

TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

-----***-----



LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG DỮ LIỆU

PHÁT TRIỂN ỨNG DỤNG WEB HỖ TRỢ
QUẢN LÝ QUÁ TRÌNH THỰC HIỆN ĐỀ TÀI
TRONG SINH VIÊN

Sinh viên: Nguyễn Thành Nghĩa
Mã số: B2004736
Khoá: 46

Học Kỳ II: 2023 – 2024

TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG
KHOA MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG DỮ LIỆU

-----***-----



LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC
MẠNG MÁY TÍNH VÀ TRUYỀN THÔNG DỮ LIỆU

PHÁT TRIỂN ỨNG DỤNG WEB HỖ TRỢ
QUẢN LÝ QUÁ TRÌNH THỰC HIỆN ĐỀ TÀI
TRONG SINH VIÊN
(DEVELOPMENT OF WEB APPLICATION FOR
STUDENT PROJECT MANAGEMENT)

Giảng viên hướng dẫn:
TS. Lâm Chí Nguyên

Sinh viên thực hiện:
Nguyễn Thành Nghĩa
Mã số: B2004736
Khóa: 46

Học Kỳ II: 2023 – 2024

LỜI CẢM ƠN

Em xin gửi lời cảm ơn chân thành đến TS. Lâm Chí Nguyên đã nhiệt tình hướng dẫn và giúp đỡ em trong quá trình hoàn thành đề tài luận văn tốt nghiệp.

Xin gửi lời cảm ơn đến Ban Giám hiệu, toàn thể thầy cô của trường Đại học Cần Thơ nói chung và trường Công nghệ thông tin và Truyền thông nói riêng, đã giảng dạy, giúp đỡ em hoàn thành tốt các môn học của chương trình đào tạo. Xin cảm ơn gia đình đã tạo điều kiện tốt nhất cho em được học tập tại trường Đại học Cần Thơ, đã động viên em vượt qua những lúc khó khăn trong quá trình học tập. Cảm ơn bạn bè đã đồng hành và giúp đỡ em, cũng như đưa ra những lời khuyên, cách giải quyết vấn đề khi em gặp khó khăn trong học tập. Cuối cùng, em xin gửi lời cảm ơn đến cộng đồng lập trình viên trong nước và quốc tế đã giúp đỡ em giải quyết những vấn đề khó khăn gặp phải khi thực hiện đề tài.

Em xin trân trọng cảm ơn.

Cần Thơ, ngày ... tháng ... năm 20...

Sinh viên thực hiện

Nguyễn Thành Nghĩa

LỜI CAM ĐOAN

Em xin cam đoan luận văn “Phát triển ứng dụng Web hỗ trợ quản lý quá trình thực hiện đề tài trong sinh viên” được hoàn thành hoàn toàn dựa trên kết quả nghiên cứu của em dưới sự hướng dẫn của TS. Lâm Chí Nguyên, các nguồn tài liệu tham khảo đã được ghi rõ trong danh mục tài liệu tham khảo.

Cần Thơ, ngày ... tháng ... năm 20...

Sinh viên thực hiện

Nguyễn Thành Nghĩa

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

Cần Thơ, ngày ... tháng ... năm 20...

Giảng viên hướng dẫn

MỤC LỤC

PHẦN GIỚI THIỆU	1
1. ĐẶT VẤN ĐỀ	1
2. LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ	3
3. MỤC TIÊU ĐỀ TÀI	4
4. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU	4
4.1. Đối tượng	4
4.2. Phạm vi nghiên cứu	5
5. BỐ CỤC LUẬN VĂN	5
PHẦN NỘI DUNG	6
CHƯƠNG 1: CƠ SỞ LÝ THUYẾT	6
1. QUY TRÌNH THỰC HIỆN ĐỀ TÀI	6
1.1. Đăng ký tài khoản	6
1.2. Đề xuất đề tài	7
1.3. Đăng ký thực hiện đề tài	8
1.4. Quản lý thực hiện đề tài	9
2. CÔNG NGHỆ MERN STACK	10
2.1. Khái quát về MERN Stack	10
2.2. Node.js	10
2.3. Express.js	11
2.4. React.js	12
2.5. Axios	12
2.6. MongoDB	13
2.7. Mongoose	14
2.8. Cách MERN Stack hoạt động	14
CHƯƠNG 2: MÔ TẢ BÀI TOÁN	16
1. MÔ TẢ CHI TIẾT BÀI TOÁN	16
1.1. Quản trị viên	16
1.2. Giảng viên	17
1.3. Sinh viên	18
2. CÁC CHỨC NĂNG CỦA HỆ THỐNG	19

2.1. Chức năng đăng nhập	19
2.2. Chức năng quản lý thông tin cá nhân	20
2.3. Chức năng quản lý giảng viên	21
2.4. Chức năng quản lý sinh viên	22
2.5. Chức năng quản lý đề tài	23
2.6. Chức năng quản lý công việc	26
3. CÁC YÊU CẦU PHI CHỨC NĂNG	28
3.1. Yêu cầu thực thi	28
3.2. Yêu cầu an toàn	28
3.3. Yêu cầu bảo mật	28
3.4. Chất lượng phần mềm	28
3.5. Các yêu cầu khác	28
CHƯƠNG 3: XÂY DỰNG VÀ PHÁT TRIỂN PHẦN MỀM	29
1. THIẾT KẾ DỮ LIỆU	29
1.1. Mô hình dữ liệu	29
1.2. Collection Users	30
1.3. Collection Projects	30
1.4. Collection Registers	31
1.5. Collection Notifications	31
1.6. Collection Messages	31
1.7. Collection Tasks	32
1.8. Collection Submissions	32
2. THIẾT KẾ THEO CHỨC NĂNG	33
2.1. Chức năng đăng nhập	33
2.2. Chức năng đăng ký	34
2.3. Chức năng đổi mật khẩu	35
2.4. Chức năng cập nhật thông tin	35
2.5. Chức năng thêm đề tài	37
2.6. Chức năng cập nhật đề tài	38
2.7. Chức năng ẩn đề tài	39
2.8. Chức năng tìm kiếm đề tài	40

2.9. Chức năng đăng ký đề tài	41
2.10. Chức năng thêm công việc	42
2.11. Chức năng nhắc nhở và gia hạn	43
2.12. Chức năng tải file và đánh giá	44
2.13. Chức năng nộp bài	45
2.14. Chức năng hoàn thành và chấm dứt đề tài	46
PHẦN KẾT LUẬN	49
1. KẾT QUẢ ĐẠT ĐƯỢC	49
1.1. Về lý thuyết	49
1.2. Về chương trình	49
2. HẠN CHẾ	49
3. HƯỚNG PHÁT TRIỂN	49
TÀI LIỆU THAM KHẢO	50

DANH MỤC HÌNH ẢNH

Hình 1-1 : Sơ đồ đăng ký tài khoản	6
Hình 1-2 : Sơ đồ đề xuất đề tài	7
Hình 1-3 : Sơ đồ đăng ký thực hiện đề tài	8
Hình 1-4 : Sơ đồ quản lý thực hiện đề tài	9
Hình 1-5 : Mô hình khái quát về MERN Stack	10
Hình 1-6 : Mô hình hoạt động của MERN Stack	15
Hình 2-1 : Sơ đồ chức năng quản trị viên	16
Hình 2-2 : Sơ đồ chức năng giảng viên	17
Hình 2-3 : Sơ đồ chức năng sinh viên	18
Hình 2-4 : Chức năng đăng nhập	19
Hình 2-5 : Chức năng quản lý thông tin cá nhân	20
Hình 2-6 : Chức năng quản lý giảng viên	21
Hình 2-7 : Chức năng quản lý sinh viên	22
Hình 2-8 : Chức năng quản lý đề tài đối với quản trị viên	23
Hình 2-9 : Chức năng quản lý đề tài đối với giảng viên	24
Hình 2-10 : Chức năng quản lý đề tài đối với sinh viên	25
Hình 2-11 : Chức năng quản lý công việc đối với giảng viên	26
Hình 2-12 : Chức năng quản lý công việc đối với sinh viên	27
Hình 3-1 : Mô hình dữ liệu của ứng dụng	29
Hình 3-2 : Bảng liên kết giữa các collections	29
Hình 3-3 : Giao diện đăng nhập	33
Hình 3-4 : Giao diện đăng ký	34
Hình 3-5 : Giao diện đổi mật khẩu	35
Hình 3-6 : Giao diện cập nhật thông tin người dùng	36
Hình 3-7 : Giao diện thêm đề tài	37
Hình 3-8 : Giao diện cập nhật đề tài	38
Hình 3-9 : Giao diện ẩn đề tài	39
Hình 3-10 : Giao diện xác nhận ẩn đề tài	39
Hình 3-11 : Giao diện tìm kiếm đề tài	40
Hình 3-12 : Giao diện đăng ký đề tài	41

Hình 3-13 : Giao diện xác nhận đăng ký đề tài	41
Hình 3-14 : Giao diện thêm công việc	42
Hình 3-15 : Giao diện nhắc nhở và gia hạn	43
Hình 3-16 : Giao diện nhắc nhở	43
Hình 3-17 : Giao diện gia hạn	43
Hình 3-18 : Giao diện tải file và đánh giá	44
Hình 3-19 : Giao diện đánh giá	45
Hình 3-20 : Giao diện nộp bài	46
Hình 3-21 : Giao diện hoàn thành và chấm dứt đề tài	47
Hình 3-22 : Giao diện hoàn thành đề tài	47
Hình 3-23 : Giao diện chấm dứt đề tài	47

DANH MỤC BẢNG

Bảng 2-1 : Mô tả chức năng đăng nhập	19
Bảng 2-2 : Mô tả chức năng quản lý thông tin cá nhân	20
Bảng 2-3 : Mô tả chức năng quản lý giảng viên	21
Bảng 2-4 : Mô tả chức năng quản lý sinh viên	22
Bảng 2-5 : Mô tả chức năng quản lý đề tài đối với quản trị viên	23
Bảng 2-6 : Mô tả chức năng quản lý đề tài đối với giảng viên	24
Bảng 2-7 : Mô tả chức năng quản lý đề tài đối với sinh viên	25
Bảng 2-8 : Mô tả chức năng quản lý công việc đối với giảng viên	26
Bảng 2-9 : Mô tả chức năng quản lý công việc đối với sinh viên	27
Bảng 3-1 : Thành phần giao diện đăng nhập	33
Bảng 3-2 : Thành phần giao diện đăng ký	34
Bảng 3-3 : Thành phần giao diện đổi mật khẩu	35
Bảng 3-4 : Thành phần giao diện cập nhật thông tin người dùng	36
Bảng 3-5 : Thành phần giao diện thêm đề tài	37
Bảng 3-6 : Thành phần giao diện cập nhật đề tài	38
Bảng 3-7 : Thành phần giao diện ẩn đề tài	39
Bảng 3-8 : Thành phần giao diện tìm kiếm đề tài	40
Bảng 3-9 : Thành phần giao diện đăng ký đề tài	41
Bảng 3-10 : Thành phần giao diện thêm công việc	42
Bảng 3-11 : Thành phần giao diện nhắc nhở và gia hạn	44
Bảng 3-12 : Thành phần giao diện tải file và đánh giá	45
Bảng 3-13 : Thành phần giao diện nộp bài	46
Bảng 3-14 : Thành phần giao diện hoàn thành và chấm dứt đề tài	48

KÝ HIỆU VÀ CÁC THUẬT NGỮ VIẾT TẮT

STT	Thuật ngữ/Viết tắt	Định nghĩa/Giải thích
1	API	Viết tắt của Application Programming Interface là một tập hợp các quy tắc và giao thức được sử dụng để xây dựng và tương tác với phần mềm ứng dụng và dịch vụ khác nhau.
5	JS (Javascript)	Ngôn ngữ lập trình Javascript.
6	CSDL	Cơ sở dữ liệu.
8	CRUD	CRUD là viết tắt của Create, Read, Update và Delete, đại diện cho các thao tác cơ bản trên dữ liệu gồm tạo, đọc, cập nhật và xóa.
2	JSON	Viết tắt của JavaScript Object Notation là một định dạng dữ liệu nhẹ, dễ đọc và dễ hiểu được sử dụng để truyền tải và lưu trữ dữ liệu.
3	DOM	DOM là viết tắt của Document Object Model (Mô hình Đối tượng Tài liệu). Trong ngữ cảnh của web, DOM là một cách tiếp cận tiêu chuẩn được sử dụng để truy cập và tương tác với các thành phần trên trang web, chẳng hạn như các phần tử HTML, CSS và JavaScript.
4	HTTP	HTTP là viết tắt của HyperText Transfer Protocol. Đây là một trong những giao thức phổ biến nhất được sử dụng trên Internet để truyền tải dữ liệu giữa máy chủ và máy khách.
7	URL	URL là viết tắt của Uniform Resource Locator, là một chuỗi ký tự được sử dụng để xác định địa chỉ của một tài nguyên trên internet.

TÓM TẮT

Trong môi trường nghiên cứu và phát triển hiện đại, việc quản lý và theo dõi tiến độ thực hiện các đề tài nghiên cứu đóng vai trò cực kỳ quan trọng để đảm bảo sự thành công và hiệu quả của dự án. Các dự án nghiên cứu thường đòi hỏi sự phối hợp chặt chẽ giữa giảng viên và sinh viên. Thêm vào đó, quá trình nghiên cứu thường diễn ra qua nhiều giai đoạn khác nhau, với các công việc phức tạp và mục tiêu rõ ràng. Do đó, việc có một hệ thống quản lý tiến độ hiệu quả là chìa khóa để đảm bảo dự án được thực hiện đúng hạn và đạt được mục tiêu đề ra.

Đề tài này ra đời với mục tiêu giải quyết những thách thức này thông qua việc phát triển một ứng dụng web sử dụng công nghệ MERN stack. MERN stack được chọn lựa vì tính linh hoạt, hiệu suất và khả năng mở rộng của nó. MongoDB cung cấp một cơ sở dữ liệu không cấu trúc linh hoạt, giúp lưu trữ dữ liệu nhanh chóng và dễ dàng. Express.js là một framework mạnh mẽ cho Node.js, giúp xây dựng các ứng dụng web một cách nhanh chóng và hiệu quả. React.js cung cấp giao diện người dùng tương tác và thân thiện, giúp người dùng tương tác với ứng dụng một cách dễ dàng và thuận tiện. Cuối cùng, Node.js là một môi trường thực thi mã JavaScript phía máy chủ, cho phép triển khai ứng dụng một cách linh hoạt và hiệu quả.

Ứng dụng Web hỗ trợ quản lý quá trình thực hiện đề tài trong sinh viên được xây dựng dựa trên công nghệ MERN stack mang lại nhiều lợi ích đối với nhà trường, giảng viên và sinh viên. Với nhà trường, việc sử dụng ứng dụng này giúp quản lý hoạt động nghiên cứu của giảng viên và sinh viên trở nên hiệu quả hơn, từ đó nâng cao chất lượng đào tạo. Với giảng viên, ứng dụng cung cấp một công cụ quản lý tiện lợi, giúp theo dõi tiến độ thực hiện đề tài của sinh viên và cung cấp phản hồi nhanh chóng. Đồng thời, sinh viên cũng được hưởng lợi từ tính linh hoạt và tiện lợi của ứng dụng, giúp họ quản lý công việc và nhận phản hồi từ giảng viên một cách dễ dàng.

ABSTRACT

In the modern research and development environment, managing and monitoring the progress of research projects plays an extremely important role to ensure the success and effectiveness of the project. Research projects often require close coordination between lecturers and students. In addition, the research process often takes place through many different stages, with complex tasks and clear goals. Therefore, having an effective schedule management system is key to ensuring the project is completed on time and achieves its goals.

This project was born with the goal of solving these challenges through developing a web application using MERN stack technology. MERN stack was chosen for its flexibility, performance and scalability. MongoDB provides a flexible unstructured database that makes data storage quick and easy. Express.js is a powerful framework for Node.js that helps build web applications quickly and efficiently. React.js provides an interactive and friendly user interface, helping users interact with the application easily and conveniently. Finally, Node.js is a server-side JavaScript code execution environment that allows for flexible and efficient application deployment.

The project progress management web application is built based on MERN stack technology, bringing many benefits to schools, lecturers and students. For schools, using this application helps manage research activities of lecturers and students more effectively, thereby improving the quality of training. For lecturers, the application provides a convenient management tool, helping to track student progress and provide quick feedback. At the same time, students also benefit from the flexibility and convenience of the app, which helps them manage their work and receive feedback from instructors easily.

PHẦN GIỚI THIỆU

1. ĐẶT VẤN ĐỀ

Trong quá trình học tập và nghiên cứu tại trường đại học, sinh viên thường phải thực hiện các đề tài như niên luận, luận văn và nghiên cứu khoa học. Trong quá trình thực hiện đề tài việc quản lý tiến độ thực hiện là một công việc quan trọng, quyết định chất lượng của kết quả thực hiện đề tài.

Việc quản lý tiến độ là một yếu tố quan trọng trong việc đảm bảo sinh viên thực hiện các bước nghiên cứu theo kế hoạch và hoàn thành đề tài đúng thời hạn. Bằng cách lập kế hoạch chi tiết, theo dõi tiến độ thường xuyên và báo cáo kết quả kịp thời, sinh viên có thể duy trì sự tập trung và đảm bảo rằng mọi giai đoạn của đề tài được thực hiện theo đúng tiến trình đã đề ra. Điều này không chỉ giúp sinh viên tránh những trở ngại không mong muốn, mà còn tăng cường chất lượng của kết quả nghiên cứu và khả năng đạt được mục tiêu đề tài.

Việc quản lý thực hiện đề tài tại trường bắt đầu từ việc giảng viên đề xuất đề tài với nhà trường. Giảng viên đưa ra ý tưởng đề tài mang tính thực tiễn và đóng góp cho lĩnh vực nghiên cứu, sau đó trình bày với nhà trường hoặc khoa liên quan. Nhà trường thành lập hội đồng đánh giá để xem xét và đánh giá đề xuất từ giảng viên, tập trung vào tính khoa học, khả thi và giá trị của đề tài. Khi đề tài được phê duyệt, nhà trường xác nhận và công bố thông tin để sinh viên có thể đăng ký thực hiện.

Sinh viên quan tâm có thể đăng ký đề tài theo quy định của trường và sau khi đăng ký, giảng viên sẽ xem xét khả năng của sinh viên để phê duyệt và hướng dẫn thực hiện đề tài. Quy trình này đảm bảo rằng sinh viên có đủ điều kiện và kỹ năng cần thiết để thực hiện đề tài một cách hiệu quả. Sự phê duyệt từ giảng viên giúp xác định sự phù hợp của sinh viên với đề tài đã chọn, đồng thời tạo cơ hội cho giảng viên hướng dẫn sinh viên trong suốt quá trình thực hiện, từ việc xây dựng kế hoạch đến quá trình nghiên cứu và hoàn thành đề tài.

Để thực hiện một đề tài thông thường giảng viên cần phân công các công việc đến sinh viên cho quá trình thực hiện bao gồm các mốc thời gian quan trọng như hoàn thành đề cương, thu thập dữ liệu, phân tích dữ liệu, viết báo cáo và bảo vệ đề tài trước hội đồng. Giúp sinh viên sắp xếp công việc một cách hợp lý và đảm bảo tiến độ nghiên cứu đúng theo lịch trình.

Đối với sinh viên cần phải theo dõi tiến độ của mình thường xuyên để đảm bảo đang đi đúng hướng và thực hiện các bước theo kế hoạch. Trao đổi thông tin định kỳ với giảng viên hướng dẫn để cập nhật tiến độ, nhận phản hồi và điều chỉnh kế hoạch nếu cần. Giảng viên có thể đưa ra lời khuyên, hỗ trợ và giúp sinh viên duy trì tiến độ nghiên cứu.

Đối với giảng viên có thể đánh giá tiến độ của sinh viên định kỳ và đưa ra phản hồi kịp thời để sinh viên biết mình đang ở đâu trong quá trình thực hiện đề tài. Phản hồi này có thể giúp sinh viên cải thiện và tiến bộ trong quá trình nghiên cứu.

Tuy nhiên, thực tế cho thấy rằng việc quản lý quá trình thực hiện đề tài đang gặp nhiều hạn chế. Các vấn đề chính bao gồm thiếu hệ thống quản lý thống nhất và khó khăn trong việc theo dõi tiến độ của sinh viên.

Đầu tiên, việc thiếu hệ thống quản lý thống nhất khiến sinh viên phải tự theo dõi tiến độ, quản lý tài liệu và cập nhật thông tin với giảng viên hướng dẫn. Điều này dẫn đến thiếu thống nhất trong việc quản lý tiến độ và tài liệu, dễ gây ra tình trạng thất lạc dữ liệu và ảnh hưởng đến tiến độ nghiên cứu.

Tiếp theo, giảng viên gặp khó khăn trong việc theo dõi tiến độ của từng sinh viên. Điều này dẫn đến việc đánh giá chất lượng và hỗ trợ sinh viên không hiệu quả, ảnh hưởng đến quá trình hoàn thành đề tài. Các giải pháp hiện tại bao gồm việc lập kế hoạch chi tiết về tiến độ nghiên cứu và sử dụng các công cụ quản lý dự án như Trello hoặc Asana để theo dõi tiến độ và quản lý tài liệu.

Ngoài ra, giảng viên và sinh viên thường thiết lập các buổi gặp định kỳ để trao đổi thông tin và cập nhật tiến độ thực hiện đề tài. Những buổi gặp này tạo điều kiện để sinh viên nhận được hướng dẫn và phản hồi trực tiếp từ giảng viên, giúp giải quyết kịp thời các vấn đề phát sinh trong quá trình nghiên cứu. Đây cũng là cơ hội để sinh viên trình bày tiến trình công việc của mình và thảo luận về những điều chỉnh cần thiết để đảm bảo tiến độ theo đúng kế hoạch. Việc duy trì liên lạc thường xuyên với giảng viên thông qua các buổi gặp định kỳ là một phần quan trọng trong việc hỗ trợ sinh viên hoàn thành đề tài thành công.

Do đó, để giải quyết các vấn đề này một cách hiệu quả hơn, nên phát triển và áp dụng một hệ thống quản lý tiến độ thống nhất, cả sinh viên và giảng viên có thể cải thiện quá trình thực hiện đề tài và đạt được kết quả tốt hơn. Hệ thống sẽ hỗ trợ trong việc quản lý tiến độ thực hiện đề tài như lưu trữ thông tin về các đề tài, bao gồm mô tả đề tài, yêu cầu, hướng dẫn thực hiện cũng như quản lý tài liệu cho phép sinh viên tải lên, lưu trữ và quản lý các tài liệu liên quan, bao gồm đề cương, báo cáo tiến độ, báo cáo kết quả và theo dõi tiến độ giúp sinh viên và giảng viên theo dõi tiến độ thực hiện đề tài, bao gồm thời gian hoàn thành từng mốc quan trọng cũng như tỷ lệ hoàn thành.

Việc xây dựng ứng dụng quản lý tiến độ thực hiện đề tài mang lại nhiều lợi ích cho sinh viên, giảng viên và nhà trường. Đối với sinh viên, ứng dụng giúp sinh viên quản lý hiệu quả quá trình thực hiện đề tài, tiết kiệm thời gian và công sức, nâng cao chất lượng kết quả. Đối với giảng viên, ứng dụng giúp giảng viên theo dõi sát sao tiến độ thực hiện đề tài của sinh viên, hỗ trợ sinh viên hiệu quả hơn, nâng cao chất lượng đào tạo. Đối với nhà trường, ứng dụng giúp nhà trường quản lý thống nhất công tác quản lý đề tài, nâng cao hiệu quả hoạt động của trường, khoa và bộ môn. Ứng dụng quản lý tiến độ thực hiện đề tài là một công cụ hữu ích giúp nâng cao chất lượng công tác quản lý và thực hiện. Việc xây dựng và triển khai ứng dụng này sẽ góp phần nâng cao chất lượng đào tạo của nhà trường. Do đó việc nghiên cứu và phát triển một ứng dụng website thực hiện quản lý tiến độ thực hiện đề tài là điều khả thi.

2. LỊCH SỬ GIẢI QUYẾT VẤN ĐỀ

Để giải quyết vấn đề quản lý tiến độ thực hiện thì hiện nay cũng đã có một số giải pháp trong cả bối cảnh quốc tế và trong nước, bao gồm:

- *Nước ngoài:*

- Asana (<https://asana.com/features/project-management>): là một công cụ quản lý dự án được phát triển bởi công ty Asana. Nó cung cấp một nền tảng tập trung để quản lý công việc và nhiệm vụ, giúp các đội nhóm và tổ chức làm việc hiệu quả hơn. Asana cho phép người dùng tạo dự án và thêm các nhiệm vụ vào dự án đó, sau đó có thể thiết lập mốc thời gian và ngày hết hạn cho từng nhiệm vụ. Người dùng có thể theo dõi tiến độ dự án thông qua các báo cáo và biểu đồ trực quan, giúp quản lý hiệu suất công việc một cách chặt chẽ.

- Microsoft Project (MS Project): là một công cụ quản lý dự án nổi tiếng được phát triển và thuộc sở hữu của tập đoàn công nghệ Microsoft. Đây là một trong những công cụ quản lý dự án lâu đời nhất trên thị trường và được nhiều doanh nghiệp trên khắp thế giới sử dụng. MS Project cung cấp các chức năng mạnh mẽ để hỗ trợ lập kế hoạch, theo dõi tiến độ và quản lý tài nguyên trong dự án. Người dùng có thể lập kế hoạch chi tiết cho dự án, bao gồm các nhiệm vụ, cột mốc và thời gian hoàn thành. Công cụ cho phép phân bổ tài nguyên như nhân lực, vật liệu và thiết bị cho các nhiệm vụ cụ thể, cũng như cung cấp biểu đồ Gantt và lịch trình để theo dõi tiến độ dự án.

- *Trong nước:*

- Base Work+ (<https://base.vn/platform/work>): là một ứng dụng quản lý công việc và dự án thuộc nền tảng Base.vn, một sản phẩm do công ty Base.vn sở hữu và phát triển. Đây là một giải pháp quản lý dự án hiệu quả, giúp các doanh nghiệp và tổ chức tối ưu hóa quy trình làm việc. Ứng dụng cho phép người dùng tạo nhiệm vụ, lập kế hoạch và phân công công việc cho các thành viên trong nhóm. Bên cạnh đó, Base Work+ cung cấp các công cụ theo dõi tiến độ thực hiện dự án, bao gồm biểu đồ Gantt, lịch trình và các báo cáo trực quan. Các tính năng này giúp người quản lý dễ dàng giám sát và điều chỉnh tiến độ dự án để đạt được mục tiêu đề ra.

- FastWork (<https://fastwork.vn>): là một ứng dụng quản lý tiến độ được phát triển bởi công ty FastWork Việt Nam. Ứng dụng này cung cấp một loạt các chức năng hỗ trợ doanh nghiệp quản lý công việc và dự án một cách hiệu quả. FastWork cho phép người dùng lập kế hoạch công việc, tạo và phân công nhiệm vụ, theo dõi thời gian và tiến độ dự án. Ứng dụng cũng cung cấp các công cụ quản lý tài nguyên, giúp phân bổ công việc cho các thành viên trong nhóm một cách tối ưu. Người dùng có thể theo dõi tiến độ dự án thông qua các báo cáo và phân tích trực quan, hỗ trợ quá trình ra quyết định và quản lý thời gian hiệu quả.

3. MỤC TIÊU ĐỀ TÀI

Xây dựng ứng dụng web quản lý đề tài nhằm hỗ trợ các hoạt động quản lý và thực hiện đề tài một cách khoa học và chuyên nghiệp, góp phần nâng cao chất lượng đào tạo của nhà trường. Ứng dụng hỗ trợ cho việc quản lý đề tài, từ việc tìm kiếm, lựa chọn đến theo dõi tiến độ và đánh giá chất lượng của đề tài nghiên cứu. Nhờ đó, ứng dụng giúp nâng cao chất lượng đề tài thực hiện, cung cấp công cụ hữu ích để cán bộ quản lý, giảng viên và sinh viên thực hiện công việc hiệu quả. Thông qua tính năng theo dõi tiến độ, hỗ trợ giao tiếp và quản lý tài liệu, ứng dụng tạo điều kiện thuận lợi cho việc hướng dẫn, đánh giá và hoàn thành đề tài đúng tiến độ và đạt kết quả tốt.

Đối với cán bộ quản lý, việc tự động hóa các quy trình quản lý thực hiện đề tài giúp giảm bớt khối lượng công việc thủ công, từ đó nâng cao hiệu quả quản lý và theo dõi tiến độ thực hiện của sinh viên. Ứng dụng web sẽ cung cấp một cơ sở dữ liệu đầy đủ và chính xác về các đề tài, giảng viên hướng dẫn và sinh viên thực hiện đề tài, tạo điều kiện thuận lợi cho việc tra cứu và quản lý thông tin một cách nhanh chóng.

Đối với giảng viên hướng dẫn, một ứng dụng quản lý đề tài hiệu quả giúp dễ dàng theo dõi tiến độ thực hiện của sinh viên, cho phép họ giám sát và hỗ trợ kịp thời trong quá trình nghiên cứu. Ứng dụng tạo điều kiện thuận lợi cho giao tiếp và trao đổi thông tin giữa giảng viên và sinh viên một cách hiệu quả, nhờ tính năng nhắn tin trực tuyến. Điều này giúp giảng viên có thể cung cấp phản hồi, hướng dẫn và giải đáp thắc mắc cho sinh viên một cách nhanh chóng và chính xác.

Đối với sinh viên, cho phép sinh viên tìm kiếm đề tài luận văn một cách dễ dàng nhờ cơ sở dữ liệu phong phú và tính năng tìm kiếm thông minh. Sau khi chọn được đề tài, sinh viên có thể đăng ký với giảng viên. Ứng dụng giúp sinh viên theo dõi tiến độ thực hiện thông qua các mốc thời gian và nhiệm vụ được phân công, đồng thời cung cấp các công cụ để nhận phản hồi từ giảng viên hướng dẫn một cách kịp thời. Nhờ vào khả năng cập nhật tiến độ và giao tiếp với giảng viên, sinh viên có thể điều chỉnh kế hoạch và nội dung thực hiện một cách linh hoạt, đảm bảo tiến độ nghiên cứu được thực hiện đúng thời hạn và đạt chất lượng cao.

4. ĐỐI TƯỢNG VÀ PHẠM VI NGHIÊN CỨU

4.1. Đối tượng

Đối tượng nghiên cứu chính của đề tài này bao gồm cán bộ quản lý, giảng viên hướng dẫn, sinh viên và các công tác liên quan đến việc thực hiện quản lý tiến độ thực hiện. Đề tài tập trung vào việc hỗ trợ sinh viên theo dõi tiến độ trong quá trình thực hiện luận văn cũng như cung cấp công cụ để giảng viên giám sát tiến độ và hỗ trợ sinh viên hiệu quả.

4.2. Phạm vi nghiên cứu

Phạm vi nghiên cứu của ứng dụng bao gồm cả các khía cạnh lý thuyết và thực hành liên quan đến việc phát triển một ứng dụng quản lý luận văn hiệu quả.

- *Về lý thuyết:*
 - Nghiên cứu công tác liên quan đến việc quản lý thực hiện đề tài.
 - Nghiên cứu về công nghệ MERN Stack.
 - Tìm hiểu về chuẩn thiết kế Restful API.
 - Kiến trúc Client-Server.
 - Thiết kế giao diện của một website quản lý.
- *Về lập trình:*
 - Ngôn ngữ lập trình: JavaScript.
 - Cơ sở dữ liệu: MongoDB.
 - Các thư viện hỗ trợ xây dựng một website quản lý.
 - Kiến trúc và tính năng cho một ứng dụng quản lý tiến độ thực hiện đề tài.

5. BỐ CỤC LUẬN VĂN

Bố cục luận văn này được chia thành ba phần chính: giới thiệu, nội dung và kết luận.

- *Phần giới thiệu:* trình bày tổng quan về vấn đề cần giải quyết, bao gồm lịch sử giải quyết vấn đề, mục tiêu của đề tài, đối tượng và phạm vi nghiên cứu. Phần này đặt nền móng cho việc hiểu rõ bối cảnh và mục đích của nghiên cứu.
- *Phần nội dung:* tập trung vào việc mô tả chi tiết bài toán cần giải quyết và các yêu cầu của ứng dụng. Nó bao gồm đặc tả các yêu cầu chức năng và phi chức năng, thiết kế và cài đặt ứng dụng, thiết kế dữ liệu và giao diện, cũng như các bước xử lý từng chức năng của ứng dụng.
- *Phần kết luận:* tổng kết những kết quả đạt được trong quá trình nghiên cứu và phát triển ứng dụng, đồng thời đưa ra phương hướng phát triển trong tương lai của đề tài. Phần này nêu rõ những thành tựu đã đạt được và các hướng cải thiện tiềm năng.

Ngoài ra, luận văn còn bao gồm phần tài liệu tham khảo, cung cấp danh mục và các bảng, hình ảnh minh họa được trình bày trong luận văn. Trước phần giới thiệu là mục lục, giúp người đọc dễ dàng theo dõi nội dung và cấu trúc của luận văn.

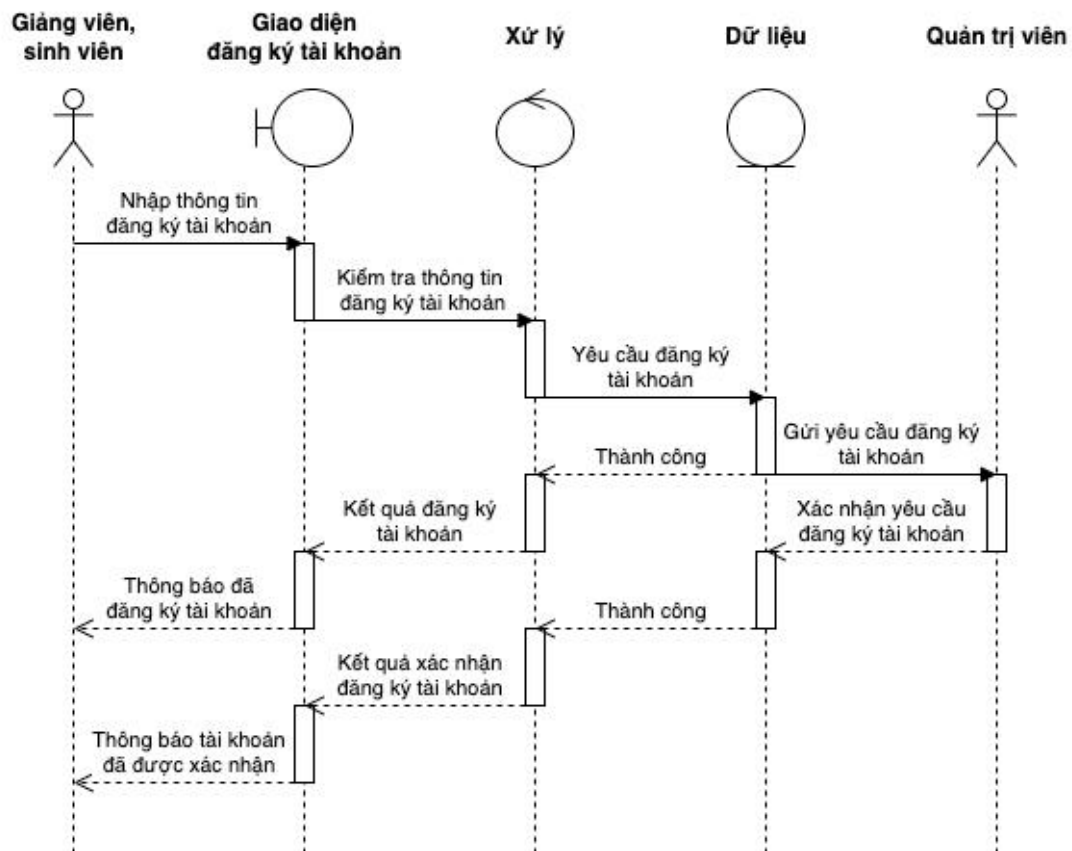
PHẦN NỘI DUNG

CHƯƠNG 1: CƠ SỞ LÝ THUYẾT

1. QUY TRÌNH THỰC HIỆN ĐỀ TÀI

1.1. Đăng ký tài khoản

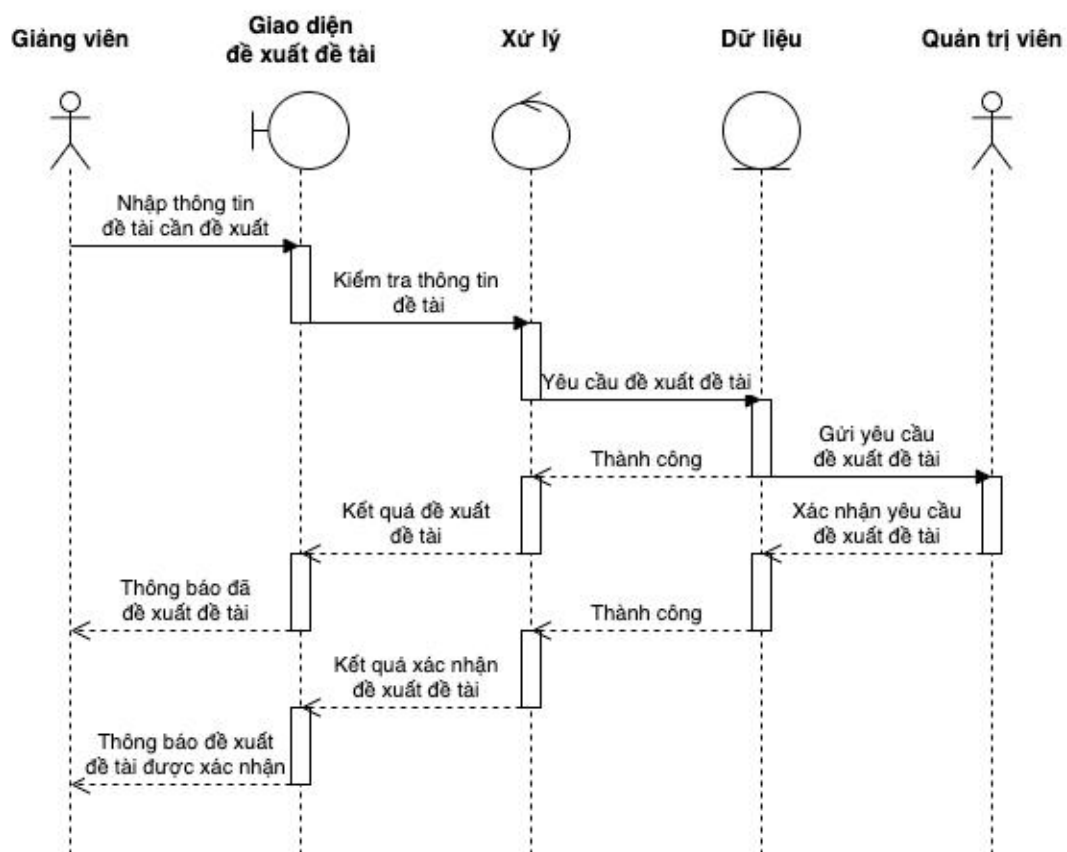
Giảng viên và sinh viên truy cập vào ứng dụng web, sau đó thực hiện đăng ký tài khoản và cập nhật các thông tin cần thiết. Sau khi hoàn thành, yêu cầu đăng ký được gửi đến quản trị viên để xem xét. Quản trị viên sẽ kiểm tra thông tin và quyết định duyệt hoặc từ chối yêu cầu. Nếu được duyệt, họ có thể truy cập vào hệ thống và sử dụng các chức năng trong phạm vi cho phép. Quy trình này đảm bảo rằng chỉ những người được ủy quyền mới có thể truy cập vào hệ thống, đồng thời tạo ra một cơ chế kiểm soát và quản lý cho việc đăng ký tài khoản.



Hình 1-1: Sơ đồ đăng ký tài khoản

1.2. Đề xuất đề tài

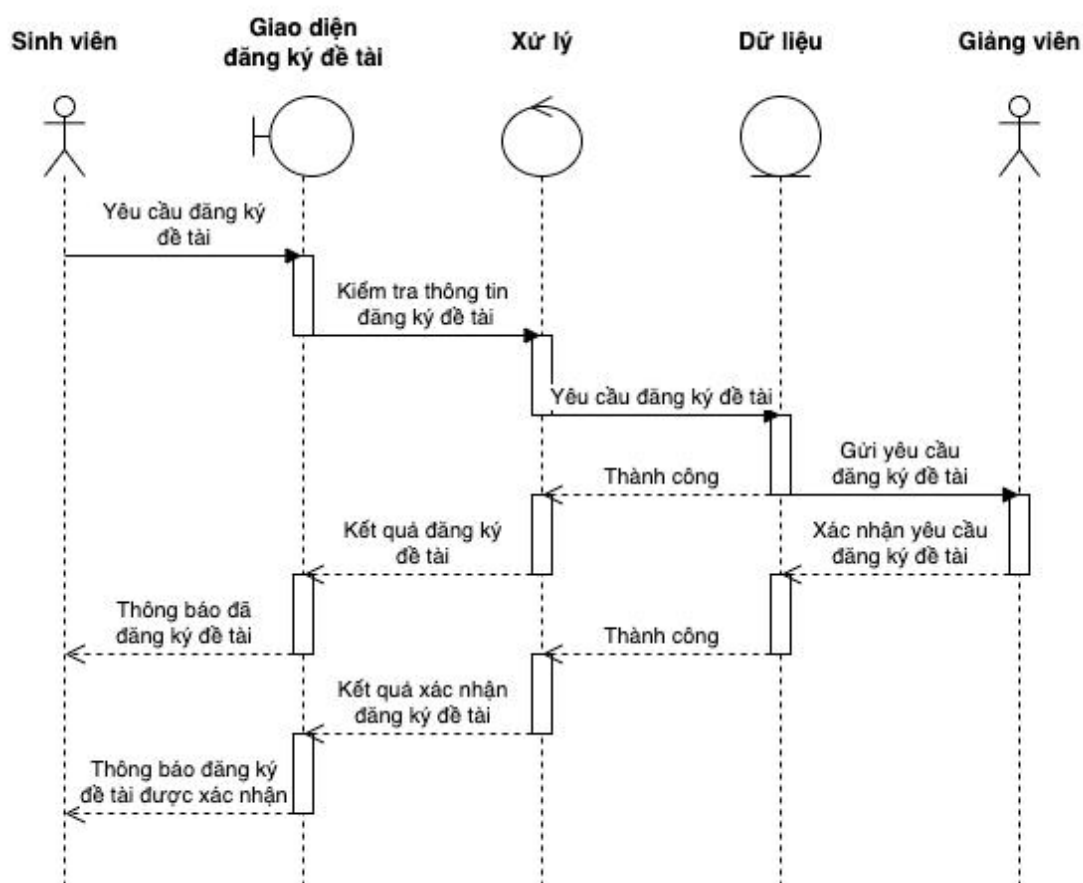
Giảng viên đăng nhập vào hệ thống và thêm thông tin về đề tài mà họ muốn đề xuất. Sau khi hoàn thành, họ gửi yêu cầu đề xuất đến quản trị viên. Quản trị viên sau đó sẽ xem xét đề xuất, kiểm tra tính phù hợp và khả thi của đề tài và quyết định duyệt hoặc từ chối yêu cầu. Kết quả của quyết định được thông báo cho giảng viên và nếu đề tài được duyệt, đề tài sẽ được thêm vào hệ thống. Quy trình này đảm bảo rằng mọi đề xuất đều được xem xét cẩn thận trước khi được thêm vào hệ thống, đồng thời giúp đảm bảo chất lượng và tính đáng tin cậy của dữ liệu trong hệ thống.



Hình 1-2: Sơ đồ đề xuất đề tài

1.3. Đăng ký thực hiện đề tài

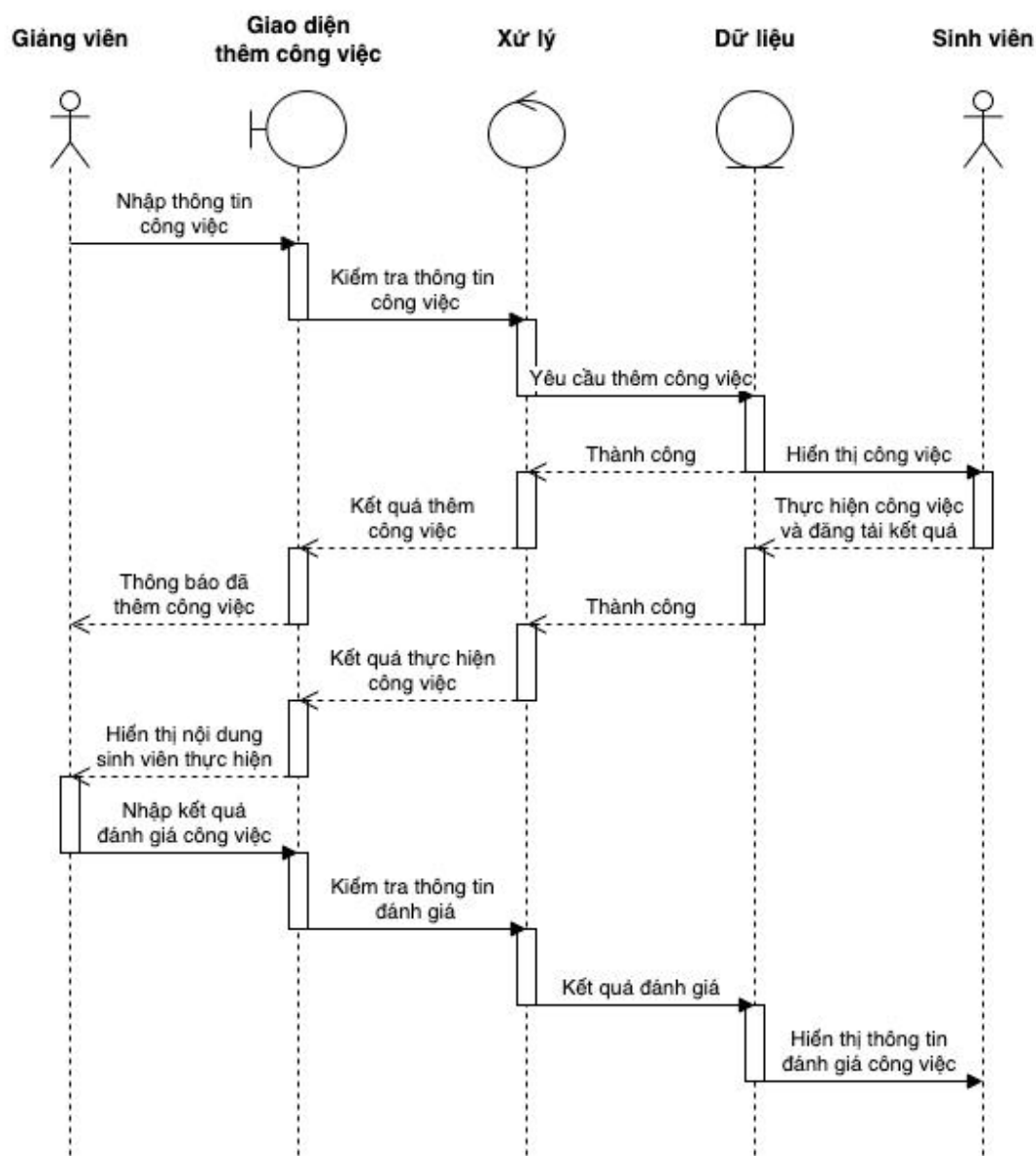
Sinh viên đăng nhập vào hệ thống và đăng ký đề tài. Yêu cầu đăng ký đề tài sau đó được gửi đến giảng viên phụ trách đề tài, lúc này sinh viên và giảng viên có thể trao đổi với nhau trực tiếp trên hệ thống. Giảng viên xem xét yêu cầu và quyết định xác nhận hoặc từ chối dựa trên năng lực và khả năng của sinh viên. Kết quả của quyết định được thông báo cho sinh viên. Nếu được xác nhận, sinh viên có thể tiến hành thực hiện đề tài. Quy trình này giúp đảm bảo rằng mỗi đề tài được xem xét kỹ lưỡng trước khi được chấp nhận và tạo điều kiện cho sự tương tác hợp tác giữa giảng viên và sinh viên.



Hình 1-3: Sơ đồ đăng ký thực hiện đề tài

1.4. Quản lý thực hiện đề tài

Trong quy trình quản lý thực hiện đề tài, giảng viên phân công nhiệm vụ cụ thể cho sinh viên thực hiện đề tài. Sinh viên nhận nhiệm vụ và tiến hành thực hiện, sau đó nộp kết quả cho giảng viên. Dựa trên kết quả này, giảng viên đánh giá và quyết định tiếp tục hoặc điều chỉnh công việc tiếp theo. Nếu sinh viên không hoàn thành hoặc không đạt yêu cầu, giảng viên có thể điều chỉnh kế hoạch hoặc ngừng công việc của sinh viên đó. Quy trình này đảm bảo sự linh hoạt và hiệu quả trong quản lý đề tài giữa giảng viên và sinh viên.



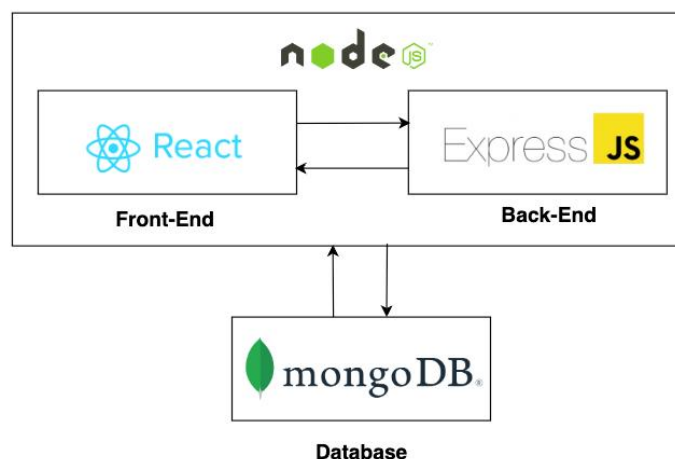
Hình 1-4: Sơ đồ quản lý thực hiện đề tài

2. CÔNG NGHỆ MERN STACK

2.1. Khái quát về MERN Stack

MERN stack là một bộ công nghệ phổ biến trong phát triển web. MERN là viết tắt của MongoDB, Express.js, React.js, Node.js, theo tên bốn công nghệ chính tạo nên ngăn xếp.

- *Node.js*: Là một môi trường thực thi JavaScript, cho phép chạy mã JavaScript trên máy chủ.
- *Express.js*: Là một framework web cho Node.js, được sử dụng để xây dựng ứng dụng web và API.
- *React.js*: Là một thư viện JavaScript phía máy khách, được sử dụng để xây dựng giao diện người dùng tương tác.
- *MongoDB*: Là một hệ quản trị cơ sở dữ liệu NoSQL, được thiết kế để lưu trữ dữ liệu dưới dạng tài liệu JSON.



Hình 1-5: Mô hình khái quát về MERN Stack

2.2. Node.js

Node.js là một môi trường thực thi mã JavaScript dựa trên Chrome's V8 JavaScript engine. Nó cho phép việc thực thi mã JavaScript ở phía máy chủ, mở ra một cách tiếp cận mới trong việc phát triển các ứng dụng web và dịch vụ mạng.

Một trong những điểm nổi bật của Node.js là mô hình bất đồng bộ (asynchronous), non-blocking I/O, cho phép xử lý đa luồng mà không cần tạo ra các luồng mới. Điều này có nghĩa là Node.js có thể xử lý hàng ngàn kết nối cùng một lúc mà không gặp phải sự chậm trễ lớn, làm cho ứng dụng trở nên linh hoạt và hiệu quả hơn.

Node.js cũng có một cộng đồng lớn và đầy năng động của các nhà phát triển, cung cấp hàng ngàn thư viện và framework để giúp xây dựng các ứng dụng một cách nhanh chóng và dễ dàng. Điều này giúp cho việc phát triển các ứng dụng web trở nên nhanh chóng và linh hoạt hơn bao giờ hết.

Một trong những ưu điểm lớn của Node.js là khả năng sử dụng JavaScript cả ở phía máy chủ và phía máy khách, tạo ra sự nhất quán trong toàn bộ ứng dụng. Điều này giúp giảm độ phức tạp và thời gian học tập khi phát triển ứng dụng web và cũng làm cho việc chia sẻ mã giữa phía máy chủ và phía máy khách trở nên dễ dàng hơn.

Tóm lại, Node.js không chỉ là một môi trường thực thi mã JavaScript, mà còn là một cách tiếp cận mới trong việc phát triển ứng dụng web, với tính linh hoạt, hiệu suất và sự nhất quán giữa phía máy chủ và phía máy khách.

2.3. Express.js

Express.js là một trong những framework web phổ biến nhất cho Node.js, cung cấp một cách tiếp cận linh hoạt và đơn giản để xây dựng các ứng dụng web và API. Với cú pháp đơn giản và các tính năng mạnh mẽ, Express.js giúp cho việc phát triển các ứng dụng web trở nên dễ dàng và hiệu quả.

Một trong những điểm nổi bật của Express.js là cách nó quản lý các route. Express.js cho phép định nghĩa các endpoint của ứng dụng và xử lý các yêu cầu HTTP tương ứng với mỗi endpoint. Điều này giúp việc xây dựng API trở nên đơn giản và dễ hiểu.

Express.js cũng có một hệ thống middleware mạnh mẽ, cho phép các tác vụ như xác thực, xử lý lỗi và ghi log được thực hiện một cách dễ dàng và linh hoạt. Middleware cho phép chúng ta tùy chỉnh cách xử lý các yêu cầu HTTP một cách linh hoạt, từ việc kiểm tra quyền truy cập cho đến việc chuyển tiếp yêu cầu đến các endpoint khác.

Ngoài ra, Express.js cũng hỗ trợ một loạt các tính năng như session management, routing và template engines, giúp việc phát triển các ứng dụng web trở nên dễ dàng hơn. Điều này làm cho Express.js trở thành một lựa chọn lý tưởng cho việc xây dựng cả các ứng dụng web đơn giản và phức tạp.

Tóm lại, Express.js là một framework web mạnh mẽ và linh hoạt cho Node.js, giúp việc xây dựng các ứng dụng web và API trở nên dễ dàng và hiệu quả hơn bao giờ hết. Với cú pháp đơn giản và các tính năng mạnh mẽ, Express.js là một công cụ không thể thiếu trong bộ công cụ phát triển của mọi nhà phát triển web Node.js.

2.4. React.js

React.js là một thư viện JavaScript phía máy khách được phát triển bởi Facebook, được sử dụng rộng rãi trong việc xây dựng giao diện người dùng tương tác trong các ứng dụng web hiện đại. Với triết lý "Learn Once, Write Anywhere", React.js tập trung vào việc xây dựng các thành phần giao diện tái sử dụng và quản lý trạng thái của ứng dụng một cách hiệu quả.

Một trong những điểm mạnh của React.js là Virtual DOM (DOM ảo). React.js sử dụng Virtual DOM để cập nhật giao diện người dùng một cách hiệu quả hơn. Thay vì cập nhật trực tiếp DOM mỗi khi có sự thay đổi, React.js tạo ra một bản sao ảo của DOM và so sánh nó với DOM thực tế, chỉ cập nhật các phần tử có sự thay đổi. Điều này giúp tăng hiệu suất của ứng dụng và giảm thời gian phản hồi cho người dùng.

Ngoài ra, React.js cũng giúp tạo ra các thành phần giao diện tái sử dụng. Thành phần trong React.js là các khối xây dựng giao diện độc lập, có thể tái sử dụng trong nhiều phần của ứng dụng mà không cần viết lại mã. Điều này giúp tăng sự linh hoạt và tiết kiệm thời gian cho việc phát triển ứng dụng.

Một trong những tính năng quan trọng khác của React.js là JSX (JavaScript XML), một phần mở rộng của JavaScript cho phép viết HTML trong mã JavaScript. JSX giúp tạo ra mã nguồn của ứng dụng trở nên dễ đọc và dễ hiểu hơn, đồng thời cung cấp một cách tiếp cận tự nhiên hơn cho việc xây dựng giao diện người dùng.

Tóm lại, React.js là một thư viện JavaScript phía máy khách mạnh mẽ và linh hoạt, được sử dụng rộng rãi trong cộng đồng phát triển web để xây dựng các ứng dụng web tương tác và dễ bảo trì. Với Virtual DOM, thành phần tái sử dụng và JSX, React.js là một công cụ không thể thiếu trong bộ công cụ phát triển của mọi nhà phát triển web.

2.5. Axios

Axios là một thư viện HTTP client phổ biến trong cộng đồng phát triển web, được sử dụng để tương tác với các API từ các ứng dụng web hoặc dịch vụ mạng khác. Với cú pháp đơn giản và dễ hiểu, Axios giúp việc gửi các yêu cầu HTTP và xử lý các phản hồi trở nên dễ dàng hơn bao giờ hết.

Một trong những điểm mạnh của Axios là khả năng hỗ trợ các yêu cầu HTTP không đồng bộ, non-blocking I/O, phù hợp với mô hình lập trình bất đồng bộ của JavaScript và Node.js. Điều này giúp việc gửi và nhận các yêu cầu

HTTP từ các API trở nên nhanh chóng và hiệu quả, mà không làm chậm lại quá trình thực thi của ứng dụng.

Axios cung cấp một API linh hoạt để tạo và tùy chỉnh các yêu cầu HTTP, bao gồm các phương thức như GET, POST, PUT, DELETE và nhiều phương thức khác. Nó cũng hỗ trợ các tính năng như interceptors (chặn) và cancellation (hủy bỏ), giúp kiểm soát quá trình gửi và nhận các yêu cầu HTTP một cách linh hoạt và dễ dàng.

Một trong những ưu điểm khác của Axios là khả năng xử lý các phản hồi HTTP tự động chuyển đổi sang dạng JSON. Điều này giúp cho việc xử lý dữ liệu từ các API trở nên thuận tiện hơn, và giảm thiểu thời gian và công sức cần thiết cho việc phân tích và chuyển đổi dữ liệu.

Tóm lại, Axios là một thư viện HTTP client mạnh mẽ và linh hoạt, là công cụ không thể thiếu trong bộ công cụ phát triển của mọi nhà phát triển web. Với khả năng hỗ trợ các yêu cầu HTTP không đồng bộ, API linh hoạt và tính năng mạnh mẽ, Axios giúp tạo ra các ứng dụng web hiệu quả và dễ bảo trì.

2.6. MongoDB

MongoDB là một hệ quản trị cơ sở dữ liệu NoSQL, được phát triển và duy trì bởi MongoDB. Với kiến trúc tài liệu linh hoạt, MongoDB cho phép lưu trữ dữ liệu dưới dạng tài liệu JSON-like gọi là BSON (Binary JSON), không yêu cầu một schema cố định như các hệ quản trị cơ sở dữ liệu truyền thống. Điều này mang lại tính linh hoạt cao cho việc lưu trữ dữ liệu, cho phép các ứng dụng thay đổi cấu trúc dữ liệu một cách dễ dàng mà không gặp phải các rắc rối về schema.

Một trong những điểm nổi bật của MongoDB là khả năng mở rộng tốt. MongoDB được thiết kế để có thể mở rộng từ một máy chủ đơn lẻ đến một cụm máy chủ lớn hơn mà không gặp vấn đề về hiệu suất. Điều này làm cho MongoDB trở thành lựa chọn phổ biến cho các ứng dụng web có yêu cầu về mở rộng và xử lý lượng dữ liệu lớn.

Ngoài ra, MongoDB cũng cung cấp một loạt các tính năng như Replication, Sharding và Indexing để tối ưu hóa hiệu suất và đảm bảo độ tin cậy của hệ thống. Điều này giúp cho MongoDB có thể xử lý các tải đồng thời lớn mà vẫn duy trì tính khả dụng và độ tin cậy cao.

Với việc tích hợp dễ dàng với các công nghệ phổ biến như Node.js trong MERN stack, MongoDB trở thành một lựa chọn lý tưởng cho việc xây dựng các ứng dụng web hiệu quả, linh hoạt và mở rộng được.

2.7. Mongoose

Mongoose là một thư viện mô hình hóa đối tượng cho MongoDB và Node.js, giúp quản lý mô hình dữ liệu trong ứng dụng Node.js một cách thuận tiện và linh hoạt. Được phát triển dựa trên driver MongoDB chính thức, Mongoose cung cấp một giao diện đơn giản và trực quan để tương tác với cơ sở dữ liệu MongoDB, đồng thời giúp đơn giản hóa việc xử lý dữ liệu và thực hiện các hoạt động như tìm kiếm, thêm mới, cập nhật và xóa dữ liệu.

Một trong những điểm mạnh của Mongoose là nó cho phép định nghĩa cấu trúc dữ liệu và quan hệ giữa các tài liệu trong cơ sở dữ liệu thông qua các Schema. Các Schema trong Mongoose giúp định nghĩa các trường dữ liệu, kiểu dữ liệu, ràng buộc và các tùy chọn khác liên quan đến dữ liệu. Điều này giúp đảm bảo tính nhất quán và đồng nhất trong dữ liệu được lưu trữ trong cơ sở dữ liệu MongoDB.

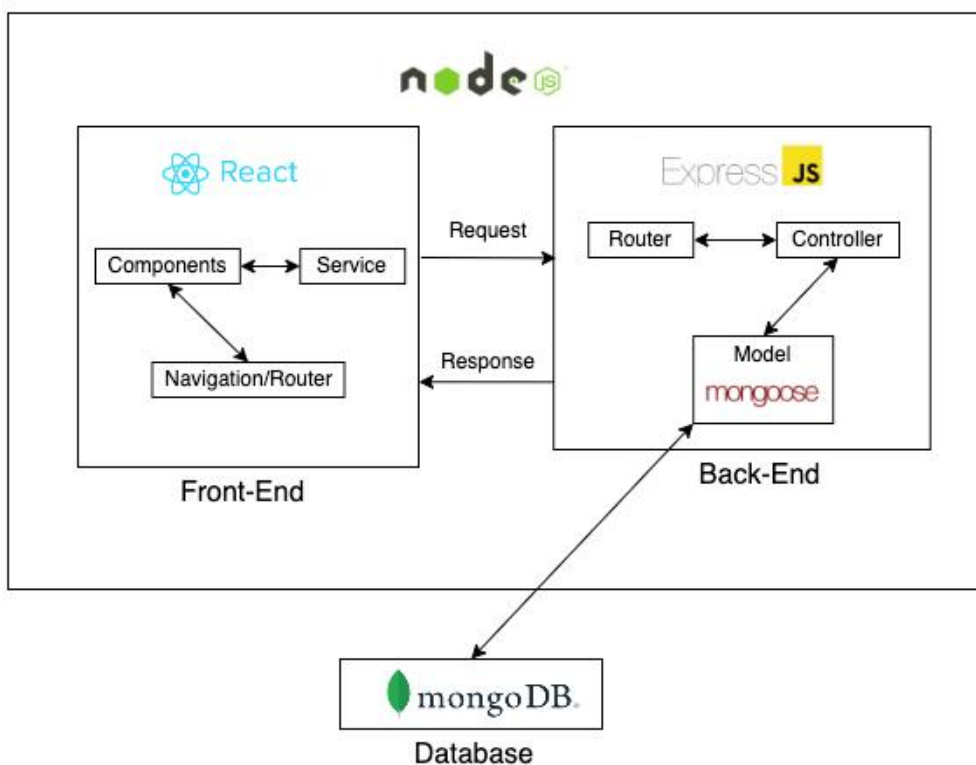
Mongoose cũng cung cấp một bộ các phương thức và phương tiện để thực hiện các hoạt động với cơ sở dữ liệu, như tìm kiếm, sắp xếp, lọc và phân trang dữ liệu. Điều này giúp việc xây dựng các ứng dụng Node.js và MongoDB trở nên dễ dàng hơn, đồng thời giảm thiểu thời gian và công sức cần thiết cho việc viết mã.

Ngoài ra, Mongoose còn hỗ trợ các tính năng như validation (kiểm tra hợp lệ), middleware (trung gian) và populate (làm phong phú dữ liệu), giúp tối ưu hóa quá trình phát triển và quản lý ứng dụng. Điều này làm cho Mongoose trở thành một công cụ mạnh mẽ và linh hoạt cho việc phát triển ứng dụng Node.js và MongoDB.

Tóm lại, Mongoose là một thư viện quan trọng và không thể thiếu cho việc phát triển ứng dụng Node.js kết hợp với cơ sở dữ liệu MongoDB. Với khả năng định nghĩa Schema, cung cấp các phương thức tiện ích và hỗ trợ các tính năng quan trọng, Mongoose giúp tạo ra các ứng dụng Node.js mạnh mẽ, linh hoạt và dễ bảo trì.

2.8. Cách MERN Stack hoạt động

Các công nghệ trong MERN Stack hoạt động cùng nhau bằng cách tạo ra một quy trình phát triển toàn diện:



Hình 1-6: Mô hình hoạt động của MERN Stack

- Front-End sử dụng React dùng để xây dựng giao diện người dùng. React Components được sử dụng để hiển thị dữ liệu và tương tác với người dùng, trong khi React Router quản lý routing trong ứng dụng web. Sử dụng thư viện Axios trong Node.js để gửi và nhận các yêu cầu HTTP từ phía máy khách của ứng dụng web.
- Back-End sử dụng Node.js và Express.js để tạo và quản lý API. Express.js xử lý các yêu cầu từ Front-End và tương tác với cơ sở dữ liệu MongoDB. Middleware của Express được sử dụng để xử lý các yêu cầu HTTP như đăng nhập, xác thực và xử lý lỗi.
- Cơ sở dữ liệu MongoDB được sử dụng để lưu trữ và truy xuất dữ liệu. Sử dụng thư viện Mongoose trong Node.js để tạo và quản lý mô hình dữ liệu MongoDB.

Tất cả các thành phần này kết hợp lại với nhau để tạo ra một hệ thống ứng dụng web đầy đủ chức năng, từ việc xây dựng giao diện người dùng đến xử lý yêu cầu máy chủ và truy xuất dữ liệu từ cơ sở dữ liệu.

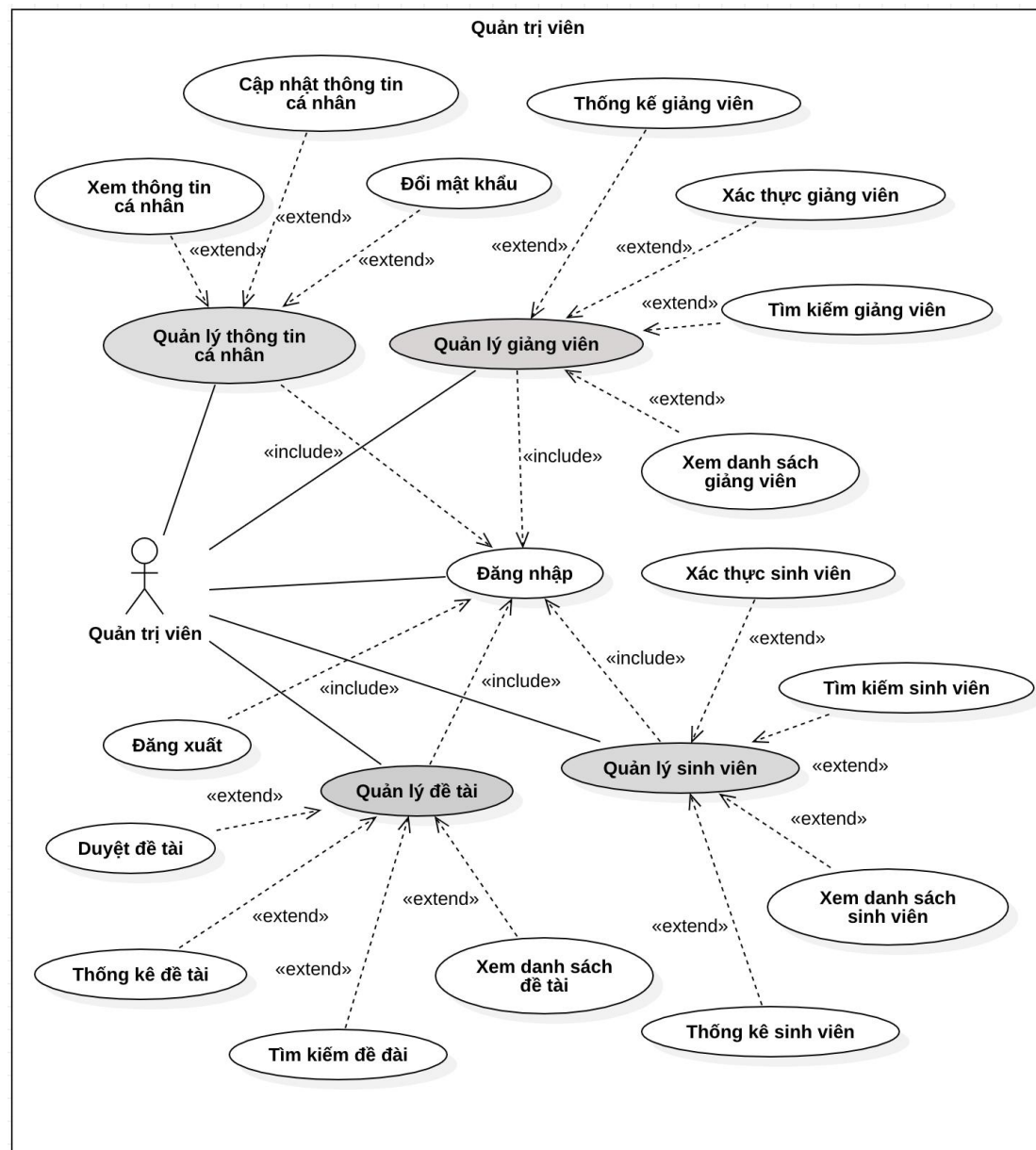
CHƯƠNG 2: MÔ TẢ BÀI TOÁN

1. MÔ TẢ CHI TIẾT BÀI TOÁN

Đối với ứng dụng web này thì sẽ bao gồm 3 nhóm người dùng chính.

1.1. Quản trị viên

Quản trị viên là người dùng có quyền cao nhất trong hệ thống, chịu trách nhiệm quản lý các hoạt động liên quan đến giảng viên, sinh viên, đề tài và phải được sự xác nhận của quản trị viên trước khi đưa vào hệ thống và được mô tả bằng sơ đồ sau đây:

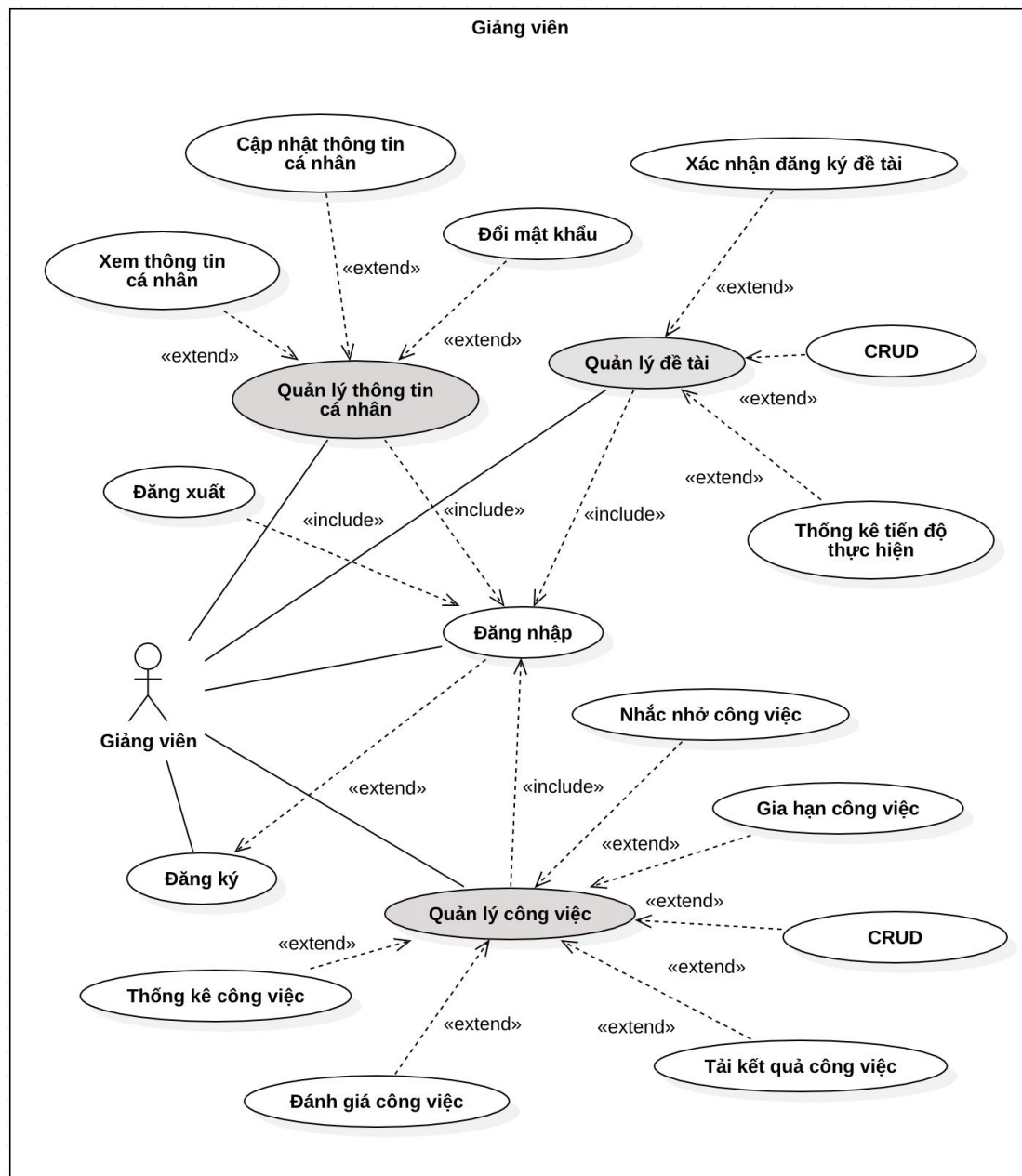


Hình 2-1: Sơ đồ chức năng quản trị viên

1.2. Giảng viên

Người dùng giảng viên có thể thực hiện các chức năng:

- Quản lý thông tin cá nhân: cho phép người dùng có thể dễ dàng xem, cập nhật các thông tin cá nhân của mình và thực hiện thay đổi mật khẩu.
- Quản lý đề tài: cho phép giảng viên đề xuất thực hiện các đề tài, xác nhận đăng ký đề tài từ sinh viên và quản lý thay đổi thông tin của đề tài.
- Quản lý công việc: cho phép phân công, đánh giá chất lượng công việc đã phân công đến sinh viên và quản lý thay đổi thông tin của công việc.

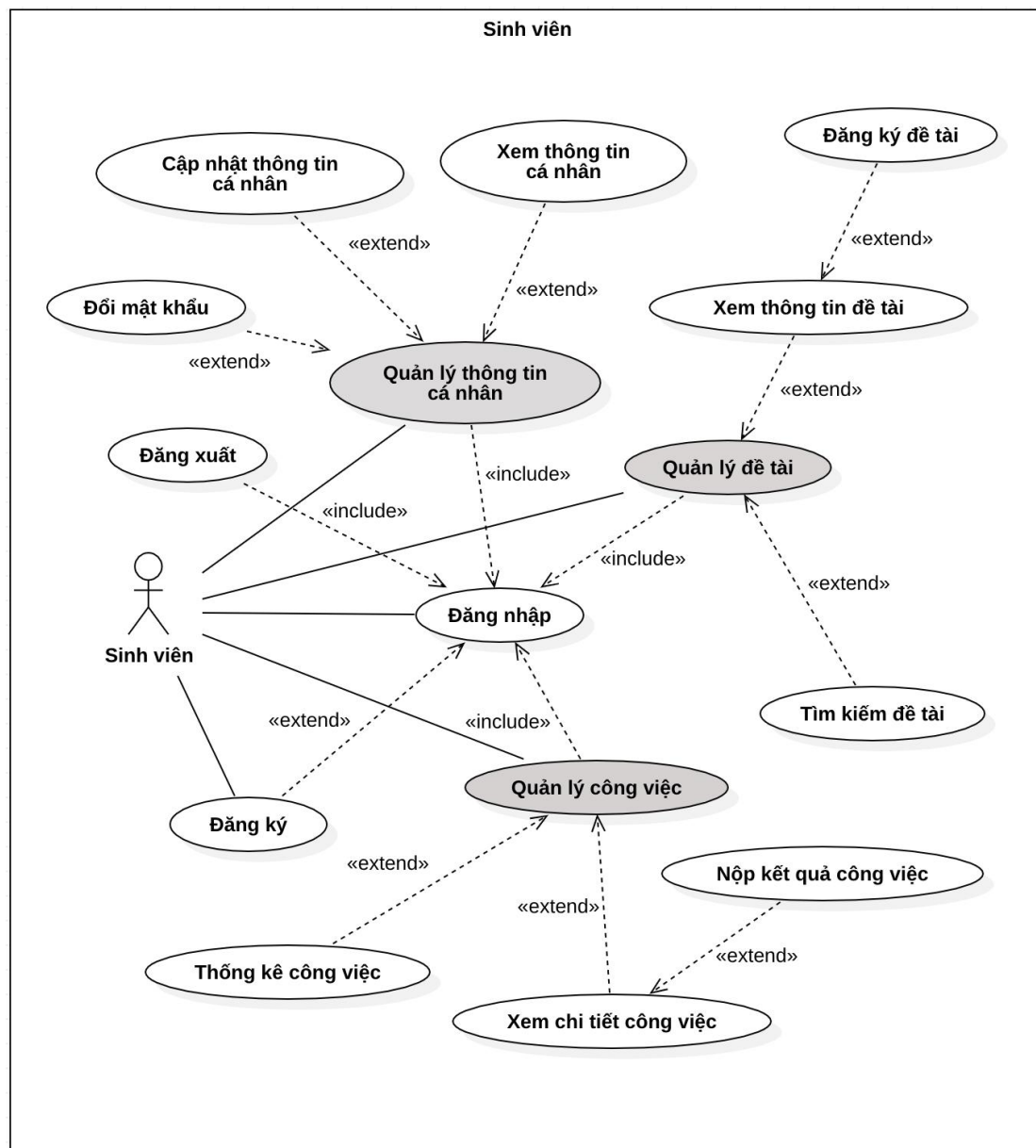


Hình 2-2: Sơ đồ chức năng giảng viên

1.3. Sinh viên

Người dùng giảng viên có thể thực hiện các chức năng:

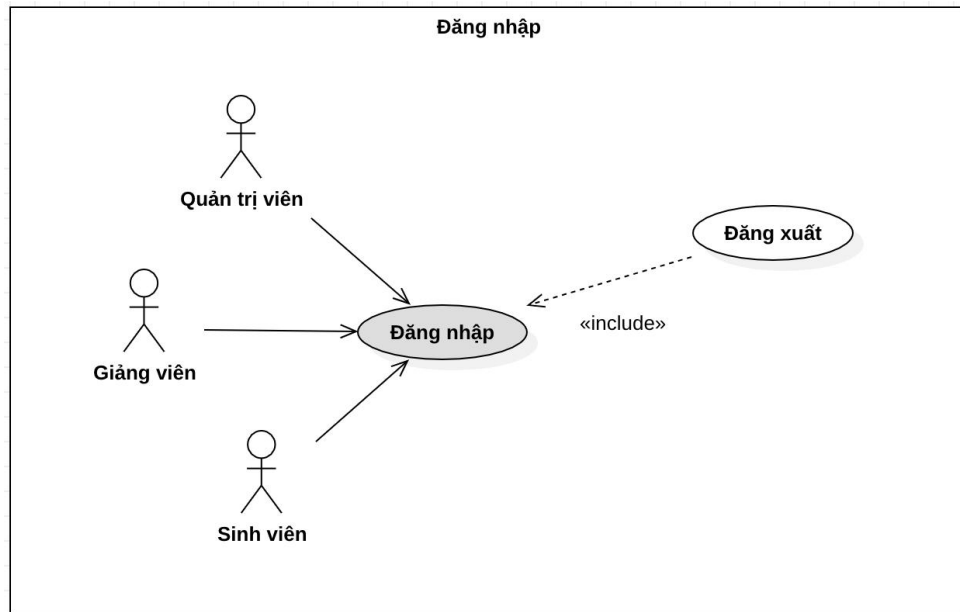
- Quản lý thông tin cá nhân: cho phép người dùng có thể dễ dàng xem, cập nhật các thông tin cá nhân của mình và thực hiện thay đổi mật khẩu.
- Quản lý đề tài: cho phép sinh viên tìm kiếm, xem thông tin và đăng ký thực hiện đề tài.
- Quản lý công việc: cho phép sinh viên thực hiện đăng tải các nội dung và thống kê danh sách các công việc được phân công.



Hình 2-3: Sơ đồ chức năng sinh viên

2. CÁC CHỨC NĂNG CỦA HỆ THỐNG

2.1. Chức năng đăng nhập

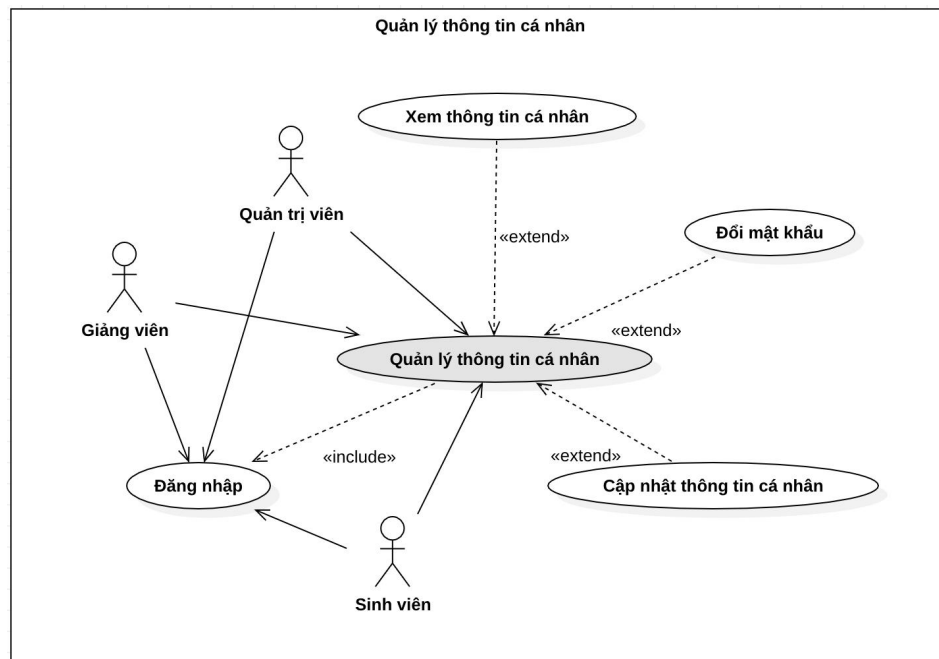


Hình 2-4: Chức năng đăng nhập

Tên chức năng	Đăng nhập
Đối tượng sử dụng	Quản trị viên, Giảng viên, Sinh viên.
Mô tả tóm tắt	Hệ thống yêu cầu phải đăng nhập trước khi thực hiện các thao tác và chức năng.
Tiền điều kiện	Phải có tài khoản trong hệ thống.
Cách xử lý	Bước 1: Nhập các thông tin yêu cầu. Bước 2: Nhấn vào nút đăng nhập. Hệ thống sẽ kiểm tra thông tin.
Kết quả	+ Nhập sai thì sẽ thông báo lỗi và nhập lại. + Nhập đúng sẽ chuyển về trang chỉ định.
Ghi chú	+ Tên tài khoản là mã số sinh viên hoặc cán bộ. + Quản trị viên mặc định là Admin.

Bảng 2-1: Mô tả chức năng đăng nhập

2.2. Chức năng quản lý thông tin cá nhân

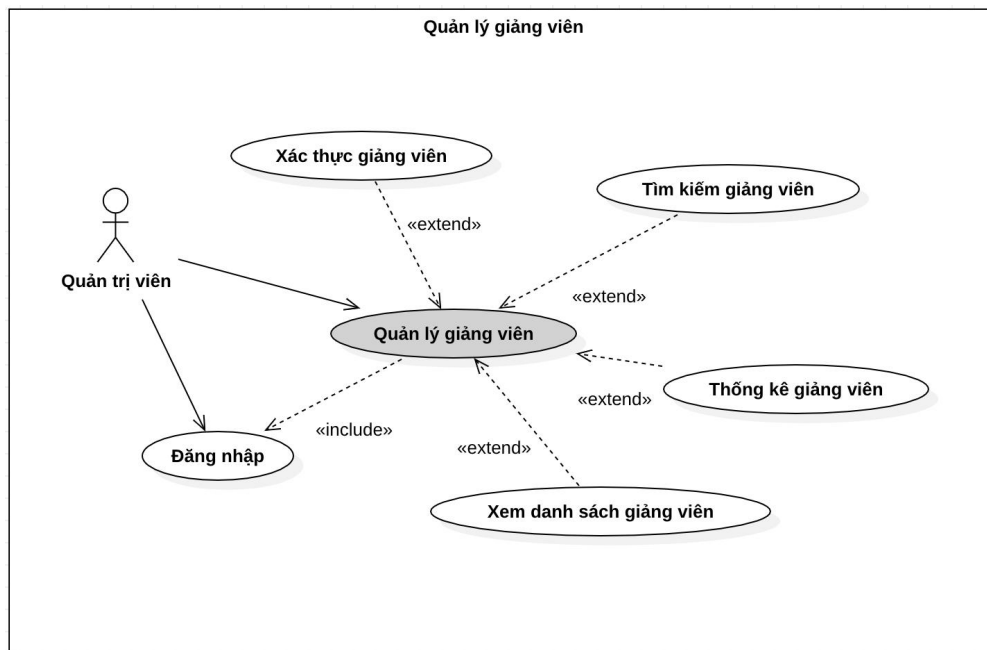


Hình 2-5: Chức năng quản lý thông tin cá nhân

Tên chức năng	Quản lý thông tin cá nhân
Đối tượng sử dụng	Quản trị viên, Giảng viên, Sinh viên.
Mô tả tóm tắt	Cho phép quản lý các thông tin cá nhân của người dùng.
Tiền điều kiện	Người dùng đã đăng nhập vào hệ thống.
Cách xử lý	<p>Thực hiện các chức năng quản lý cho phép cập nhật thông tin cá nhân và đổi mật khẩu.</p> <ol style="list-style-type: none"> Khi thực hiện chức năng “Cập nhật thông tin cá nhân” sẽ hiển thị form cập nhật thông tin cá nhân với các thông tin hiện tại và có thể tùy chỉnh được các thông tin cho phép, sau đó nhấn chọn để xác nhận. Khi thực hiện chức năng “Đổi mật khẩu” sẽ hiển thị form đổi mật khẩu và nhập mật khẩu cũ và mật khẩu mới, sau đó nhấn chọn để xác nhận.
Kết quả	Lưu thông tin vào hệ thống khi thực hiện thành công.
Ghi chú	Không có.

Bảng 2-2: Mô tả chức năng quản lý thông tin cá nhân

2.3. Chức năng quản lý giảng viên

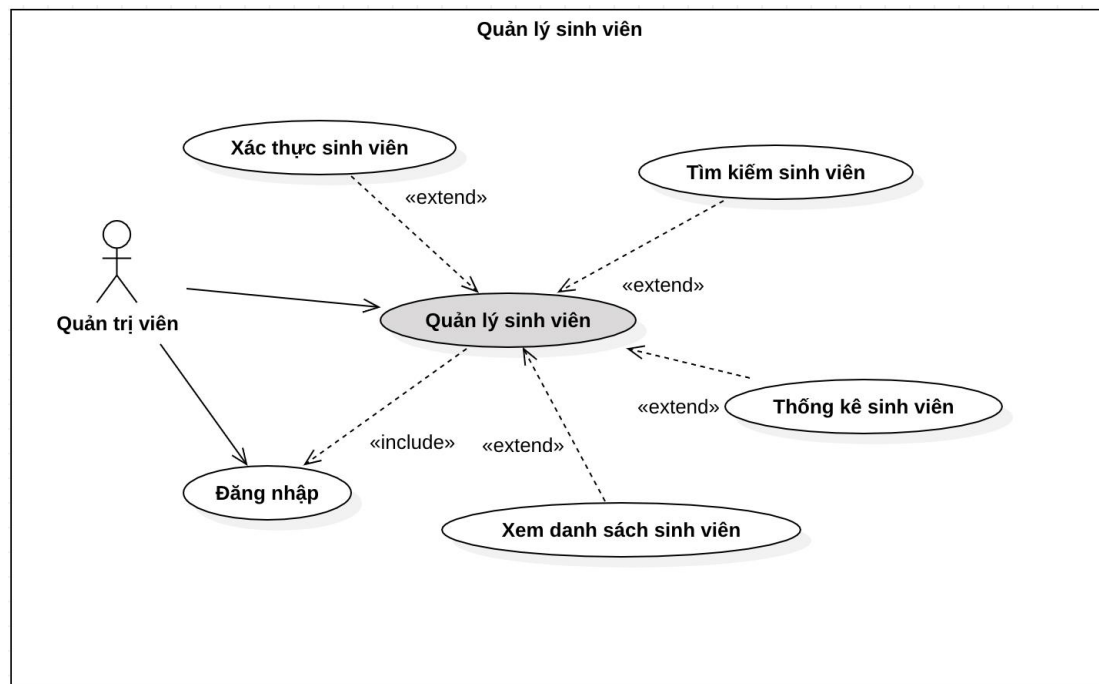


Hình 2-6: Chức năng quản lý giảng viên

Tên chức năng	Quản lý giảng viên
Đối tượng sử dụng	Quản trị viên.
Mô tả tóm tắt	Cho phép quản lý xác thực và thống kê danh sách giảng viên trong hệ thống.
Tiền điều kiện	Đã đăng nhập vào hệ thống với tài khoản Quản trị viên.
Cách xử lý	Thực hiện các chức năng quản lý cho phép xác thực, tìm kiếm và thống kê danh sách giảng viên. 1. Khi thực hiện chức năng “Xác thực giảng viên” sẽ dựa vào thông tin Giảng viên cung cấp để xác thực. 2. Khi thực hiện chức năng “Tìm kiếm giảng viên” sẽ hiển thị form tìm kiếm và nhập thông tin cần tìm kiếm, sau đó hệ thống sẽ hiển thị danh sách giảng viên phù hợp.
Kết quả	Lưu thông tin vào hệ thống khi thực hiện thành công.
Ghi chú	Không có.

Bảng 2-3: Mô tả chức năng quản lý giảng viên

2.4. Chức năng quản lý sinh viên



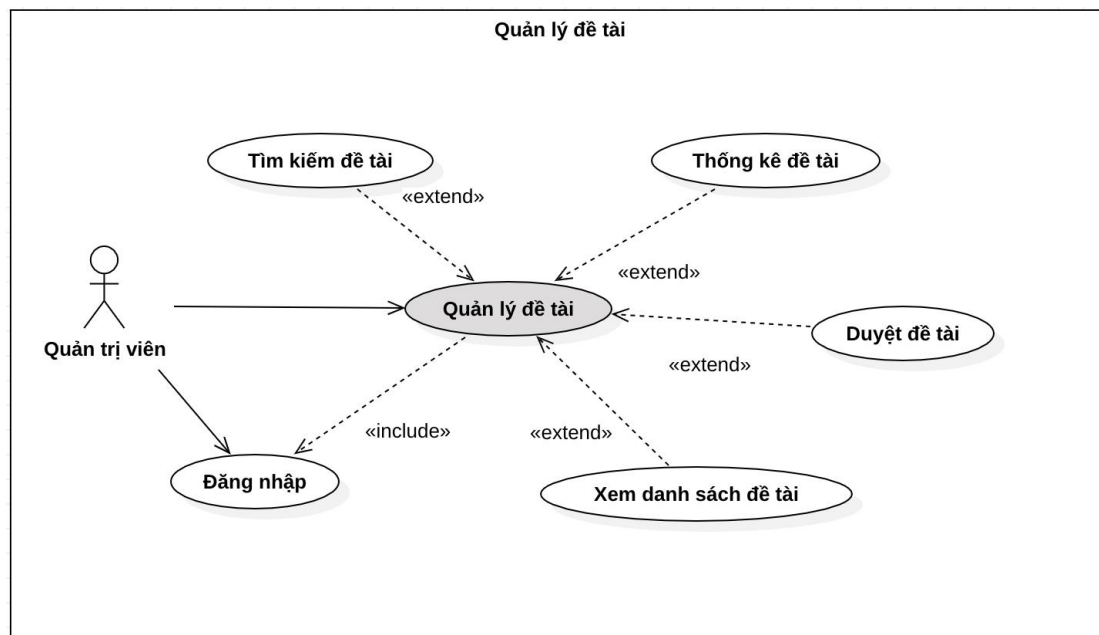
Hình 2-7: Chức năng quản lý sinh viên

Tên chức năng	Quản lý sinh viên
Đối tượng sử dụng	Quản trị viên.
Mô tả tóm tắt	Cho phép quản lý xác thực và thống kê danh sách sinh viên trong hệ thống.
Tiền điều kiện	Đã đăng nhập vào hệ thống với tài khoản Quản trị viên.
Cách xử lý	Thực hiện các chức năng quản lý cho phép xác thực, tìm kiếm và thống kê danh sách sinh viên. 1. Khi thực hiện chức năng “Xác thực sinh viên” sẽ dựa vào thông tin Sinh viên cung cấp để xác thực. 2. Khi thực hiện chức năng “Tìm kiếm sinh viên” sẽ hiển thị form tìm kiếm và nhập thông tin cần tìm kiếm, sau đó hệ thống sẽ hiển thị danh sách sinh viên phù hợp.
Kết quả	Lưu thông tin vào hệ thống khi thực hiện thành công.
Ghi chú	Không có.

Bảng 2-4: Mô tả chức năng quản lý sinh viên

2.5. Chức năng quản lý đề tài

- Đối với quản trị viên

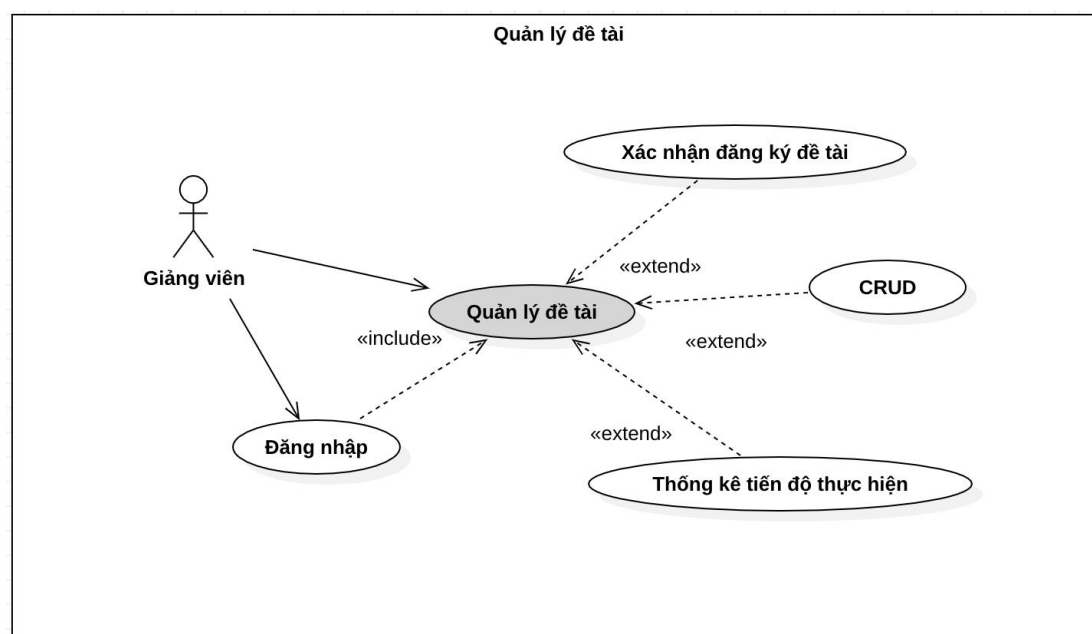


Hình 2-8: Chức năng quản lý đề tài đối với quản trị viên

Tên chức năng	Quản lý đề tài
Đối tượng sử dụng	Quản trị viên.
Mô tả tóm tắt	Cho phép quản lý duyệt đề tài trong hệ thống.
Tiền điều kiện	Đã đăng nhập vào hệ thống với tài khoản Quản trị viên.
Cách xử lý	Thực hiện các chức năng quản lý cho phép duyệt, tìm kiếm và thống kê danh sách đề tài. 1. Khi thực hiện chức năng “Duyệt đề tài” sẽ dựa vào tính khả thi của đề tài để xác thực. 2. Khi thực hiện chức năng “Tìm kiếm đề tài” sẽ hiển thị form tìm kiếm và nhập thông tin cần tìm kiếm, sau đó hệ thống sẽ hiển thị danh sách đề tài phù hợp.
Kết quả	Lưu thông tin vào hệ thống khi thực hiện thành công.
Ghi chú	Không có.

Bảng 2-5: Mô tả chức năng quản lý đề tài đối với quản trị viên

• Đối với giảng viên

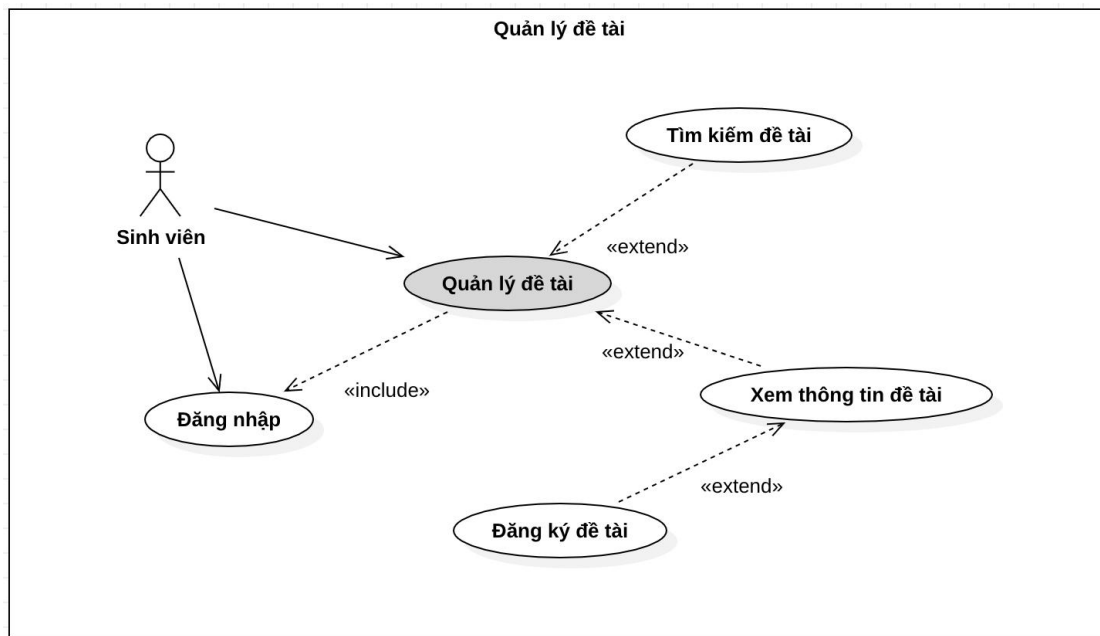


Hình 2-9: Chức năng quản lý đề tài đối với giảng viên

Tên chức năng	Quản lý đề tài
Đối tượng sử dụng	Giảng viên.
Mô tả tóm tắt	Cho phép quản lý thông tin đề tài và xác nhận đăng ký từ sinh viên.
Tiền điều kiện	Đã đăng nhập vào hệ thống với tài khoản Giảng viên và đã được xác thực từ Quản trị viên.
Cách xử lý	Thực hiện các chức năng quản lý thêm, sửa, xoá và xác nhận đăng ký đề tài. 1. Khi thực hiện chức năng “CRUD” sẽ cho phép giảng viên có thể thêm, sửa, xoá đề tài của mình. 2. Khi thực hiện chức năng “Xác nhận đăng ký đề tài” sẽ cho phép sinh viên tham gia thực hiện đề tài.
Kết quả	Lưu thông tin vào hệ thống khi thực hiện thành công.
Ghi chú	Không có.

Bảng 2-6: Mô tả chức năng quản lý đề tài đối với giảng viên

• Đối với sinh viên



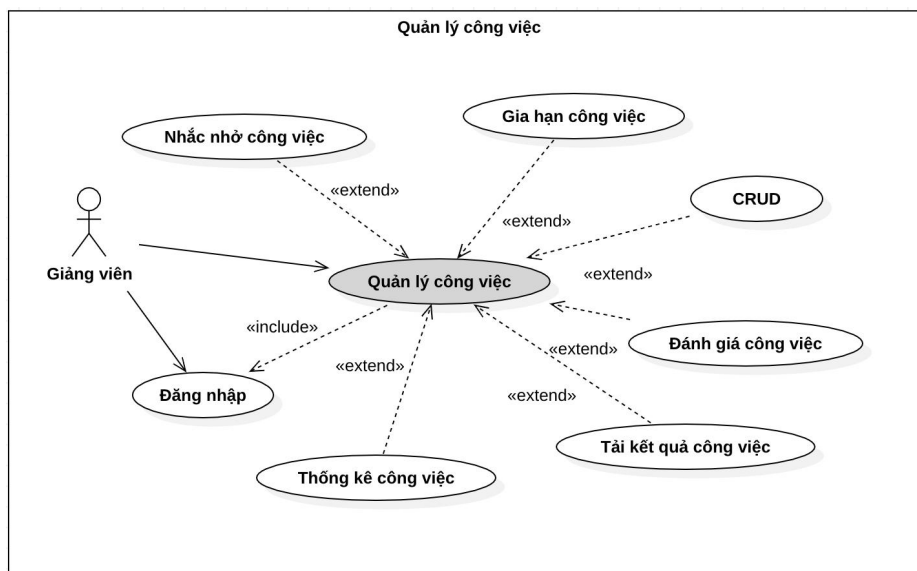
Hình 2-10: Chức năng quản lý đề tài đối với sinh viên

Tên chức năng	Quản lý đề tài
Đối tượng sử dụng	Sinh viên.
Mô tả tóm tắt	Cho phép quản lý tìm kiếm và đăng ký thực hiện đề tài.
Tiền điều kiện	Đã đăng nhập vào hệ thống với tài khoản Sinh viên và đã được xác thực từ Quản trị viên.
Cách xử lý	Thực hiện các chức năng tìm kiếm và đăng ký đề tài. 1. Khi thực hiện chức năng “Tìm kiếm đề tài” sẽ hiển thị form tìm kiếm và nhập thông tin cần tìm kiếm, sau đó hệ thống sẽ hiển thị danh sách đề tài phù hợp. 2. Khi thực hiện chức năng “Đăng ký đề tài” sẽ hiển thị form xác nhận đăng ký và nhấn chọn để xác nhận.
Kết quả	Lưu thông tin vào hệ thống khi thực hiện thành công.
Ghi chú	Không có.

Bảng 2-7: Mô tả chức năng quản lý đề tài đối với sinh viên

2.6. Chức năng quản lý công việc

- Đối với giảng viên

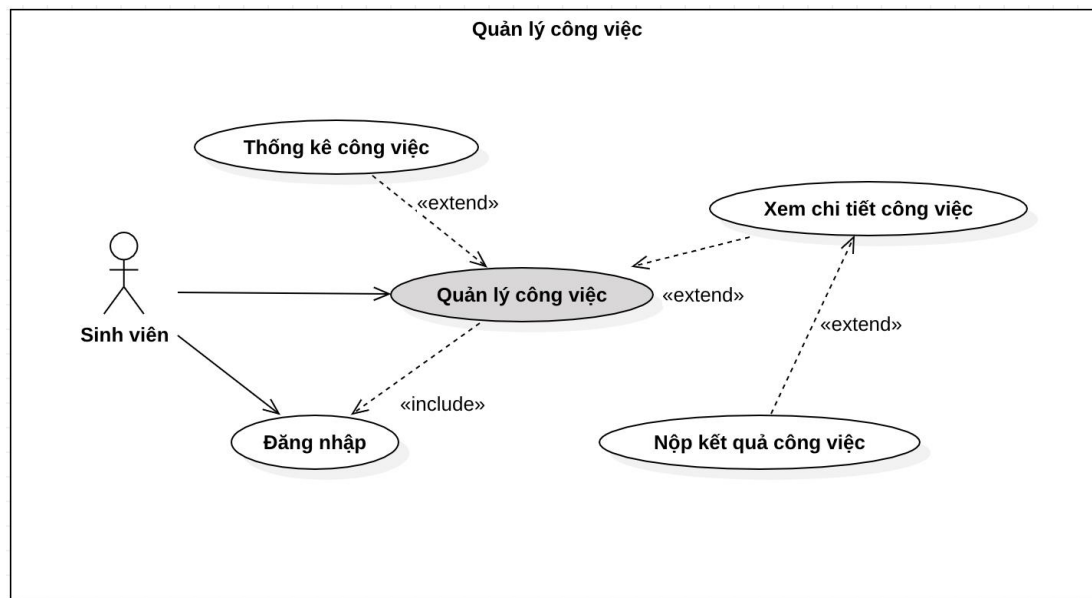


Hình 2-11: Chức năng quản lý công việc đối với giảng viên

Tên chức năng	Quản lý công việc
Đối tượng sử dụng	Giảng viên.
Mô tả tóm tắt	Cho phép thực hiện phân công và đánh giá các công việc.
Tiền điều kiện	Đã đăng nhập vào hệ thống với tài khoản Giảng viên và đã được xác thực từ Quản trị viên.
Cách xử lý	<p>Chọn vào sinh viên đã được xác nhận đăng ký đề tài.</p> <p>1. Khi thực hiện chức năng “CRUD” sẽ cho phép giảng viên có thể thêm, sửa, xóa công việc.</p> <p>2. Khi thực hiện chức năng “Nhắc nhở” và “Gia hạn” sẽ hiển thị form và nhập các thông tin cần thiết, sau đó nhấn chọn để xác nhận.</p> <p>3. Khi thực hiện chức năng “Tải kết quả” và “Đánh giá”, file kết quả công việc sẽ được lưu về máy và từ đó đánh giá tỷ lệ hoàn thành, sau đó nhấn chọn để xác nhận.</p>
Kết quả	Lưu thông tin vào hệ thống khi thực hiện thành công.
Ghi chú	Không có.

Bảng 2-8: Mô tả chức năng quản lý công việc đối với giảng viên

- Đối với sinh viên



Hình 2-12: Chức năng quản lý công việc đối với sinh viên

Tên chức năng	Quản lý công việc
Đối tượng sử dụng	Sinh viên.
Mô tả tóm tắt	Cho phép thực hiện đăng tải các nội dung được phân công và thống kê kết quả các công việc.
Tiền điều kiện	Đã đăng nhập vào hệ thống với tài khoản Sinh viên và đã được xác thực từ Quản trị viên.
Cách xử lý	Khi thực hiện chức năng “Nộp kết quả công việc” sẽ hiển thị form nộp bài và thực hiện đăng tải các nội dung giảng viên yêu cầu và nhập các ghi chú mô tả, sau đó nhấn chọn để xác nhận.
Kết quả	Lưu thông tin vào hệ thống khi thực hiện thành công.
Ghi chú	Không có.

Bảng 2-9: Mô tả chức năng quản lý công việc đối với sinh viên

3. CÁC YÊU CẦU PHI CHỨC NĂNG

3.1. Yêu cầu thực thi

- Không bị lỗi phần mềm khi chạy chương trình.
- Máy sử dụng cần có kết nối Internet.

3.2. Yêu cầu an toàn

- Đảm bảo về mặt lưu trữ dữ liệu.
- Đảm bảo an toàn dữ liệu trên hệ thống.

3.3. Yêu cầu bảo mật

- Người dùng đảm bảo phải đăng nhập trước khi có thể thao tác các chức năng cho phép trong phạm vi.
- Thông tin về người dùng khi đăng ký tài khoản phải được mã hoá và bảo mật trong cơ sở dữ liệu.

3.4. Chất lượng phần mềm

- Tính đúng đắn: Chức năng của hệ thống phải hoạt động đúng theo yêu cầu.
- Tính khoa học: Cách xây dựng, tổ chức các chức năng khoa học. Xây dựng cơ sở dữ liệu hợp lý, giảm tải nguyên lưu trữ dữ liệu trong cơ sở dữ liệu.
- Tính tin cậy: Hệ thống phải đảm bảo các chức năng thực hiện tốt.
- Tính thích nghi: Hệ thống phải đảm bảo hoạt động tốt.
- Khả năng bảo trì: Lập trình, thiết kế phải tuân thủ các tiêu chuẩn. Cần bố trí mô hình các gói thuận tiện cho việc bảo trì và phát triển.
- Khả năng tái sử dụng: Các thành phần của hệ thống phải được thiết kế riêng biệt và có các chức năng cụ thể riêng biệt. Sẵn sàng thay đổi hoặc sử dụng cho hệ thống khác.

3.5. Các yêu cầu khác

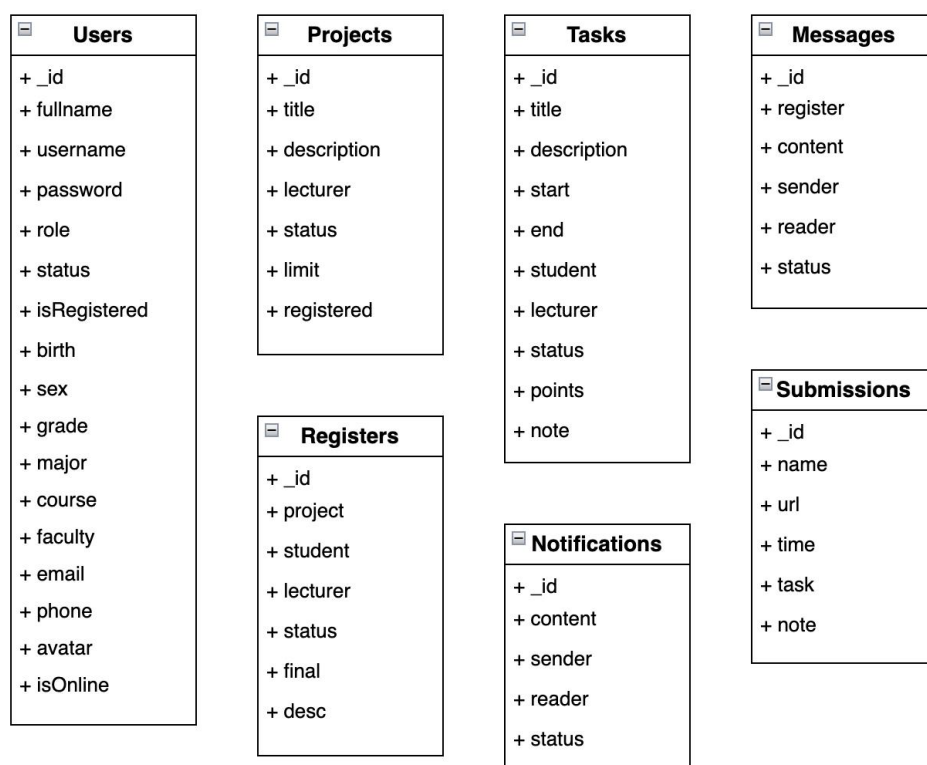
- Ngôn ngữ thuần tiếng Việt, từ ngữ đơn giản, dễ hiểu.
- Giao diện hiện đại, dễ sử dụng và các nút lệnh được sắp xếp hợp lý.
- Mức độ bảo mật cao, đáng tin cậy.
- Tốc độ xử lý nhanh chóng và chính xác.
- Dễ bảo trì và nâng cấp.

CHƯƠNG 3: XÂY DỰNG VÀ PHÁT TRIỂN PHẦN MỀM

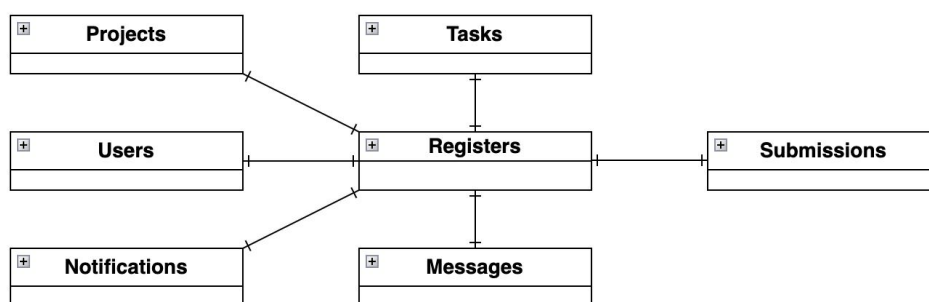
1. THIẾT KẾ DỮ LIỆU

1.1. Mô hình dữ liệu

Hệ thống thực hiện việc lưu trữ dữ liệu trên hệ quản trị cơ sở dữ liệu MongoDB là một dạng cơ sở dữ liệu phi quan hệ cho phép người dùng lưu trữ dữ liệu và truy vấn dữ liệu một cách dễ dàng và nhanh chóng dưới dạng các collection, mỗi đối tượng của MongoDB sẽ được tự động tạo ra thuộc tính `_id` để phân biệt và tìm kiếm cũng như cập nhật. Điều này giúp việc xây dựng hệ thống tiết kiệm thời gian trong khâu xây dựng cơ sở dữ liệu và phát triển sau này khi có mong muốn hoàn thiện hệ thống hơn.



Hình 3-1: Mô hình dữ liệu của ứng dụng



Hình 3-2: Bảng liên kết giữa các collections

1.2. Collection Users

Collection users chứa các thông tin về các người dùng trong ứng dụng.

```
{
  _id:      <ObjectId>    // 65f9704083d8def4dc33226a
  fullname: <String>      // Nguyễn Thành Nghĩa
  username: <String>      // B2004736
  password: <String>      // $2b$10$PoFNduu0R72RuXukofzooOfb
  role:     <Number>      // 1
  status:   <Boolean>     // true
  isRegistered: <Boolean> // true
  isOnline: <Boolean>     // false

  info: {
    birth:    <String> // 07/07/2002
    sex:      <String> // Nam
    grade:    <String> // DI20T9A1
    major:    <String> // Mạng máy tính và truyền thông dữ liệu
    course:   <String> // 46
    faculty:  <String> // Khoa Công nghệ thông tin và Truyền thông
    email:    <String> // nghiab2004736@student.ctu.edu.vn
    phone:    <String> // 0902641448
    avatar:   <String> // https://img.vn/Images/125547.jpg
  }
}
```

1.3. Collection Projects

Collection projects chứa thông tin về các đề tài trong ứng dụng.

```
{
  _id:      <ObjectId>    // 65f9704083d8def4dc39678a
  title:    <String>      // Phát triển ứng dụng web ...
  description: <String>    // Xây dựng ứng dụng hỗ trợ ...
  lecturer: <ObjectId>    // 65f9704083d8def4dc10486a
  status:   <Boolean>     // true
  limit:    <Number>      // 4
  registered: <Number>    // 1
}
```

1.4. Collection Registers

Collection registers chứa thông tin về các đăng ký thực hiện đề tài trong ứng dụng.

```
{
  _id:      <ObjectId>    // 85f9704083d8def4dc39609b
  project:  <ObjectId>    // 45f7704083d8def4dc39658c
  student:  <ObjectId>    // 76f9704083d8def4dc396b8d
  lecturer: <ObjectId>    // 68f2704083d8def4dc39642p
  status:   <Boolean>     // false
  final:    <Boolean>     // false
  desc:     <String>      //
}
```

1.5. Collection Notifications

Collection notifications chứa thông tin về các thông báo trong ứng dụng.

```
{
  _id:      <ObjectId>    // 65f9704083d8def4dc393d8d
  content:   <String>     // Bạn đã nhận được thông báo mới!
  sender:    <ObjectId>    // 85f9704083d8def4dc39b06k
  reader:    <ObjectId>    // 35f9704083d8def4dc39600p
  status:    <Boolean>     // true
}
```

1.6. Collection Messages

Collection messages chứa thông tin về các tin nhắn trong ứng dụng.

```
{
  _id:      <ObjectId>    // 65f9704083d8def4dc39678a
  register: <ObjectId>    // 65f9704083d8def4dc39678a
  sender:    <ObjectId>    // 65f9704083d8def4dc39678a
  reader:    <ObjectId>    // 65f9704083d8def4dc39678a
  content:   <String>     // Em chào thầy ạ!
  status:    <String>     // false
}
```

1.7. Collection Tasks

Collection tasks chứa thông tin về các công việc trong ứng dụng.

```
{
  _id:      <ObjectId>    // 85f9704083dsdef4dh39123r
  title:    <String>      // Phân tích hệ thống
  description: <String>    // Sử dụng các sơ đồ UML để ...
  start:    <Date>        // 2024-02-01
  end:      <Date>        // 2024-02-08
  student:  <ObjectId>    // 65f9704083d8def4dc39569n
  lecturer: <ObjectId>    // 65f9704083d8def4dc39607m
  status:   <Boolean>     // true
  points:   <Number>      // 99
  note:     <String>      // Hoàn thành tốt
}
```

1.8. Collection Submissions

Collection submissions chứa thông tin về các kết quả sinh viên thực hiện của đề tài trong ứng dụng.

```
{
  _id:      <ObjectId>    // 67f9704083d8def4dc39y95b
  name:     <String>      // phan-tich-he-thong.docx
  url:      <String>      // src/uploads/phan-tich-he-thong.docx
  time:     <Date>        // 2024-02-01
  task:     <ObjectId>    // 42f9704083d8def4dc35j7b0g
  note:     <String>      // Cần chỉnh sửa phần 2 thành ...
}
```

2. THIẾT KẾ THEO CHỨC NĂNG

2.1. Chức năng đăng nhập

- Mục đích: Đăng nhập tài khoản với các thông tin đã đăng ký trong ứng dụng.
- Giao diện đăng nhập:

Đề tài: Phát triển ứng dụng quản lý tiến độ đề tài

Xây dựng một ứng dụng web cho phép người dùng theo dõi, kiểm soát và cập nhật tiến độ của các đề tài bằng công nghệ MERN Stack.

Các chức năng chính:

- Quản lý đề tài và công việc
- Theo dõi tiến độ và đánh giá
- Thông báo và nhắc nhở
- Báo cáo và thống kê

Sinh viên thực hiện
Nguyễn Thành Nghĩa B2004736

Giảng viên hướng dẫn
Lâm Chí Nguyễn 001708

Đăng nhập

Tên đăng nhập

Mật khẩu

☐ Ghi nhớ đăng nhập

Đăng nhập

Đăng ký

Năm thực hiện: 2024

Hình 3-3: Giao diện đăng nhập

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Input	Tài khoản/Mã số	
2	Input	Mật khẩu	
3	Button	Nút đăng nhập	
4	Link	Đăng ký tài khoản	

Bảng 3-1: Thành phần giao diện đăng nhập

- Các thao tác: Ở giao diện này để đăng nhập ta nhập tên tài khoản, mật khẩu. Sau đó nhấp chọn vào nút Đăng nhập.

2.2. Chức năng đăng ký

- Mục đích: Đăng ký tài khoản để lưu trữ thông tin trong ứng dụng.
- Giao diện đăng ký:

The screenshot shows a web application interface for registration. At the top, there is a header with the logo of Truong Dai Hoc Can Tho (CTU) and the text 'TRƯỜNG ĐẠI HỌC CẦN THƠ' and 'Trường Công nghệ thông tin và Truyền thông'. To the right, it says 'ỨNG DỤNG Quản lý tiến độ đề tài'. The main content area is titled 'Đăng ký' (Registration). Below the title, there is a description of the application: 'Đề tài: Phát triển ứng dụng quản lý tiến độ đề tài' and 'Xây dựng một ứng dụng web cho phép người dùng theo dõi, kiểm soát và cập nhật tiến độ của các đề tài bằng công nghệ MERN Stack.' Below this, there are 'Các chức năng chính:' (Main functions:) listed as: 'Quản lý đề tài và công việc', 'Theo dõi tiến độ và đánh giá', 'Thông báo và nhắc nhở', and 'Báo cáo và thống kê'. There are two sections for user roles: 'Sinh viên thực hiện' (Student performing) with the name 'Nguyễn Thành Nghĩa B2004736' and 'Giảng viên hướng dẫn' (Instructor) with the name 'Lâm Chí Nguyễn 001708'. The registration form itself has six numbered annotations: 1 points to the 'Họ và tên' (Last name and first name) input field; 2 points to the 'Tên đăng nhập' (Username) input field; 3 points to the 'Mật khẩu' (Password) input field; 4 points to the radio buttons for 'Sinh viên' (Student) and 'Giảng viên' (Instructor); 5 points to the 'Đăng ký' (Register) button; and 6 points to the 'Đăng nhập' (Login) link. The footer of the page says 'Năm thực hiện: 2024'.

Hình 3-4: Giao diện đăng ký

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Input	Họ và tên	
2	Input	Tài khoản/Mã số	
3	Input	Mật khẩu	
4	Radio	Vai trò sinh viên/giảng viên	
5	Button	Nút đăng ký	
6	Link	Đăng nhập tài khoản	

Bảng 3-2: Thành phần giao diện đăng ký

- Các thao tác: Ở giao diện này để đăng ký ta nhập họ tên, tên tài khoản, mật khẩu và chọn vai trò. Sau đó nhấp chọn vào nút Đăng ký.

2.3. Chức năng đổi mật khẩu

- Mục đích: Đổi mật khẩu tài khoản người dùng trong ứng dụng.
- Giao diện đổi mật khẩu:

The screenshot shows the 'Đổi mật khẩu' (Change password) page. The header includes the logo of Trường Đại học Cần Thơ (CTU) and the text 'TRƯỜNG ĐẠI HỌC CẦN THƠ' and 'Trường Công nghệ thông tin và Truyền thông'. The user is logged in as 'Nguyễn Thành Nghĩa'. The page has tabs for 'Tổng quan', 'Cập nhật thông tin', and 'Đổi mật khẩu'. The 'Đổi mật khẩu' tab is active. The form contains three input fields: 'Mật khẩu hiện tại', 'Mật khẩu mới', and 'Nhập lại mật khẩu mới'. A blue button 'Đổi mật khẩu' is at the bottom. Red numbers 1, 2, 3, and 4 point to the 'Đổi mật khẩu' button, the 'Mật khẩu mới' field, the 'Nhập lại mật khẩu mới' field, and the 'Mật khẩu hiện tại' field respectively.

Hình 3-5: Giao diện đổi mật khẩu

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Input	Mật khẩu hiện tại	
2	Input	Mật khẩu mới	
3	Input	Nhập lại mật khẩu mới	
4	Button	Đổi mật khẩu	

Bảng 3-3: Thành phần giao diện đổi mật khẩu

- Các thao tác: Ở giao diện này để đổi mật khẩu ta nhập mật khẩu cũ và mật khẩu mới 2 lần. Sau đó nhấp chọn vào nút Đổi mật khẩu.

2.4. Chức năng cập nhật thông tin

- Mục đích: Cập nhật thông tin người dùng trong ứng dụng.

- Giao diện cập nhật thông tin:

Hình 3-6: Giao diện cập nhật thông tin người dùng

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	File	Ảnh đại diện	
2	Icon	Gỡ ảnh đại diện	
3	Input	Ngày sinh	
4	Input	Giới tính	
5	Input	Lớp	
6	Input	Ngành học	
7	Input	Khoá học	
8	Input	Khoa	
9	Button	Nút lưu thông tin	

Bảng 3-4: Thành phần giao diện cập nhật thông tin người dùng

- Các thao tác: Ở giao diện này để cập nhật thông tin ta nhập các thông tin cần thiết. Sau đó nhấp chọn vào nút Lưu thông tin.

2.5. Chức năng thêm đề tài

- Mục đích: Giảng viên đề xuất đề tài đến quản trị viên trong ứng dụng.
- Giao diện thêm đề tài:

TRƯỜNG ĐẠI HỌC CẦN THƠ
Trường Công nghệ thông tin và Truyền thông

ỨNG DỤNG
Quản lý tiến độ đề tài

Lâm Chí Nguyên

Trang chủ / Thêm đề tài

Thêm đề tài 1

Tên đề tài

Mô tả đề tài

Normal B I U

Thêm đề tài 3 **Trở lại** 4

Năm thực hiện: 2024

Hình 3-7: Giao diện thêm đề tài

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Input	Tên đề tài	
2	Text Editor	Mô tả đề tài	
3	Button	Nút thêm đề tài	
4	Button	Nút trở lại	

Bảng 3-5: Thành phần giao diện thêm đề tài

- Các thao tác: Ở giao diện này để thêm đề tài ta nhập tên đề tài và mô tả đề tài. Sau đó nhấp chọn vào nút Tạo đề tài.

2.6. Chức năng cập nhật đề tài

- Mục đích: Giảng viên cập nhật lại thông tin đề tài trong ứng dụng.
- Giao diện cập nhật đề tài:

The screenshot shows the 'Cập nhật đề tài' (Update Topic) form. At the top, there's a header with the school logo and name 'TRƯỜNG ĐẠI HỌC CẦN THƠ' and the application name 'ỨNG DỤNG Quản lý tiến độ đề tài'. Below the header, there's a navigation bar with a home icon and the user name 'Lâm Chí Nguyễn'. The main content area has a breadcrumb 'Trang chủ / Cập nhật đề tài' and a title 'Cập nhật đề tài'. There are two input fields: 'Tên đề tài' (Topic Name) with the value 'Phát triển ứng dụng web quản lý luận văn' (1), and 'Mô tả đề tài' (Topic Description) which is a rich text editor containing the text 'Xây dựng ứng dụng web để quản lý luận văn' and a bulleted list (2). At the bottom, there are two buttons: 'Cập nhật đề tài' (3) and 'Trở lại' (4). The footer shows 'Năm thực hiện: 2024'.

Hình 3-8: Giao diện cập nhật đề tài

- Các thành phần giao diện:

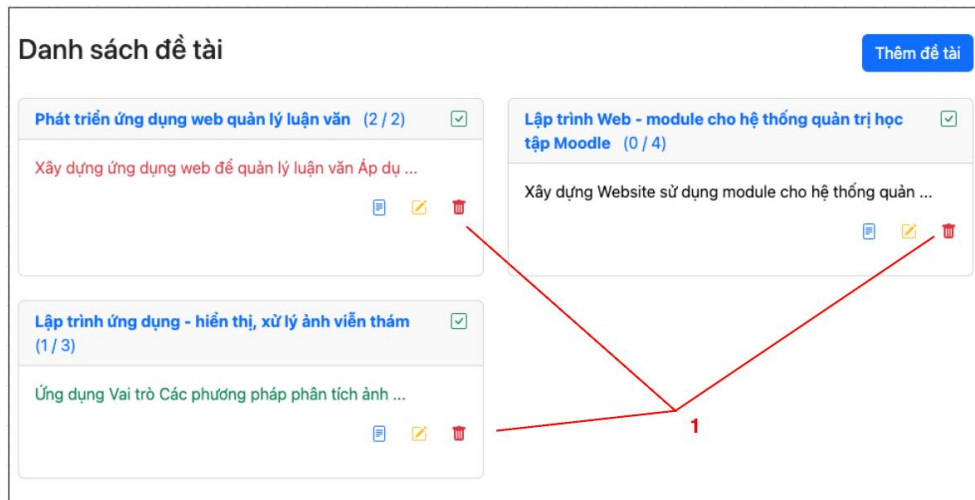
STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Input	Tên đề tài hiện tại	
2	Text Editor	Mô tả đề tài hiện tại	
3	Button	Nút cập nhật đề tài	
4	Button	Nút trở lại	

Bảng 3-6: Thành phần giao diện cập nhật đề tài

- Các thao tác: Ở giao diện này để cập nhật đề tài ta thay đổi các thông tin cần thiết. Sau đó nhấp chọn vào nút Cập nhật đề tài.

2.7. Chức năng ẩn đề tài

- Mục đích: Giảng viên ẩn đề tài trong ứng dụng.
- Giao diện ẩn đề tài:



Hình 3-9: Giao diện ẩn đề tài



Hình 3-10: Giao diện xác nhận ẩn đề tài

- Các thành phần giao diện:

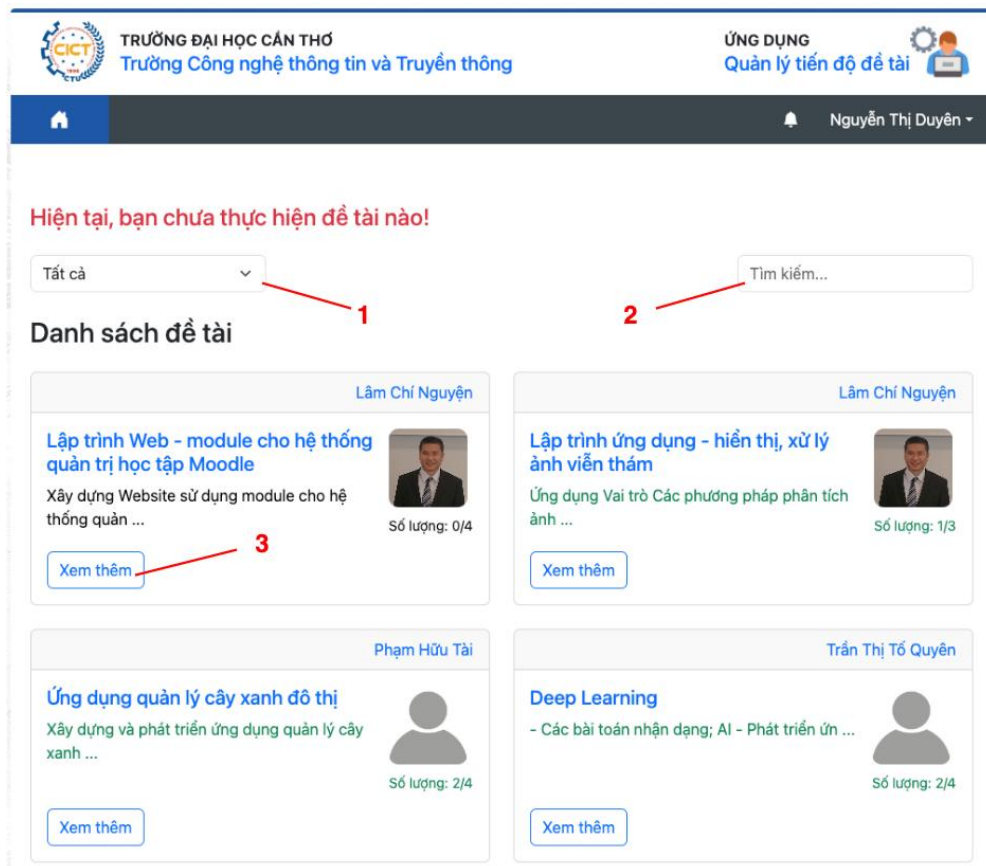
STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Icon	Ẩn đề tài	
2	Button	Xác nhận ẩn đề tài	
3	Button	Nút hủy bỏ	
4	Icon	Tắt ẩn đề tài	

Bảng 3-7: Thành phần giao diện ẩn đề tài

- Các thao tác: Ở giao diện này để ẩn đề tài ta nhấp vào biểu tượng thùng rác màu đỏ trong thẻ đề tài. Sau đó nhấp chọn vào nút Xóa.

2.8. Chức năng tìm kiếm đề tài

- Mục đích: Sinh viên chưa có đăng ký tìm kiếm đề tài trong ứng dụng.
- Giao diện tìm kiếm đề tài:



Hình 3-11: Giao diện tìm kiếm đề tài

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Select	Tên giảng viên	
2	Input	Nội dung tìm kiếm	
3	Button	Xem thông tin đề tài	

Bảng 3-8: Thành phần giao diện tìm kiếm đề tài

- Các thao tác: Ở giao diện này ta có thể lọc đề tài theo tên giảng viên bằng cách chọn vào danh sách bên trái hoặc nhập từ khoá tìm kiếm vào ô tìm kiếm bên phải.

2.9. Chức năng đăng ký đề tài

- Mục đích: Sinh viên đăng ký đề tài đến giảng viên trong ứng dụng.
- Giao diện đăng ký đề tài:

Hình 3-12: Giao diện đăng ký đề tài

Hình 3-13: Giao diện xác nhận đăng ký đề tài

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Button	Nút đăng ký đề tài	
2	Button	Nút quay lại	
3	Button	Xác nhận đăng ký đề tài	
4	Button	Nút hủy bỏ	
5	Icon	Tắt đăng ký đề tài	

Bảng 3-9: Thành phần giao diện đăng ký đề tài

- Các thao tác: Ở giao diện này để đăng ký đề tài ta nhấp vào nút đăng ký bên dưới tên giảng viên. Sau đó nhấp chọn vào nút Đăng ký.

2.10. Chức năng thêm công việc

- Mục đích: Giảng viên thêm công việc đến sinh viên thực hiện đề tài.
- Giao diện thêm công việc:

The screenshot shows a web application interface for adding a task. At the top, there's a header with the university logo and name 'TRƯỜNG ĐẠI HỌC CẦN THƠ' and 'Trường Công nghệ thông tin và Truyền thông'. Below the header, there's a navigation bar with a home icon and a user profile 'Lâm Chí Nguyễn'. The main content area is titled 'Phân công đến "Nguyễn Thành Nghĩa"'. It contains a form with the following elements: a text input field for 'Tên công việc' (Task Name) labeled with a red '1'; a text editor for 'Mô tả công việc' (Task Description) labeled with a red '2'; two date pickers for 'Ngày bắt đầu' (Start Date) and 'Ngày kết thúc' (End Date) labeled with red '3' and '4' respectively; and two buttons at the bottom, 'Phân công' (Assign) labeled with a red '5' and 'Trở lại' (Back) labeled with a red '6'. The year '2024' is shown at the bottom right.

Hình 3-14: Giao diện thêm công việc

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Input	Tên công việc	
2	Text Editor	Mô tả công việc	
3	Date	Ngày bắt đầu	
4	Date	Ngày kết thúc	
5	Button	Nút phân công	
6	Button	Nút trở lại	

Bảng 3-10: Thành phần giao diện thêm công việc

- Các thao tác: Ở giao diện này để thêm công việc ta nhập tên công việc, mô tả công việc và thời gian thực hiện. Sau đó nhấp chọn vào nút Phân công.

2.11. Chức năng nhắc nhở và gia hạn

- Mục đích: Nhắc nhở hoặc gia hạn công việc sinh viên chưa hoàn thành.
- Giao diện nhắc nhở và gia hạn công việc:

Tên công việc: *Phân tích thiết kế hệ thống* Chưa có đánh giá!

Mô tả

Sử dụng các sơ đồ để phân tích các chức năng

- Use Case
- Class
- Sequence

Ngày bắt đầu: 2024-04-02 Ngày kết thúc: 2024-04-09

Sinh viên chưa đăng tải nội dung

Nhắc nhở Gia hạn

Hình 3-15: Giao diện nhắc nhở và gia hạn

Nhắc nhở công việc

Nhập nội dung:

Trở lại Nhắc nhở

Hình 3-16: Giao diện nhắc nhở

Gia hạn công việc

Nhập số ngày gia hạn:

Trở lại Gia hạn

Hình 3-17: Giao diện gia hạn

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Button	Nút nhắc nhở	
2	Button	Nút gia hạn	
3	Textarea	Nội dung nhắc nhở	
4	Button	Xác nhận nhắc nhở	
5	Button	Tắt nhắc nhở	
6	Input	Số ngày gia hạn	
7	Button	Xác nhận gia hạn	
8	Button	Tắt gia hạn	

Bảng 3-11: Thành phần giao diện nhắc nhở và gia hạn

- Các thao tác:

- Để nhắc nhở ta nhấp vào nút nhắc nhở màu vàng. Sau đó nhập nội dung và nhấn vào nút nhắc nhở.
- Để gia hạn công việc ta nhấp vào nút gia hạn. Sau đó nhập số ngày và nhấn vào nút gia hạn.

2.12. Chức năng tải file và đánh giá

- Mục đích: Tải file để xem xét kết quả công việc và đánh giá kết quả công việc sinh viên đã hoàn thành.

- Giao diện tải file và đánh giá công việc:

Hình 3-18: Giao diện tải file và đánh giá

Hình 3-19: Giao diện đánh giá

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Button	Nút tải file	
2	Button	Nút đánh giá	
3	Input	Tỷ lệ hoàn thành	
4	Textarea	Ghi chú	
5	Button	Đánh giá	
6	Button	Tắt đánh giá	

Bảng 3-12: Thành phần giao diện tải file và đánh giá

- Các thao tác:

- Để tải file ta nhấp vào nút tải file màu vàng. Sau đó file sẽ được tự động lưu về máy.
- Để đánh giá công việc ta nhấp vào nút đánh giá. Sau đó nhập số điểm, ghi chú và nhấn vào nút đánh giá.

2.13. Chức năng nộp bài

- Mục đích: Gửi kết quả công việc đã hoàn thành đến giảng viên hướng dẫn trong ứng dụng.

- Giao diện nộp bài:

Tên công việc: Phát triển module bằng ngôn ngữ lập trình python Chưa có đánh giá!

Mô tả

Dựa trên các sơ đồ thiết kế UML phát triển chi tiết các thành phần của ứng dụng

Ngày bắt đầu: 2024-05-22 **Ngày kết thúc:** 2024-07-22

Đăng tải nội dung theo yêu cầu

Choose File

No file chosen

Ghi chú:

Nộp bài

Hình 3-20: Giao diện nộp bài

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	File	Kết quả công việc	
2	Textarea	Ghi chú	
3	Button	Nút nộp bài	

Bảng 3-13: Thành phần giao diện nộp bài

- Các thao tác: Ở giao diện này để nộp bài ta đăng tải các nội dung được yêu cầu từ công việc và ghi chú. Sau đó nhấp vào nút Nộp bài.

2.14. Chức năng hoàn thành và chấm dứt đề tài

- Mục đích: Xác nhận đề tài đã hoàn thành hoặc bị chấm dứt.
- Giao diện hoàn thành và chấm dứt đề tài:

Tiến độ thực hiện

Chế độ xem: ☐ Ngày ☒ Tuần ☐ Tháng

W06 W07 February, 2024 W08 W09 W10 W11 March, 2024 W12 W13 W14 W

Viết mô tả đề tài

Demo MERN Stack

Xây dựng các chức năng cho hệ thống

Tích hợp hệ thống cho mobile

Viết báo cáo và bảo vệ đề tài

Tên công việc: **Viết mô tả đề tài** **Đã hoàn thành (50%)!**

Mô tả

Viết mô tả đề tài đặt vấn đề khoảng 10 trang

- Đặt vấn đề
- Lịch sử giải quyết vấn đề
- Mục tiêu
- Đối tượng và phạm vi

Ngày bắt đầu: 2024-02-01 Ngày kết thúc: 2024-02-08

Đã nộp

1 2

Tổng số công việc hoàn thành: 5/5

Hoàn thành Kết thúc Trở lại

Hình 3-21: Giao diện hoàn thành và chấm dứt đề tài

Hoàn thành đề tài

Nhập nội dung:

Trở lại Hoàn thành

Hình 3-22: Giao diện hoàn thành đề tài

Kết thúc đề tài

Nhập lý do kết thúc:

Trở lại Kết thúc

Hình 3-23: Giao diện chấm dứt đề tài

- Các thành phần giao diện:

STT	Loại điều khiển	Giá trị mặc định	Lưu ý
1	Button	Nút hoàn thành	
2	Button	Nút kết thúc	
3	Textarea	Nội dung đánh giá	
4	Button	Nút xác nhận hoàn thành	
5	Button	Nút ẩn hoàn thành	
6	Textarea	Lý do kết thúc	
7	Button	Nút xác nhận kết thúc	
8	Button	Nút ẩn kết thúc	

Bảng 3-14: Thành phần giao diện hoàn thành và chấm dứt đề tài

- Các thao tác:

- Để hoàn thành đề tài ta kiểm tra tiến độ thực hiện của sinh viên và nhấp vào nút hoàn thành. Sau đó nhập nội dung đánh giá và nhấp chọn vào nút hoàn thành.
- Để chấm dứt đề tài ta kiểm tra tiến độ thực hiện của sinh viên và nhấp vào nút kết thúc. Sau đó nhập lý do kết thúc và nhấp chọn vào nút kết thúc.

PHẦN KẾT LUẬN

1. KẾT QUẢ ĐẠT ĐƯỢC

1.1. Về lý thuyết

Sau quá trình nghiên cứu, tìm tòi, học hỏi về công nghệ và kiến thức chuyên môn để thực hiện đề tài thì bản thân đã có được cái nhìn tổng quan hơn trong việc xây dựng một website từ phân tích thiết kế hệ thống và cuối cùng là lập trình.

Bên cạnh đó, bản thân cũng học được nhiều kiến thức và công nghệ mới cũng như việc sử dụng các phần mềm hỗ trợ như Visual Studio Code, Postman, MongoDB Atlas.

1.2. Về chương trình

Kết quả đạt được sau khi thực hiện đề tài là bản thân đã xây dựng thành công một hệ thống quản lý tiến độ đề tài bao gồm các chức năng cơ bản như đăng nhập, đăng ký, đổi mật khẩu, cập nhật thông tin người dùng và các tính năng cơ bản của một website quản lý như quản lý giảng viên, sinh viên, đề tài, tìm kiếm, đánh giá các đề tài thực hiện và các công việc phân công.

2. HẠN CHẾ

Vì thời gian có hạn và kiến thức của bản thân còn hạn chế nên vẫn chưa hoàn thành ở những vấn đề sau:

- Chưa responsive tốt ứng dụng để sử dụng trên tất cả các thiết bị khác nhau.
- Tính năng đăng ký người dùng chưa hiệu quả.
- Chưa có chức năng hậu hoàn thành đề tài.

3. HƯỚNG PHÁT TRIỂN

Một số tính năng có thể phát triển thêm:

- Xây dựng chức năng gợi đề tài cho sinh viên, các tài liệu liên quan đến đề tài.
- Xây dựng tính năng hỗ trợ giảng viên có thể đánh giá luận văn cho buổi báo cáo khi các giảng viên tham gia báo cáo.
- Xây dựng tính năng cho phép thêm người dùng tự động.

TÀI LIỆU THAM KHẢO

- [1]. Quick Start - React. [Online]. Có sẵn tại: <https://react.dev/learn>. [Lần cuối truy cập ngày 7/5/2024]
- [2]. Nodejs.org About Node.js. [Online]. Có sẵn tại: <https://nodejs.org/en/about>. [Lần cuối truy cập ngày 7/5/2024]
- [3]. Express - Node.js web application framework. [Online]. Có sẵn tại: <https://expressjs.com>. [Lần cuối truy cập ngày 7/5/2024]
- [4]. MongoDB Documentation. [Online]. Có sẵn tại: <https://www.mongodb.com/docs>. [Lần cuối truy cập ngày 7/5/2024]
- [5]. Mongoose v8.3.4: Documents. [Online]. Có sẵn tại <https://mongoosejs.com/docs/documents.html>. [Lần cuối truy cập ngày 7/5/2024]
- [6]. Get started with Bootstrap. [Online]. Có sẵn tại: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>. [Lần cuối truy cập ngày 7/5/2024]
- [7]. Interactive Gantt Chart for React. [Online]. Có sẵn tại: <https://www.npmjs.com/package/gantt-task-react>. [Lần cuối truy cập ngày 7/5/2024]
- [8]. Doughnut and Pie Charts. [Online]. Có sẵn tại: <https://www.chartjs.org/docs/latest/charts/line.html>. [Lần cuối truy cập ngày 7/5/2024]
- [9] JWT, “JSON Web Token Introduction,” JWT. [Online]. Có sẵn tại: <https://jwt.io/introduction/>. [Lần cuối truy cập ngày 7/5/2024]