thống chưa có tính năng đa ngôn ngữ, làm giảm khả năng tiếp cận người dùng ở phạm vi rộng hơn hoặc khi muốn mở rộng dự án trong tương lai.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

- Trang thống kê còn đơn giản: Dashboard thống kê chỉ hiển thị số liệu cơ bản, chưa có biểu đồ trực quan, thống kê theo thời gian hoặc phân tích xu hướng để hỗ trợ quản trị viên trong việc ra quyết định.

- Khó khôi phục dữ liệu đã xóa: Các thao tác xóa hiện tại đều là hard delete, dữ liệu bị xóa vĩnh viễn và không thể khôi phục, gây rủi ro mất dữ liệu nếu thao tác nhầm.

- Tối ưu tệp tải lên chưa tốt: Các tệp (đặc biệt là hình ảnh) chưa được nén hoặc tối ưu trước khi lưu, dẫn đến tiêu tốn dung lượng lưu trữ và có thể làm chậm tốc độ tải.

- Frontend chưa tối ưu: Ứng dụng chưa áp dụng kỹ thuật code splitting hoặc lazy loading cho các thành phần lớn, khiến thời gian tải trang ban đầu bị chậm.

- Thiếu cơ chế hạn chế tần suất gửi yêu cầu: Hệ thống chưa triển khai rate limiting, dễ bị tấn công brute force ở các chức năng đăng nhập và đăng ký.

- Chính sách mật khẩu còn yếu: Mật khẩu chỉ yêu cầu tối thiểu 6 ký tự, không bắt buộc ký tự đặc biệt, chữ hoa hoặc số, dẫn đến mức độ bảo mật thấp.

- Token chưa an toàn: JWT được lưu trong localStorage hoặc sessionStorage, dễ bị tấn công XSS. Ngoài ra, hệ thống chưa triển khai cơ chế refresh token, làm tăng rủi ro khi token bị đánh cắp.

5.3. Hướng phát triển

Để mở rộng và nâng cao chất lượng hệ thống trong tương lai, một số hướng phát triển sau có thể được triển khai:

- **Tích hợp AI và các tính năng hỗ trợ thông minh:** Bổ sung khả năng gợi ý tài liệu phù hợp, đề xuất câu hỏi liên quan hoặc chatbot hỗ trợ người dùng tra cứu nhanh thông tin.

- **Hỗ trợ đa ngôn ngữ:** Mở rộng giao diện sang nhiều ngôn ngữ khác nhau để tăng khả năng tiếp cận và phục vụ tốt hơn cho người dùng ngoài phạm vi tiếng Việt.

Thiết kế hệ thống chia sẻ tài liệu học tập và hỏi đáp giữa sinh viên

- **Nâng cấp hệ thống thông báo:** Xây dựng cơ chế thông báo trong ứng dụng và thông báo theo thời gian thực để người dùng nắm bắt ngay các hoạt động như có câu trả lời mới, tài liệu được duyệt hoặc báo cáo được xử lý.
- **Tối ưu bảo mật:** Tăng cường cơ chế bảo vệ tài khoản, áp dụng chính sách mật khẩu mạnh, cơ chế refresh token và hệ thống phát hiện tấn công nhằm đảm bảo an toàn cho người dùng.
- **Cải thiện trang thống kê dành cho quản trị viên:** Cải thiện trang thống kê với biểu đồ theo thời gian, thống kê tăng trưởng nội dung và phân tích xu hướng giúp quản trị viên đưa ra quyết định chính xác hơn.
- **Triển khai lưu trữ đám mây:** Chuyển tệp sang các dịch vụ cloud storage (Amazon S3, Cloudflare R2...) để tăng khả năng mở rộng và đảm bảo hiệu suất lưu trữ.
- **Tối ưu hiệu suất hệ thống:** Áp dụng nén ảnh, tối ưu tệp tải lên, sử dụng code splitting và lazy loading để giảm thời gian tải trang và cải thiện trải nghiệm sử dụng.
- **Phát triển ứng dụng di động:** Xây dựng phiên bản dành cho Android và iOS nhằm nâng cao sự tiện lợi và khả năng tiếp cận của người dùng.

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] TopDev, "Kiến trúc phân lớp (Layered Architecture)," [Online]. Available: <https://topdev.vn/blog/kien-truc-phan-lop-layered-architecture/>. [Accessed 18 11 2025].
- [2] D. Abramov, "React Documentation," 16 3 2023. [Online]. Available: <https://react.dev/>. [Accessed 18 11 2025].
- [3] W3Schools, "W3Schools Online Web Tutorials," [Online]. Available: <https://www.w3schools.com>. [Accessed 18 11 2025].
- [4] OpenJS Foundation – Node.js Contributors, "Node.js Documentation," 27 5 2009. [Online]. Available: <https://nodejs.org/en/docs>. [Accessed 18 11 2025].
- [5] MySQL Documentation Team, "MySQL Reference Manual," [Online]. Available: <https://dev.mysql.com/doc>. [Accessed 18 11 2025].