

ỦY BAN NHÂN DÂN TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC SÀI GÒN

NGUYỄN HOÀNG TIẾN – 3121411206
NGUYỄN MINH TRÍ - 3121411212

XÂY DỰNG TRANG WEB QUẢN LÝ HOẠT ĐỘNG
NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN

KHÓA LUẬN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN
TRÌNH ĐỘ ĐÀO TẠO: ĐẠI HỌC

NGƯỜI HƯỚNG DẪN: TS. CAO THÁI PHƯƠNG THANH

TP. HỒ CHÍ MINH, THÁNG 5 NĂM 2025

ỦY BAN NHÂN DÂN TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC SÀI GÒN

NGUYỄN HOÀNG TIẾN – 3121411206
NGUYỄN MINH TRÍ - 3121411212

XÂY DỰNG TRANG WEB QUẢN LÝ HOẠT ĐỘNG
NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN

KHÓA LUẬN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN
TRÌNH ĐỘ ĐÀO TẠO: ĐẠI HỌC

NGƯỜI HƯỚNG DẪN: TS. CAO THÁI PHƯƠNG THANH

TP. HỒ CHÍ MINH, THÁNG 5 NĂM 2025

LỜI CAM ĐOAN

Chúng em xin cam đoan đây là công trình nghiên cứu của riêng chúng em, các số liệu và kết quả nghiên cứu nêu trong khóa luận là trung thực và chưa từng được công bố trong bất kì một công trình nào khác.

Tác giả khóa luận

**Nguyễn Hoàng Tiến
Trí**

Nguyễn Minh

LỜI CẢM ƠN

Đầu tiên, chúng em xin chân thành cảm ơn các quý Thầy Cô giảng viên khoa Công nghệ thông tin Trường Đại học Sài Gòn đã tận tâm giảng dạy, truyền đạt và xây dựng cho chúng em một nền tảng khá vững chắc để em có thể thực hiện Khóa luận tốt nghiệp này.

Đặc biệt, chúng em xin gửi lời cảm ơn sâu sắc đến thầy đã cung cấp các gợi ý, đóng góp các ý kiến, chỉ ra ưu và khuyết điểm của Khóa luận tốt nghiệp này giúp chúng em cải tiến và hoàn thành nó tốt hơn.

Sau khoảng thời gian 15 tuần thực hiện, chúng em đã nỗ lực hết sức và sử dụng tất cả kiến thức mà mình có để xây dựng Khóa luận tốt nghiệp một cách tốt nhất. Tuy vậy, trong quá trình thực hiện không thể không xảy ra sai sót nên tôi rất mong các quý Thầy, Cô thông cảm và góp ý để sản phẩm của Khóa luận tốt nghiệp này có thể phát triển tốt hơn trong tương lai.

Nhóm sinh viên thực hiện:

Nguyễn Hoàng Tiến

Nguyễn Minh Trí

[illegible]

Hồ Chí Minh, ngày....tháng....năm 2025
Giảng viên hướng dẫn

Mục lục

LỜI CAM ĐOAN	3
LỜI CẢM ƠN	4
NHẬN XÉT CỦA GIẢNG VIÊN.....	5
DANH MỤC HÌNH ẢNH.....	10
DANH MỤC BẢNG.....	11
DANH MỤC CÁC CHỮ VIẾT TẮT.....	13
CHƯƠNG 1 : TỔNG QUAN.....	16
1.1. Đặt vấn đề và tính cấp thiết	16
1.2. Mục tiêu của đề tài.....	17
1.3. Đối tượng và phạm vi nghiên cứu	19
1.4. Ý nghĩa của đề tài.....	21
1.5. Phương pháp nghiên cứu	21
1.6. Cấu trúc báo cáo	22
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ.....	23
2.1. Tổng quan về hệ thống	23
2.2. Kiến trúc hệ thống	24
2.2.1. Khái niệm về MERN Stack	24
2.2.2. Các thành phần trong MERN Stack.....	25
2.2.3.Cách MERN Stack hoạt động.....	26
2.2.4. Lợi thế của MERN Stack	27
2.3.Công nghệ Back-End	27
2.3.1.Khái niệm NodeJS.....	27
2.3.2. Đặc điểm của NodeJS	27
2.3.3.Khái niệm NestJS	29
2.3.4.Đặc điểm của Nest JS.....	29
2.4.Công Nghệ Front-End.....	31
2.4.1.Khái niệm ReactJS.....	31
2.4.2.Tại sao lại chọn ReactJS.....	31
2.5.Cơ sở dữ liệu MongoDB.....	33

2.5.1. Khái niệm MongoDB	33
2.5.2. Đặc điểm của MongoDB	33
2.6. PostMan	34
2.7. JSON Web Token	35
2.7.1. JWT-Định nghĩa và ứng dụng trong bảo mật Client-Server	35
2.7.2. Cấu trúc của JWT	35
2.7.3. Ưu điểm của JWT	36
CHƯƠNG 3 : PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	37
3.1. Phân tích yêu cầu hệ thống	37
3.1.2. Yêu cầu phi chức năng	40
3.2. Tác nhân hệ thống	41
3.3. Lược đồ Use Case	41
3.3.1. Lược đồ Use Case Sinh viên	41
3.3.2. Lược đồ Use Case Giảng Viên	42
3.3.3. Lược đồ Use Case Giáo vụ	45
3.4. Đặc tả Use Case	46
3.4.1. Use case chung	46
3.4.2. Sinh Viên	48
3.4.3. Giảng viên	53
3.4.4. Giáo vụ	62
CHƯƠNG 4 : THIẾT KẾ PHẦN MỀM	77
4.1. MÔ TẢ CƠ SỞ DỮ LIỆU	
4.1.1. Thông tin các trường trong bảng	77
4.1.2. Mô hình ERD	83
4.2. Hiện thực thông qua MongoDB	84
4.2.1. Bảng Topics	84
4.2.2. Bảng User	84
4.2.3. Bảng UserNotifications	85
4.2.4. Bảng TopicCategories	86

4.2.5. Bảng Semesters	86
4.2.6. Bảng ScoreBoards.....	87
4.2.7. Bảng RubricEvaluations	87
4.2.8. Bảng Rubrics	88
4.2.9. Bảng RegistrationPeriods	89
4.2.10. Bảng Majors	89
4.2.11. Bảng GroupStudents	90
4.2.12. Bảng Faculties	90
4.2.13. Bảng Assemblies.....	91
CHƯƠNG 5 : TỔNG QUAN HỆ THỐNG.....	92
5.1. VAI TRÒ CỦA ỨNG DỤNG	92
5.1.1. Mô hình tổng quan của ứng dụng	92
5.1.2. Giới thiệu các ứng dụng.....	93
5.2. Ứng dụng web dành cho sinh viên.....	94
5.2.1. Công nghệ sử dụng.....	94
5.2.2. Mô tả luồng hoạt động	95
5.2.2.1. Luồng đăng nhập	95
5.2.2.2. Luồng đề xuất đề tài.....	96
5.2.2.3. Luồng ghi danh đề tài	98
5.2.2.4. Luồng hủy ghi danh đề tài	100
5.2.2.5. Luồng cập nhật thông tin bổ sung đề tài	102
5.2.2.6. Luồng xem thông tin chi tiết đề tài đã đăng ký thực hiện	103
CHƯƠNG 6 : KIỂM THỬ HỆ THỐNG	105
6.1.MỤC ĐÍCH KIỂM THỬ.....	105
6.2. PHẠM VI KIỂM TRA	105
6.3. CHIẾN LƯỢC KIỂM THỬ ÁP DỤNG	105
6.4. CÁC PHƯƠNG PHÁP KIỂM THỬ	106
6.5. KIỂM THỬ CHỨC NĂNG.....	107

6.5.1. Kiểm thử chức năng Đăng ký, Đăng nhập	107
6.5.2. Kiểm thử chức năng Cập nhật profile người dùng.....	110
6.5.3. Kiểm thử chức năng Xem danh sách đề tài NCKH, sinh viên, giảng viên, chi tiết đề tài NCKH	113
6.5.4. Kiểm thử chức năng Quản lý đề tài NCKH ..	116
6.5.5. Kiểm thử chức năng Quản lý phiếu đánh giá	120
6.5.6. Kiểm thử chức năng Quản lý hội đồng	123
6.5.7. Kiểm thử chức năng Chấm điểm.....	126
6.6. KẾT QUẢ KIỂM THỬ	130
CHƯƠNG 7 : KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	132
7.1. KẾT QUẢ ĐẠT ĐƯỢC	132
7.2. ĐÁNH GIÁ HỆ THỐNG.....	133
7.2.1.Ưu điểm	133
7.2.2.Nhược điểm	133
7.2.3. Hướng phát triển.....	134
7.3. KẾT LUẬN.....	135
TÀI LIỆU KHAM THẢO	136

DANH MỤC HÌNH ẢNH

Hình 2.1 Giải thích MERN Stack	17
Hình 2.2 Sơ đồ hoạt động MERN Stack	19
Hình 2.3 Virtual DOM	25
Hình 2.4 One-Way Data binding	25
Hình 3.1 Lược đồ Use Case Sinh viên	35
Hình 3.2 Lược đồ Use Case Giảng Viên	37
Hình 3.3 Lược đồ Use Case Giáo vụ	39
Hình 4.1 Mô hình ERD	74
Hình 4.2 Cấu trúc Bảng Topics trên MongoDB	75
Hình 4.3 Cấu trúc Bảng Users trên MongoDB	76
Hình 4.4 Cấu trúc Bảng UserNotifications trên MongoDB	77
Hình 4.5 Cấu trúc Bảng TopicCategories trên MongoDB.....	77
Hình 4.6 Cấu trúc Bảng Semesters trên MongoDB	78
Hình 4.7 Cấu trúc Bảng ScoreBoards trên MongoDB	78
Hình 4.8 Cấu trúc Bảng RubricEvaluations trên MongoDB	79
Hình 4.9 Cấu trúc Bảng Rubrics trên MongoDB	79
Hình 4.10 Cấu trúc Bảng RegistrationPeriods trên MongoDB	80
Hình 4.11 Cấu trúc Bảng Majors trên MongoDB	80
Hình 4.12 Cấu trúc Bảng GroupStudents trên MongoDB	81
Hình 4.13 Cấu trúc Bảng Faculties trên MongoDB	81
Hình 4.14 Cấu trúc Bảng Assemblies trên MongoDB	82
Hình 5.1 Giao diện trang chủ khi sinh viên chưa đăng nhập	95
Hình 5.2 Giao diện đăng nhập.....	96
Hình 5.3 Sơ đồ luồng đăng nhập.....	96
Hình 5.4 Giao diện Form mẫu đăng ký đề tài	97
Hình 5.5 Lược đồ mô tả trình tự đề xuất đề tài của sinh viên.....	98
Hình 5.6 Giao diện danh sách đề tài đang mở trong đợt đăng ký	99
Hình 5.7 Giao diện thông tin chi tiết đề tài	99
Hình 5.8 Lược đồ mô tả trình tự đề xuất đề tài của sinh viên.....	100
Hình 5.9: Giao diện hủy ghi danh đề tài đã được đăng ký.....	101
Hình 5.10: <i>Lược đồ mô tả trình tự hủy đề tài của sinh viên</i>	101
Hình 5.11: Giao diện cập nhật thông tin bổ sung đề tài.....	102
Hình 5.12: Lược đồ mô tả trình tự cập nhật bổ sung thông tin.....	103
Hình 5.13: Giao diện xem thông tin chi tiết đề tài đã đăng ký thực hiện	104
Hình 5.14:Lược đồ mô tả trình tự xem chi tiết đề tài đã đăng ký.....	104

DANH MỤC BẢNG

Bảng 3.1 Yêu cầu chức năng nghiệp vụ của giảng viên	30
Bảng 3.2 Yêu cầu chức năng nghiệp vụ của sinh viên	31
Bảng 3.3 Yêu cầu chức năng nghiệp vụ của giáo vụ	33
Bảng 3.4 Yêu cầu phi chức năng	34
Bảng 3.5 Mô tả Use case Sinh viên	35
Bảng 3.6 Mô tả Use Case Giảng Viên	36
Bảng 3.7 Mô tả Use Case Giáo vụ	38
Bảng 3.8 Use case Đăng nhập	40
Bảng 3.9 Use Case Xem thông tin cá nhân	41
Bảng 3.10 Use Case Cập nhập thông tin cá nhân	41
Bảng 3.11 Use Case Xem danh sách đề tài được phép đăng ký	42
Bảng 3.12 Use Case Ghi danh vào đề tài	43
Bảng 3.13 Use Case Hủy đề tài	43
Bảng 3.14 Use Case Đề xuất đề tài	44
Bảng 3.15 Use Case Xem chi tiết đề tài	45
Bảng 3.16 Use Case Cập nhập thông tin cho đề tài	46
Bảng 3.17 Use Case Thêm đề tài (Giảng viên)	47
Bảng 3.18 Use Case Sửa đề tài (Giảng viên)	47
Bảng 3.19 Use Case Xóa đề tài (Giảng viên)	48
Bảng 3.20 Use Case Gửi đề tài lên giáo vụ	49
Bảng 3.21 Use Case Xóa đề tài (Quản lý đề tài - Giảng viên)	49
Bảng 3.22 Use Case Yêu cầu cập nhật bảng điểm	50
Bảng 3.23 Use Case Chọn nhóm sinh viên thực hiện đề tài	51
Bảng 3.24 Use Case Chấm điểm đề tài hướng dẫn	51
Bảng 3.25 Use Case Chấm điểm đề tài phản biện	52
Bảng 3.26 Use Case Hội đồng chấm điểm đề tài	53
Bảng 3.27 Use Case Xem thông tin nhóm	54
Bảng 3.28 Use Case Xem lịch sử đề tài	54
Bảng 3.29 Use Case Xem thông tin chi tiết đề tài (Giáo vụ)	55
Bảng 3.30 Use Case Xác nhận đề tài thực hiện	55
Bảng 3.31 Use Case Phân công GVPB	56
Bảng 3.32 Use Case Tải xuống danh sách đề tài	57
Bảng 3.33 Use Case Xem danh sách sinh viên thực hiện	57
Bảng 3.34 Use case Xem điểm chi tiết	58
Bảng 3.35 Use case Xóa người dùng	58
Bảng 3.36 Use case Thêm mới người dùng	59
Bảng 3.37 Use case Thêm danh sách người dùng bằng file Excel	60
Bảng 3.38 Use case Thêm mới học kì	60
Bảng 3.39 Use case Xóa học kì	61
Bảng 3.40 Use case Cập nhập học kì	62
Bảng 3.41 Use case Thêm mới đợt đăng ký	62
Bảng 3.42 Use case Xóa đợt đăng ký	63

Bảng 3.43 Use case Cập nhật đợt đăng ký	63
Bảng 3.44 Use case Thêm mới phiếu đánh giá.....	64
Bảng 3.45 Use Case Xóa phiếu đánh giá	64
Bảng 3.46 Use case Thêm mới tiêu chí.....	65
Bảng 3.47 Use case Cập nhật tiêu chí.....	66
Bảng 3.48 Use case Thêm mới hội đồng	66
Bảng 3.49 Use Case Xóa hội đồng	67
Bảng 3.50 Use case Gán đề tài vào hội đồng	68
Bảng 4.1 Thông tin các trường trong bảng Users	69
Bảng 4.2 Thông tin các trường trong bảng Topics	69
Bảng 4.3 Thông tin các trường trong bảng UserNotifications	70
Bảng 4.4 Thông tin các trường trong bảng TopicCategory	71
Bảng 4.5 Thông tin các trường trong bảng Semesters	71
Bảng 4.6 Thông tin các trường trong bảng ScoreBoards	71
Bảng 4.7 Thông tin các trường trong bảng RubricEvaluations	72
Bảng 4.8 Thông tin các trường trong bảng Rubrics	72
Bảng 4.9 Thông tin các trường trong bảng RegistrationPeriods	72
Bảng 4.10 Thông tin các trường trong bảng Majors	73
Bảng 4.11 Thông tin các trường trong bảng GroupStudent	73
Bảng 4.12 Thông tin các trường trong bảng Faculties	74
Bảng 4.13 Thông tin các trường trong bảng Assemblies	74
Bảng 6.5.1 Kiểm thử chức năng Đăng ký, Đăng nhập	97
Bảng 6.5.2 Kiểm thử chức năng Cập nhật profile người dùng	101
Bảng 6.5.3 Kiểm thử chức năng Xem danh sách đề tài NCKH, sinh viên, giảng viên, chi tiết đề tài NCKH	103
Bảng 6.5.4 Kiểm thử chức năng Quản lý đề tài NCKH	106
Bảng 6.5.5 Kiểm thử chức năng Quản lý phiếu đánh giá	110
Bảng 6.5.6 Kiểm thử chức năng Quản lý hội đồng	113
Bảng 6.5.7 Kiểm thử chức năng Chấm điểm	116

DANH MỤC CÁC CHỮ VIẾT TẮT

Từ/Cụm từ đầy đủ	Viết tắt	Ý nghĩa/Giải thích ngắn gọn
Nghiên cứu Khoa học	NCKH	Hoạt động tìm tòi, phát hiện, giải quyết vấn đề khoa học của sinh viên dưới sự hướng dẫn của giảng viên.
Công nghệ Thông tin	CNTT	Ngành học và lĩnh vực liên quan đến việc sử dụng máy tính và phần mềm để xử lý, lưu trữ, và truyền tải thông tin.
Thành phố Hồ Chí Minh	TP. HCM	Đơn vị hành chính cấp tỉnh của Việt Nam, nơi trường Đại học Sài Gòn tọa lạc.
Sinh viên	SV	Người đang theo học tại các trường đại học, cao đẳng.
Giảng viên	GV	Người làm công tác giảng dạy và nghiên cứu tại các cơ sở giáo dục đại học.
Giảng viên Hướng dẫn	GVHD	Giảng viên trực tiếp hướng dẫn sinh viên trong quá trình thực hiện đề tài nghiên cứu khoa học.
Giảng viên Phản biện	GVPB	Giảng viên được phân công đọc, nhận xét và đánh giá chất lượng đề tài nghiên cứu khoa học của sinh viên.
Đề tài	ĐT	Một vấn đề khoa học cụ thể được sinh viên lựa chọn để nghiên cứu và giải quyết.
Phiếu đánh giá	PDG	Biểu mẫu chứa các tiêu chí và thang điểm dùng để đánh giá chất lượng của một đề tài nghiên cứu khoa học.
Hội đồng	HD	Nhóm các giảng viên được thành lập để đánh giá, nghiệm thu các đề tài nghiên cứu.

		cứu khoa học.
Cơ sở dữ liệu	CSDL	Tập hợp dữ liệu có tổ chức, được lưu trữ và truy cập bằng hệ thống máy tính.
Giao diện Lập trình Ứng dụng	API	Tập hợp các quy tắc và giao thức cho phép các thành phần phần mềm khác nhau tương tác với nhau.
Giao thức Truyền siêu văn bản	HTTP	Giao thức nền tảng cho việc truyền tải dữ liệu trên World Wide Web.
JavaScript Object Notation	JSON	Một định dạng trao đổi dữ liệu nhẹ, dễ đọc và dễ viết cho con người, dễ dàng cho máy tính phân tích và tạo ra.
Asynchronous JavaScript and XML	AJAX	Kỹ thuật phát triển web cho phép tạo các ứng dụng web tương tác bằng cách trao đổi lượng nhỏ dữ liệu với máy chủ ở chế độ nền.
JSON Web Token	JWT	Một tiêu chuẩn mở (RFC 7519) định nghĩa một cách nhỏ gọn và tự chứa để truyền thông tin an toàn giữa các bên dưới dạng đối tượng JSON.
Ngôn ngữ Mô hình hóa Thống nhất	UML	Một ngôn ngữ trực quan tiêu chuẩn được sử dụng để chỉ định, trực quan hóa, xây dựng và tài liệu hóa các tạo phẩm của một hệ thống phần mềm.
Not only SQL (Thường được hiểu là Cơ sở dữ liệu phi quan hệ)	NoSQL	Một loại cơ sở dữ liệu cung cấp cơ chế lưu trữ và truy xuất dữ liệu khác với các mô hình quan hệ được sử dụng trong các CSDL truyền thống.
Node Package Manager	NPM	Trình quản lý gói mặc định cho Node.js, giúp các nhà phát triển JavaScript dễ

		dành chia sẻ và tái sử dụng mã nguồn.
Sơ đồ Quan hệ Thực thể	ERD	Một loại sơ đồ khái niệm mô tả cấu trúc logic của một cơ sở dữ liệu.
Trường hợp kiểm thử	TC	Một tập hợp các điều kiện hoặc biến số mà theo đó người kiểm thử sẽ xác định xem một hệ thống có hoạt động đúng như mong đợi hay không.
Kiểm thử Chấp nhận Người dùng	UAT	Giai đoạn cuối cùng của quá trình kiểm thử phần mềm, nơi người dùng thực tế kiểm tra phần mềm để đảm bảo nó đáp ứng nhu cầu của họ.

CHƯƠNG 1 : TỔNG QUAN

1.1. Đặt vấn đề và tính cấp thiết

Trong bối cảnh sự phát triển mạnh mẽ của nền công nghiệp CNTT, việc Xây dựng trang web quản lý hoạt động nghiên cứu khoa học của sinh viên không chỉ giúp tối ưu hóa quy trình quản lý hoạt động này mà còn mang lại nhiều lợi ích lớn cho cả sinh viên và giảng viên.

Một trong những ưu điểm lớn nhất của việc triển khai một hệ thống quản lý hoạt động nghiên cứu khoa học (NCKH) trực tuyến là khả năng tiếp cận thông tin mọi nơi, mọi lúc. Sinh viên có thể dễ dàng tra cứu và cập nhật thông tin về hoạt động NCKH của mình, tiếp cận tài liệu liên quan và theo dõi tiến độ công việc một cách thuận lợi. Đồng thời, giảng viên cũng có thể quản lý và đánh giá hoạt động NCKH của sinh viên một cách hiệu quả hơn, giảm thiểu thời gian và công sức cho việc theo dõi và báo cáo.

Ngoài ra, hệ thống quản lý hoạt động NCKH còn giúp tăng cường tính minh bạch và công bằng trong quá trình đánh giá. Việc có một cơ sở dữ liệu trực tuyến giúp đảm bảo rằng tất cả các thông tin liên quan đều được lưu trữ một cách an toàn và minh bạch, từ quá trình giao nhiệm vụ, tiến độ thực hiện đến kết quả đánh giá. Điều này tạo ra một môi trường công bằng cho sinh viên, nâng cao chất lượng của quá trình học tập và nghiên cứu.

Cuối cùng, việc triển khai website quản lý hoạt động NCKH cũng góp phần nâng cao uy tín và tiếng tăm của Khoa CNTT. Khả năng áp dụng công nghệ vào quá trình quản lý và giáo dục không chỉ thể hiện sự tiến bộ mà còn làm tăng giá trị thương hiệu của khoa, thu hút sự quan tâm từ cả sinh viên và doanh nghiệp. Điều này đồng nghĩa với việc tạo ra một môi trường học tập chất lượng và làm tăng cơ hội nghề nghiệp cho sinh viên Khoa CNTT trong tương lai.

Do đó, cần thiết phải có một hệ thống quản lý nghiên cứu khoa học tự động, tích hợp các tính năng quan trọng như tìm kiếm, phản hồi, và báo cáo.

1.2. Mục tiêu của đề tài

Mục tiêu chính của đề tài là xây dựng một website quản lý nghiên cứu khoa học cho sinh viên và giảng viên, với các mục tiêu cụ thể như sau:

1. Tự động hóa quy trình: Giảm bớt thời gian xử lý thông tin.

- Hiện trạng (thủ công): Quy trình NCKH thường liên quan đến nhiều giấy tờ, email, và các bước thủ công như:
 - Sinh viên nộp đề cương/báo cáo bằng bản cứng hoặc email.
 - Giảng viên/Bộ môn phải tổng hợp, phân loại, chuyển tiếp các bản nộp này.
 - Theo dõi tiến độ bằng file Excel hoặc sổ sách.
 - Thông báo kết quả, lịch bảo vệ qua nhiều kênh (bảng tin, email, điện thoại).
 - Lưu trữ tài liệu phân tán, khó tìm kiếm.
- Website tự động hóa như thế nào:
 - Nộp đề tài/báo cáo trực tuyến: Sinh viên submit file trực tiếp lên hệ thống qua form chuẩn hóa.
 - Luồng phê duyệt tự động: Hệ thống tự động chuyển đề cương/báo cáo đến giảng viên hướng dẫn hoặc hội đồng để xem xét, nhận xét, phê duyệt/từ chối ngay trên web.
 - Theo dõi tiến độ trực quan: Cả sinh viên và giảng viên đều thấy được trạng thái hiện tại của đề tài (ví dụ: Mới nộp, Đang xem xét, Cần chỉnh sửa, Đã duyệt, Đang thực hiện, Đã báo cáo...).
 - Thông báo tự động: Hệ thống gửi email/thông báo trên web cho các bên liên quan khi có cập nhật mới (ví dụ: có nhận xét mới, sắp đến

hạn nộp, đề tài được duyệt...).

- Quản lý và lưu trữ tập trung: Mọi tài liệu, nhận xét, kết quả đánh giá được lưu trữ gắn liền với từng đề tài trên hệ thống, dễ dàng truy xuất.

2. Minh bạch thông tin: Dễ dàng theo dõi và đánh giá tiến độ nghiên cứu.

- Hiện trạng (thiếu minh bạch):
 - Sinh viên khó biết chính xác đề tài của mình đang ở giai đoạn nào, ai đang xử lý, khi nào có kết quả.
 - Tiêu chí đánh giá, nhận xét đôi khi không được ghi lại rõ ràng, khó đối chiếu.
 - Thông tin về các đề tài khác, các hướng nghiên cứu trong khoa khó tiếp cận.
- Website minh bạch hóa như thế nào:
 - Dashboard cá nhân: Sinh viên và giảng viên có trang tổng quan riêng, hiển thị rõ ràng trạng thái các đề tài mình tham gia/hướng dẫn, các hạn nộp sắp tới, thông báo mới.
 - Lịch sử hoạt động: Mọi thao tác quan trọng (nộp bài, nhận xét, duyệt, đánh giá) đều được ghi lại kèm thời gian và người thực hiện, giúp dễ dàng xem lại quá trình.
 - Truy cập thông tin tập trung: Các quy định, biểu mẫu, tiêu chí đánh giá, danh sách đề tài được duyệt/đang thực hiện (trong phạm vi cho phép) được công bố rõ ràng trên website.
 - Phản hồi rõ ràng: Nhận xét, góp ý của giảng viên được ghi lại trực tiếp trên hệ thống, sinh viên dễ dàng xem và phản hồi nếu cần.

3. Tăng tính tương tác: Tạo môi trường trao đổi giữa sinh viên và giảng viên.

- Hiện trạng (tương tác hạn chế): Việc trao đổi chủ yếu qua email, gặp mặt trực tiếp, đôi khi thông tin bị trôi hoặc khó sắp xếp thời gian.
- Website tăng tương tác như thế nào:

- Hệ thống bình luận/nhắn tin: Cho phép giảng viên và sinh viên trao đổi trực tiếp ngay trên giao diện của đề tài cụ thể (ví dụ: giảng viên comment vào báo cáo tiến độ, sinh viên đặt câu hỏi về nhận xét đó).
- Diễn đàn/Kênh trao đổi chung: Có thể tạo các chủ đề thảo luận chung về phương pháp nghiên cứu, các vấn đề kỹ thuật, hoặc thông báo chung liên quan đến NCKH.
- Thông báo đẩy (Notifications): Hệ thống thông báo khi có tin nhắn mới, bình luận mới, giúp các bên phản hồi nhanh chóng hơn.
- Chia sẻ tài liệu: Giảng viên dễ dàng đính kèm tài liệu tham khảo, hướng dẫn cho sinh viên ngay trên hệ thống.

4. Hỗ trợ tìm kiếm: Giúp sinh viên nhanh chóng tìm kiếm đề tài liên quan.

- Hiện trạng (khó tìm kiếm): Sinh viên muốn tìm hiểu các đề tài đã/đang làm trong khoa thường phải hỏi dò, tìm trên các nguồn rời rạc, mất thời gian.
- Website hỗ trợ tìm kiếm như thế nào:
 - Cơ sở dữ liệu đề tài: Lưu trữ thông tin các đề tài (tên đề tài, sinh viên thực hiện, giảng viên hướng dẫn, tóm tắt, từ khóa, lĩnh vực, trạng thái, file báo cáo cuối cùng - nếu được phép công khai).
 - Bộ lọc và tìm kiếm nâng cao: Cung cấp công cụ tìm kiếm theo từ khóa, tên giảng viên, lĩnh vực nghiên cứu, năm học...; bộ lọc để thu hẹp kết quả.
 - Gợi ý đề tài: Có thể có mục giới thiệu các hướng nghiên cứu, các đề tài đang cần sinh viên thực hiện do giảng viên đề xuất.

1.3. Đối tượng và phạm vi nghiên cứu

- **Đối tượng nghiên cứu:**
 - **Sinh viên:** Là người sử dụng hệ thống với mục đích đăng ký đề tài mới để hoặc đề tài có sẵn, quản lý đề tài đang thực hiện, nộp báo cáo, thông tin cần thiết
 - **Giảng viên:** Là người sử dụng hệ thống, với mục đích đăng kí các đề tài mới, lựa chọn nhóm sinh viên thực hiện đề tài, quản lý các đề tài mà mình đang hướng dẫn và phân biện, tham gia hội đồng bảo vệ đề tài, chấm điểm các đề tài

- **Giáo vụ :** Là người quản trị của hệ thống , với mục đích mở các đợt đăng ký , quản lý các phiếu chấm điểm , các hội đồng , người dùng của hệ thống ,tải xuống thông tin các đề tài , bảng điểm của sinh viên
- **Phạm vi nghiên cứu:**
 - **Phạm vi Công nghệ:**
 - Xây dựng hệ thống dưới dạng ứng dụng web (Web Application).
 - Sử dụng ReactJS làm thư viện/framework chính cho phần giao diện người dùng (Front-end).
 - Sử dụng NodeJS (cụ thể là framework như ExpressJS) để xây dựng phần xử lý logic phía máy chủ (Back-end) và API.
 - Sử dụng hệ quản trị cơ sở dữ liệu MongoDB để lưu trữ dữ liệu.
 - **Phạm vi chức năng:**
 - **Quản lý NCKH cấp sinh viên:** Tập trung vào quy trình NCKH dành cho sinh viên, bao gồm:
 - Đăng ký/đề xuất đề tài NCKH.
 - Phân công hoặc sinh viên chọn giảng viên hướng dẫn.
 - Nộp và quản lý các phiên bản đề cương, báo cáo tiến độ, báo cáo tổng kết.
 - Giảng viên thực hiện nhận xét, góp ý, và cho điểm đánh giá trực tuyến.
 - Theo dõi trạng thái và tiến độ thực hiện của đề tài.
 - Hỗ trợ quản lý nghiên cứu theo **cá nhân** và theo **nhóm sinh viên**.
 - **Quản lý thông tin:**
 - Quản lý cơ sở dữ liệu các đề tài NCKH của sinh viên trong Khoa (tên đề tài, mô tả, sinh viên thực hiện, GVHD, trạng thái, tài liệu đính kèm...).
 - Quản lý thông tin người dùng cơ bản (sinh viên, giảng viên, admin) liên quan đến hoạt động NCKH.
 - **Tương tác và Tìm kiếm:**
 - Cung cấp chức năng tìm kiếm, lọc đề tài dựa trên các tiêu chí cơ bản (tên, từ khóa, giảng viên, năm học...).
 - Hỗ trợ các tính năng tương tác cơ bản như thông báo, bình luận (nếu có) giữa sinh viên và giảng viên liên quan đến đề tài cụ thể.
 -

- **Phạm vi Kiểm thử và Triển khai:**
 - Thực hiện kiểm thử chức năng (functional testing), kiểm thử tích hợp (integration testing) trong môi trường phát triển.
 - Tiến hành **triển khai thử nghiệm** hệ thống trên một máy chủ (có thể là máy chủ nội bộ của Khoa hoặc dịch vụ đám mây) để đánh giá trong **môi trường gần với thực tế** với sự tham gia của một nhóm giảng viên và sinh viên thuộc Khoa CNTT. Thu thập phản hồi để đánh giá và cải thiện.

1.4. Ý nghĩa của đề tài

- **Ý nghĩa thực tiễn:**
 - Nâng cao hiệu quả trong quản lý nghiên cứu khoa học.
 - Giảm bớt gánh nặng hành chính cho giảng viên.
 - Tăng tính tự giác nghiên cứu cho sinh viên.
- **Ý nghĩa khoa học:**
 - Đóng góp tài liệu tham khảo về cách quản lý nghiên cứu hiệu quả.

1.5. Phương pháp nghiên cứu

- **Phương pháp thu thập dữ liệu:**
 - **Phỏng vấn:** Thực hiện phỏng vấn trực tiếp giảng viên và sinh viên để thu thập thông tin về các khó khăn trong việc quản lý và thực hiện đề tài nghiên cứu khoa học.
 - **Khảo sát:** Xây dựng bảng câu hỏi khảo sát trực tuyến và giấy để lấy ý kiến từ đối tượng người dùng về nhu cầu và đề xuất tính năng cần thiết.
 - **Nghiên cứu hệ thống tương tự:** Phân tích các hệ thống quản lý nghiên cứu khoa học hiện có để học hỏi và áp dụng những tính năng hiệu quả.
- **Phương pháp phân tích hệ thống:**

- **Phân tích yêu cầu:** Thu thập và phân loại yêu cầu từ người dùng cuối, bao gồm cả giảng viên và sinh viên.
- **Sử dụng UML:** Xây dựng các biểu đồ Use Case, Sequence, và Class để mô tả và trực quan hóa hệ thống.
- **Mô hình MVC:** Áp dụng mô hình Model-View-Controller để tách biệt phần xử lý dữ liệu, giao diện và điều khiển.
- **Phương pháp phát triển phần mềm:**
 - **Phương pháp Agile:** Áp dụng phương pháp phát triển phần mềm linh hoạt, cho phép thay đổi và cải tiến trong quá trình phát triển.
 - **Framework Scrum:** Chia nhỏ công việc thành các sprint ngắn (2-4 tuần) để dễ dàng quản lý tiến độ và điều chỉnh khi cần thiết.
 - **Kiểm thử phần mềm:** Thực hiện kiểm thử đơn vị (Unit Test) và kiểm thử tích hợp (Integration Test) để đảm bảo chất lượng và độ tin cậy của hệ thống.

1.6. Cấu trúc báo cáo

Báo cáo bao gồm 6 chương:

- **Chương 1:** Tổng Quan
- **Chương 2:** Cơ sở lý thuyết và công nghệ.
- **Chương 3:** Phân tích và thiết kế hệ thống.
- **Chương 4:** Thiết kế phần mềm.
- **Chương 5:** Tổng quan hệ thống.
- **Chương 6:** Kiểm thử hệ thống.
- **Chương 7:** Kết luận và hướng phát triển

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

2.1. Tổng quan về hệ thống

- **Các thành phần cơ bản của hệ thống:**
 - **Frontend:** Giao diện người dùng bao gồm các trang đăng nhập, quản lý đề tài, thống kê, thông báo... giúp người dùng tương tác và thực hiện các chức năng cần thiết.
 - **Backend:** Là nơi xử lý các logic nghiệp vụ như xác thực người dùng, xử lý dữ liệu đề tài, lưu trữ và truy xuất thông tin, đảm bảo tính toàn vẹn dữ liệu.
 - **Cơ sở dữ liệu:** Lưu trữ tất cả thông tin liên quan đến người dùng, đề tài nghiên cứu, tiến độ thực hiện và báo cáo.
- **Quy trình cơ bản của các chức năng:**
 - **Đăng nhập và xác thực người dùng:**
 - Người dùng nhập thông tin tài khoản.
 - Hệ thống xác thực thông tin và phân quyền truy cập (sinh viên, giảng viên, quản trị viên).
 - **Quản lý đề tài:**
 - Sinh viên đăng ký đề tài thông qua biểu mẫu.
 - Giảng viên hoặc hội đồng xét duyệt, phản hồi và phê duyệt.
 - Theo dõi tiến độ, cập nhật tình trạng và nộp báo cáo.
 - **Thông báo và báo cáo:**
 - Hệ thống tự động gửi thông báo về hạn nộp báo cáo, cập nhật tình trạng.
 - Cho phép xuất báo cáo tổng hợp và chi tiết.
- **Các vấn đề thường gặp:**
 - Quản lý giấy tờ thủ công dễ gây thất lạc, khó theo dõi tiến độ.
 - Thiếu minh bạch trong việc xét duyệt đề tài.

- Khó khăn trong việc truy xuất dữ liệu và tổng hợp báo cáo.
- **Các giải pháp hiện có và hạn chế:**
 - Một số hệ thống hiện tại giải quyết được lưu trữ và tra cứu nhưng thiếu tính tương tác và khả năng theo dõi tiến độ theo thời gian thực.
 - Hệ thống cần cải thiện giao diện người dùng và hỗ trợ tốt trên đa nền tảng.

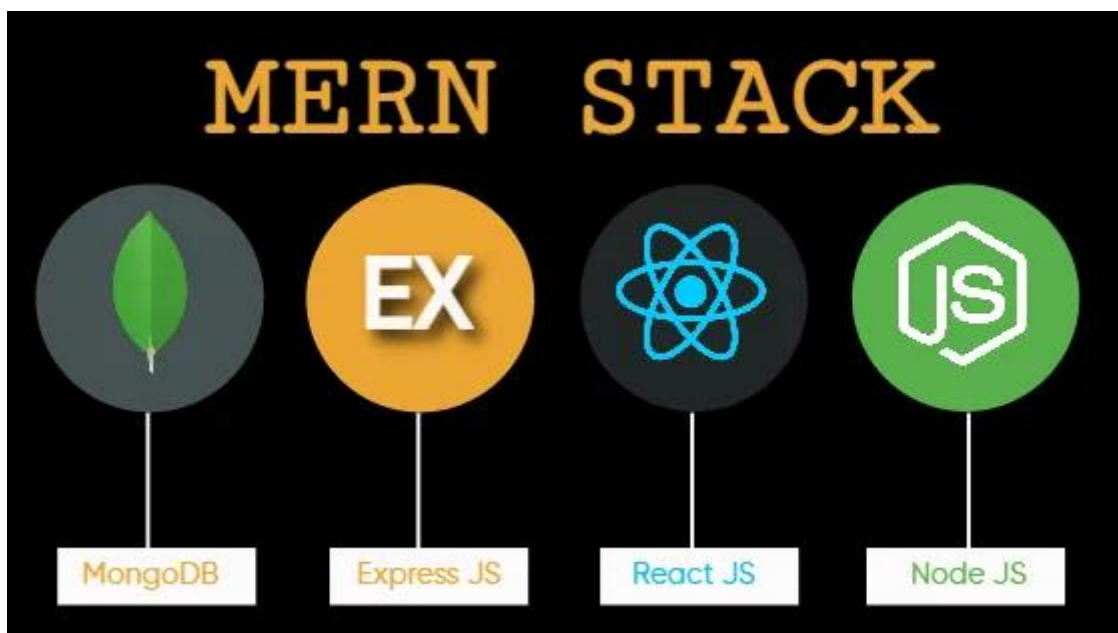
2.2. Kiến trúc hệ thống

2.2.1. Khái niệm về MERN Stack

MERN được viết tắt cho MongoDB, Express, React, Node, và bốn từ khóa công nghệ đã tạo nên Stack này:

- ✓ MongoDB: Cơ sở dữ liệu dạng tài liệu (Document Database).
- ✓ ExpressJS/NestJS: Framework cho website viết bằng Node.js.
- ✓ ReactJS: Thư viện JavaScript phía client (Client-side).
- ✓ NodeJS: Máy chủ web viết bằng JavaScript.

MERN Stack được dùng để xây dựng ứng dụng React Universal App.



Hình 2.1 Giải thích MERN Stack

2.2.2. Các thành phần trong MERN Stack

MERN đại diện cho MongoDB (M), ExpressJS (E) hoặc NestJS (N), React (R) và NodeJS (N).

✓ React:

- Được biết đến như React.js hoặc ReactJS, là một thư viện JavaScript mã nguồn mở dành cho front-end để xây dựng giao diện người dùng trên các thành phần UI (User Interface).
- Được phát triển bởi Meta (Facebook), React là một trong những công cụ phổ biến trong giới lập trình web, đặc biệt cho các ứng dụng Single-page và Mobile.
- React cũng hỗ trợ server-side rendering với framework như Next.js.

✓ MongoDB:

- Là một hệ quản trị cơ sở dữ liệu NoSQL, đặc điểm của nó là không có ràng buộc quan hệ, phân tán, mã nguồn mở.
- MongoDB có khả năng lưu trữ lượng dữ liệu lớn một cách hiệu quả.
- Trong MERN stack, MongoDB đóng vai trò là nơi lưu trữ dữ liệu, kết nối với NodeJS và ExpressJS để xử lý thông tin.

✓ NodeJS:

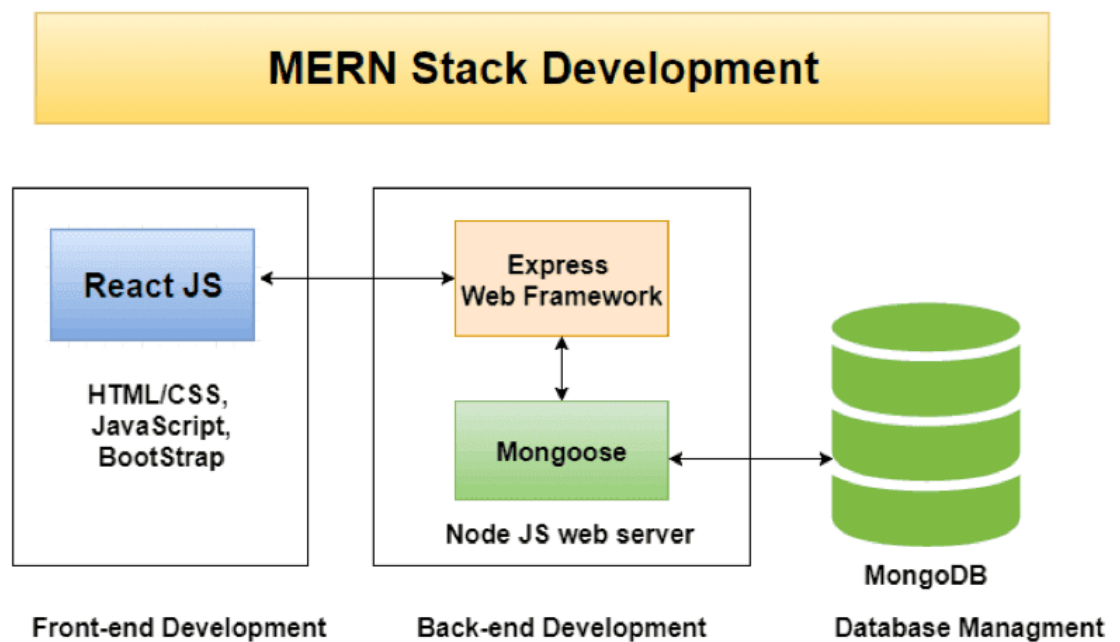
- Là môi trường chạy JavaScript phía server, không cần trình duyệt.
- NodeJS có nhiệm vụ xử lý các Request và phản hồi Response từ phía client.
- Trong mô hình MERN stack, khi có yêu cầu (Request) từ React ở phía client, NodeJS xử lý và phản hồi kết quả.

- NodeJS cũng đóng vai trò quan trọng trong việc tính toán logic, xác thực người dùng, phân quyền và thao tác với tập tin, cơ sở dữ liệu MongoDB.

✓ ExpressJS là một framework của NodeJS, nó cung cấp thêm những chức năng giúp cho quá trình làm việc bằng NodeJS được dễ dàng hơn. Cụ thể, nó giúp đơn giản hóa việc xử lý các Request đến từ nhiều địa chỉ khác nhau từ phía người dùng. Nói chung, ExpressJS giúp cho công việc xử lý ở phía backend trở nên dễ dàng và nhanh chóng hơn.

✓ NestJS là một framework của NodeJS cho phép xây dựng ứng dụng phía server. Nest mở rộng các framework NodeJS như Express hay Fastify để bổ sung thêm nhiều module hay thư viện hỗ trợ xử lý tác vụ. Đây là một framework mã nguồn mở, sử dụng TypeScript và rất linh hoạt để xây dựng các hệ thống backend.

2.2.3.Cách MERN Stack hoạt động



Hình 2.2.Sơ đồ hoạt động MERN Stack

Tại mỗi client, React thực hiện xử lý và tạo ra giao diện người dùng. React cũng thực thi single-page-application trên mỗi client.

Tại server, NodeJS và ExpressJS có nhiệm vụ xử lý logic, nghiệp vụ và xác

thực. MongoDB ở server có nhiệm vụ tổ chức và lưu trữ dữ liệu.

Client và server sẽ giao tiếp bằng cách gửi các Request và Response, đây là các Request và Response gửi ngầm bằng công nghệ AJAX giúp người dùng không cần reload lại trang web. Trong các Request và Response này, dữ liệu được tổ chức thành dạng JSON – là một dạng dữ liệu được tổ chức ngắn gọn và có dễ đọc hiểu.

2.2.4. Lợi thế của MERN Stack

MERN Stack sử dụng một ngôn ngữ chung là JavaScript cho cả client-side và server-side, giúp cho quá trình phát triển trang web trở nên dễ tiếp cận.

NodeJS có khả năng chạy trên cả Linux, Windows và OS X nên nó không phụ thuộc vào hệ điều hành. Cơ chế non-blocking I/O giúp server có thể đáp ứng hàng ngàn connection đồng thời.

MERN Stack sử dụng JSON là kiểu dữ liệu trao đổi giữa tất cả các thành phần nên không cần thêm thư viện để chuyển đổi dữ liệu trong quá trình tương tác giữa phía client và phía server.

MongoDB có khả năng quản lý một lượng dữ liệu lớn, có thể thêm thuộc tính mới vào các bảng mà không cần cập nhật lại toàn bộ bảng.

Các công nghệ sử dụng trong MERN Stack đều là mã nguồn mở và miễn phí. Cộng đồng sử dụng rộng lớn thuận tiện cho việc học hỏi kinh nghiệm hay giải đáp thắc mắc.

2.3. Công nghệ Back-End

2.3.1. Khái niệm NodeJS

Node.js là một môi trường thực thi JavaScript dựa trên Chrome V8 Engine, được xây dựng để xử lý các tác vụ I/O không đồng bộ. Điều này cho phép Node.js xử lý hàng ngàn kết nối đồng thời, làm cho nó lựa chọn lý tưởng cho các ứng dụng web thời gian thực.

2.3.2. Đặc điểm của NodeJS

Node.js được biết đến với một số đặc điểm quan trọng làm nổi bật nó trong phát triển ứng dụng web. Dưới đây là những đặc điểm chính của Node.js:

✓ **Bất đồng bộ (Asynchronous):**

Giải Quyết Vấn Đề IO Blocking: Node.js sử dụng mô hình bất đồng bộ, giúp tránh tình trạng chờ đợi khi thực hiện các thao tác đầu vào/ra (I/O). Điều này tối ưu hóa sự sử dụng tài nguyên và đảm bảo hiệu suất cao trong khi xử lý nhiều kết nối đồng thời.

✓ **Kiến trúc sự kiện (Event-Driven):**

Xử Lý Sự Kiện Đồng Thời: Node.js dựa trên mô hình kiến trúc sự kiện, cho phép xử lý nhiều sự kiện cùng một lúc mà không chờ đợi hoàn tất. Điều này giúp ứng dụng thực hiện các tác vụ đồng thời và linh hoạt.

✓ **JavaScript Everywhere:**

Chia sẻ mã nguồn giữa Máy Chủ và Máy Khách: Sự đồng nhất của ngôn ngữ JavaScript giữa máy chủ và máy khách giúp đơn giản hóa quá trình phát triển và quản lý mã nguồn.

✓ **V8 JavaScript Engine:**

Hiệu suất cao: Node.js sử dụng Chrome V8 JavaScript Engine, giúp thực thi mã nhanh chóng và có hiệu suất cao, đặc biệt là trong việc xử lý mã JavaScript.

✓ **Modules và NPM:**

Quản lý mã nguồn và thư viện: Hỗ trợ khái niệm module, cho phép chia nhỏ mã nguồn thành các phần nhỏ hơn để dễ quản lý. NPM (Node Package Manager) cung cấp một cách hiệu quả để quản lý và chia sẻ thư viện.

✓ **Xử lý đa nhiệm và Luồng sự kiện:**

Đa nhiệm và Luồng sự kiện: Node.js có khả năng xử lý đa nhiệm thông qua sự kiện và các luồng sự kiện, giúp ứng dụng quản lý nhiều kết nối đồng thời mà không làm giảm hiệu suất.

✓ **Môi trường thực thi đa nền tảng:**

Đa Nền Tảng: Node.js hỗ trợ trên nhiều nền tảng như Windows, macOS, và Linux, làm cho nó trở thành một lựa chọn linh hoạt cho các dự án đa nền tảng.

✓ **Ứng dụng Real-time:**

Truyền Thông Real-time: Node.js thích hợp cho các ứng dụng đòi hỏi truyền thông real-time như chat, video call, và cập nhật dữ liệu đồng bộ ngay lập tức.

2.3.3. Khái niệm NestJS

NestJS là một framework Node.JS cho phép xây dựng ứng dụng phía server. Nest mở rộng các framework Node.js như Express hay Fastify để bổ sung thêm nhiều module hay thư viện hỗ trợ việc xử lý tác vụ. Đây là một framework mã nguồn mở, sử dụng TypeScript và rất linh hoạt để xây dựng các hệ thống backend. Là framework Node.js phát triển mạnh nhất trong 3 năm trở lại đây. Tương tự như framework Express JS.

NestJS là một NodeJS framework dùng để phát triển server-side applications hiệu quả và có thể mở rộng. NestJS là sự kết hợp bởi OOP(Object Oriented Programming), FP (Functional Programming), FRP (Functional Reactive Programming). NestJS sử dụng TypeScript để phát triển nhưng nó cũng hỗ trợ cả Javascript.

Mỗi dự án đều có một kiến trúc phù hợp khác nhau như MVC, kiến trúc thành phần hoặc các kiến trúc khác. Với một nhà phát triển mới làm việc với NodeJS thì họ sẽ không có tư duy về architecture, scalable hay maintainable thì trong quá trình phát triển có thể gây ra việc tạo ra kiến trúc code không chất lượng làm xáo trộn dự án. NestJS cung cấp cho chúng ta một kiến trúc cụ thể và rõ ràng để giải quyết vấn đề này.

2.3.4. Đặc điểm của Nest JS

NestJS là một framework Node.js mạnh mẽ, dựa trên TypeScript và thiết kế để xây dựng ứng dụng server-side hiện đại và dễ bảo trì. Dưới đây là những đặc

điểm chính của NestJS:

- ✓ **TypeScript và Decorators:** Kiểu dữ liệu mạnh mẽ: Sử dụng TypeScript giúp xác định kiểu dữ liệu một cách rõ ràng, giảm lỗi trong quá trình phát triển.
- ✓ **Decorators:** Sử dụng decorators để đánh dấu và mô tả metadata, hỗ trợ trong Dependency Injection và Routing.
- ✓ **Module và Dependency Injection:** Tổ chức mã nguồn hiệu quả: Sử dụng module để tổ chức ứng dụng thành các phần chức năng riêng biệt.
- ✓ **Dependency Injection (DI):** NestJS tích hợp DI giúp quản lý và tái sử dụng thành phần một cách dễ dàng.
- ✓ **Controller và Routing:** Quản lý Endpoint linh hoạt: Sử dụng controllers để định nghĩa các endpoint, giúp quản lý routing một cách hiệu quả.
- ✓ **Decorators cho Routing:** Sử dụng decorators để định nghĩa route một cách ngắn gọn và rõ ràng.
- ✓ **Middleware và Pipes:** Xử lý yêu cầu một cách linh hoạt: Middleware giúp xử lý các yêu cầu trước khi chúng đến tới controller, cung cấp khả năng mở rộng và tùy chỉnh.
- ✓ **Pipes cho xử lý dữ liệu:** Sử dụng pipes để kiểm soát và xử lý dữ liệu trước khi nó đến tới controller.
- ✓ **Provider và Service:** Quản lý thành phần dễ dàng: Provider là một cách NestJS gọi các thành phần như service, controller, middleware.
- ✓ **Service để thực hiện logic:** Service giúp tách biệt logic business và tạo ra các service có thể tái sử dụng.
- ✓ **ORM Integration:** Tích hợp dễ dàng với cơ sở dữ liệu: Hỗ trợ tích hợp với các thư viện ORM như TypeORM, giúp tương tác với cơ sở dữ liệu một cách thuận tiện.
- ✓ **Hệ thống Middleware mạnh mẽ:** Middleware có sẵn: Bao gồm nhiều middleware tích hợp sẵn như logger, compression, và nhiều middleware khác

giúp tối ưu hóa ứng dụng.

- ✓ **Hỗ trợ CLI (Command Line Interface):** CLI đa nhiệm: NestJS có một Command Line Interface mạnh mẽ giúp tạo mới và quản lý module, controller, service, và nhiều thành phần khác.

- ✓ **Cộng đồng phát triển mạnh mẽ:** Cộng đồng chủ động: Sự phát triển mạnh mẽ với cộng đồng tích cực, cung cấp sự hỗ trợ và tài nguyên phong phú.

2.4.Công Nghệ Front-End

2.4.1.Khái niệm ReactJS

Đầu tiên React không phải là Javascript Framework mà nó là một thư viện javascript được phát triển bởi Jordan Walke, nó giúp tạo ra giao diện người dùng. React được dùng lần đầu tiên trên Facebook's Newsfeed vào năm 2011 và sau đó là trên Instagram, Whatsapp vào năm 2012, sau đó phát hành vào năm 2013.

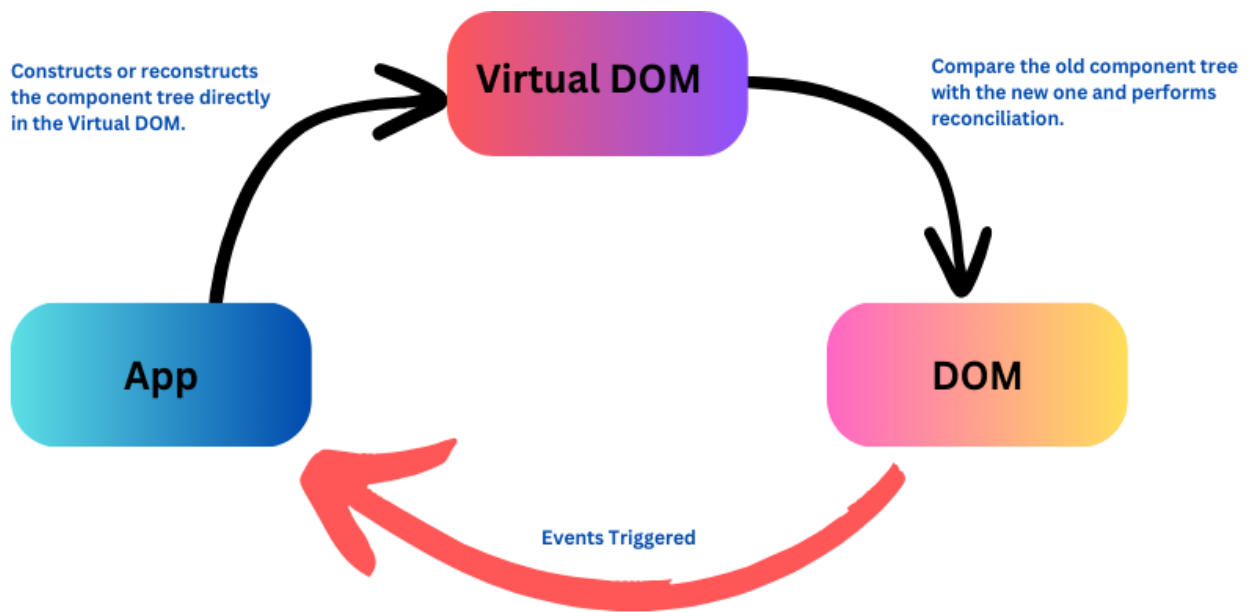
Trong React mọi thứ đều là thành phần và mỗi thành phần chịu trách nhiệm xuất một đoạn code HTML nhỏ và có thể tái sử dụng. Nó thường được sử dụng để xây dựng những thành phần có thể tái sử dụng và nó giảm thiểu quá trình render của DOM.

2.4.2.Tại sao lại chọn ReactJS

Hiện nay có nhiều Javascript Framework như là Angular, jQuery. Nhưng với những gì React làm được đã khiến nó trở nên phổ biến và là một trong những sự lựa chọn hàng đầu của các nhà phát triển giao diện. Sau đây hãy cùng tìm hiểu một số tính năng của React.

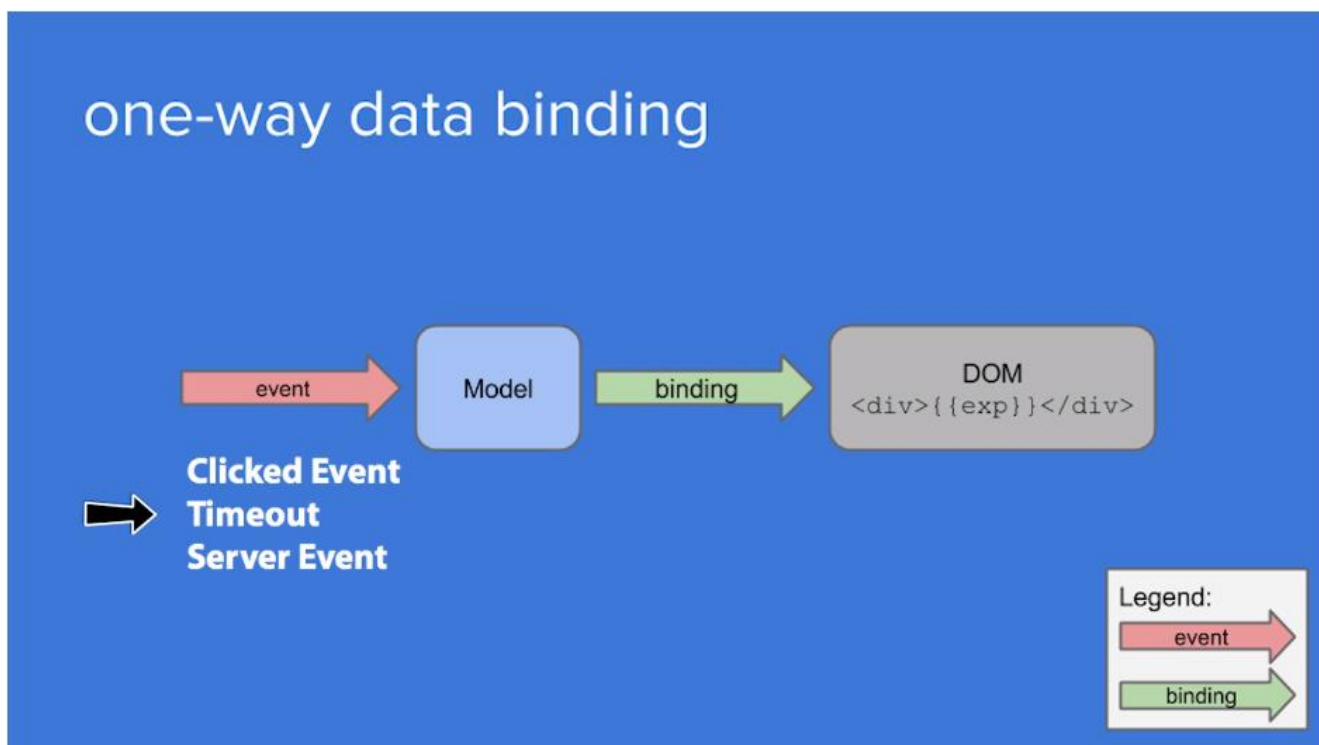
- ✓ **JSX:** được viết tắt của Javascript XML. Đó là cú pháp giống XML HTML nhưng được dùng bởi React. Nó nhanh hơn Javascript thông thường vì nó được tối ưu trong quá trình dịch sang Javascript thông thường.

- ✓ **Virtual DOM:** React sẽ chỉ cho trình duyệt render một lần và chỉ khi có sự thay đổi thì React mới cho trình duyệt render lại. Cách này giúp giảm thiểu quá trình render và làm cho nó nhanh hơn.



Hình 2.3: Virtual DOM

✓ **One-way data binding:** với phương pháp này, dòng dữ liệu chỉ trong một hướng nghĩa là view sẽ không tự động cập nhật khi dữ liệu của model được thay đổi. Tính năng này giúp bạn dễ dàng kiểm soát ứng dụng hơn.



Hình 2.4: One-Way Data binding

✓ Native Support : React cũng có phiên bản gốc được gọi là React Native. Nó cung cấp những gì tốt nhất thế giới cho nền tảng phát triển ứng dụng di động. React Native hỗ trợ xây dựng đồng thời là cả trên IOS và Android.

2.5.Cơ sở dữ liệu MongoDB

2.5.1.Khái niệm MongoDB

MongoDB hệ quản trị cơ sở dữ liệu mã nguồn mở (open-source) thuộc họ cơ sở dữ liệu phi quan hệ (NoSQL).

MongoDB được xây dựng theo hướng là một cơ sở dữ liệu hướng đối tượng vì vậy nó không có các ràng buộc toàn vẹn dữ liệu như ở các Cơ sở dữ liệu quan hệ mà cho phép lưu trữ dữ liệu linh hoạt (stores data in flexible) có dữ liệu khác nhau trong cùng một Collection (tương đương với Table ở SQL). Do chính đặc thù này mà MongoDB Có thể được sử dụng để lưu các dữ liệu đa dạng và phức tạp mà không bị hạn chế khi có thay đổi về cấu trúc lưu trữ dữ liệu.

MongoDB là một cơ sở dữ liệu phân tán vì vậy nó sẽ có những đặc trưng của cơ sở dữ liệu phân tán như: tính sẵn sàng cao, khả năng mở rộng theo chiều ngang,

Dữ liệu JSON được tạo ra từ ReactJS sau đó được gửi đến ExpressJS để xử lý và sau đó có thể lưu trữ trực tiếp đến MongoDB.

2.5.2.Đặc điểm của MongoDB

MongoDB là cơ sở dữ liệu No-SQL loại Document Database với nhiều yếu tố được sung để nâng cao hiệu suất sử dụng như Index, Schema,....

MongoDB được lập trình bằng ngôn ngữ C++ và hỗ trợ tốt với các Javascript Framework.

Các khái niệm tiêu biểu khi sử dụng MongoDB:

✓ **Document - tài liệu:** là đơn vị cơ bản của dữ liệu trong MongoDB, một document tương đương với một dòng trong bảng của SQL. Mỗi một document đều có cấu trúc là một đoạn chuỗi json độc lập.

✓ **Collection:** là tập hợp các document có các thuộc tính tương tự nhau.

Collection tương đương với một bảng (Table) trong SQL. Điểm khác nhau rõ rệt nhất ở đây chính là việc bảng trong SQL có cấu trúc được quy định sẵn và dữ liệu vào luôn phải tuân theo cấu trúc đó, còn Collection có thể tiếp nhận tự do mọi document có các thuộc tính khác nhau vào mà không hề phát sinh lỗi.

- ✓ **Database:** là các cơ sở dữ liệu độc lập lẫn nhau của MongoDB, một Database của MongoDB bao gồm một hay nhiều Collection và các cài đặt phân quyền kèm theo Database đó.

- ✓ **_id:** là một khóa định danh đặc biệt được quy định tự động bởi MongoDB khi thêm một Document vào Collection. _id của bất cứ Document nào đều luôn là duy nhất trong Collection chứa document đó.

- ✓ **Index - chỉ mục:** là một cấu trúc dữ liệu thu thập thông tin về các trường (field) trong Document của Collection từ đó mà nó giúp cho các câu truy vấn (query) đến các trường này tăng đáng kể tốc độ truy vấn. Tuy nhiên việc tạo thêm các Index cũng làm tăng đáng kể dung lượng bộ nhớ tiêu hao cho Database MongoDB. Trường _id là trường chỉ mục luôn luôn được khởi tạo mặc định bởi MongoDB khi thêm mới một Document

- ✓ **Query - truy vấn:** cũng giống như tất cả cơ sở dữ liệu khác, MongoDB hỗ trợ truy vấn dữ liệu và nhà phát triển của cơ sở dữ liệu này cũng đã cung cấp số lượng lớn các câu lệnh hỗ trợ giúp đơn giản hóa việc truy vấn dữ liệu hơn rất nhiều lần so với các ngôn ngữ truy vấn dữ liệu truyền thống mà vẫn đảm bảo được tốc độ truy vấn nhanh.

2.6.PostMan

Postman là một công cụ phổ biến được sử dụng để thử nghiệm và kiểm thử các API đặc biệt là các API theo yêu cầu cấu trúc REST. Dưới đây là các điểm chính:

- ✓ Mục đích chính: Postman được thiết kế để hỗ trợ lập trình viên và nhóm phát triển trong việc tương tác và kiểm thử các dịch vụ web và API.

- ✓ Đặc điểm nổi bật:

- Hỗ trợ phương thức HTTP: Hỗ trợ đầy đủ các phương thức HTTP như GET,
- POST, PUT, DELETE, PATCH.
- Quản lý dữ liệu linh hoạt Có khả năng lưu trữ dữ liệu và biến, giúp tạo các request động và linh hoạt.
- Hỗ trợ mọi loại dữ liệu: Có khả năng gửi và nhận dữ liệu ở nhiều định dạng khác nhau như JSON, XML, HTML.
- Quản lý môi trường: Cho phép quản lý nhiều môi trường khác nhau để dễ dàng chuyển đổi giữa chúng.
- Tiện ích cho lập trình viên: Postman cung cấp giao diện người dùng thân thiện và dễ sử dụng, đồng thời hỗ trợ trên nhiều nền tảng; Có thể chia sẻ bộ kiểm thử với đồng đội, tạo sự đồng bộ trong quá trình kiểm thử và phát triển.
- Real-time Testing: Cung cấp khả năng kiểm thử API real-time để theo dõi và xem kết quả trực tiếp từ các request.
- Hỗ trợ đa nền tảng: Postman hỗ trợ trên nhiều hệ điều hành như Windows, macOS, và Linux, cũng như trên các trình duyệt web.

2.7.JSON Web Token

2.7.1.JWT-Định nghĩa và ứng dụng trong bảo mật Client-Server

JSON Web Token (JWT) là một tiêu chuẩn mở được áp dụng để đảm bảo tính an toàn của dữ liệu truyền trên mô hình Client-Server. Thông tin trong JWT được biểu diễn dưới định dạng JSON và bảo mật bằng chữ ký điện tử, sử dụng cặp khóa riêng (Private Key) và khóa công khai (Public Key) với các thuật toán như HMAC.

2.7.2. Cấu trúc của JWT

JWT bao gồm ba thành phần chính:

- ✓ Header: Xác định loại mã token và thuật toán mã hóa được sử dụng.
- ✓ Payload: Chứa các thuộc tính xác định cần mã hóa, mang thông tin quan trọng.
- ✓ Signature: Chữ ký điện tử giúp xác minh tính toàn vẹn của thông tin, đảm bảo rằng dữ liệu không bị thay đổi trong quá trình truyền tin.

2.7.3. Ưu điểm của JWT

JWT mang lại nhiều lợi ích cho hệ thống sử dụng:

- ✓ Tăng cường Bảo mật: Chữ ký điện tử giúp xác minh tính toàn vẹn và nguồn gốc của dữ liệu, đảm bảo tính an toàn trong quá trình truyền tin.
- ✓ Phổ biến và Rộng rãi: JWT được sử dụng rộng rãi trong nhiều ứng dụng và hệ thống, tạo ra một tiêu chuẩn chung trong việc xác thực và truyền thông tin.
- ✓ Nhỏ gọn và Dễ xử lý: Định dạng JSON của JWT làm cho nó nhỏ gọn và dễ dàng xử lý trong cả quá trình tạo và xác minh token.

CHƯƠNG 3 : PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1. Phân tích yêu cầu hệ thống

3.1.1. Yêu cầu chức năng

3.1.1.1 Yêu cầu chức năng nghiệp vụ

✓ Giảng viên

STT	Yêu cầu	Loại YC	Mô tả/ Ràng buộc/ Công thức
1	Thêm đề tài	Lưu trữ	Thêm đề tài vào cơ sở dữ liệu.
2	Duyệt nhóm sinh viên thực hiện đề tài	Lưu trữ	Xác nhận nhóm sinh viên thực hiện đề tài được đăng ký.
3	Chấm điểm đề tài	Lưu trữ	Lưu danh sách điểm đề tài vào cơ sở dữ liệu.
4	Xem danh sách hội đồng tham gia	Kết xuất	Hiện tất cả các hội đồng có phân trang.
5	Xem danh sách người dùng	Kết xuất	Hiện tất cả các người dùng có phân trang.
6	Sửa đề tài chưa được duyệt	Lưu trữ	Chọn đề tài cần sửa, nhập các thông tin sửa chữa và lưu lại.
7	Sửa thông tin người dùng	Lưu trữ	Chọn người dùng cần sửa, nhập các thông tin sửa chữa và lưu lại.
8	Xóa đề tài chưa được duyệt	Lưu trữ	Xóa đề tài khỏi cơ sở dữ liệu.
9	Xem lịch sử đề tài	Kết xuất	Hiện tất cả các đề tài có phân trang.

10	Gửi thông báo	Lưu trữ	Gửi thông báo đến sinh viên.
----	---------------	---------	------------------------------

Bảng 3.1. Yêu cầu chức năng nghiệp vụ của giảng viên

✓ Sinh viên:

STT	Yêu cầu	Loại YC	Mô tả/ Ràng buộc/ Công thức
1	Đề xuất đề tài	Lưu trữ	Thêm đề tài vào cơ sở dữ liệu.
2	Hủy đề xuất đề tài	Lưu trữ	Xóa đề tài khỏi cơ sở dữ liệu.
3	Đăng ký thực hiện đề tài	Lưu trữ	Đăng ký thực hiện đề tài.
4	Xem thông tin đề tài thực hiện	Lưu trữ	Hiển thị thông tin đề tài đang thực hiện.
5	Cập nhật báo cáo quan trọng	Lưu trữ	Lưu các báo cáo quan trọng vào đề tài.
6	Xem điểm đề tài	Kết xuất	Hiển thị thông tin điểm của bản thân.

Bảng 3.2. Yêu cầu chức năng nghiệp vụ của sinh viên

✓ Giáo vụ:

STT	Yêu cầu	Loại YC	Mô tả/ Ràng buộc/ Công thức
1	Thêm học kỳ	Lưu trữ	Danh mục của học kỳ phải tồn tại ở cơ sở dữ liệu.
2	Thêm đợt đăng ký	Lưu trữ	Danh mục của đợt đăng ký phải tồn tại ở cơ sở dữ liệu.
3	Thêm phiếu đánh giá	Lưu trữ	Danh mục của phiếu đánh giá phải tồn tại ở cơ sở dữ liệu.
4	Thêm tiêu chí	Lưu trữ	Danh mục của tiêu chí đánh giá

	đánh giá		phải tồn tại ở cơ sở dữ liệu.
5	Thêm hội đồng	Lưu trữ	Danh mục của hội đồng phải tồn tại ở cơ sở dữ liệu.
6	Thêm người dùng	Lưu trữ	Danh mục của người dùng phải tồn tại ở cơ sở dữ liệu.
7	Xem danh sách học kỳ	Kết xuất	Hiện tất cả các học kỳ có phân trang.
8	Xem danh sách đợt đăng ký	Kết xuất	Hiện tất cả các đợt đăng ký có phân trang.
9	Xem danh sách phiếu đánh giá	Kết xuất	Hiện tất cả các phiếu đánh giá có phân trang.
10	Xem danh sách tiêu chí đánh giá	Kết xuất	Hiện tất cả các tiêu chí đánh giá có phân trang.
11	Xem danh sách hội đồng	Kết xuất	Hiện tất cả các hội đồng có phân trang.
12	Xem danh sách người dùng	Kết xuất	Hiện tất cả các người dùng có phân trang.
13	Sửa học kỳ bất kỳ	Lưu trữ	Chọn học kỳ cần sửa, nhập các thông tin sửa chữa và lưu lại.
14	Sửa đợt đăng ký bất kỳ	Lưu trữ	Chọn đợt đăng ký cần sửa, nhập các thông tin sửa chữa và lưu lại.
15	Sửa tiêu chí đánh giá bất kỳ	Lưu trữ	Chọn tiêu chí cần sửa, nhập các thông tin sửa chữa và lưu lại.
16	Sửa thông tin người dùng	Lưu trữ	Chọn người dùng cần sửa, nhập các thông tin sửa chữa và lưu lại.
17	Xóa học kỳ	Lưu trữ	Xóa học kỳ khỏi cơ sở dữ liệu.
18	Xóa đợt đăng ký	Lưu trữ	Xóa đợt đăng ký khỏi cơ sở dữ liệu.

19	Xóa phiếu đánh giá	Lưu trữ	Xóa phiếu đánh giá khỏi cơ sở dữ liệu.
20	Xóa hội đồng	Lưu trữ	Xóa người dùng khỏi cơ sở dữ liệu.
21	Phân công GVPP	Lưu trữ	Phân công GVPP cho đề tài.
22	Xem lịch sử đề tài	Kết xuất	Hiện tất cả các đề tài có phân trang.
23	Xuất danh sách đề tài	Kết xuất	Tải xuống danh sách đề tài.
24	Gửi thông báo	Lưu trữ	Gửi thông báo đến người dùng.
25	Cập nhật thông tin hội đồng bảo vệ	Lưu trữ	Cập nhật thông tin hội đồng vào cơ sở dữ liệu.
26	Xuất bảng điểm của người dùng	Kết xuất	Tải xuống bảng điểm người dùng.
27	Duyệt đề tài thực hiện	Lưu trữ	Xác nhận cho đề tài được thực hiện.
28	Xóa người dùng	Lưu trữ	Xóa người dùng ra khỏi hệ thống dữ liệu

Bảng 3.3. Yêu cầu chức năng nghiệp vụ của giáo vụ

3.1.1.2 Yêu cầu chức năng hệ thống

3.1.2. Yêu cầu phi chức năng

STT	Yêu cầu	Loại yêu cầu
1	Giao diện thân thiện, dễ sử dụng	Tính tiện dụng
2	Có khả năng mở rộng , nâng cấp , cải tiến trong tương lai	Tính tiến hóa
3	Tốc độ tải trang ổn định , các thao tác phản hồi nhanh chóng	Tính hiệu quả

4	Bảo mật tốt	Tính bảo mật
---	-------------	--------------

Bảng 3.4. Yêu cầu phi chức năng

3.2. Tác nhân hệ thống

- ✓ Sinh viên : Là người có thể đăng ký thực hiện đề tài mới hoặc đề tài đã có sẵn, quản lý đề tài đang được thực hiện , nộp các báo cáo , thông tin cần thiết
- ✓ Giảng viên : Là người quản lý các đề tài được sinh viên đề xuất , tạo ra các đề tài cho sinh viên vào đăng ký , thực hiện gửi phê duyệt các đề tài đến giáo vụ để được phê duyệt. Đóng vai trò chính trong hệ thống để hoàn tất các quy trình mà Giáo vụ đưa ra từ chấm điểm theo form cũng như chấm điểm hội đồng phản biện , phản biện với sinh viên.
- ✓ Giáo vụ : là quản trị viên của hệ thống , có thể quản lý các học kỳ, đợt đăng ký, quản lý các phiếu chấm điểm , các hội đồng , người dùng hệ thống , tải xuống thông tin các đề tài , bảng điểm của sinh viên , gửi yêu cầu cập nhật các báo cáo , thông tin cần thiết.

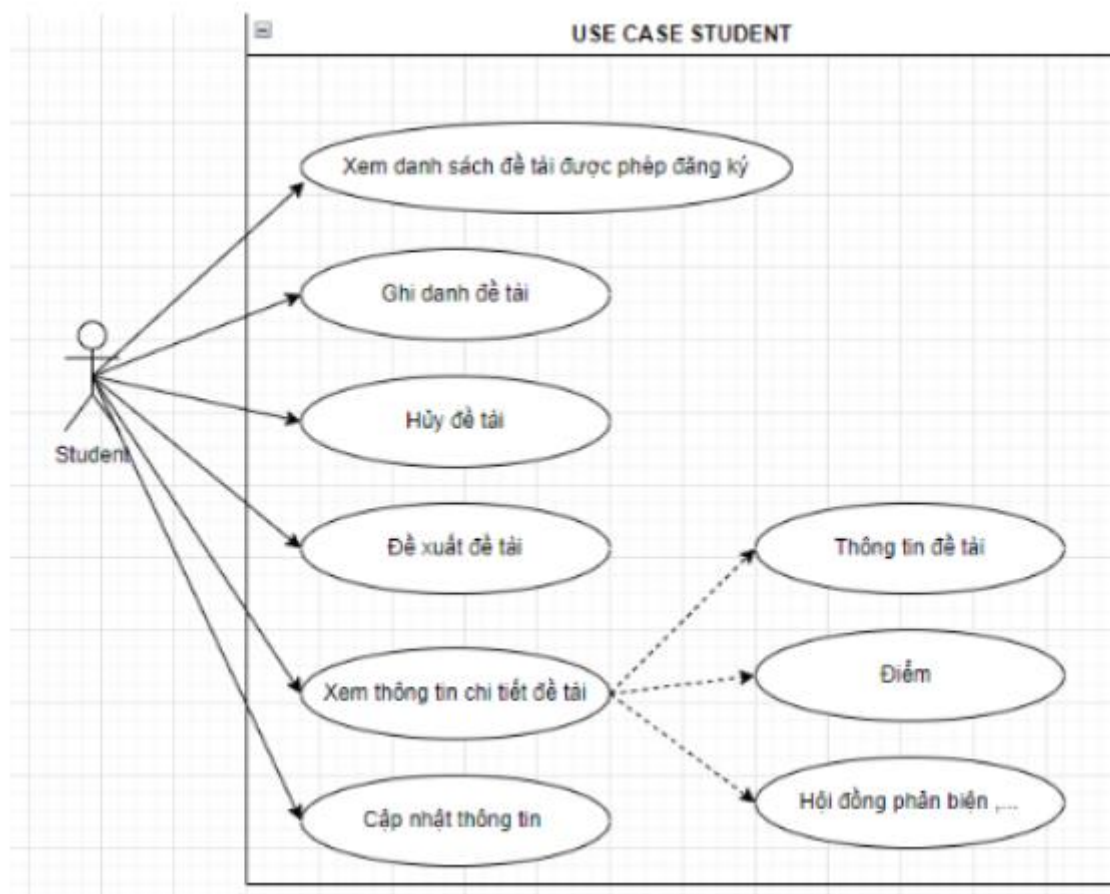
3.3. Lược đồ Use Case

3.3.1. Lược đồ Use Case Sinh viên

STT	Tên use case	Ý nghĩa
1	Xem danh sách đề tài được phép đăng ký	Sinh viên có thể xem danh sách đề tài đang mở để đăng ký trong đợt đăng ký đề tài.
2	Ghi danh đề tài	Sinh viên chọn và ghi danh vào đề tài mà họ mong muốn thực hiện và đăng ký.
3	Hủy đề tài	Sinh viên có thể hủy đăng ký đề tài nếu đăng ký chưa được chấp nhận bởi giảng viên hướng dẫn.
4	Đề xuất đề tài	Sinh viên nhập thông tin đề tài mà họ muốn được hướng dẫn và đề xuất với giảng viên.
5	Xem thông tin chi	Sinh viên xem chi tiết về đề tài mà họ đã đăng ký,

	tiết đề tài	bao gồm thông tin đề tài, điểm, hội đồng phản biện, v.v.
6	Cập nhật thông tin cho đề tài	Sinh viên tiến hành cập nhật các thông tin cần thiết cho đề tài trong quá trình thực hiện đề tài

Bảng 3.5 Mô tả Use case Sinh viên



Hình 3.1 Lược đồ Use case Sinh viên

3.3.2. Lược đồ Use Case Giảng Viên

STT	Tên use case	Ý nghĩa
1	Quản lý đề tài	Giảng viên quản lý đề tài của giảng viên đó, bao gồm thêm đề tài, sửa đề tài, xóa đề tài, duyệt đề tài và gửi đề tài lên giáo vụ.
2	Quản lý đề tài sinh viên đề xuất	Quản lý tất cả các đề tài mà sinh viên đề xuất được làm.

3	Duyệt nhóm thực hiện	Giảng viên có thể xem thông tin nhóm đăng kí đề tài và chọn duyệt ra 1 nhóm tốt nhất để làm đề tài.
4	Quản lí đề tài hướng dẫn	Giảng viên quản lí các đề tài hướng dẫn sau khi xác nhận và đánh giá cho điểm cho nhóm.
5	Quản lí đề tài phản biện	Giảng viên quản lí các đề tài được phân công để phản biện và cho điểm các thành viên trên hệ thống.
6	Quản lí các đề tài tham gia trong hội đồng phản biện	Giảng viên quản lí các đề tài được tham gia vào hội đồng và chấm điểm theo phiếu điểm.

Bảng 3.6 Mô tả Use Case Giảng Viên

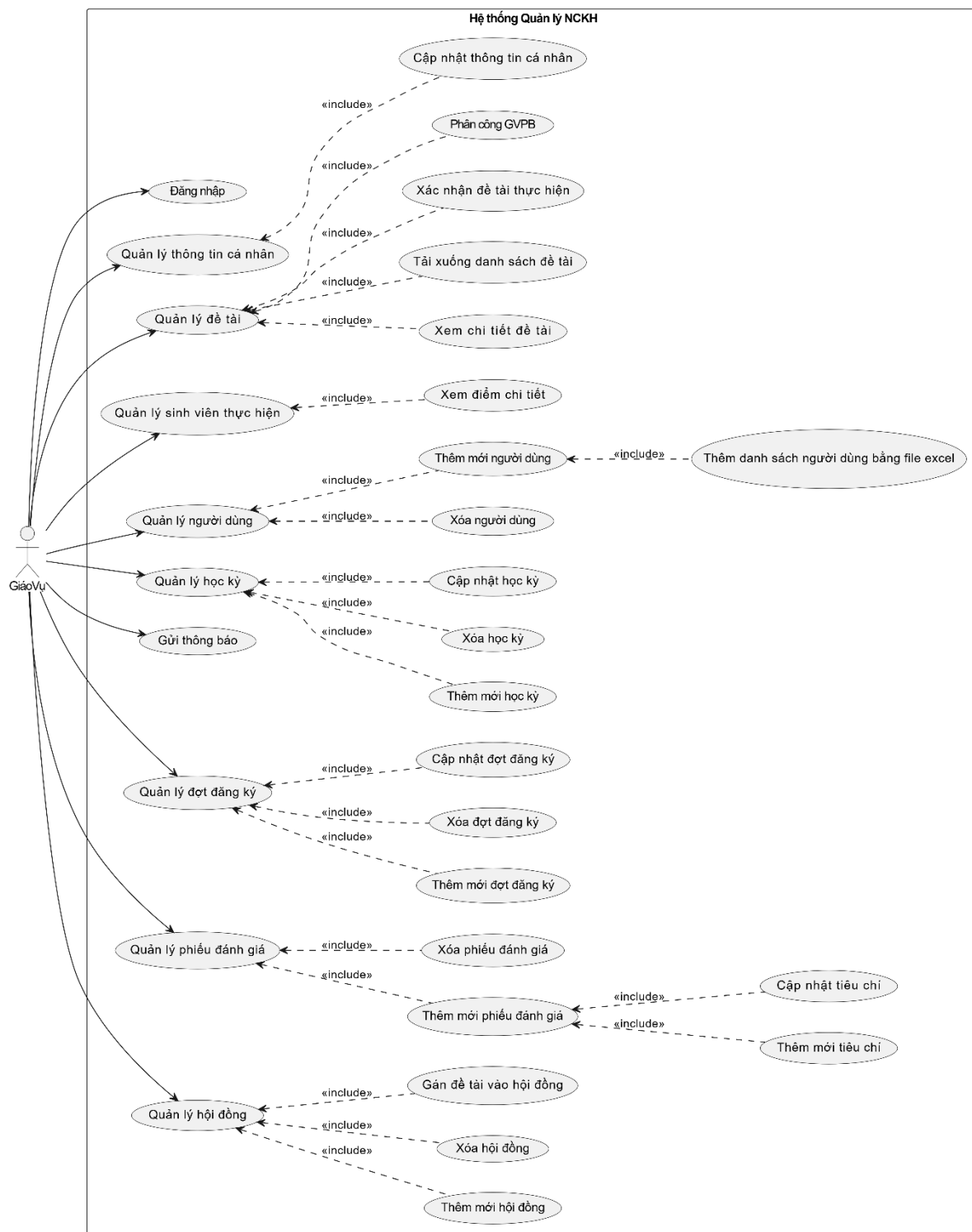


Hình 3.2. Lược đồ Use case Giảng Viên

3.3.3. Lược đồ Use Case Giáo vụ

STT	Tên use case	Ý nghĩa
1	Quản lý thông tin cá nhân	Giáo vụ quản lý thông tin cá nhân trên hệ thống.
2	Quản lý đề tài	Giáo vụ quản lý tất cả các đề tài trên hệ thống.
3	Quản lý sinh viên thực hiện	Giáo vụ quản lý nhóm sinh viên đang thực hiện đề tài.
4	Quản lý người dùng	Giáo vụ quản lý tất cả người dùng trên hệ thống.
5	Quản lý học kì	Giáo vụ quản lý thông tin học kì trên hệ thống.
6	Quản lý đợt đăng ký	Giáo vụ quản lý đợt đăng ký trên hệ thống.
7	Quản lý phiếu đánh giá	Giáo vụ quản lý các phiếu đánh giá trên hệ thống.
8	Quản lý hội đồng	Giáo vụ quản lý thông tin hội đồng bảo vệ.
9	Gửi thông báo	Giáo vụ gửi thông báo đến giảng viên và sinh viên.

Bảng 3.7. Mô tả Use Case Giáo vụ



3.4. Đặc tả Use Case

3.4.1. Use case chung

3.4.1.1 Đăng nhập

Use case	Đăng nhập
Short Description	Người dùng có thể đăng nhập vào tài khoản thông qua email và mật khẩu đã tạo
Actors	Sinh viên, Giảng viên, Giáo vụ
Pre-Conditions	Thông tin đăng nhập phải tồn tại trong hệ thống.
Post Conditions	Người dùng đăng nhập thành công.
Main Flow	(1) Người dùng truy cập vào website (2) Hệ thống tự động đưa người dùng đến trang đăng nhập (3) Người dùng điền tài khoản và mật khẩu. (4) Người dùng nhấp vào “Đăng nhập” (5) Hệ thống xác thực tài khoản. [E1] (6) Hệ thống điều hướng trang đến màn hình trang chủ. [A1]
Alternate Flow	(A1) Hệ thống điều hướng đến trang đúng với phân quyền của tài khoản.
Exception Flow	(E1) Tài khoản hoặc mật khẩu người dùng vừa nhập không hợp lệ, quay lại bước (3)

Bảng 3.8 Use case Đăng nhập

3.4.1.2 Xem thông tin cá nhân

Use case	Xem thông tin cá nhân
Short Description	Người dùng có thể xem thông tin của mình trên website
Actor	Sinh viên , Giảng viên , Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống
Post Conditions	Người dùng xem thông tin cá nhân thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào "Thông tin cá nhân"

	(3) Hệ thống điều hướng đến trang thông tin cá nhân
Alternate Flow	
Exception Flow	

Bảng 3.9: Use Case Xem thông tin cá nhân

3.4.1.3 Cập nhập thông tin cá nhân

Use case	Cập nhập thông tin cá nhân
Short Description	Người dùng có thể cập nhật thông tin cá nhân của mình
Actors	Sinh viên, Giảng viên, Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống Người dùng đang xem thông tin cá nhân của mình
Post Conditions	Người dùng cập nhật thông tin cá nhân thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào “Thông tin cá nhân” (3) Hệ thống điều hướng đến trang thông tin cá nhân (4) Người dùng thay đổi thông tin cá nhân (5) Người dùng ấn nút “Cập nhật” (6) Hệ thống xác thực thông tin (7) Hệ thống thông báo kết quả thành công. [A1]
Alternate Flow	(A1) Hệ thống thông báo kết quả thất bại.
Exception Flow	

Bảng 3.10: Use Case Cập nhập thông tin cá nhân

3.4.2.Sinh Viên

3.4.2.1.Xem danh sách đề tài được đăng ký

Use case	Xem danh sách đề tài được phép đăng ký
Short Description	Sinh viên xem danh sách các đề tài mà họ có thể đăng ký thực hiện

Actors	Sinh viên
Pre-Conditions	Sinh viên truy cập vào hệ thống, đăng nhập thành công và trong thời gian đăng ký đề tài
Post Conditions	Sinh viên xem được danh sách đề tài và có thể thực hiện đăng ký nếu muốn
Main Flow	(1) Sinh viên đăng nhập vào hệ thống (2) Sinh viên chọn mục "Đề tài" (3) Hệ thống hiển thị danh sách các đề tài được phép đăng ký (4) Sinh viên xem thông tin chi tiết của từng đề tài.
Alternate Flow	(A1) Sinh viên có thể quay lại danh sách đề tài hoặc thực hiện các hành động khác trên hệ thống
Exception Flow	Không có trường hợp ngoại lệ trong quá trình xem danh sách đề tài được phép đăng ký.

Bảng 3.11: Use Case Xem danh sách đề tài được phép đăng ký

3.4.2.2. Ghi danh vào đề tài

Use case	Ghi danh vào đề tài
Short Description	Người dùng đăng nhập vào hệ thống để ghi danh vào một đề tài cụ thể.
Actors	Sinh viên
Pre-Conditions	Sinh viên truy cập vào hệ thống, đăng nhập thành công và vào mục "Đề tài" để xem danh sách đề tài
Post Conditions	Sinh viên ghi danh vào đề tài thành công.
Main Flow	(1) Sinh viên truy cập vào trang web. (2) Sinh viên đăng nhập thành công (3) Sinh viên thực hiện xem danh sách đề tài ở mục "Đề tài" (4) Sinh viên nhấn vào xem chi tiết đề tài họ muốn ghi

	<p>danh</p> <p>(5) Sinh viên tiến hành chọn các sinh viên khác cùng thực hiện (nếu cần)</p> <p>(6) Sinh viên nhấn nút “Ghi danh”</p> <p>(7) Hệ thống xác nhận việc ghi danh vào đề tài [A1].</p>
Alternate Flow	(A1) Hệ thống điều hướng đến trang chủ
Exception Flow	Sinh viên thực hiện quay trở lại danh sách, kết thúc quá trình ghi danh đề tài

Bảng 3.12: Use Case Ghi danh vào đề tài

3.4.2.3. Hủy đề tài

Use case	Hủy đề tài
Short Description	Sinh viên thực hiện hủy đề tài đã ghi danh trước đó.
Actors	Sinh viên
Pre-Conditions	Sinh viên truy cập vào hệ thống, đăng nhập thành công, đã ghi danh vào một đề tài chưa được giảng viên hướng dẫn phê duyệt nhóm thực hiện
Post Conditions	Sinh viên thực hiện hủy đề tài thành công
Main Flow	<p>(1) Sinh viên truy cập vào hệ thống và đăng nhập thành công.</p> <p>(2) Sinh viên truy cập vào “Trang chủ”.</p> <p>(3) Hệ thống hiển thị thông tin đề tài mà sinh viên đã ghi danh.</p> <p>(4) Sinh viên nhấn vào nút “Hủy đề tài”.</p> <p>(5) Hệ thống hiển thị xác nhận việc hủy đề tài và yêu cầu xác nhận từ sinh viên.</p> <p>(6) Sinh viên xác nhận việc hủy đề tài.</p> <p>(7) Hệ thống xác nhận hủy đề tài thành công</p>
Alternate Flow	Sinh viên hủy đề tài đã ghi danh thành công

Exception Flow	Sinh viên hủy xác nhận, hệ thống đóng thông báo xác nhận hủy đề tài
----------------	---

Bảng 3.13: Use Case Hủy đề tài

3.4.2.4. Đề xuất đề tài

Use case	Đề xuất đề tài
Short Description	Sinh viên đề xuất đề tài mong muốn được thực hiện với giảng viên hướng dẫn
Actors	Sinh viên
Pre-Conditions	Sinh viên truy cập vào hệ thống và đăng nhập thành công
Post Conditions	Sinh viên có thể xem lại thông tin về đề tài đã đề xuất và theo dõi quá trình xử lý.
Main Flow	<p>(1) Sinh viên truy cập vào hệ thống và đăng nhập thành công.</p> <p>(2) Sinh viên chọn mục "Đề xuất đề tài".</p> <p>(3) Hệ thống hiển thị giao diện đề xuất đề tài, trong đó sinh viên điền thông tin về đề tài mong muốn.</p> <p>(4) Sinh viên nhập thông tin cần thiết và chọn "Gửi đề xuất".</p> <p>(5) Hệ thống lưu thông tin đề xuất và thông báo cho sinh viên về việc gửi đề xuất thành công.[A1]</p>
Alternate Flow	(A1) Nếu sinh viên quay lại trang chủ hoặc thực hiện các hành động khác trên hệ thống.
Exception Flow	Sinh viên đề xuất thất bại hệ thống thông báo không thành công

Bảng 3.14: Use Case Đề xuất đề tài

3.4.2.5. Xem chi tiết đề tài

Use case	Xem chi tiết đề tài
----------	---------------------

Short Description	Sinh viên xem chi tiết về đề tài mà họ đã đăng ký, bao gồm thông tin đề tài, điểm, hội đồng phản biện, v.v.
Actors	Sinh viên
Pre-Conditions	Sinh viên truy cập vào hệ thống, đăng nhập thành công, đã ghi danh vào một đề tài
Post Conditions	Xem được đầy đủ các thông tin của một đề tài
Main Flow	<p>(1) Sinh viên truy cập vào hệ thống và đăng nhập thành công</p> <p>(2) Sinh viên truy cập vào trang chủ</p> <p>(3) Hệ thống hiển thị thông tin đầy đủ về đề tài họ đã ghi danh và cung cấp các tùy chọn để xem chi tiết hơn.</p> <p>a. Xem thông tin đề tài:</p> <ul style="list-style-type: none"> - Hiển thị tên đề tài, mô tả chi tiết, loại đề tài, thuộc chuyên ngành,... - Cung cấp thông tin về người hướng dẫn người phản biện <p>b. Xem điểm [E1]:</p> <ul style="list-style-type: none"> - Hiển thị tùy chọn xem điểm của giảng viên hướng dẫn, giảng viên phản biện, và điểm của hội đồng. - Sinh viên nhấn vào tùy chọn tương ứng để xem điểm <p>c. Xem hội đồng phản biện [E2]:</p> <ul style="list-style-type: none"> - Hiển thị tùy chọn xem thông tin hội đồng - Sinh viên nhấn vào tùy chọn xem thông tin hội đồng - Hệ thống liệt kê các thành viên trong hội đồng phản biện.
Alternate Flow	Sinh viên có thể thực hiện các hành động khác trên hệ thống
Exception Flow	<p>[E1] Chưa có thông tin bảng điểm thì hệ thống hiển thị thông báo chưa có bảng điểm</p> <p>[E2] Chưa có thông tin hội</p>

	đồng thì hệ thống hiển thị thông báo chưa có thông tin hội đồng
--	---

Bảng 3.15: Use Case Xem chi tiết đề tài

3.4.2.6.Cập nhập thông tin cho đề tài

Use case	Cập nhập thông tin cho đề tài
Short Description	Sinh viên tiến hành cập nhật các thông tin cần thiết cho đề tài trong quá trình thực hiện
Actors	Sinh viên
Pre-Conditions	Sinh viên truy cập vào hệ thống, đăng nhập thành công và đang thực hiện 1 đề tài
Post Conditions	Sinh viên có thể cập nhật các thông tin quan trọng và cần thiết cho đề tài
Main Flow	<p>(1) Sinh viên truy cập vào hệ thống và đăng nhập thành công</p> <p>(2) Sinh viên truy cập vào trang chủ</p> <p>(3) Sinh viên cung cấp các thông tin cần thiết vào từng tùy chọn cập nhật thông tin</p> <p>(4) Sinh viên chọn Cập nhật [A1]</p> <p>(5) Hệ thống xác nhận thông tin và lưu vào cơ sở dữ liệu. [E1]</p>
Alternate Flow	(A1) Sinh viên không cập nhật, sinh viên có thể thực hiện các hành động khác
Exception Flow	(E1) Sinh viên cung cấp thông tin không hợp lệ thì hệ thống thông báo lỗi thông tin không hợp lệ (3)

Bảng 3.16.Use Case Cập nhập thông tin cho đề tài

3.4.3.Giảng viên

3.4.3.1.Thêm đề tài

Use case	Thêm đề tài
Short Description	Người dùng cần tạo tài khoản để đăng nhập và sử dụng các dịch vụ của website
Actors	Giảng viên
Pre-Conditions	Người dùng phải vào trang web và cung cấp đầy đủ thông tin để có thể thực hiện tạo đề tài
Post Conditions	Giảng viên tạo thành công đề tài và hiển thị lên hệ thống
Main Flow	(1) Giảng viên truy cập vào website (2) Giảng viên chọn vào mục Đăng nhập (3) Giảng viên chọn Quản lý đề tài (4) Giảng viên nhấn vào nút “Thêm đề tài” (5) Giảng viên nhập thông tin vào form[A1] (6) Giảng viên nhấn vào nút “Thêm đề tài” (7) Hiển thị thông báo thêm thành công đề tài
Alternate Flow	(A1) Người dùng hủy đăng kí đề tài
Exception Flow	

Bảng 3.17: Use Case Thêm đề tài

3.4.3.2.Sửa đề tài

Use case	Sửa đề tài
Short Description	Giảng viên sửa đề tài vừa tạo ra
Actors	Giảng viên
Pre-Conditions	Đề tài phải tồn tại trên hệ thống
Post Conditions	Đề tài được chỉnh sửa thành công
Main Flow	(1) Giảng viên truy cập vào website (2) Giảng viên chọn vào mục “Quản lý đề tài”. (3) Giảng viên chọn đề tài muốn chỉnh sửa. (4) Giảng viên nhấn vào nút “Cập nhật đề tài”

	(5) Hiện thị form chỉnh sửa đề tài (6) Nhập thông tin vào form chỉnh sửa [A1] (7) Nhấn vào nút “Cập nhật”
Alternate Flow	(A1) Huỷ cập nhật và đóng form cập nhật.
Exception Flow	

Bảng 3.18: Use Case Sửa đề tài

3.4.3.3.Xóa đề tài

Use case	Xoá đề tài
Short Description	Giảng viên có thể xoá đề tài mà mình quản lí.
Actors	Giảng viên
Pre-Conditions	Phải đăng nhập thành công vào hệ thống, đề tài không có nhóm thực hiện.
Post Conditions	Đề tài bị xoá khỏi hệ thống và không còn hiển thị trên màn hình.
Main Flow	(1) Chọn vào mục Quản lí đề tài . (2) Chọn đề tài mong muốn xoá (3) Chọn nút “Xoá”
Alternate Flow	
Exception Flow	

Bảng 3.19: Use Case Xóa đề tài

3.4.3.4.Gửi đề tài lên giáo vụ

Use case	Gửi đề tài lên giáo vụ
Short Description	Giảng viên gửi đề xuất đề tài của bản thân hoặc sinh viên lên đội giáo vụ duyệt
Actors	Giảng viên
Pre-Conditions	Phải đăng nhập thành công vào hệ thống

	Đề tài phải ở status registered hoặc là ready – ready
Post Conditions	Đề tài hiển thị status pending
Main Flow	(1) Người dùng chọn vào mục quản lí đề tài . (2) Người dùng chọn đề tài muốn gửi lên Giáo vụ (3) Người dùng chọn nút “Gửi” để đưa thông tin đề tài lên giáo vụ.
Alternate Flow	(1) Người dùng chọn vào Đề tài đề xuất. (2) Người dùng chọn đề tài muốn gửi lên Giáo vụ (3) Người dùng chọn nút “Gửi” để đưa thông tin đề tài lên giáo vụ.
Exception Flow	

Bảng 3.20: Use Case Gửi đề tài lên giáo vụ

3.4.3.5.Xóa đề tài (Quản lý đề tài)

Use case	Xoá đề tài
Short Description	Giảng viên xoá đề tài của bản thân
Actors	Giảng viên
Pre-Conditions	Quản trị viên phải đăng nhập thành công vào hệ thống.Đề tài chưa được gửi lên giáo vụ
Post Conditions	Đề tài không còn hiển thị trên hệ thống
Main Flow	(1) Chọn vào Quản lý đề tài (2) Chọn đề tài cần xoá (3) Nhấn nút xoá để xoá đi đề tài
Alternate Flow	(A1) Chọn “X” để hủy bỏ thao tác thêm mới và quay về trang Quản lý đề tài
Exception Flow	

Bảng 3.21: Use Case Xóa đề tài (Quản lý đề tài)

3.4.3.6.Yêu cầu cập nhập bảng điểm

Use case	Yêu cầu cập nhật bảng điểm
Short Description	Giảng viên yêu cầu nhóm sinh viên đăng kí đề tài do mình hướng dẫn cập nhật thêm bảng điểm lên hệ thống để giảng viên cân nhắc giữa nhiều nhóm sinh viên đăng kí để chọn ra nhóm phù hợp
Actors	Giảng viên
Pre-Conditions	Quản trị viên phải đăng nhập thành công vào hệ thống. Đề tài đó đã có nhóm đăng kí đề tài
Post Conditions	Hiện thị thông báo gửi tin nhắn thành công
Main Flow	(1) Chọn vào “Duyệt đề tài” (2) Chọn một đề tài cần duyệt nhóm làm (3) Lựa chọn 1 nhóm bất kì trong danh sách các nhóm đã đăng kí đề tài (4) Nhấn vào biểu tượng xem thông tin nhóm (5) Chuyển hướng đến trang chi tiết các thành viên trong nhóm (6) Nhấn biểu tượng gửi tin nhắn (7) Nhập nội dung cần nhắn gửi đến sinh viên (8) Nhấn vào nút “Gửi”.[A1]
Alternate Flow	(A1) Chọn “X” để huỷ bỏ thao tác gửi tin nhắn
Exception Flow	

Bảng 3.22: Use Case Yêu cầu đăng nhập bảng điểm

3.4.3.7. Chọn nhóm sinh viên thực hiện đề tài

Use case	Chọn nhóm thực hiện đề tài
Short Description	Giảng viên chọn 1 nhóm phù hợp trong nhiều nhóm đăng kí để thực hiện đề tài .
Actors	Giảng viên
Pre-Conditions	Giảng viên phải đăng nhập thành công vào hệ thống.

Post Conditions	- Đề tài bị xoá khỏi mục duyệt đề tài - Đề tài hiển thị bên mục “Đề tài hướng dẫn”
Main Flow	(1) Chọn vào “Duyệt đề tài” (2) Chọn đề tài mong muốn duyệt. (3) Chọn ra một nhóm ưu tú trong danh sách các nhóm đăng kí. (4) Nhấn nút “Xác nhận”
Alternate Flow	
Exception Flow	

Bảng 3.23: Use Case Chọn nhóm sinh viên thực hiện đề tài

3.4.3.8.Chấm điểm đề tài hướng dẫn

Use case	Chấm điểm đề tài hướng dẫn
Short Description	Giảng viên chấm điểm cho nhóm sinh viên mà mình hướng dẫn.
Actors	Giảng viên
Pre-Conditions	Giảng viên phải đăng nhập thành công vào hệ thống. Đề tài của giảng viên đó phải có nhóm sinh viên đăng kí và được xác nhận hướng dẫn bởi giảng viên đó. Đã có form chấm điểm dành cho giảng viên hướng dẫn.
Post Conditions	Hiển thị thông báo cập nhật thông tin bảng điểm thành công
Main Flow	(1) Chọn vào mục “Đề tài hướng dẫn” (2) Người dùng chọn vào đề tài bất kì . (3) Người dùng chọn vào biểu tượng edit ở từng dòng sinh viên (4) Người dùng nhập thông tin điểm theo form đã có sẵn để chấm điểm. (5) Người dùng nhấn vào nút “ Lưu thay đổi “ để cập nhật

	thông tin điểm cho sinh viên
Alternate Flow	
Exception Flow	

Bảng 3.24: Use Case Chấm điểm đề tài hướng dẫn

3.4.3.9.Chấm điểm đề tài phản biện

Use case	Chấm điểm đề tài phản biện
Short Description	Giảng viên chấm điểm cho nhóm sinh viên mà mình phụ trách phản biện
Actors	Giảng viên
Pre-Conditions	Giảng viên phải đăng nhập thành công vào hệ thống. Đề tài phụ trách phản biện phải có nhóm sinh viên đăng kí và được xác nhận hướng dẫn bởi giảng viên hướng dẫn. Đã có form chấm điểm dành cho giảng viên phản biện.
Post Conditions	Hiện thị thông báo cập nhật thông tin bảng điểm thành công
Main Flow	(1) Chọn vào mục “Đề tài phản biện” (2) Người dùng chọn vào đề tài bất kì . (3) Người dùng chọn vào biểu tượng edit ở từng dòng sinh viên. (4) Người dùng nhập thông tin điểm theo form đã có sẵn để chấm điểm. (5) Người dùng nhấn vào nút “ Lưu thay đổi “ để cập nhật thông tin điểm cho sinh viên
Alternate Flow	
Exception Flow	

Bảng 3.25: Use Case Chấm điểm đề tài phản biện

3.4.3.10.Hội đồng chấm điểm đề tài

Use case	Chấm điểm đề tài hội đồng
Short Description	Giảng viên chấm điểm cho đề tài mà mình tham gia trong hội đồng phản biện
Actors	Giảng viên
Pre-Conditions	Giảng viên phải đăng nhập thành công vào hệ thống. Giảng viên nằm trong hội đồng phản biện của đề tài Đã có form chấm điểm dành cho hội đồng
Post Conditions	Hiển thị thông báo cập nhật thông tin bảng điểm thành công
Main Flow	(1) Chọn vào mục “Hội đồng” [A1]< (2) Người dùng chọn vào đề tài bất kì . (3) Người dùng chọn vào biểu tượng edit ✎ để hiển thị form chấm điểm dành cho hội đồng. [A2] (4) Người dùng nhập thông tin điểm theo form đã có sẵn để chấm điểm . (5) Người dùng nhấn vào nút “ Lưu thay đổi “ để cập nhật thông tin điểm cho sinh viên
Alternate Flow	[A1] Người dùng chọn vào mục “ Hướng dẫn đề tài “
Exception Flow	[A2] Người dùng chọn đề tài bất kì để hiển thị thông tin hội đồng phản biện của nhóm sinh viên. Người dùng chọn button hội đồng . Nếu người dùng nằm trong bất kì một vai trò nào trong hội đồng sẽ hiển thị form điểm như [A2] để tiếp tục như các bước tiếp theo.

Bảng 3.26: Use Case Hội đồng chấm điểm đề tài

3.4.3.11.Xem thông tin nhóm

Use case	Xem thông tin nhóm
----------	--------------------

Short Description	Xem chi tiết thông tin nhóm làm đề tài
Actors	Giảng viên
Pre-Conditions	Người dùng phải đăng nhập thành công vào hệ thống quản lý Phải có sinh viên đăng kí đề tài hoặc được phân công phản biện một đề tài
Main Flow	(1) Chọn vào mục “Đề tài hướng dẫn” hoặc “Đề tài phản biện” (2) Chọn vào một đề tài bất kì. (3) Chọn vào biểu tượng view đề xem chi tiết thông tin nhóm
Alternate Flow	
Exception Flow	

Bảng 3.27: Use Case Xem thông tin nhóm

3.4.3.12. Xem lịch sử đề tài

Use Case	Xem lịch sử đề tài hướng dẫn
Short Description	Giảng viên xem lại lịch sử đề tài bản thân đã hướng dẫn qua các năm
Actors	Giảng viên
Pre-Conditions	Người dùng phải đăng nhập thành công vào hệ thống quản lý
Post Conditions	Hiện thị danh sách đề tài bản thân đã hướng dẫn
Main Flow	(1) Chọn vào Lịch sử đề tài (2) Hệ thống hiển thị danh sách đề tài mà giảng viên hướng dẫn
Alternate Flow	
Exception Flow	

Bảng 3.28: Use Case Xem lịch sử đề tài

3.4.4. Giáo vụ

3.4.4.1. Xem chi tiết đề tài

Use Case	Xem thông tin chi tiết đề tài
Short Description	Người dùng có thể xem thông tin chi tiết của đề tài đang thực hiện
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống Hệ thống có đề tài đang thực hiện
Post Conditions	Người dùng xem thông tin chi tiết đề tài thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đề tài đang thực hiện” (3) Hệ thống hiển thị danh sách đề tài đang thực hiện (4) Người dùng chọn vào đề tài muốn xem chi tiết (5) Hệ thống hiển thị chi tiết thông tin đề tài (A1)
Alternate Flow	(A1) Hệ thống thông báo lỗi khi tải dữ liệu
Exception Flow	

Bảng 3.29: Use Case Xem thông tin chi tiết đề tài

3.4.4.2. Xác nhận đề tài thực hiện

Use Case	Người dùng xác nhận đề tài thực hiện
Short Description	Người dùng xác nhận cho phép đề tài đăng kí thực hiện
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống Hệ thống có tồn tại đề tài đang chờ duyệt
Post Conditions	Người dùng duyệt đề tài thành công
Main Flow	(1) Người dùng truy cập vào web (2) Người dùng truy cập vào mục “Đề tài chờ duyệt”

	(3) Người dùng chọn vào đề tài muốn duyệt (4) Hệ thống hiển thị Dialog duyệt đề tài (5) Người dùng chọn duyệt đề tài [A1] (6) Hệ thống thông báo duyệt đề tài thành công [E1]
Alternate Flow	
Exception Flow	(E1) Hệ thống thông báo duyệt đề tài thất bại

Bảng 3.30: Use Case Xác nhận đề tài thực hiện

3.4.4.3. Phân công GVPB

Use Case	Người dùng phân công GVPB cho đề tài
Short Description	Người dùng có thể phân công GVPB cho đề tài
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống
Post Conditions	Người dùng phân công GVPB thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đề tài đang thực hiện” (3) Hệ thống hiển thị danh sách đề tài đang thực hiện (4) Người dùng chọn vào đề tài muốn xem chi tiết (5) Hệ thống hiển thị chi tiết thông tin đề tài (6) Người dùng chọn vào mục GVPB (7) Hệ thống hiển thị Dialog để người dùng chọn GVPB (8) Người dùng chọn GVPB và ấn nút xác nhận (9) Hệ thống thông báo cập nhật GVPB thành công (A1)
Alternate Flow	(A1) Hệ thống thông báo cập nhật GVPB thất bại
Exception Flow	

Bảng 3.31: Use Case Phân công GVPB

3.4.4.4. Tải xuống danh sách đề tài

Use Case	Người dùng tải xuống danh sách đề tài
Short Description	Người dùng có thể tải xuống danh sách đề tài hiện có trên website
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống Hệ thống đã có danh sách các đề tài
Post Conditions	Người dùng tải xuống danh sách đề tài thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đề tài đang thực hiện” (A1) (3) Hệ thống hiển thị danh sách đề tài (4) Người dùng chọn “Tải xuống” (5) Hệ thống xác thực thông tin và tải xuống danh sách đề tài
Alternate Flow	(A1) Người dùng chọn vào mục “Lịch sử đề tài”, tiếp tục bước 3
Exception Flow	

Bảng 3.32: Use Case Tải xuống danh sách đề tài

3.4.4.5. Xem danh sách sinh viên thực hiện

Use Case	Người dùng xem danh sách sinh viên thực hiện đề tài
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống Hệ thống có đề tài đang thực hiện
Post Conditions	Người dùng xem danh sách sinh viên thực hiện đề tài thành công

Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đề tài đang thực hiện” (3) Hệ thống hiển thị danh sách đề tài đang thực hiện (4) Người dùng chọn vào đề tài muốn xem (5) Hệ thống hiển thị thông tin chi tiết đề tài (bao gồm thông tin nhóm sinh viên đang thực hiện đề tài)
Actors	Giáo vụ
Alternate Flow	
Exception Flow	

Bảng 3.33: Use Case Xem danh sách sinh viên thực hiện

3.4.4.6. Xem điểm chi tiết

Use case	Đăng nhập
Short Description	Người dùng có thể đăng nhập vào tài khoản thông qua tài khoản và mật khẩu đã tạo
Actors	Sinh viên, Giảng viên, Giáo vụ
Pre-Conditions	Thông tin đăng nhập phải tồn tại trong hệ thống.
Post Conditions	Người dùng đăng nhập thành công.
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đề tài đang thực hiện” (3) Hệ thống hiển thị danh sách đề tài đang thực hiện (4) Người dùng chọn vào đề tài muốn xem (5) Hệ thống hiển thị thông tin chi tiết đề tài (bao gồm thông tin nhóm sinh viên đang thực hiện đề tài) (6) Người dùng chọn vào biểu tượng “Mất” (7) Hệ thống hiển thị bảng điểm của sinh viên (A1)
Alternate Flow	(A1) Hệ thống hiển thị thông báo chưa có bảng điểm
Exception Flow	

Bảng 3.34. Use case Xem điểm chi tiết

3.4.4.7.Xóa người dùng

Use case	Xóa người dùng
Short Description	Giáo vụ xóa một tài khoản người dùng khỏi hệ thống.
Actors	Giáo vụ
Pre-Conditions	1. Giáo vụ đã đăng nhập thành công vào hệ thống 2. Tài khoản người dùng cần xóa tồn tại trong hệ thống.
Post Conditions	Người dùng đăng nhập thành công.
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đề tài đang thực hiện” (3) Hệ thống hiển thị danh sách đề tài đang thực hiện (4) Người dùng chọn vào đề tài muốn xem (5) Hệ thống hiển thị thông tin chi tiết đề tài (bao gồm thông tin nhóm sinh viên đang thực hiện đề tài) (6) Người dùng chọn vào biểu tượng “Mất” (7) Hệ thống hiển thị bảng điểm của sinh viên (A1)
Alternate Flow	(A1) Hệ thống hiển thị thông báo chưa có bảng điểm
Exception Flow	

Bảng 3.35.Use case Xóa người dùng

3.4.4.8.Thêm mới người dùng

Use case	Người dùng thêm mới người dùng
Short Description	Người dùng muốn thêm người dùng vào hệ thống
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống
Post Conditions	Người dùng thêm mới thành công

Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Sinh viên” (A1) (3) Hệ thống hiển thị danh sách người dùng (4) Người dùng chọn vào chức năng “Thêm người dùng” (5) Hệ thống hiển thị trang thêm mới người dùng (6) Người dùng điền các thông tin được yêu cầu (7) Người dùng chọn “Thêm mới” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	(A1) Người dùng chọn vào mục “Giảng viên”; tiếp tục bước 3
Exception Flow	(E1) Hệ thống thông báo lỗi và quay lại bước 5

Bảng 3.36. Use case Thêm mới người dùng

3.4.4.9. Thêm danh sách người dùng bằng file Excel

Use case	Người dùng thêm người dùng bằng file excel
Short Description	Người dùng muốn thêm mới nhiều người dùng bằng file excel
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống
Post Conditions	Người dùng thêm mới người dùng bằng file excel thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Sinh viên” (A1) (3) Hệ thống hiển thị danh sách người dùng (4) Người dùng chọn vào chức năng “Thêm người dùng” (5) Hệ thống hiển thị trang thêm mới người dùng (6) Người dùng tải lên file danh sách (7) Người dùng chọn “Tải lên” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	(A1) Người dùng chọn vào mục “Giảng viên” và tiếp tục bước 3
Exception Flow	(E1) Hệ thống thông báo lỗi

Bảng 3.37. Use case Thêm danh sách người dùng bằng file Excel

3.4.4.10. Thêm mới học kỳ

Use case	Người dùng thêm mới học kỳ
Short Description	Người dùng muốn tạo học kỳ mới
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống
Post Conditions	Người dùng tạo mới học kỳ thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Học kỳ” (3) Hệ thống hiển thị danh sách học kỳ (4) Người dùng chọn vào “Thêm học kỳ” (5) Hệ thống hiển thị Sidebar thêm học kỳ (6) Người dùng điền các thông tin cần thiết (7) Người dùng chọn vào “Tạo học kỳ” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (6)

Bảng 3.38. Use case Thêm mới học kỳ

3.4.4.11. Xóa học kỳ

Use case	Người dùng xóa Học kỳ
Short Description	Người dùng muốn xóa Học kỳ
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống Hệ thống có danh sách học kỳ
Post Conditions	Người dùng xóa học kỳ thành công

Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Học kỳ” (3) Hệ thống hiển thị danh sách Học kỳ (4) Người dùng chọn biểu tượng “Thùng rác” (5) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (3)

Bảng 3.39. Use case Xóa học kỳ

3.4.4.12. Cập nhật học kỳ

Use case	Người dùng cập nhật Học kỳ
Short Description	Người dùng muốn cập nhật Học kỳ
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống Đã có thông tin Học kỳ
Post Conditions	Người dùng cập nhật Học kỳ thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Học kỳ” (3) Hệ thống hiển thị danh sách Học kỳ (4) Người dùng chọn Học kỳ cập nhật (5) Hệ thống hiển thị Dialog cập nhật Học kỳ (6) Người dùng cập nhật các thông tin muốn thay đổi (7) Người dùng chọn vào “Xác nhận” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống thông báo lỗi, quay lại bước (6)

Bảng 3.40. Use case Cập nhật học kỳ

3.4.4.13. Thêm mới đợt đăng ký

Use case	Người dùng thêm mới đợt đăng ký
Short Description	Người dùng muốn tạo đợt đăng ký mới
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống
Post Conditions	Người dùng tạo mới đợt đăng ký thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đợt đăng ký” (3) Hệ thống hiển thị danh sách Đợt đăng ký (4) Người dùng chọn vào “Thêm Đợt đăng ký” (5) Hệ thống hiển thị Sidebar thêm đợt đăng ký (6) Người dùng điền các thông tin cần thiết (7) Người dùng chọn vào “Tạo đợt đăng ký” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (6)

Bảng 3.41. Use case Thêm mới đợt đăng ký

3.4.4.14. Xóa đợt đăng ký

Use case	Người dùng xóa Đợt đăng ký
Short Description	Người dùng muốn xóa Đợt đăng ký
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống Hệ thống có danh sách Đợt đăng ký
Post Conditions	Người dùng xóa Đợt đăng ký thành công

Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đợt đăng ký” (3) Hệ thống hiển thị danh sách Đợt đăng ký (4) Người dùng chọn biểu tượng “Thùng rác” (5) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (3)

Bảng 3.42. Use case Xóa đợt đăng ký

3.4.4.15. Cập nhật đợt đăng ký

Use case	Người dùng cập nhật Đợt đăng ký
Short Description	Người dùng muốn cập nhật Đợt đăng ký
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống Đã có thông tin đợt đăng ký
Post Conditions	Người dùng cập nhật Đợt đăng ký thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Đợt đăng ký” (3) Hệ thống hiển thị danh sách Đợt đăng ký (4) Người dùng chọn Đợt đăng ký cập nhật (5) Hệ thống hiển thị Dialog cập nhật Đợt đăng ký (6) Người dùng cập nhật các thông tin muốn thay đổi (7) Người dùng chọn vào “Xác nhận” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống thông báo lỗi, quay lại bước (6)

Bảng 3.43. Use case Cập nhật đợt đăng ký

3.4.4.16. Thêm mới phiếu đánh giá

Use case	Người dùng thêm mới phiếu đánh giá
Short Description	Người dùng muốn tạo phiếu đánh giá mới
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống
Post Conditions	Người dùng tạo mới phiếu đánh giá thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Phiếu đánh giá” (3) Hệ thống hiển thị danh sách Phiếu đánh giá (4) Người dùng chọn vào “Thêm Phiếu đánh giá” (5) Hệ thống hiển thị Sidebar thêm Phiếu đánh giá (6) Người dùng điền các thông tin cần thiết (7) Người dùng chọn vào “Tạo Phiếu đánh giá” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (6)

Bảng 3.44. Use case Thêm mới phiếu đánh giá

3.4.4.17. Xóa phiếu đánh giá

Use case	Người dùng xóa Phiếu đánh giá
Short Description	Người dùng muốn xóa Phiếu đánh giá
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống Hệ thống có danh sách Phiếu đánh giá
Post Conditions	Người dùng xóa Phiếu đánh giá thành công

Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Phiếu đánh giá” (3) Hệ thống hiển thị danh sách Phiếu đánh giá (4) Người dùng chọn biểu tượng “Thùng rác” (5) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (3)

Bảng 3.45. Use Case Xóa phiếu đánh giá

3.4.4.18. Thêm mới tiêu chí

Use case	Người dùng thêm mới tiêu chí
Short Description	Người dùng muốn thêm mới tiêu chí vào phiếu đánh giá
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống Phiếu đánh giá chưa có tiêu chí
Post Conditions	Người dùng thêm mới tiêu chí thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Phiếu đánh giá” (3) Hệ thống hiển thị danh sách Phiếu đánh giá (4) Người dùng chọn Phiếu đánh giá cần thêm tiêu chí (5) Hệ thống hiển thị Dialog thêm tiêu chí (6) Người dùng điền các thông tin cần thiết (7) Người dùng chọn vào “Tạo tiêu chí” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (6)

Bảng 3.46. Use case Thêm mới tiêu chí

3.4.4.19.Cập nhật tiêu chí

Use case	Người dùng cập nhật tiêu chí phiếu đánh giá
Short Description	Người dùng muốn cập nhật tiêu chí của phiếu đánh giá
Actors	Giáo vụ
Pre-Conditions	Người dùng đã đăng nhập vào hệ thống Phiếu đánh giá chưa được phân công
Post Conditions	Người dùng cập nhật tiêu chí thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Phiếu đánh giá” (3) Hệ thống hiển thị danh sách Phiếu đánh giá (4) Người dùng chọn Phiếu đánh giá cần cập nhật tiêu chí(5) Hệ thống hiển thị Dialog cập nhật tiêu chí (6) Người dùng cập nhật các tiêu chí muốn thay đổi (7) Người dùng chọn vào “Xác nhận” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống thông báo lỗi, quay lại bước (6)

Bảng 3.47.Use case Cập nhật tiêu chí

3.4.4.20.Thêm mới hội đồng

Use case	Người dùng thêm mới hội đồng
Short Description	Người dùng muốn tạo hội đồng mới
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống
Post Conditions	Người dùng tạo mới hội đồng thành công

Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Hội đồng” (3) Hệ thống hiển thị danh sách Hội đồng (4) Người dùng chọn vào “Thêm Hội đồng” (5) Hệ thống hiển thị Sidebar thêm Hội đồng (6) Người dùng điền các thông tin cần thiết (7) Người dùng chọn vào “Tạo Hội đồng” (8) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (6)

Bảng 3.48. Use case Thêm mới hội đồng

3.4.4.21.Xóa hội đồng

Use case	Người dùng xóa Hội đồng bảo vệ
Short Description	Người dùng muốn xóa Hội đồng bảo vệ
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống Hệ thống có danh sách hội đồng bảo vệ
Post Conditions	Người dùng xóa Hội đồng bảo vệ thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Hội đồng” (3) Hệ thống hiển thị danh sách Hội đồng (4) Người dùng chọn biểu tượng “Thùng rác” (5) Hệ thống xác thực thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (3)

Bảng 3.49. Use Case Xóa hội đồng

3.4.4.22. Gán đề tài vào hội đồng

Use case	Người dùng gán đề tài cho hội đồng bảo vệ
Short Description	Người dùng muốn gán đề tài cho hội đồng bảo vệ
Actors	Giáo vụ
Pre-Conditions	Người dùng phải đăng nhập vào hệ thống Đề tài đã cập nhật đủ các thông tin
Post Conditions	Người dùng gán đề tài cho hội đồng bảo vệ thành công
Main Flow	(1) Người dùng truy cập vào website (2) Người dùng chọn vào mục “Hội đồng” (3) Hệ thống hiển thị danh sách Hội đồng (4) Người dùng chọn vào biểu tượng “Bút chì” (5) Hệ thống hiển thị Dialog gán đề tài (6) Người dùng chọn các đề tài muốn thực hiện gán (7) Người dùng chọn vào “Xác nhận” (8) Hệ thống hiển thị Dialog nhập các thông tin cần thiết cho hội đồng bảo vệ (9) Người dùng điền các thông tin cần thiết (10) Người dùng chọn vào “Xác nhận” (11) Hệ thống xác nhận thông tin (E1)
Alternate Flow	
Exception Flow	(E1) Hệ thống hiển thị thông báo lỗi, quay lại bước (9)

Bảng 3.50. Use case Gán đề tài vào hội đồng

CHƯƠNG 4 : THIẾT KẾ PHẦN MỀM

4.1. MÔ TẢ CƠ SỞ DỮ LIỆU

4.1.1. Thông tin các trường trong bảng

Bảng 4.1. Thông tin các trường trong bảng Users

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
1	Users	Lưu trữ thông tin người dùng
1.1	email	Email người dùng
1.2	password	Mật khẩu người dùng
1.3	user_CCCD:	Căn cước công dân
1.4	user_name	Họ và tên người dùng
1.5	user_avatar	Ảnh đại diện của người dùng
1.6	user_date_of_birth	Ngày sinh của người dùng
1.7	user_phone	Số điện thoại của người dùng
1.8	user_permanent_address	Đại chỉ thường trú
1.9	user_temporary_address	Địa chỉ tạm trú
1.10	user_department	Phòng làm việc
1.11	user_faculty	Khoa của người dùng
1.12	user_major	Chuyên ngành của người dùng
1.13	role	Phân quyền của người dùng
1.14	user_status	Trạng thái tài khoản người dùng
1.15	user_average_grade	Điểm trung bình của người dùng
1.16	user_transcript	Bảng điểm của người dùng
1.17	user_id	ID người dùng
1.18	created_at	Ngày tạo người dùng
1.19	update_at	Ngày cập nhật người dùng

Bảng 4.2. Thông tin các trường trong bảng Topics

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
2	Topics	Lưu trữ thông tin đề tài
2.1	topic_registration_period	Đợt đăng ký
2.2	topic_title	Tiêu đề của đề tài
2.3	topic_description	Mô tả chi tiết đề tài
2.4	topic_category	Loại đề tài
2.5	topic_major	Chuyên ngành của đề tài
2.5	topic_creator	Người tạo đề tài
2.6	topic_max_members	Tối đa sinh viên thực hiện đề tài
2.7	topic_group_student	Nhóm sinh viên đăng ký thực hiện
2.8	topic_instructor	Người hướng dẫn đề tài
2.8	topic_reviewer	Người phản biện đề tài
2.9	topic_teacher_status	Trạng thái đề tài đối với giảng viên
2.10	topic_leader_status	Trạng thái đề tài đối với giáo vụ
2.11	topic_advisor_request	Đơn xin giảng viên hướng dẫn
2.12	topic_defense_request	Đơn xin nghiệm thu đề tài NCKH
2.13	topic_final_report	Báo cáo cuối cùng
2.14	rubric_instructor	Rubric chấm điểm dành cho giảng viên hướng dẫn
2.15	rubric_reviewer	Rubric chấm điểm dành cho giảng viên phản biện
2.16	rubric_assembly	Rubric chấm điểm dành cho hội đồng phản biện
2.17	topic_assembly	Hội đồng phản biện của đề tài
2.18	topic_room	Địa điểm phản biện
2.19	topic_time_start	Thời gian bắt đầu phản biện
2.20	topic_time_end	Thời gian kết thúc phản biện
2.21	topic_date	Ngày phản biện
2.22	topic_block	Trạng thái khóa đề tài

2.23	created_at	Ngày tạo đề tài
2.24	update_at	Ngày cập nhật đề tài

Bảng 4.3. Thông tin các trường trong bảng UserNotifications

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
3	UserNotifications	Lưu trữ thông báo của người dùng
3.1	user_notification_title	Tiêu đề thông báo
3.2	user_notification_sender	Người gửi thông báo
3.3	user_notification_recipient	Người nhận thông báo
3.4	user_notification_content	Nội dung thông báo
3.5	user_notification_type	Loại thông báo
3.6	user_notification_isRead	Trạng thái đã đọc của đề tài
3.7	user_notification_topic	Thể hiện thông báo thuộc đề tài nào
3.8	created_at	Ngày tạo thể loại đề tài
3.9	update_at	Ngày cập nhật thể loại đề tài

Bảng 4.4. Thông tin các trường trong bảng TopicCategory

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
4	TopicCategories	Lưu trữ thông tin loại đề tài
4.1	topic_category_title	Tên thể loại đề tài
4.2	topic_category_description	Mô tả thể loại đề tài
4.3	created_at	Ngày tạo thể loại đề tài
4.4	update_at	Ngày cập nhật thể loại đề tài

Bảng 4.5. Thông tin các trường trong bảng Semesters

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
5	Semesters	Lưu trữ thông tin học kỳ
5.1	school_year_start	Năm học bắt đầu

5.2	school_year_end	Năm học kết thúc
5.3	semester	Học kỳ
5.4	created_at	Ngày tạo học kỳ
5.5	update_at	Ngày cập nhật học kỳ

Bảng 4.6. Thông tin các trường trong bảng ScoreBoards

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
6	ScoreBoards	Lưu trữ thông tin bảng điểm
6.1	rubric_id	ID rubric chấm điểm
6.2	rubric_category	ID loại rubric
6.3	topic_id	ID đề tài
6.4	grader	Người chấm điểm
6.5	student_id	ID sinh viên được chấm điểm
6.6	rubric_student_evaluations	Danh sách các tiêu chí chấm điểm của rubric
6.7	total_score	Tổng điểm
6.8	created_at	Ngày tạo bảng điểm
6.9	update_at	Ngày cập nhật bảng điểm

Bảng 4.7. Thông tin các trường trong bảng RubricEvaluations

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
7	RubricEvaluations	Lưu trữ thông tin các tiêu chí chấm điểm của rubric
7.1	rubric_id	ID của rubric
7.2	serial	Thứ tự tiêu chí
7.3	evaluation_criteria	Tên tiêu chí đánh giá
7.4	grading_scale	Thang điểm
7.5	weight	Trọng số

7.6	note	Mô tả chi tiết tiêu chí
7.7	created_at	Ngày tạo tiêu chí
7.8	update_at	Ngày cập nhật tiêu chí

Bảng 4.8. Thông tin các trường trong bảng Rubrics

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
8	Rubrics	Lưu trữ thông tin các rubric chấm điểm
8.1	rubric_name	Tên rubric
8.2	rubric_category	Loại rubric
8.3	rubric_topic_category	Loại đề tài
8.4	rubric_evaluations	Danh sách tiêu chí đánh giá
8.5	rubric_note	Ghi chú mô tả chi tiết rubric
8.6	rubric_template	Mẫu rubric
8.7	created_at	Ngày tạo rubric
8.8	update_at	Ngày cập nhật rubric

Bảng 4.9. Thông tin các trường trong bảng RegistrationPeriods

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
9	RegistrationPeriods	Lưu trữ thông tin các đợt đăng ký
9.1	registration_period_semester	ID học kỳ
9.2	registration_period_start	Thời gian bắt đầu đợt đăng ký
9.3	registration_period_end	Thời gian kết thúc đợt đăng ký
9.4	registration_period_status	Trạng thái đợt đăng ký
9.5	created_at	Ngày tạo đợt đăng ký
9.6	update_at	Ngày cập nhật đợt đăng ký

Bảng 4.10. Thông tin các trường trong bảng Majors

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
-----	---------------------	-------------------------------

10	Majors	Lưu trữ thông tin các chuyên ngành
10.1	major_faculty	ID khoa
10.2	major_title	Tên chuyên ngành
10.3	major_description	Mô tả chi tiết chuyên ngành
10.4	created_at	Ngày tạo chuyên ngành
10.5	update_at	Ngày cập nhật chuyên ngành

Bảng 4.11. Thông tin các trường trong bảng GroupStudent

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
11	GroupStudents	Lưu trữ thông tin các nhóm sinh viên đăng ký thực hiện đề tài
11.1	group_name	Tên nhóm
11.2	group_member	Danh sách thành viên trong nhóm
11.3	group_max_member	Tối đa thành viên trong nhóm
11.4	created_at	Ngày tạo nhóm
11.5	update_at	Ngày cập nhật nhóm

Bảng 4.12. Thông tin các trường trong bảng Faculties

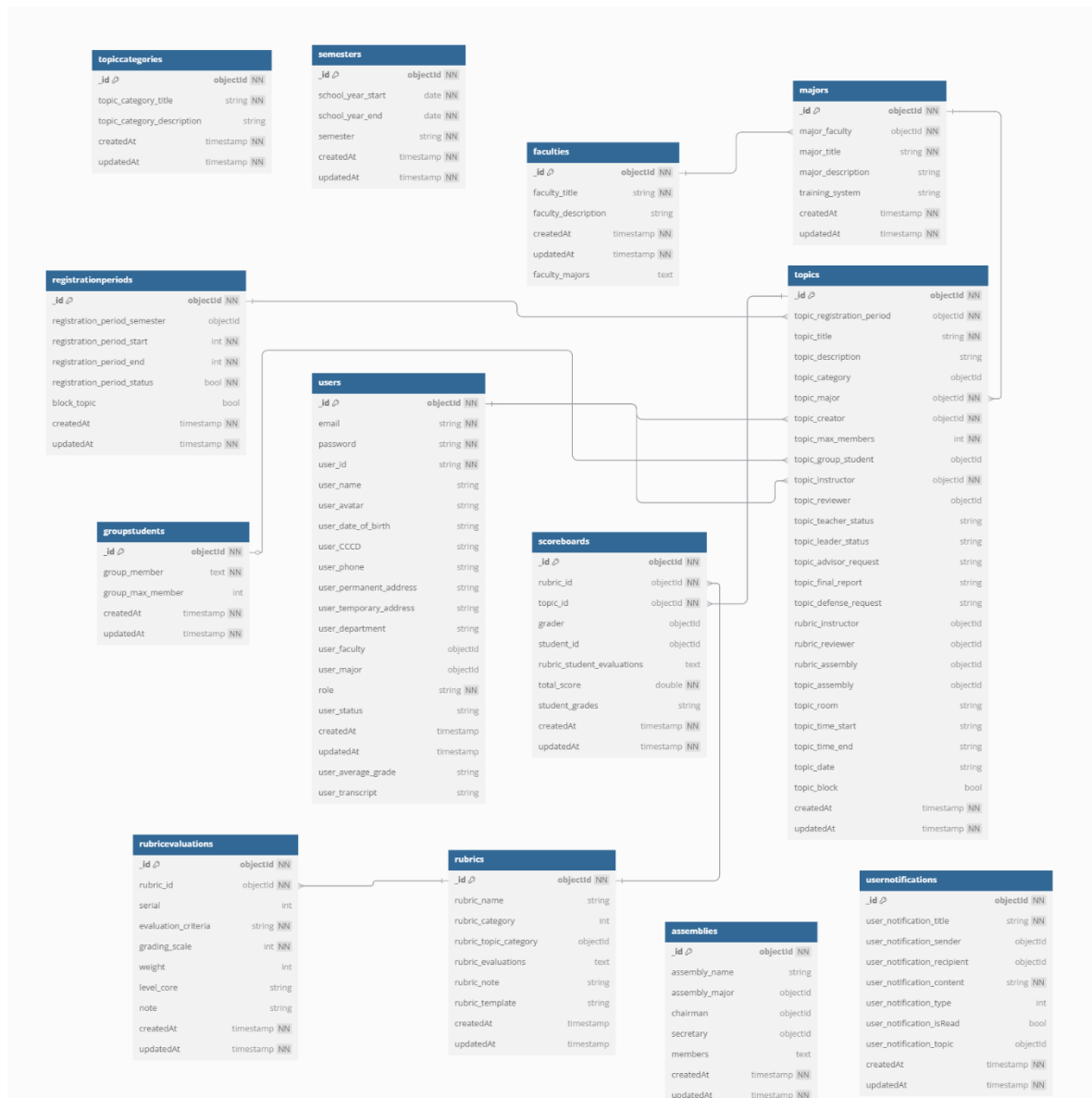
STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
12	Faculties	Lưu trữ thông tin khoa
12.1	faculty_title	Tên khoa
12.2	faculty_description	Mô tả chi tiết khoa
12.3	created_at	Ngày tạo khoa
12.4	update_at	Ngày cập nhật khoa

Bảng 4.13. Thông tin các trường trong bảng Assemblies

STT	Tên bảng/Tên trường	Mục đích bảng/Mục đích trường
-----	---------------------	-------------------------------

13	Assemblies	Lưu trữ thông tin hội đồng chấm điểm cho đề tài
13.1	assembly_name	Tên đề tài
13.2	assembly_major	ID chuyên ngành
13.3	chairman	Chủ tịch hội đồng
13.4	secretary	Thư ký hội đồng
13.4	members	Danh sách các ủy viên
13.5	created_at	Ngày tạo hội đồng

4.1.2. Mô hình ERD



4.2. Hiện thực thông qua MongoDB

4.2.1. Bảng Topics

topics		
_id	?	objectId NN
topic_registration_period	?	objectId NN
topic_title		string NN
topic_description		string
topic_category	?	objectId
topic_major	?	objectId NN
topic_creator	?	objectId NN
topic_max_members		int NN
topic_group_student	?	objectId
topic_instructor	?	objectId NN
topic_reviewer	?	objectId
topic_teacher_status		string
topic_leader_status		string
topic_advisor_request		string
topic_final_report		string
topic_defense_request		string
rubric_instructor	?	objectId
rubric_reviewer	?	objectId
rubric_assembly	?	objectId
topic_assembly	?	objectId
topic_room		string
topic_time_start		string
topic_time_end		string
topic_date		string
topic_block		bool
createdAt		timestamp NN
updatedAt		timestamp NN

Hình 4.2. Bảng Topics trên MongoDB

4.2.2. Bảng User

users	
_id 🔗	objectId NN
email	string NN
password	string NN
user_id	string NN
user_name	string
user_avatar	string
user_date_of_birth	string
user_CCCD	string
user_phone	string
user_permanent_address	string
user_temporary_address	string
user_department	string
user_faculty 🔗	objectId
user_major 🔗	objectId
role	string NN
user_status	string
createdAt	timestamp
updatedAt	timestamp
user_average_grade	string
user_transcript	string

Hình 4.3. Bảng Users trên MongoDB

4.2.3. Bảng UserNotifications

usertifications	
_id 🔗	objectId NN
user_notification_title	string NN
user_notification_sender 🔗	objectId
user_notification_recipient 🔗	objectId NN
user_notification_content	string NN
user_notification_type	int
user_notification_isRead	bool
user_notification_topic 🔗	objectId
createdAt	timestamp NN
updatedAt	timestamp NN

Hình 4.4. Bảng UserNotifications trên MongoDB

4.2.4. Bảng TopicCategories

topiccategories	
_id 🔗	objectId NN
topic_category_title	string NN
topic_category_description	string
createdAt	timestamp NN
updatedAt	timestamp NN


4.2.5. Bảng Semesters

semesters		
_id 🔗	objectId	NN
school_year_start	date	NN
school_year_end	date	NN
semester	string	NN
createdAt	timestamp	NN
updatedAt	timestamp	NN

4.2.6. Bảng ScoreBoards

scoreboards		
_id 🔗	objectId	NN
rubric_id 🔗	objectId	NN
topic_id 🔗	objectId	NN
grader 🔗	objectId	NN
student_id 🔗	objectId	NN
rubric_student_evaluations	text	
total_score	double	NN
student_grades	string	
createdAt	timestamp	NN
updatedAt	timestamp	NN

4.2.7. Bảng RubricEvaluations

rubricevaluations	
<u>_id</u> 	objectId NN
rubric_id 	objectId NN
serial	int
evaluation_criteria	string NN
grading_scale	int NN
weight	int
level_core	string
note	string
createdAt	timestamp NN
updatedAt	timestamp NN

4.2.8. Bảng Rubrics

rubrics	
_id 🔗	objectId NN
rubric_name	string
rubric_category	int
rubric_topic_category 🔗	objectId
rubric_evaluations	text
rubric_note	string
rubric_template	string
createdAt	timestamp
updatedAt	timestamp

4.2.9. Bảng RegistrationPeriods

registrationperiods	
_id 🔗	objectId NN
registration_period_semester 🔗	objectId NN
registration_period_start	int NN
registration_period_end	int NN
registration_period_status	bool NN
block_topic	bool
createdAt	timestamp NN
updatedAt	timestamp NN

4.2.10. Bảng Majors

majors		
_id 🔗	objectId	NN
major_faculty 🔗	objectId	NN
major_title	string	NN
major_description	string	
training_system	string	
createdAt	timestamp	NN
updatedAt	timestamp	NN

4.2.11. Bảng GroupStudents

groupstudents		
_id 🔗	objectId	NN
group_member	text	NN
group_max_member	int	
createdAt	timestamp	NN
updatedAt	timestamp	NN

4.2.12. Bảng Faculties

faculties	
_id 🔗	objectId NN
faculty_title	string NN
faculty_description	string
createdAt	timestamp NN
updatedAt	timestamp NN
faculty_majors	text

4.2.13. Bảng Assemblies

assemblies	
_id 🔗	objectId NN
assembly_name	string
assembly_major 🔗	objectId NN
chairman 🔗	objectId NN
secretary 🔗	objectId NN
members	text
createdAt	timestamp NN
updatedAt	timestamp NN

CHƯƠNG 5 : TỔNG QUAN HỆ THỐNG

5.1. VAI TRÒ CỦA ỨNG DỤNG

5.1.1. Mô hình tổng quan của ứng dụng

Hệ thống Quản lý hoạt động nghiên cứu khoa học (NCKH) của sinh viên, sử dụng công nghệ NodeJS (NestJS) và ReactJS, là một giải pháp công nghệ hiện đại. Hệ thống này giúp cải thiện quy trình quản lý, theo dõi tiến độ và tương tác liên quan đến các đề tài NCKH của sinh viên. Hệ thống được xây dựng với 2 nhóm ứng dụng chính là:

- Nhóm ứng dụng tương tác với người dùng - các ứng dụng web (dành cho sinh viên, giảng viên, và giáo vụ).
- Nhóm ứng dụng phục vụ cho các ứng dụng tương tác người dùng – server (API backend).

Sử dụng NodeJS, một nền tảng JavaScript hiệu suất cao, hệ thống này cho phép xử lý đồng bộ và không đồng bộ hiệu quả, phù hợp với nhu cầu xử lý dữ liệu đa dạng và các tác vụ phức tạp thường gặp trong quản lý hoạt động NCKH. NestJS, một framework của NodeJS, mang đến cấu trúc rõ ràng và mạch lạc, hỗ trợ tốt cho việc xây dựng các dịch vụ back-end linh hoạt và mạnh mẽ, đảm bảo quy trình quản lý NCKH được vận hành trơn tru.

Ở phía giao diện người dùng, ReactJS được sử dụng để xây dựng một giao diện trực quan và thân thiện, giúp sinh viên và giảng viên dễ dàng thao tác. Kết hợp với công nghệ này, REST API cho phép hệ thống quản lý hoạt động NCKH tương tác mượt mà và linh hoạt với các dịch vụ back-end. Điều này cung cấp khả năng truy xuất và quản lý thông tin đề tài NCKH (như tiến độ, báo cáo, đánh giá) một cách nhanh chóng và chính xác.

Nhờ vào sự kết hợp của NodeJS và ReactJS, hệ thống không chỉ đảm bảo hiệu suất cao trong việc xử lý dữ liệu mà còn mang lại trải nghiệm người dùng ưu việt. Người dùng (sinh viên, giảng viên, giáo vụ) có thể dễ dàng truy cập, đăng

ký đề tài, nộp báo cáo, cập nhật thông tin, theo dõi tiến độ, và thực hiện các tương tác khác một cách linh hoạt và hiệu quả. Sự kết hợp này cũng giúp đơn giản hóa quy trình quản lý NCKH, tạo điều kiện phát triển, cải thiện khả năng bảo trì và nâng cấp hệ thống, đồng thời mở rộng khả năng tích hợp với các công nghệ và dịch vụ khác trong tương lai để phục vụ tốt hơn cho hoạt động nghiên cứu khoa học tại khoa/trường.

5.1.2. Giới thiệu các ứng dụng

Hệ thống có thể được phân chia thành hai nhóm chính: nhóm ứng dụng tương tác trực tiếp với người dùng (client) và nhóm ứng dụng phục vụ cho các ứng dụng người dùng (server). Nếu ta tiếp tục phân loại theo mục đích và đối tượng sử dụng, các ứng dụng trong hệ thống có thể được phân chia như sau:

1. Ứng dụng phục vụ cho Sinh viên:

- **Ứng dụng web dành cho Sinh viên (Client-side):** Xây dựng bằng ngôn ngữ JavaScript chạy trên trình duyệt, sử dụng thư viện ReactJS để tạo giao diện người dùng tương tác. Sinh viên sử dụng ứng dụng này để tìm kiếm thông tin đề tài, đăng ký/đề xuất đề tài, nộp báo cáo tiến độ, xem phản hồi và kết quả.
- **Ứng dụng server phục vụ cho ứng dụng web của Sinh viên (Server-side API):** Phát triển bằng ngôn ngữ JavaScript trên nền tảng NodeJS (sử dụng framework NestJS). Cung cấp các API để xử lý logic nghiệp vụ, xác thực, tương tác với cơ sở dữ liệu liên quan đến các hoạt động NCKH của sinh viên.

2. Ứng dụng phục vụ cho Giảng viên:

- **Ứng dụng web dành cho Giảng viên (Client-side):** Xây dựng bằng ngôn ngữ JavaScript chạy trên trình duyệt, sử dụng thư viện ReactJS. Giảng viên sử dụng ứng dụng này để quản lý các đề tài mình hướng dẫn/phản biện, đề xuất đề tài mới, duyệt đăng ký của sinh viên, nhận xét, chấm điểm và

theo dõi tiến độ.

- **Ứng dụng server phục vụ cho ứng dụng web của Giảng viên (Server-side API):** Phát triển bằng ngôn ngữ JavaScript trên nền tảng NodeJS (sử dụng framework NestJS). Cung cấp các API để xử lý logic nghiệp vụ, xác thực, tương tác với cơ sở dữ liệu liên quan đến vai trò của giảng viên trong hoạt động NCKH.

3. Ứng dụng phục vụ cho Giáo vụ (Quản trị viên hệ thống - Admin):

- **Ứng dụng web dành cho Giáo vụ (Client-side):** Xây dựng bằng ngôn ngữ JavaScript chạy trên trình duyệt, sử dụng thư viện ReactJS. Giáo vụ sử dụng ứng dụng này để quản lý toàn bộ hệ thống như quản lý người dùng (sinh viên, giảng viên), quản lý các đợt đăng ký NCKH, quản lý danh mục đề tài, quản lý chuyên ngành, khoa, cấu hình hệ thống và xem báo cáo tổng hợp.
- **Ứng dụng server phục vụ cho ứng dụng web của Giáo vụ (Server-side API):** Phát triển bằng ngôn ngữ JavaScript trên nền tảng NodeJS (sử dụng framework NestJS). Cung cấp các API quản trị, xử lý logic nghiệp vụ, xác thực, tương tác với cơ sở dữ liệu cho các chức năng quản trị hệ thống NCKH.

5.2. Ứng dụng web dành cho sinh viên

5.2.1. Công nghệ sử dụng

Ứng dụng được xây dựng dựa trên ngôn ngữ lập trình JavaScript và giao diện được dựng lên bởi công nghệ ReactJS.

- Sử dụng thư viện axios (một thư viện HTTP Client dựa trên Promise) để xây dựng thư viện API riêng của website axiosClient với việc trong header của request sẽ được bổ sung thêm thuộc tính Authorization cần thiết cho

những request cần mang tính xác thực ở server. Và thư viện axios cũng hỗ trợ đầy đủ các lệnh các API cơ bản như GET, POST, PUT, DELETE.

- Sử dụng Tailwind CSS: Được sử dụng để xây dựng giao diện người dùng của ứng dụng. Tailwind CSS là một thư viện CSS mạnh mẽ, được thiết kế để tạo ra giao diện người dùng linh hoạt, dễ bảo trì và tùy chỉnh.
- Sử dụng thư viện react-toast-notifications để thông báo các thông tin.

5.2.2. Mô tả luồng hoạt động

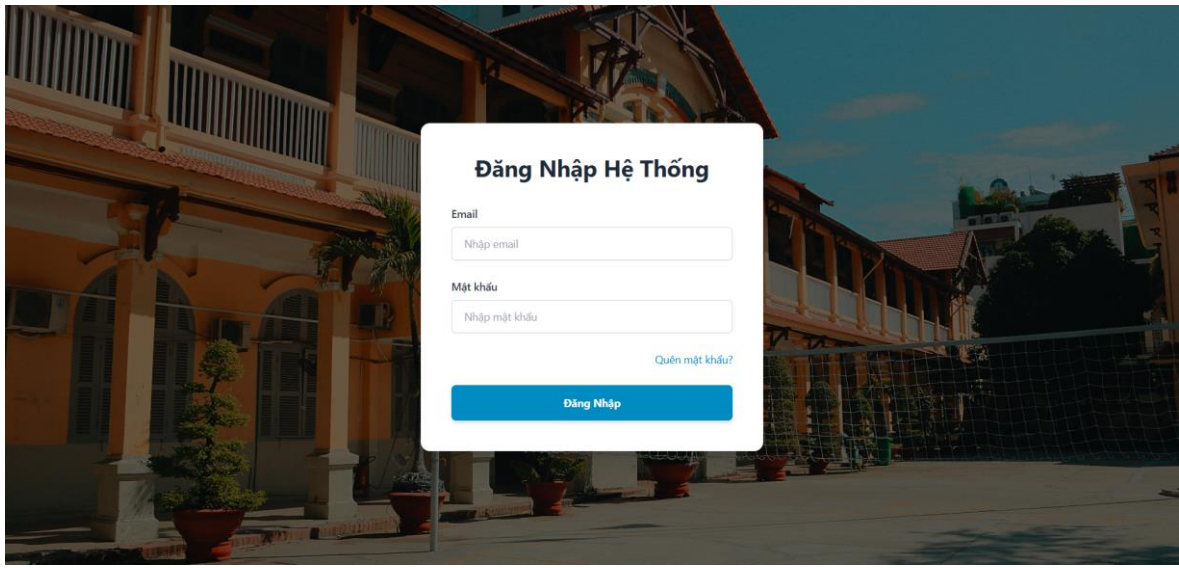
5.2.2.1. Luồng đăng nhập

Mô tả : Sinh viên đăng nhập email và password do hệ thống cung cấp sẵn và nhấn nút đăng nhập.

Giao diện:

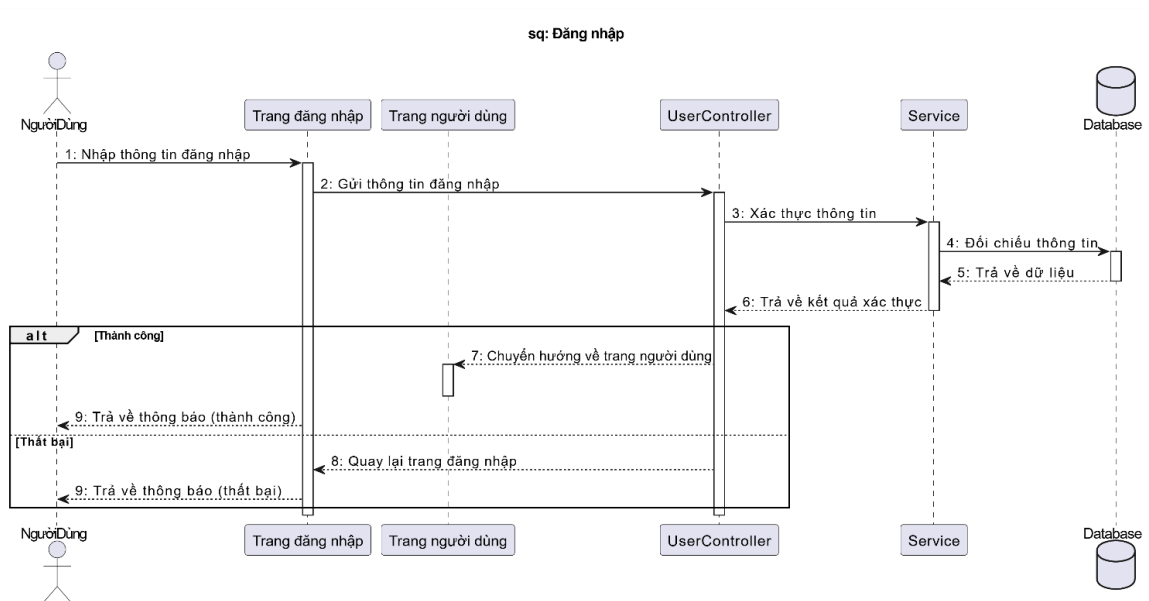


Hình 5.1. Giao diện trang chủ khi sinh viên chưa đăng nhập



Hình 5.2: Giao diện đăng nhập

Sơ đồ luồng đăng nhập



Hình 5.3: Sơ đồ luồng đăng nhập

5.2.2.2. Luồng đề xuất đề tài

Mô tả:

- Sinh viên có các ý tưởng đề tài muốn đề xuất và mong muốn được giảng viên hướng dẫn để thực hiện hoạt động nghiên cứu khoa học

(NCKH).

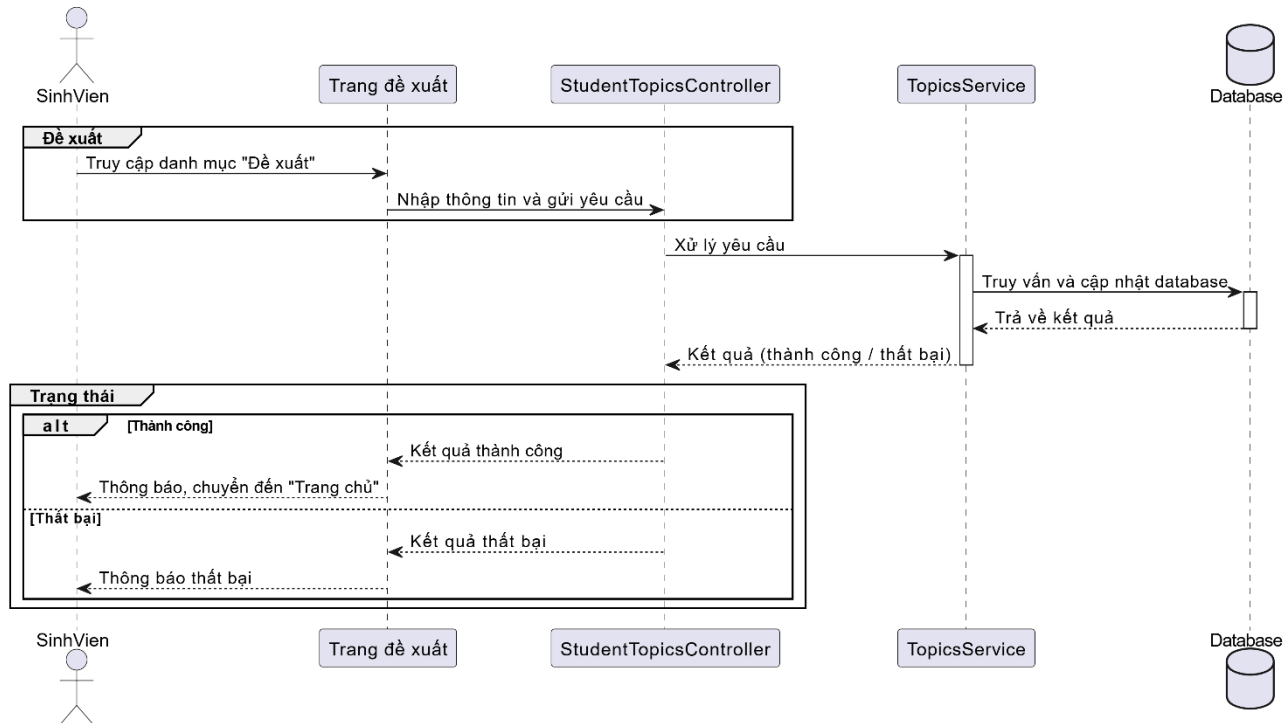
- Sinh viên có thể đăng kí nhóm gồm nhiều thành viên để thực hiện hoạt động nghiên cứu khoa học (NCKH).

Giao diện:

Hình 5.4: Giao diện Form mẫu đăng ký đề tài

- Bước 1: Sinh viên nhập các thông tin theo biểu mẫu để nộp ý tưởng làm đề tài lên giảng viên để xác nhận.
- Bước 2: Sinh viên có thể chọn các sinh viên phù hợp với bản thân để làm chung đề tài trong học kì này.
- Bước 3: Sau khi đăng kí đề tài thành công thì đề tài sẽ được gửi trực tiếp lên giáo viên hướng dẫn để xem xét.

Sơ đồ luồng đề xuất đề tài:

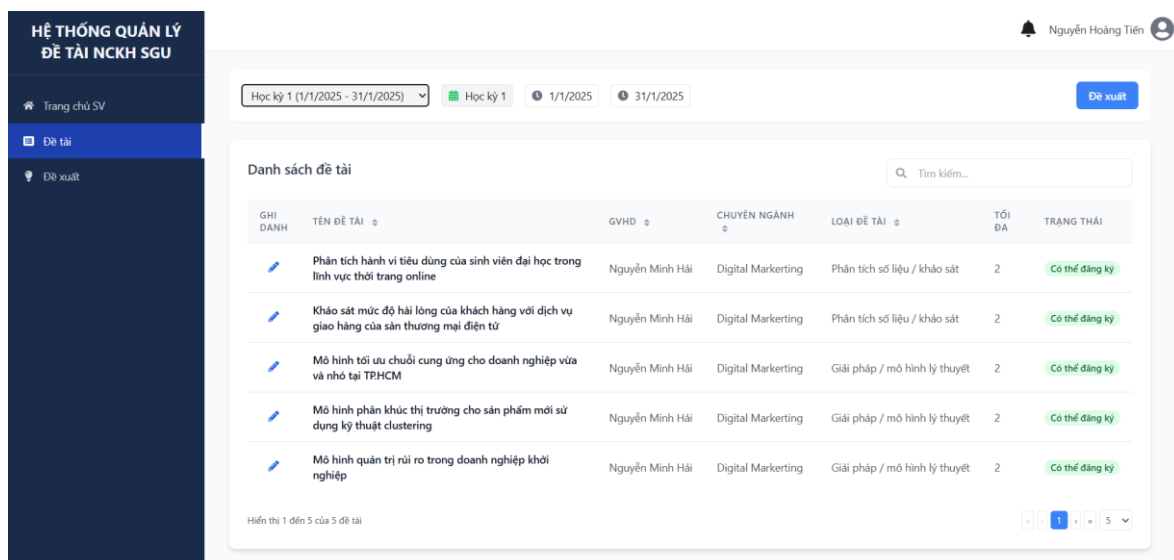


Hình 5.5: Lược đồ mô tả trình tự đề xuất đề tài của sinh viên

5.2.2.3. Luồng ghi danh đề tài

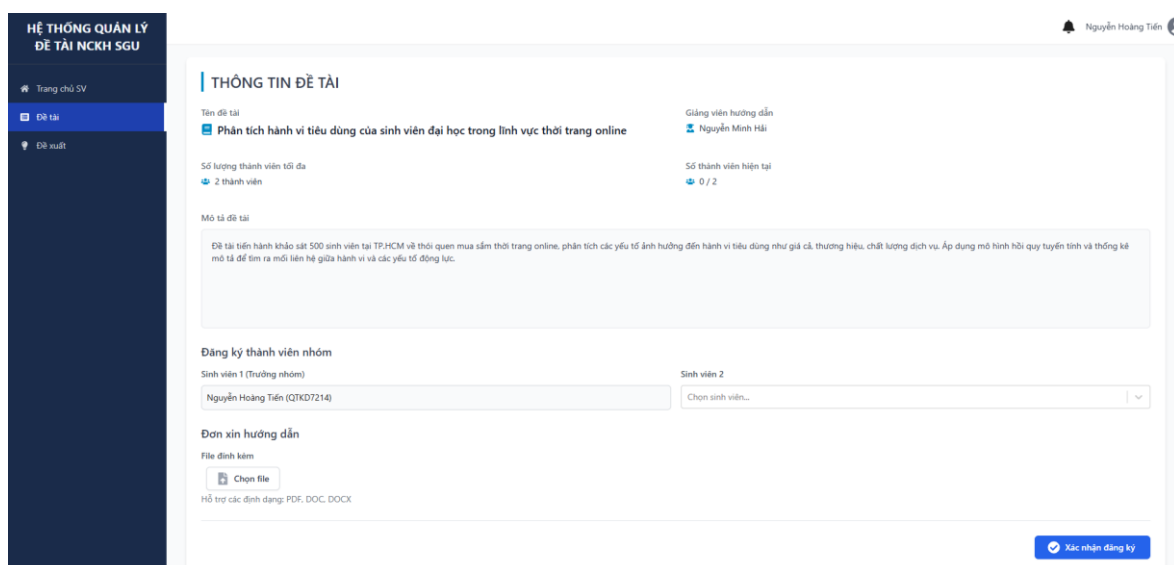
Mô tả: Sinh viên chưa có ý tưởng về đề tài nghiên cứu khoa học. Sinh viên có thể vào tham khảo các đề tài được các giảng viên đề xuất và có thể chọn một đề tài nghiên cứu khoa học phù hợp để thực hiện.

Giao diện:



Hình 5.6. Giao diện danh sách đề tài đang mở trong đợt đăng ký

- Bước 1: Ở giao diện này sinh viên có thể xem hết tất cả các đề tài của các giảng viên đang có trong đợt đăng ký nghiên cứu khoa học này.
- Bước 2: Sinh viên chọn vào biểu tượng cây bút để xem chi tiết đề tài cũng như các yêu cầu của các giảng viên.
- Bước 3: Sau đó giao diện chi tiết thông tin đề tài sẽ hiện thị lên.



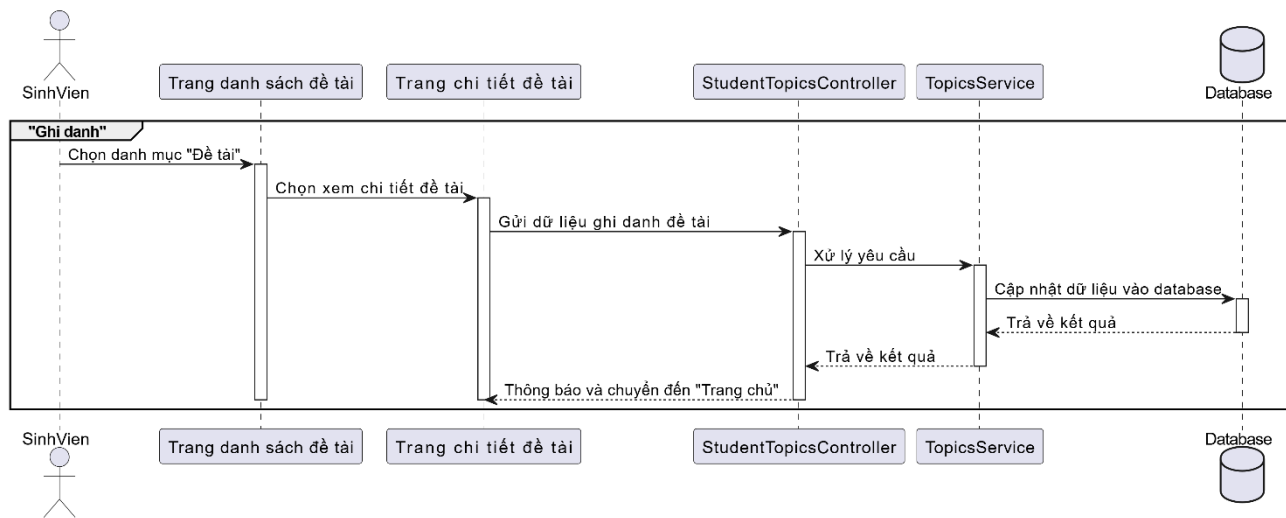
Hình 5.7: Giao diện thông tin chi tiết đề tài

- Bước 1: Ở màn hình này hiện thị tất cả các thông tin cơ bản của một đề tài và yêu cầu chi tiết khi đăng ký đề tài như: số lượng sinh viên tối đa thực hiện đề tài,

Chuyên ngành thực hiện đề tài ,

-Bước 2: Sau khi tham khảo đề tài và sinh viên có mong muốn thực hiện đề tài này thì tiến hành chọn thành viên làm cùng nếu cần, sau đó nhấn nút “Đăng kí đề tài” để gửi thông tin nhóm sinh viên đăng kí cũng như thông báo cho giảng viên hướng dẫn đề tài.

Sơ đồ luồng ghi danh đề tài:

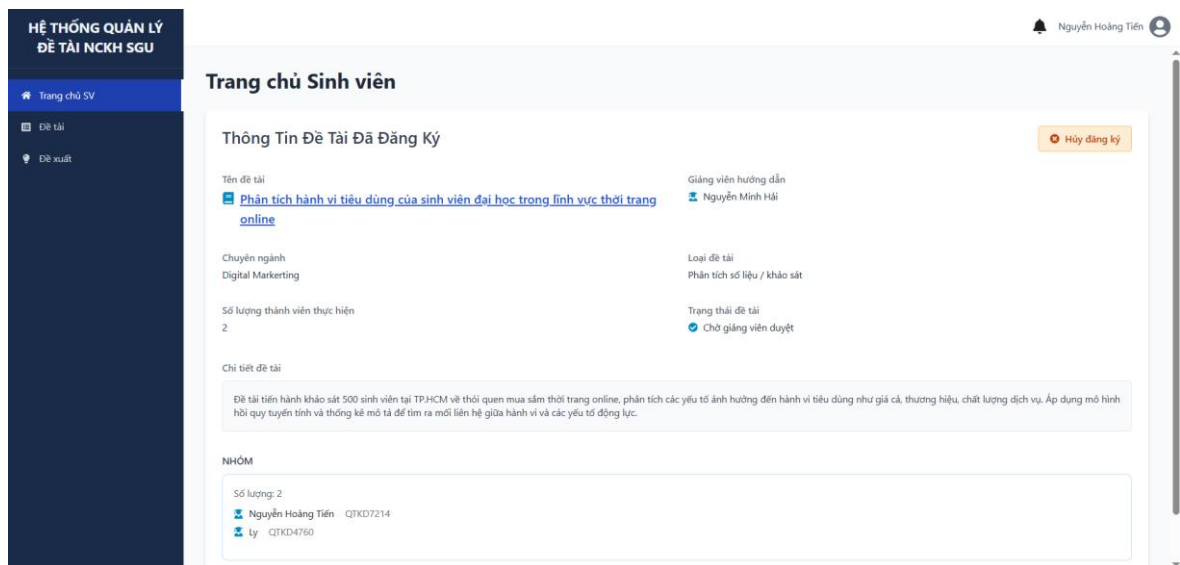


Hình 5.8: Lược đồ mô tả trình tự đề xuất đề tài của sinh viên

5.2.2.4. Luồng hủy ghi danh đề tài

Mô tả: Sinh viên thực hiện hủy đề tài đã đăng ký, chỉ được phép hủy ghi danh đề tài trước khi được giảng viên hướng dẫn xác nhận nhóm sinh viên thực hiện.

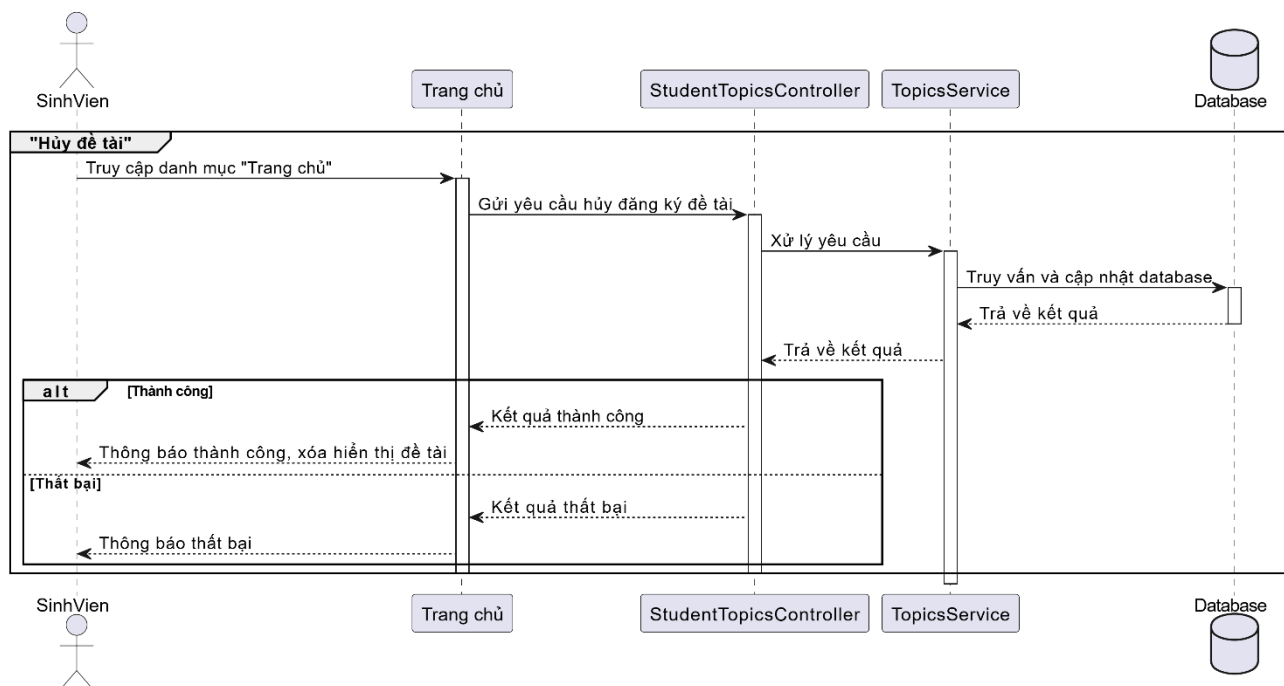
Giao diện:



Hình 5.9: Giao diện hủy ghi danh đề tài đã được đăng ký

- Bước 1: Ở đây sinh viên tìm đến nút “Hủy đề tài” để hủy ghi danh đề tài nếu không muốn thực hiện đề tài nữa
- Bước 2: Buton “Hủy đề tài” chỉ hiển thị khi đề tài chưa được giảng viên hướng dẫn xác nhận nhóm thực hiện đề tài

Sơ đồ luồng hủy ghi danh đề tài:



Hình 5.10: Sơ đồ mô tả trình tự hủy đề tài của sinh viên

5.2.2.5. Luồng cập nhật thông tin bổ sung đề tài

Mô tả:

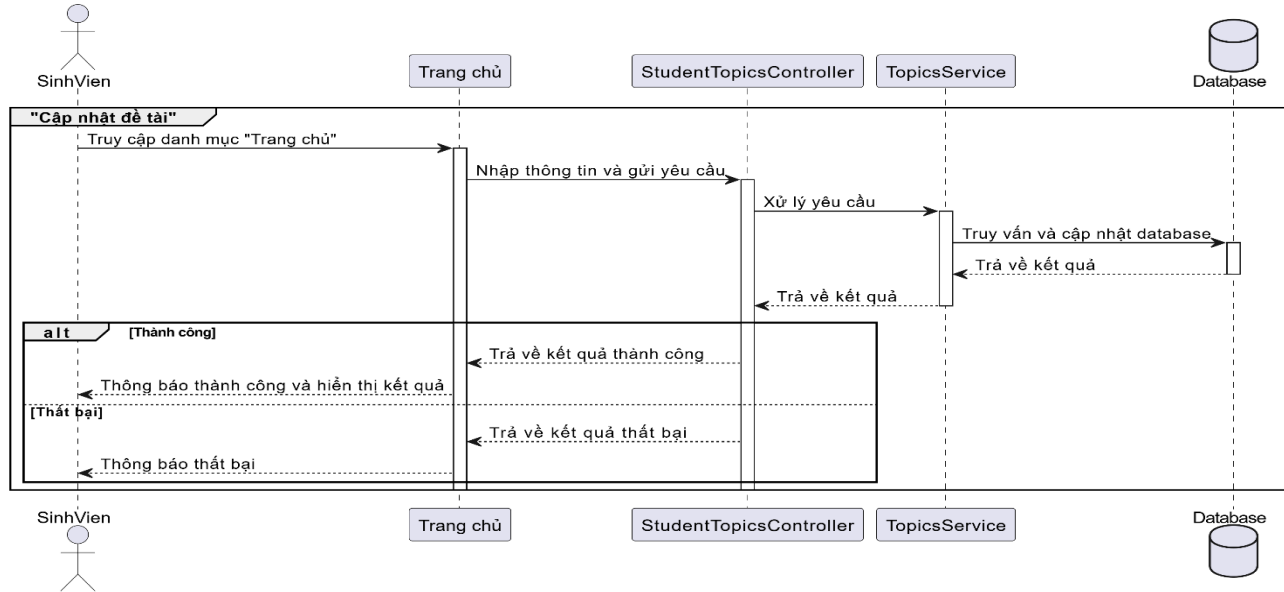
- Sau khi sinh viên được giảng viên hướng dẫn chấp nhận để thực hiện đề tài nghiên cứu khoa học .
- Ở từng giai đoạn trong quá trình thực hiện đề tài, giảng viên hướng dẫn hoặc giáo vụ sẽ gửi thông báo đến nhóm sinh viên thực hiện để cập nhật các thông tin/tài liệu quan trọng cần thiết cho đề tài.

Giao diện:

Hình 5.11: Giao diện cập nhật thông tin bổ sung đề tài

- Bước 1: Sinh viên có thể tải xuống mẫu tài liệu ở nút tải xuống tương ứng để tải về file mẫu
- Bước 2: Sinh viên tìm đến danh mục tài liệu quan trọng và nhấn nút “Chọn file” để chọn file cần thiết để tải lên.
- Bước 3: Sau khi tải từ máy lên thì nhấn nút “Tải lên” để cập nhật file.

Sơ đồ luồng cập nhật thêm các thông tin bổ sung:



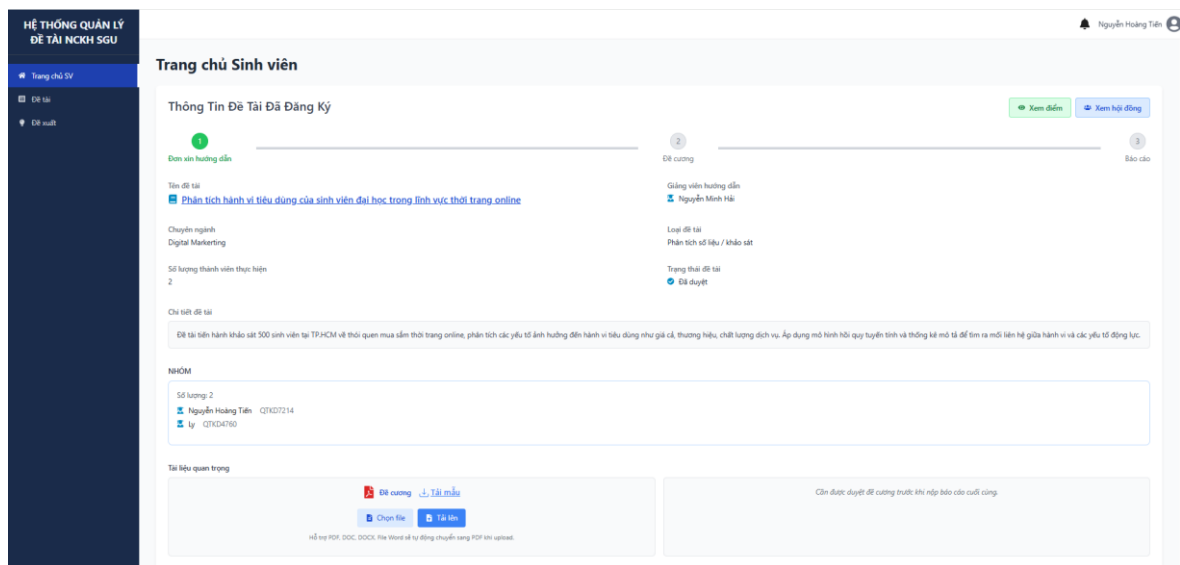
Hình 5.12: Lược đồ mô tả trình tự cập nhật bổ sung thông tin

5.2.2.6. Luồng xem thông tin chi tiết đề tài đã đăng ký thực hiện

Mô tả:

- Sau khi sinh viên được giảng viên hướng dẫn chấp nhận để thực hiện đề tài nghiên cứu khoa học.
- Sinh viên có thể xem chi tiết đề tài mà mình đang thực hiện và các thông tin liên quan đề đề tài như: bảng điểm, hội đồng phản biện,... ở từng giai đoạn thực hiện đề tài.

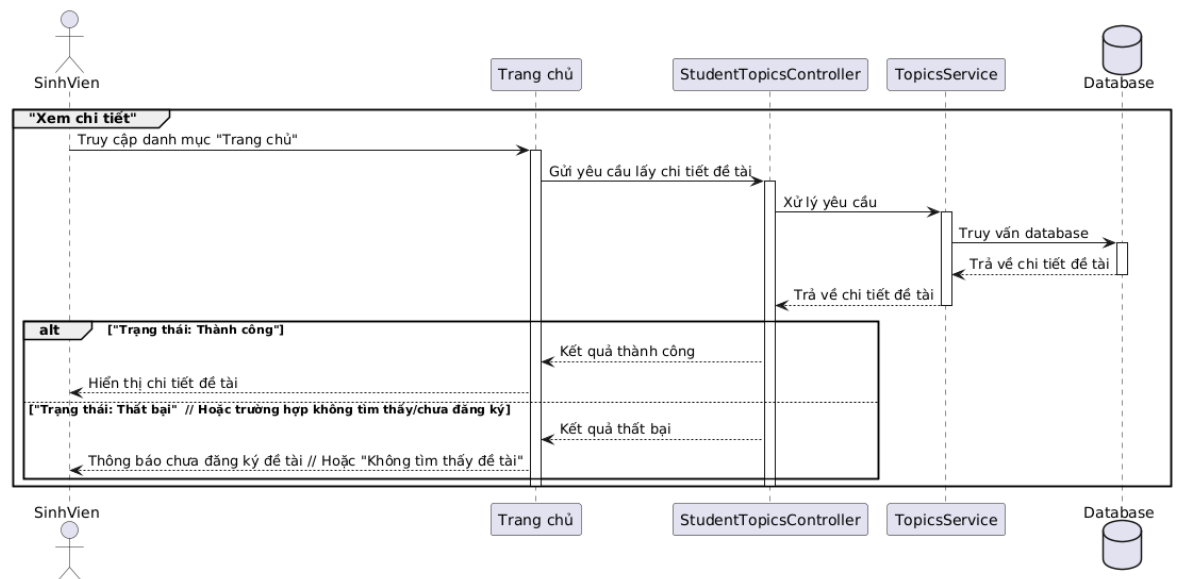
Giao diện:



Hình 5.13: Giao diện xem thông tin chi tiết đề tài đã đăng ký thực hiện

- Bước 1: Sinh viên có thể nhấn nút “Xem điểm” để xem điểm giảng viên đã đánh giá cho mình.
- Bước 2: Sinh viên có thể nhấn nút “Xem hội đồng” để xem thông tin chi tiết hội đồng phản biện.

Sơ đồ luồng xem chi tiết đề tài đã đăng ký:



Hình 5.14: Lược đồ mô tả trình tự xem chi tiết đề tài đã đăng ký

CHƯƠNG 6 : KIỂM THỬ HỆ THỐNG

6.1.MỤC ĐÍCH KIỂM THỬ

- Mục tiêu của quá trình kiểm thử là để xác nhận tính đúng đắn, độ tin cậy và bảo mật của hệ thống Quản lý Hoạt động Nghiên cứu Khoa học của Sinh viên, qua đó:
- Đánh giá toàn diện việc hệ thống có tuân thủ các yêu cầu nghiệp vụ liên quan đến quản lý Nghiên cứu Khoa học đã định trước hay không.
- Kiểm tra các chức năng chính như đăng ký/đề xuất đề tài Nghiên cứu Khoa học, đăng nhập, quản lý hồ sơ cá nhân của sinh viên và giảng viên, cũng như các chức năng quản trị của giáo vụ.
- Xác định và loại bỏ các lỗi có thể ảnh hưởng đến chất lượng, hiệu suất và trải nghiệm người dùng của hệ thống quản lý Nghiên cứu Khoa học.

6.2. PHẠM VI KIỂM TRA

- Kế hoạch kiểm thử sẽ tập trung vào các khía cạnh sau của hệ thống quản lý NCKH:
 - Chức năng quản lý vòng đời của một đề tài NCKH, từ khâu tạo mới, đề xuất, duyệt, đến cập nhật thông tin, báo cáo tiến độ và hoàn thành.
 - Quy trình phê duyệt đề tài NCKH bởi giảng viên hướng dẫn, (nếu có) giảng viên phản biện và các cấp quản lý (ví dụ: giáo vụ, hội đồng khoa học).
 - Khả năng theo dõi tiến độ thực hiện và quản lý các mốc thời gian quan trọng của đề tài NCKH.
 - Chức năng tương tác (ví dụ: bình luận, trao đổi, phản hồi) giữa sinh viên và giảng viên liên quan đến từng đề tài NCKH.
 - Các chức năng quản trị hệ thống của giáo vụ (quản lý người dùng, quản lý học kỳ, đợt đăng ký, v.v.).

6.3. CHIẾN LƯỢC KIỂM THỬ ÁP DỤNG

Quá trình kiểm thử sẽ bao gồm các chiến lược sau:

- **Kiểm Thử Chức Năng (Functional Testing):** Đánh giá mọi chức năng cốt lõi của hệ thống quản lý NCKH, đảm bảo chúng hoạt động đúng theo đặc tả yêu cầu.
- **Kiểm Thử Giao Diện Người Dùng (UI Testing):** Đảm bảo giao diện người dùng (UI) trực quan, thân thiện, dễ sử dụng và dễ điều hướng cho tất cả các đối tượng người dùng (sinh viên, giảng viên, giáo vụ). Kiểm tra tính nhất quán của thiết kế.
- **Kiểm Thử Bảo Mật (Security Testing):** Xác minh tính an toàn của thông tin cá nhân, dữ liệu đề tài NCKH, và các tài liệu liên quan. Đánh giá khả năng chống lại các mối đe dọa bảo mật phổ biến và đảm bảo phân quyền truy cập chính xác.
- **Kiểm Thử Tính Khả Dụng (Usability Testing):** Đánh giá mức độ dễ dàng và hiệu quả khi người dùng tương tác với hệ thống để hoàn thành các tác vụ NCKH.

6.4. CÁC PHƯƠNG PHÁP KIỂM THỬ

Các phương pháp kiểm thử cụ thể bao gồm:

- **Unit Testing:** Phân tích và kiểm tra các thành phần nhỏ nhất của hệ thống như các hàm, lớp và thành phần giao diện.
- **Integration Testing:** Đánh giá sự tương tác và truyền dữ liệu giữa các module và lớp đối tượng.
- **System Testing:** Kiểm thử chức năng toàn diện của hệ thống, bao gồm cả hiệu suất và độ ổn định.
- **Black-box Testing:** Kiểm thử dựa trên yêu cầu và đặc tả chức năng mà không cần quan tâm đến cấu trúc bên trong.
- **White-box Testing:** Kiểm thử cấu trúc bên trong của mã nguồn, dựa trên kiến thức về logic và cấu trúc mã nguồn.
- **Boundary Testing:** Kiểm tra các giới hạn của dữ liệu đầu vào như số lượng ký tự của tên đề tài NCKH.

6.5. KIỂM THỬ CHỨC NĂNG

Kiểm thử chức năng chính là một bước quan trọng trong quá trình xây dựng và phát triển trang web, nhằm đảm bảo trang web hoạt động đúng, ổn định và đáp ứng đầy đủ các yêu cầu đã đề cập. Tại Chương Kiểm thử và đánh giá này, nhóm sẽ trình bày quá trình kiểm thử chức năng cho trang web nghiên cứu khoa học cho sinh viên và đánh giá mức độ hoàn thiện của các chức năng.

Các trường hợp kiểm thử (Test case) được xây dựng dựa trên các chức năng đã phân tích từ giai đoạn thiết kế và yêu cầu người dùng. Kiểm thử được thực hiện theo phương pháp Kiểm thử hộp đen (Black-box testing), kiểm thử thủ công giao diện người dùng để đảm bảo các chức năng hoạt động ổn định.

6.5.1. Kiểm thử chức năng Đăng ký, Đăng nhập

Mã Test case	Mục đích kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
DN_1	Đăng nhập thành công với tài khoản hợp lệ (Sinh viên).	1. Mở trang đăng nhập. 2. Nhập email/username SV hợp lệ. 3. Nhập mật khẩu SV hợp lệ. 4. Nhấn nút "Đăng nhập".	Chuyển hướng đến trang chủ của Sinh viên. Hiển thị thông báo đăng nhập thành công.	PASS
DN_2	Đăng nhập thành công với tài khoản hợp lệ (Giảng viên).	1. Mở trang đăng nhập. 2. Nhập email/username	Chuyển hướng đến trang chủ của Giảng viên. Hiển thị thông báo đăng nhập thành	PASS

		GV hợp lệ. 3. Nhập mật khẩu GV hợp lệ. 4. Nhấn nút "Đăng nhập".	công.	
DN_3	Đăng nhập thành công với tài khoản hợp lệ (Giáo vụ).	1. Mở trang đăng nhập. 2. Nhập email/username Giáo vụ hợp lệ. 3. Nhập mật khẩu Giáo vụ hợp lệ. 4. Nhấn nút "Đăng nhập".	Chuyển hướng đến trang quản trị của Giáo vụ. Hiện thị thông báo đăng nhập thành công.	PASS
DN_4	Đăng nhập thất bại - Mật khẩu sai.	1. Mở trang đăng nhập. 2. Nhập email/username hợp lệ. 3. Nhập mật khẩu sai. 4. Nhấn nút "Đăng nhập".	Hiện thị thông báo lỗi "Sai tên đăng nhập hoặc mật khẩu". Giữ nguyên ở trang đăng nhập.	PASS
DN_5	Đăng nhập thất bại - Email/Username sai.	1. Mở trang đăng nhập. 2. Nhập email/username không tồn tại. 3. Nhập mật khẩu	Hiện thị thông báo lỗi "Sai tên đăng nhập hoặc mật khẩu". Giữ nguyên ở trang đăng nhập.	PASS

		bất kỳ. 4. Nhấn nút "Đăng nhập".		
DN_6	Đăng nhập thất bại - Bỏ trống Email/Username.	1. Mở trang đăng nhập. 2. Để trống trường email/username. 3. Nhập mật khẩu. 4. Nhấn nút "Đăng nhập".	Hiển thị thông báo yêu cầu nhập Email/Username.	PASS
DN_7	Đăng nhập thất bại - Bỏ trống Mật khẩu.	1. Mở trang đăng nhập. 2. Nhập email/username hợp lệ. 3. Để trống trường mật khẩu. 4. Nhấn nút "Đăng nhập".	Hiển thị thông báo yêu cầu nhập Mật khẩu.	PASS
DN_8	Đăng nhập thất bại - Bỏ trống cả Email/Username và Mật khẩu.	1. Mở trang đăng nhập. 2. Để trống cả hai trường. 3. Nhấn nút "Đăng nhập".	Hiển thị thông báo yêu cầu nhập đầy đủ thông tin.	PASS
DN_9	Kiểm tra phân biệt chữ hoa/thường cho Username	1. Mở trang đăng nhập. 2. Nhập username "UserA" (nếu	Kết quả phụ thuộc vào thiết kế: Đăng nhập thành thất bại	PASS

		username đúng là "usera"). 3. Nhập mật khẩu đúng. 4. Nhấn nút "Đăng nhập".		
DN_10	Kiểm tra phân biệt chữ hoa/thường cho Mật khẩu	1. Mở trang đăng nhập. 2. Nhập username đúng. 3. Nhập mật khẩu "PasswordA" (nếu mật khẩu đúng là "passworda"). 4. Nhấn nút "Đăng nhập".	Đăng nhập thất bại	PASS
DN_11	Kiểm tra chức năng "Quên mật khẩu" với email không tồn tại.	1. Mở trang đăng nhập. 2. Nhấn link "Quên mật khẩu". 3. Nhập email không có trong hệ thống. 4. Nhấn "Gửi".	Hiển thị thông báo "Email không tồn tại trong hệ thống" hoặc một thông báo chung chung để bảo mật.	PASS

6.5.2. Kiểm thử chức năng Cập nhật profile người dùng

Mã Test	Mục đích kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả
---------	-------------------	--------------------	-------------------	---------

case				kiểm thử
PF_1	Cập nhật thành công tất cả các trường thông tin hợp lệ.	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Thay đổi Họ tên, Ngày sinh, SĐT, Địa chỉ, Avatar với dữ liệu hợp lệ. 5. Nhấn "Lưu".	Thông tin được cập nhật chính xác. Avatar mới hiển thị. Thông báo thành công.	PASS
PF_2	Cập nhật thất bại - Email sai định dạng (nếu cho sửa email).	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Nhập email sai định dạng (ví dụ: abc@xyz). 5. Nhấn "Lưu".	Hiện thị thông báo lỗi "Email không hợp lệ". Thông tin không được lưu.	PASS
PF_3	Cập nhật thất bại - Số điện thoại chứa ký tự chữ hoặc quá dài/ngắn.	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Nhập SĐT chứa ký tự chữ (090abcde) hoặc SĐT không đúng độ dài quy định. 5. Nhấn "Lưu".	Hiện thị thông báo lỗi "Số điện thoại không hợp lệ". Thông tin không được lưu.	PASS
PF_4	Cập nhật thất bại - Ngày sinh không hợp lệ (ngày tương lai,	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Chọn ngày sinh ở	Hiện thị thông báo lỗi "Ngày sinh không hợp lệ". Thông tin	PASS

	định dạng sai).	tương lai hoặc nhập ngày sai định dạng. 5. Nhấn "Lưu".	không được lưu.	
PF_5	Hủy bỏ thao tác cập nhật.	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Thay đổi một vài thông tin. 5. Nhấn "Hủy".	Các thay đổi không được lưu. Thông tin trở về trạng thái ban đầu trước khi chỉnh sửa.	PASS
PF_6	Cập nhật thành công khi chỉ thay đổi một trường.	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Chỉ thay đổi Số điện thoại. 5. Nhấn "Lưu".	Số điện thoại được cập nhật, các thông tin khác giữ nguyên. Thông báo thành công.	PASS
PF_7	Tải lên avatar với định dạng không được hỗ trợ.	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Chọn tải lên file avatar có định dạng không được phép (ví dụ: .txt, .exe). 5. Nhấn "Lưu".	Hiển thị thông báo lỗi "Định dạng ảnh không được hỗ trợ". Avatar không thay đổi.	PASS
PF_8	Tải lên avatar với kích thước quá lớn.	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Chọn tải lên file avatar có kích thước vượt quá giới hạn cho	Hiển thị thông báo lỗi "Kích thước ảnh quá lớn". Avatar không thay đổi.	Failse

		phép. 5. Nhấn "Lưu".		
PF_9	Cập nhật thông tin khi không thay đổi gì.	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Không thay đổi thông tin nào. 5. Nhấn "Lưu".	Hệ thống chấp nhận, không báo lỗi, không có thay đổi dữ liệu.	PASS
PF_10	Kiểm tra các trường bắt buộc	1. Đăng nhập. 2. Vào trang Profile. 3. Nhấn "Chỉnh sửa". 4. Xóa dữ liệu ở một trường bắt buộc (ví dụ: Họ tên). 5. Nhấn "Lưu".	Hiển thị thông báo lỗi yêu cầu nhập trường bắt buộc.	PASS

6.5.3. Kiểm thử chức năng Xem danh sách đề tài NCKH, sinh viên, giảng viên, chi tiết đề tài NCKH

Mã Test case	Mục đích kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
XDT_1	Hiển thị danh sách đề tài NCKH (cho SV).	1. Đăng nhập vai trò Sinh viên. 2. Truy cập "Danh sách đề tài NCKH".	Hiển thị danh sách đề tài được phép xem. Thông tin cơ bản: tên, GVHD, trạng thái.	PASS
XDT_2	Hiển thị danh	1. Đăng nhập vai	Hiển thị danh sách	PASS

	sách đề tài NCKH (cho GV).	trò Giảng viên. 2. Truy cập "Danh sách đề tài NCKH"	đề tài GV hướng dẫn, đề tài GV đề xuất, hoặc toàn bộ đề tài	
XDT_3	Hiển thị danh sách đề tài NCKH (cho Giáo vụ).	1. Đăng nhập vai trò Giáo vụ. 2. Truy cập "Quản lý đề tài NCKH".	Hiển thị toàn bộ danh sách đề tài NCKH với đầy đủ thông tin và các tùy chọn quản lý.	PASS
XDT_4	Xem chi tiết một đề tài NCKH cụ thể.	1. Từ danh sách đề tài, chọn một đề tài. 2. Nhấn xem chi tiết.	Hiển thị đầy đủ thông tin: mô tả, SV, GVHD, GVPB, tiến độ, tài liệu, lịch sử, nhận xét...	PASS
XDT_5	Kiểm tra khi danh sách đề tài rỗng.	1. Trong trường hợp không có đề tài nào phù hợp với vai trò/bộ lọc. 2. Truy cập danh sách đề tài.	Hiển thị thông báo "Không có đề tài nào" hoặc giao diện trống nhưng không lỗi.	PASS
XDT_6	Kiểm tra phân trang danh sách đề tài	1. Truy cập danh sách đề tài có nhiều hơn số lượng hiển thị/trang. 2. Điều hướng qua các trang (Next, Previous, số trang).	Dữ liệu hiển thị chính xác cho từng trang. Các nút điều hướng hoạt động đúng.	PASS

XDT_7	Tìm kiếm đề tài theo tên đề tài.	1. Vào danh sách đề tài. 2. Nhập từ khóa tên đề tài vào ô tìm kiếm. 3. Nhấn "Tìm kiếm".	Hiển thị các đề tài có tên chứa từ khóa.	PASS
XDT_8	Tìm kiếm đề tài theo tên GVHD.	1. Vào danh sách đề tài. 2. Nhập tên GVHD vào ô tìm kiếm/lọc. 3. Nhấn "Tìm kiếm".	Hiển thị các đề tài do GVHD đó hướng dẫn.	PASS
XDT_9	Lọc đề tài theo trạng thái	1. Vào danh sách đề tài. 2. Chọn một trạng thái từ bộ lọc. 3. Áp dụng bộ lọc.	Hiển thị các đề tài có trạng thái tương ứng.	PASS
XDT_10	Lọc đề tài theo chuyên ngành/khoa	1. Vào danh sách đề tài. 2. Chọn chuyên ngành/khoa từ bộ lọc. 3. Áp dụng bộ lọc.	Hiển thị các đề tài thuộc chuyên ngành/khoa đã chọn.	PASS
XSV_1	Hiển thị danh sách sinh viên (cho Giáo vụ/GV).	1. Đăng nhập vai trò Giáo vụ/GV 2. Truy cập "Danh sách sinh viên".	Hiển thị danh sách SV với thông tin: Mã SV, Họ tên, Lớp, Chuyên ngành...	PASS

XGV_1	Hiển thị danh sách giảng viên (cho Giáo vụ).	1. Đăng nhập vai trò Giáo vụ 2. Truy cập "Danh sách giảng viên".	Hiển thị danh sách GV với thông tin: Mã GV, Họ tên, Khoa, Email...	PASS
-------	--	---	--	------

6.5.4. Kiểm thử chức năng Quản lý đề tài NCKH

Mã Test case	Mục đích kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
QLDT_1	GV thêm mới đề tài NCKH thành công.	1. Đăng nhập vai trò GV. 2. Vào "Quản lý đề tài" -> "Thêm mới". 3. Nhập đầy đủ thông tin hợp lệ (tên, mô tả, chuyên ngành, số lượng SV tối đa...) 4. Nhấn "Lưu".	Đề tài mới được tạo, hiển thị trong danh sách với trạng thái "Chờ duyệt". Thông báo thành công.	PASS
QLDT_2	GV thêm mới đề tài NCKH thất bại - Thiếu thông tin bắt buộc.	1. Đăng nhập vai trò GV. 2. Vào "Thêm mới đề tài". 3. Bỏ trống tên đề tài hoặc chuyên	Hiển thị thông báo lỗi yêu cầu nhập đủ thông tin bắt buộc. Đề tài không được tạo.	PASS

		ngành. 4. Nhấn "Lưu".		
QLDT_3	GV thêm mới đề tài NCKH thất bại - Số lượng SV không hợp lệ.	1. Đăng nhập vai trò GV 2. Vào "Thêm mới đề tài". 3. Nhập số lượng SV tối đa là 0 hoặc số âm. 4. Nhấn "Lưu".	Hiện thị thông báo lỗi "Số lượng sinh viên không hợp lệ". Đề tài không được tạo.	PASS
QLDT_4	Sinh viên đề xuất đề tài thành công.	1. Đăng nhập vai trò Sinh viên. 2. Vào "Đề xuất đề tài". 3. Nhập thông tin (tên, mô tả, GV mong muốn...) 4. Nhấn "Gửi đề xuất".	Đề xuất được gửi. Trạng thái đề tài là "Chờ duyệt đề xuất". GV nhận được thông báo.	PASS
QLDT_5	Sinh viên đăng ký đề tài có sẵn thành công (còn slot).	1. Đăng nhập vai trò Sinh viên. 2. Vào "Danh sách đề tài". 3. Chọn đề tài đang mở đăng ký và còn slot. 4. Nhấn "Đăng ký". 5. (Nếu đăng ký	Đăng ký thành công. Tên SV/nhóm SV hiển thị ở đề tài. Trạng thái đề tài có thể là "Chờ GVHD duyệt nhóm". Thông báo thành công.	PASS

		nhóm) Mời thêm thành viên hợp lệ.		
QLDT_6	Sinh viên đăng ký đề tài thất bại - Đề tài đã đủ số lượng.	1. Đăng nhập vai trò Sinh viên. 2. Vào "Danh sách đề tài". 3. Chọn đề tài đã đủ số lượng SV đăng ký. 4. Cố gắng nhấn "Đăng ký".	Nút "Đăng ký" bị vô hiệu hóa	PASS
QLDT_7	Sinh viên đăng ký đề tài thất bại - Ngoài thời gian đăng ký.	1. Đăng nhập vai trò Sinh viên. 2. Vào "Danh sách đề tài" ngoài thời gian cho phép đăng ký. 3. Cố gắng nhấn "Đăng ký" vào một đề tài.	Nút "Đăng ký" bị vô hiệu hóa	PASS
QLDT_8	GVHD duyệt đề xuất/đăng ký của sinh viên thành công.	1. Đăng nhập vai trò GVHD. 2. Vào danh sách đề tài chờ duyệt (đề xuất hoặc đăng ký nhóm). 3. Chọn đề tài/nhóm SV. 4. Nhấn "Duyệt".	Trạng thái đề tài cập nhật thành "Đã duyệt". Sinh viên nhận thông báo.	PASS

QLDT_9	GVHD từ chối đề xuất/đăng ký của sinh viên (kèm lý do).	<ol style="list-style-type: none"> 1. Đăng nhập vai trò GVHD. 2. Vào danh sách đề tài chờ duyệt. 3. Chọn đề tài/nhóm SV. 4. Nhấn "Từ chối". 5. Nhập lý do từ chối 	Trạng thái đề tài cập nhật "Bị từ chối". Sinh viên nhận thông báo và lý do.	PASS
QLDT_10	Giáo vụ/GV cập nhật thông tin đề tài NCKH (đã duyệt/đang thực hiện).	<ol style="list-style-type: none"> 1. Đăng nhập vai trò Giáo vụ/GV. 2. Tìm và chọn đề tài cần cập nhật. 3. Chỉnh sửa mô tả, gia hạn thời gian, thay đổi trạng thái (ví dụ: Hoàn thành, Tạm dừng). 4. Nhấn "Lưu". 	Thông tin/trạng thái đề tài được cập nhật. Các bên liên quan nhận thông báo	PASS
QLDT_11	Sinh viên cập nhật tiến độ/nộp báo cáo cho đề tài đang thực hiện.	<ol style="list-style-type: none"> 1. Đăng nhập vai trò Sinh viên. 2. Vào "Đề tài của tôi". 3. Chọn đề tài đang thực hiện. 4. Sử dụng chức năng "Nộp báo 	Tiến độ được cập nhật, file báo cáo được tải lên. GVHD nhận thông báo.	PASS

		cáo". 5. Đính kèm file 6. Nhấn "Nộp".		
QLDT_12	GVHD/Giáo vụ xóa đề tài NCKH (chưa có SV đăng ký/chưa duyệt).	1. Đăng nhập vai trò GVHD/Giáo vụ. 2. Tìm đề tài cần xóa (đảm bảo điều kiện được xóa). 3. Nhấn "Xóa". 4. Xác nhận xóa.	Đề tài bị xóa khỏi hệ thống. Thông báo thành công.	PASS
QLDT_13	Hệ thống không cho xóa đề tài NCKH đã có SV thực hiện hoặc đã hoàn thành.	1. Đăng nhập vai trò GVHD/Giáo vụ. 2. Tìm đề tài đã có SV đang thực hiện hoặc đã hoàn thành. 3. Cố gắng nhấn "Xóa".	Hiện thị thông báo "Không thể xóa đề tài đã có sinh viên thực hiện/đã hoàn thành".	PASS

6.5.5. Kiểm thử chức năng Quản lý phiếu đánh giá

Mã Test case	Mục đích kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
QLPDG_1	Giáo vụ thêm mới phiếu	1. Đăng nhập vai trò Giáo vụ.	PDG mới được tạo với các tiêu chí	PASS

	đánh giá (PĐG) thành công.	2. Vào "Quản lý PĐG". 3. Nhấn "Thêm mới phiếu". 4. Nhập tên PĐG, mô tả. 5. Thêm các tiêu chí (tên, thang điểm, trọng số). 6. Nhấn "Lưu".	chính xác. Hiện thị trong danh sách PĐG. Thông báo thành công.	
QLPDG_2	Giáo vụ thêm mới PĐG thất bại - Thiếu tên phiếu.	1. Đăng nhập vai trò Giáo vụ. 2. Vào "Thêm mới phiếu". 3. Bỏ trống tên PĐG. 4. Nhấn "Lưu".	Hiện thị thông báo lỗi yêu cầu nhập tên phiếu. PĐG không được tạo.	PASS
QLPDG_3	Giáo vụ thêm mới PĐG thất bại - Tiêu chí có trọng số không hợp lệ.	1. Đăng nhập vai trò Giáo vụ. 2. Vào "Thêm mới phiếu". 3. Nhập tên PĐG. 4. Thêm tiêu chí với trọng số là số âm hoặc tổng trọng số các tiêu chí > 100% 5. Nhấn "Lưu".	Hiện thị thông báo lỗi "Trọng số không hợp lệ" hoặc "Tổng trọng số vượt quá 100%". PĐG không được tạo.	PASS
QLPDG_4	Giáo vụ cập nhật thông	1. Đăng nhập vai trò Giáo vụ.	Thông tin PĐG và các tiêu chí được	PASS

	tin/tiêu chí của PĐG thành công.	2. Vào "Quản lý PĐG". 3. Chọn một PĐG để sửa. 4. Thay đổi tên, mô tả, thêm/sửa/xóa tiêu chí. 5. Nhấn "Lưu".	cập nhật chính xác.	
QLPDG_5	Giáo vụ xóa PĐG chưa được sử dụng thành công.	1. Đăng nhập vai trò Giáo vụ. 2. Vào "Quản lý PĐG". 3. Chọn một PĐG chưa được gán cho bất kỳ hoạt động chấm điểm nào. 4. Nhấn "Xóa". 5. Xác nhận xóa.	PĐG được xóa thành công khỏi hệ thống.	PASS
QLPDG_6	Hệ thống không cho Giáo vụ xóa PĐG đã được sử dụng (đã có điểm).	1. Đăng nhập vai trò Giáo vụ. 2. Vào "Quản lý PĐG". 3. Chọn một PĐG đã được sử dụng để chấm điểm cho ít nhất một đề tài. 4. Cố gắng nhấn "Xóa".	Hiển thị thông báo "Không thể xóa phiếu đánh giá đã được sử dụng"	PASS
QLPDG_7	Giáo vụ xem	1. Đăng nhập vai trò	Hiển thị đầy đủ	PASS

	chi tiết một PĐG.	Giáo vụ. 2. Vào "Quản lý PĐG". 3. Nhấn vào tên một PĐG.	thông tin của PĐG bao gồm tên, mô tả và danh sách các tiêu chí chi tiết (tên, thang điểm, trọng số).	
--	-------------------	---	--	--

6.5.6. Kiểm thử chức năng Quản lý hội đồng

Mã Test case	Mục đích kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
QLHD_1	Giáo vụ tạo mới hội đồng (HĐ) thành công với đầy đủ thông tin.	1. Đăng nhập vai trò Giáo vụ. 2. Vào "Quản lý HĐ". 3. Nhấn "Tạo mới HĐ". 4. Nhập tên HĐ, chọn chuyên ngành, ngày bảo vệ/nghiệm thu, địa điểm. 5. Chọn các thành viên (Chủ tịch, Thư ký, các Ủy viên). 6. Nhấn "Lưu".	HĐ mới được tạo, hiển thị trong danh sách HĐ. Thông tin thành viên, ngày, địa điểm chính xác. Thông báo thành công.	PASS

QLHD_2	Giáo vụ tạo mới HĐ thất bại - Thiếu thông tin bắt buộc.	<ol style="list-style-type: none"> 1. Đăng nhập vai trò Giáo vụ. 2. Vào "Tạo mới HĐ". 3. Bỏ trống tên HĐ hoặc không chọn Chủ tịch. 4. Nhấn "Lưu". 	Hiện thị thông báo lỗi yêu cầu nhập đủ thông tin bắt buộc. HĐ không được tạo.	PASS
QLHD_3	Giáo vụ tạo mới HĐ thất bại - Thành viên được chọn không hợp lệ.	<ol style="list-style-type: none"> 1. Đăng nhập vai trò Giáo vụ. 2. Vào "Tạo mới HĐ". 3. Cố gắng chọn một người không phải là Giảng viên làm thành viên HĐ 	Hệ thống không cho phép chọn hoặc báo lỗi khi lưu.	PASS
QLHD_4	Giáo vụ cập nhật thông tin HĐ thành công.	<ol style="list-style-type: none"> 1. Đăng nhập vai trò Giáo vụ. 2. Vào "Quản lý HĐ". 3. Chọn một HĐ để sửa. 4. Thay đổi ngày, địa điểm, hoặc thành viên. 5. Nhấn "Lưu". 	Thông tin HĐ được cập nhật chính xác.	PASS
QLHD_5	Giáo vụ phân công/gán đề tài	<ol style="list-style-type: none"> 1. Đăng nhập vai trò Giáo vụ. 	Các đề tài được gán thành công cho HĐ.	PASS

	NCKH cho HĐ thành công.	2. Vào chi tiết một HĐ hoặc chức năng "Phân công đề tài". 3. Chọn các đề tài NCKH phù hợp (ví dụ: đã hoàn thành, chờ bảo vệ). 4. Gán các đề tài đó cho HĐ. 5. Nhấn "Lưu".	Thông tin hiển thị trên trang của HĐ và đề tài.	
QLHD_6	Giáo vụ xóa HĐ chưa có đề tài nào được gán.	1. Đăng nhập vai trò Giáo vụ. 2. Vào "Quản lý HĐ". 3. Chọn một HĐ mới tạo, chưa gán đề tài. 4. Nhấn "Xóa". 5. Xác nhận xóa.	HĐ được xóa thành công khỏi hệ thống.	PASS
QLHD_7	Hệ thống không cho Giáo vụ xóa HĐ đã có đề tài được gán hoặc đã hoạt động.	1. Đăng nhập vai trò Giáo vụ. 2. Vào "Quản lý HĐ". 3. Chọn một HĐ đã có đề tài được gán hoặc đã diễn ra buổi bảo vệ.	Hiển thị thông báo "Không thể xóa hội đồng đã có hoạt động".	PASS

		4. Cố gắng nhấn "Xóa".		
QLHD_8	Xem danh sách các HĐ và chi tiết từng HĐ.	1. Đăng nhập vai trò Giáo vụ. 2. Vào "Quản lý HĐ". 3. Nhấn vào một HĐ để xem chi tiết.	Danh sách HĐ hiển thị đúng. Thông tin chi tiết HĐ hiển thị chính xác.	PASS

6.5.7. Kiểm thử chức năng Chấm điểm

Mã Test case	Mục đích kiểm thử	Các bước thực hiện	Kết quả mong muốn	Kết quả kiểm thử
CD_1	GVHD chấm điểm và nhận xét thành công cho đề tài NCKH đang hướng dẫn.	1. Đăng nhập vai trò GVHD. 2. Truy cập đề tài NCKH cần chấm (đã đến hạn/giai đoạn chấm điểm). 3. Mở giao diện/phiếu chấm điểm dành cho GVHD. 4. Nhập điểm cho từng tiêu chí, viết nhận xét.	Điểm và nhận xét được lưu chính xác. Trạng thái chấm điểm của GVHD được cập nhật. SV có thể xem điểm	PASS

		5. Nhấn "Lưu điểm"/"Hoàn thành".		
CD_2	GVPB chấm điểm và nhận xét thành công cho đề tài NCKH được phân công.	1. Đăng nhập vai trò GVPB. 2. Truy cập đề tài NCKH cần phản biện. 3. Mở giao diện/phiếu chấm điểm dành cho GVPB. 4. Nhập điểm cho từng tiêu chí, viết nhận xét. 5. Nhấn "Lưu điểm"/"Hoàn thành".	Điểm và nhận xét được lưu chính xác. Trạng thái chấm điểm của GVPB được cập nhật.	PASS
CD_3	Thành viên Hội đồng chấm điểm và nhận xét thành công cho đề tài NCKH	1. Đăng nhập vai trò thành viên HĐ. 2. Truy cập đề tài NCKH trong phiên làm việc của HĐ. 3. Mở giao diện/phiếu chấm điểm dành cho HĐ.	Điểm và nhận xét của thành viên được lưu chính xác.	PASS

		<p>4. Nhập điểm cho từng tiêu chí, viết nhận xét.</p> <p>5. Nhấn "Lưu điểm".</p>		
CD_4	Chấm điểm thất bại - Nhập điểm ngoài thang điểm cho phép.	<p>1. Thực hiện các bước như CD_1, CD_2 hoặc CD_3.</p> <p>2. Tại một tiêu chí, nhập điểm vượt quá thang điểm (ví dụ: 11/10) hoặc điểm âm.</p> <p>3. Nhấn "Lưu điểm".</p>	Hệ thống hiển thị thông báo lỗi "Điểm không hợp lệ". Điểm không được lưu.	PASS
CD_5	Chấm điểm thất bại - Bỏ trống tiêu chí bắt buộc chấm .	<p>1. Thực hiện các bước như CD_1, CD_2 hoặc CD_3.</p> <p>2. Bỏ trống điểm ở một tiêu chí được đánh dấu là bắt buộc.</p> <p>3. Nhấn "Lưu điểm".</p>	Hệ thống hiển thị thông báo lỗi yêu cầu chấm đủ các tiêu chí bắt buộc.	PASS
CD_6	GV/Thành viên HĐ không có quyền chấm	1. Đăng nhập vai trò GV bất kỳ.	Hệ thống không cho phép truy	PASS

	điểm cho đề tài không được phân công.	2. Cố gắng truy cập và chấm điểm cho một đề tài không phải mình hướng dẫn/phản biện/thuộc HĐ của mình.	cập giao diện chấm điểm hoặc không hiển thị tùy chọn chấm điểm.	
CD_7	Kiểm tra tính toán điểm trung bình/tổng kết tự động chính xác	1. Sau khi tất cả các GV/thành viên HĐ đã cho điểm theo vai trò của họ. 2. Truy cập xem điểm tổng hợp của đề tài	Hệ thống tính toán và hiển thị điểm trung bình một cách chính xác theo công thức đã định.	PASS
CD_8	GVHD/GVPB/Thành viên HĐ cập nhật lại điểm/nhận xét đã cho	1. Đăng nhập với vai trò tương ứng. 2. Truy cập lại đề tài đã chấm. 3. Chọn sửa lại điểm/nhận xét. 4. Nhập thông tin mới. 5. Nhấn "Lưu cập nhật".	Điểm/nhận xét mới được lưu. Lịch sử thay đổi có thể được ghi nhận	Failse
CD_9	Kiểm tra khóa chức năng chấm điểm sau thời hạn quy định	1. Cố gắng thực hiện chấm điểm cho một đề tài đã quá hạn chấm	Hệ thống không cho phép chấm điểm.	PASS

		điểm theo cấu hình hệ thống.		
CD_10	Xem lại lịch sử/chi tiết điểm của các lần chấm	1. Đăng nhập với vai trò phù hợp (GV, Giáo vụ). 2. Truy cập chi tiết đề tài hoặc mục quản lý điểm. 3. Tìm và xem lịch sử chấm điểm hoặc chi tiết điểm từ các người chấm khác nhau.	Hệ thống hiển thị đầy đủ, chính xác thông tin điểm, người chấm, thời gian chấm.	PASS

6.6. KẾT QUẢ KIỂM THỬ

Từ các kiến thức quý báu đó, nhóm đã đúc kết và áp dụng kiểm thử cho quản lý đề tài Nghiên cứu Khoa học như sau:

- Chức năng đăng ký, đăng nhập
- Chức năng cập nhật profile người dùng
- Chức năng xem danh sách đề tài, sinh viên, giảng viên, chi tiết đề tài, ...
- Chức năng quản lý đề tài: thêm đề tài, duyệt đề tài, cập nhật đề tài
- Chức năng quản lý phiếu đánh giá: thêm phiếu đánh giá, cập nhật phiếu đánh giá, xóa phiếu đánh giá,...
- Chức năng quản lý hội đồng, phân chia hội đồng, ...
- Chức năng chấm điểm dành cho giảng viên hướng dẫn, giảng viên phản biện và hội đồng phản biện

Trong quá trình này, nhóm đã áp dụng thành công các phương pháp kiểm thử

phổ biến như Kiểm thử hộp trắng (White-box Testing) để kiểm tra logic mã nguồn và Kiểm thử hộp đen (Black-box Testing) để xác minh chức năng dựa trên yêu cầu người dùng. Các chức năng chưa được kiểm thử sâu là những tính năng mở rộng hoặc các trường hợp đặc biệt chưa được ưu tiên trong phạm vi kiểm thử ban đầu. Sau khi thực hiện các kịch bản kiểm thử, nhóm đã phát hiện một số lỗi còn tồn tại trong hệ thống. Các lỗi này đã được ghi nhận chi tiết (bug report) và nhóm đã nỗ lực khắc phục thành công, qua đó góp phần cải thiện chất lượng và độ ổn định của hệ thống.

CHƯƠNG 7 : KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

7.1. KẾT QUẢ ĐẠT ĐƯỢC

Trong quá trình học tập, chúng em đã tiến hành lập kế hoạch xây dựng và thực hiện kiểm thử dự án với sự hướng dẫn nhiệt tình và chi tiết từ TS. Cao Thái Phương Thanh. Nhờ vào sự hỗ trợ đặc lực của cô, nhóm chúng em đã đạt được những thành tựu đáng kể:

- Phát triển kỹ năng làm việc nhóm và phân công công việc:
 - Chúng em đã học được cách tổ chức và phân chia công việc sau khi nhận đề tài nghiên cứu khoa học. Thêm vào đó, kiến thức về xây dựng và triển khai hệ thống website đã được chúng em nắm bắt, nhờ vào sự hướng dẫn chi tiết của cô. Cô còn đóng góp những ý kiến quý báu và chân thành để chúng em có thể hoàn thiện sản phẩm đề tài nghiên cứu khoa học.
- Phát triển kỹ năng mềm:
 - Chúng em đã không chỉ học được kỹ năng cụ thể liên quan đến lập kế hoạch và giải quyết vấn đề mà còn phát triển kỹ năng làm việc nhóm. Ngoài ra, việc viết báo cáo cũng là một kỹ năng quan trọng mà chúng em đã rèn luyện được.
- Áp dụng kiến thức vào thực tế thông qua kiểm thử:
 - Từ những kiến thức đã học, nhóm chúng em đã áp dụng kiểm thử cho website quản lý hoạt động nghiên cứu khoa học của sinh viên của mình. Điều này bao gồm việc thử nghiệm và đánh giá chức năng của website, đảm bảo rằng nó hoạt động mượt mà và đáp ứng đúng yêu cầu.

Những thành tựu trên không chỉ là kết quả của sự nỗ lực cá nhân mà còn nhờ vào sự hỗ trợ và hướng dẫn chân thành của TS. Cao Thái Phương Thanh. Chúng em rất biết ơn và tự hào về quãng thời gian học tập có ý nghĩa này.

Từ các kiến thức quý báu đó, nhóm đã đúc kết và áp dụng cho việc xây dựng website Quản lý Hoạt động Nghiên cứu Khoa học của Sinh viên của mình với các

chức năng nghiệp vụ mà nhóm đã đề ra.

7.2. ĐÁNH GIÁ HỆ THỐNG

7.2.1.Ưu điểm

Sau quá trình thiết kế và xây dựng, hệ thống "Trang web Quản lý Hoạt động Nghiên cứu Khoa học của Sinh viên" đã đạt được những ưu điểm nổi bật, góp phần nâng cao hiệu quả quản lý và trải nghiệm người dùng:

- Giao diện thân thiện, dễ sử dụng: Hệ thống được thiết kế với giao diện trực quan, logic, giúp người dùng (bao gồm sinh viên, giảng viên và giáo vụ) dễ dàng làm quen và thao tác hiệu quả ngay từ lần đầu tiếp cận. Các chức năng được sắp xếp khoa học, giảm thiểu thời gian tìm kiếm và học cách sử dụng.
- Đáp ứng đầy đủ các chức năng nghiệp vụ cốt lõi: Hệ thống đã triển khai thành công các module chức năng cơ bản và quan trọng, bao gồm quản lý thông tin người dùng, quản lý đề tài NCKH (từ khâu đăng ký, duyệt, đến theo dõi tiến độ và đánh giá), quản lý các đợt đăng ký, học kỳ, và các biểu mẫu liên quan.
- Tối thiểu hóa lỗi xảy ra: Thông qua quá trình kiểm thử cẩn thận, các lỗi nghiêm trọng đã được phát hiện và khắc phục. Hệ thống vận hành tương đối ổn định, đảm bảo tính toàn vẹn dữ liệu và giảm thiểu gián đoạn trong quá trình sử dụng của người dùng, mang lại sự tin cậy cần thiết.

7.2.2.Nhược điểm

Hệ thống vẫn còn một số nhược điểm cần được nhìn nhận để có kế hoạch cải thiện trong tương lai.

- Hệ thống còn lỗi: Mặc dù đã nỗ lực kiểm thử, hệ thống có thể vẫn còn một số lỗi nhỏ hoặc các trường hợp đặc biệt chưa được bao quát hết.
- Tốc độ hệ thống chưa được tối ưu tối đa: Hiện tại, tốc độ phản hồi của một số chức năng hoặc khi xử lý lượng lớn dữ liệu có thể chưa đạt mức tối ưu nhất. Cần có thêm thời gian để rà soát, tối ưu hóa truy vấn cơ sở dữ

liệu.

- Chưa responsive toàn bộ hệ thống: Giao diện người dùng hiện tại có thể chưa hoàn toàn tương thích và hiển thị tối ưu trên tất cả các kích thước màn hình khác nhau (đặc biệt là thiết bị di động và máy tính bảng). Điều này có thể gây bất tiện cho người dùng khi truy cập hệ thống từ các thiết bị không phải máy tính.

7.2.3. Hướng phát triển

Để nâng cao hơn nữa giá trị và tính ứng dụng của hệ thống, nhóm đề xuất một số hướng phát triển tiềm năng trong tương lai:

- **Mở rộng và nâng cao chức năng báo cáo và tương tác:**
 - Phát triển module báo cáo tiến độ chi tiết hơn, cho phép sinh viên cập nhật thường xuyên và giảng viên dễ dàng theo dõi, phản hồi trực tiếp trên hệ thống.
 - Tích hợp tính năng cho phép giảng viên phản biện và hội đồng đánh giá, cho điểm trực tuyến, giảm thiểu quy trình giấy tờ thủ công.
 - Xây dựng hệ thống thông báo (notifications) thông minh hơn, nhắc nhở các hạn nộp, lịch bảo vệ, và các cập nhật quan trọng khác.
- **Cải thiện trải nghiệm người dùng đa nền tảng:**
 - Hoàn thiện thiết kế đáp ứng (Responsive Design) để đảm bảo giao diện hiển thị tốt và dễ sử dụng trên mọi loại thiết bị (desktop, tablet, mobile).
 - Phát triển ứng dụng di động (Mobile App) riêng biệt cho hệ thống, tận dụng các tính năng của thiết bị di động như thông báo đẩy (push notifications), tăng tính tiện lợi và khả năng truy cập mọi lúc mọi nơi cho người dùng.
- **Tăng cường tính năng và tối ưu hóa hệ thống:**
 - Bổ sung hỗ trợ đa ngôn ngữ để phục vụ cho nhiều đối tượng người dùng hơn, đặc biệt trong bối cảnh hội nhập quốc tế.
 - Tiếp tục tối ưu hóa hiệu suất hệ thống, tốc độ tải trang và xử lý dữ

liệu.

- Nghiên cứu và áp dụng các biện pháp bảo mật nâng cao để đảm bảo an toàn thông tin cho dữ liệu nghiên cứu khoa học nhạy cảm.
- Tích hợp các công cụ phân tích dữ liệu để đưa ra các thống kê, báo cáo hữu ích về tình hình hoạt động NCKH, hỗ trợ công tác quản lý và định hướng của khoa/trường.

7.3. KẾT LUẬN

Sau thời gian thực hiện, dự án "Xây dựng Trang web Quản lý Hoạt động Nghiên cứu Khoa học của Sinh viên" của chúng em về cơ bản đã đạt được các mục tiêu chính. Chúng em thấy rằng hệ thống này thực sự mang lại một bước tiến, giúp việc quản lý nghiên cứu khoa học không còn quá nhiều giấy tờ thủ công như trước. Nhờ đó, cả sinh viên, thầy cô và phòng giáo vụ đều có thể làm việc và tương tác với nhau một cách thuận lợi hơn trên một nền tảng chung. Những chức năng quan trọng như quản lý đề tài từ đầu đến cuối, theo dõi tiến độ, hay quản lý người dùng cũng đã được chúng em đưa vào hoạt động, và bước đầu cho thấy nhiều điểm tích cực.

Dĩ nhiên, dù rất cố gắng, sản phẩm của chúng em không thể nào hoàn hảo ngay được. Vẫn còn đó những điểm cần cải thiện, ví dụ như tốc độ xử lý đôi khi chưa nhanh như kỳ vọng, một vài tính năng có thể cần bổ sung thêm, hay việc hiển thị trên các thiết bị khác nhau chưa thực sự tối ưu. Nhưng chính việc nhận ra những điều này, cùng với các ý tưởng phát triển đã có, sẽ giúp chúng em có hướng đi rõ ràng để làm cho hệ thống tốt hơn nữa trong tương lai.

Quá trình làm đề tài này thực sự là một chặng đường học hỏi rất nhiều điều đối với chúng em. Không chỉ là cơ hội để áp dụng kiến thức đã học vào thực tế, rèn luyện thêm về code và cách thiết kế một hệ thống, mà quan trọng hơn, chúng em đã hiểu sâu hơn về cách một quy trình nghiên cứu khoa học vận hành, cũng như những gì cần thiết để một hệ thống quản lý thông tin hoạt động hiệu quả. Với những gì đã làm được và các kế hoạch cụ thể cho tương lai, chúng em tin rằng trang web quản lý hoạt động nghiên cứu khoa học này hoàn toàn có thể trở thành một công cụ hữu ích, góp phần nâng cao chất lượng nghiên cứu khoa học tại trường.

TÀI LIỆU KHAM THẢO

- [1] The MERN stack. (2023).
Retrieved from MongoDB: <https://www.mongodb.com/mern-stack>
- [2] What is the stack? (2023).
Retrieved from Quora: <https://www.quora.com/What-is-the-stack>
- [3] MEAN stack developer. (2023).
Retrieved from Guru99: <https://www.guru99.com/mean-stack-developer.html>
- [4] NestJS documentation. (2023).
Retrieved from NestJS: <https://docs.nestjs.com/>
- [5] @hapi/joi. (2019).
Retrieved from npm.com: <https://www.npmjs.com/package/@hapi/joi>
- [6] Concurrently. (2021).
Retrieved from Npm: <https://www.npmjs.com/package/concurrently>
- [7] JSON Web Token (JWT) là gì ? (2020).
Retrieved from topdev.vn: <https://topdev.vn/blog/jwt-la-gi/>
- [8] Khóa học ReactJS (2019 – 2024)
Retrieved from F8: [Khóa học ReactJS](#)
- [9] Giới thiệu Reactjs (2019)
Retrieved from viblo.asia: [Giới thiệu ReactJS](#)
- [10] Hướng dẫn sử dụng React Route và Redux Toolkit (2021).
Retrieved from techmaster.vn: [Hướng dẫn sử dụng React Router và Redux Toolkit \(techmaster.vn\)](#)
- [11] API là gì ? (2021).
Retrieved from aws.amazon.com: [API là gì? - Hướng dẫn về API cho người mới bắt đầu - AWS \(amazon.com\)](#)