

TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP

ĐỀ TÀI **QUẢN LÝ ĐỒ ÁN SINH VIÊN**

Giảng viên hướng dẫn	: TS. Nguyễn Hiếu Cường
Sinh viên thực hiện	: Trần Minh Đức
Mã sinh viên	: 201210096
Lớp	: CNTT 5
Khóa	: 61

HÀ NỘI - 2024

MỤC LỤC

DANH MỤC BẢNG BIỂU	3
DANH MỤC HÌNH ẢNH	4
LỜI GIỚI THIỆU.....	6
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI.....	7
1.1. Khảo sát	7
1.1.1. Khảo sát hiện trạng.....	7
1.1.2. Khảo sát thực tế	7
1.2. Mục tiêu và yêu cầu của hệ thống	10
1.3. Phạm vi đề tài	11
CHƯƠNG 2: CÁC CÔNG NGHỆ SỬ DỤNG.....	12
2.1. Nhận định nền tảng sử dụng	12
2.2. Công nghệ hệ thống	12
2.2.1. Các công nghệ sử dụng ở phía Frontend	12
2.2.2. Các công nghệ sử dụng ở phía Backend.....	16
CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG.....	18
3.1. Tóm tắt yêu cầu của hệ thống	18
3.2. Thiết kế hệ thống	19
3.2.1 Mô tả tác nhân	19
3.2.2 Mô tả chức năng hệ thống	20
3.2.3. Mô tả các chức năng tự động của hệ thống	21
3.2.4. Yêu cầu phi chức năng	22
3.3. Sơ đồ UseCase.....	23
3.3.1. Sơ đồ use case Tổng quát	23
3.3.2. Sơ đồ usecase Đăng nhập, đăng ký	24
3.3.3. Sơ đồ usecase quản lý sinh viên	25
3.3.4. Sơ đồ usecase quản lý giảng viên.....	25
3.3.5. Sơ đồ usecase quản lý chuyên ngành	26
3.3.6. Sơ đồ usecase quản lý học vấn	26

3.3.7. Sơ đồ usecase quản lý biểu mẫu	27
3.3.8. Sơ đồ usecase quản lý đề cương	27
3.3.9. Sơ đồ usecase quản lý lịch nộp tuần	28
3.3.10. Sơ đồ usecase đánh giá đồ án	28
3.3.11. Sơ đồ usecase quản lý học kì	29
3.3.12. Sơ đồ usecase quản lý hội đồng	29
3.3.13. Sơ đồ usecase quản lý nhóm xét duyệt	30
3.4. Sơ đồ tuần tự	31
3.4.1. Sơ đồ tuần tự đăng nhập	31
3.4.2. Sơ đồ tuần tự cấp lại mật khẩu	32
3.4.3. Sơ đồ tuần tự đổi mật khẩu	32
3.4.4. Sơ đồ tuần tự quản lý sinh viên, giảng viên	33
3.4.5. Sơ đồ tuần tự gán giảng viên hướng dẫn	33
3.4.6. Sơ đồ tuần tự quản lý học kỳ	34
3.4.7. Sơ đồ tuần tự quản lý nhóm xét duyệt	35
3.4.8. Sơ đồ tuần tự quản lý hội đồng	35
3.4.9. Sơ đồ tuần tự giảng viên cho điểm sinh viên	36
3.4.10. Sơ đồ tuần tự quản lý sinh viên cho giảng viên hướng dẫn	36
3.5 Thiết kế cơ sở dữ liệu	37
3.5.1. Tổng quan	37
3.5.2 Mô hình thực thể liên kết	38
3.5.3 Mô hình quan hệ	39
3.5.4 Mô tả dữ liệu	40
3.6. Triển khai hệ thống	47
CHƯƠNG 4: KẾT QUẢ SẢN PHẨM	50
4.1. Kiến trúc hệ thống	50
4.2. Cấu trúc dự án	50
4.3. Giao diện người dùng	51

DANH MỤC BẢNG BIỂU

Bảng 1.1. Bảng kết quả phỏng vấn.....	8
Bảng 3.1. Bảng mô tả tác nhân.....	19
Bảng 3.2. Mô tả bảng Student	40
Bảng 3.3. Mô tả bảng Teacher.....	41
Bảng 3.4. Mô tả bảng Council.....	42
Bảng 3.5. Mô tả bảng Comment.....	42
Bảng 3.6. Mô tả bảng Teaching.....	42
Bảng 3.7. Mô tả bảng Project	43
Bảng 3.8. Mô tả bảng ProjectOutline	44
Bảng 3.9. Mô tả bảng GroupReviewOutline	44
Bảng 3.10. Mô tả bảng ScheduleSemester	45
Bảng 3.11. Mô tả bảng Major.....	45
Bảng 3.12. Mô tả bảng ScheduleWeek	46
Bảng 3.13. Mô tả bảng DetailScheduleWeek.....	46
Bảng 3.14. Mô tả bảng Semester.....	47

DANH MỤC HÌNH ẢNH

Hình 3.1. Sơ đồ usecase tổng quát.....	23
Hình 3.2. Sơ đồ usecase đăng nhập, đăng ký	24
Hình 3.3. Sơ đồ usecase quản lý sinh viên	25
Hình 3.4. Sơ đồ usecase quản lý giảng viên.....	25
Hình 3.5. Sơ đồ usecase quản lý chuyên ngành	26
Hình 3.6. Sơ đồ usecase quản lý học vấn	26
Hình 3.7. Sơ đồ usecase quản lý biểu mẫu.....	27
Hình 3.8. Sơ đồ usecase quản lý đề cương.....	27
Hình 3.9. Sơ đồ usecase quản lý lịch nộp tuần.....	28
Hình 3.10. Sơ đồ usecase đánh giá đồ án	28
Hình 3.11. Sơ đồ usecase quản lý học kì.....	29
Hình 3.12. Sơ đồ usecase quản lý hội đồng	29
Hình 3.13. Sơ đồ usecase quản lý nhóm xét duyệt.....	30
Hình 3.14. Sơ đồ tuần tự đăng nhập	31
Hình 3.15. Sơ đồ tuần tự cấp lại mật khẩu	32
Hình 3.16. Sơ đồ tuần tự đổi mật khẩu.....	32
Hình 3.17. Sơ đồ tuần tự quản lý sinh viên, giảng viên	33
Hình 3.18. Sơ đồ tuần tự gán giảng viên hướng dẫn.....	33
Hình 3.19. Sơ đồ tuần tự quản lý học kỳ	34
Hình 3.20. Sơ đồ tuần tự quản lý nhóm xét duyệt.....	35
Hình 3.21. Sơ đồ tuần tự quản lý hội đồng.....	35
Hình 3.22. Sơ đồ tuần tự cấp lại mật khẩu	36
Hình 3.23. Sơ đồ tuần tự cấp lại mật khẩu	36
Hình 3. 24. Mô hình thực thể liên kết.....	38
Hình 3.25. Mô hình quan hệ	39
Hình 3.26. Thông tin cấu hình máy chủ VPS.....	47
Hình 3.27. Cấu hình máy chủ nginx.....	48
Hình 3.28. Trạng thái của máy chủ nginx	48

Hình 3.29. Cấu hình Nginx.....	48
Hình 3.30. Cấu hình service	48
Hình 3.31. Kiểm tra trạng thái của service	49
Hình 3.32. Thông tin cấu hình Hosting	49
Hình 4.1. Kiến trúc hệ thống	50
Hình 4.2. Cấu trúc dự án.....	51
Hình 4.3. Chức năng đăng nhập	52
Hình 4.4. Chức năng quên mật khẩu	52
Hình 4.5. Chức năng quản lý học kỳ	53
Hình 4.6. Chức năng quản lý chi tiết học kỳ	53
Hình 4.7. Chức năng lập kế hoạch học kỳ.....	54
Hình 4.8. Chức năng quản lý sinh viên	54
Hình 4.9. Chức năng thêm sinh viên	55
Hình 4.10. Chức năng thêm danh sách sinh viên	55
Hình 4.11. Chức năng quản lý giảng viên	56
Hình 4.12. Chức năng thêm giảng viên.....	56
Hình 4.13. Chức năng quản lý chuyên ngành	57
Hình 4.14. Chức năng quản lý học vấn	57
Hình 4.15. Chức năng quản lý biểu mẫu	58
Hình 4.16. Chức năng quản lý thông tin cá nhân của giảng viên.....	58
Hình 4.17. Chức năng cập nhật thông tin cá nhân của giảng viên	59
Hình 4.18. Chức năng quản lý kế hoạch khoa.....	59
Hình 4.19. Chức năng quản lý sinh viên hướng dẫn	60
Hình 4.20. Chức năng quản lý nhóm xét duyệt.....	60
Hình 4.21. Chức năng quản lý đề cương.....	61
Hình 4.22. Chức năng cập nhật điểm giảng viên hướng dẫn	61
Hình 4.23. Chức năng cập nhật điểm của giảng viên của đọc duyệt đồ án.....	62
Hình 4.24. Chức năng quản lý trang cá nhân của sinh viên	62
Hình 4.25. Chức năng quản lý hội đồng bảo vệ và lịch tuần của sinh viên	63
Hình 4.26. Chức năng theo dõi lịch báo cáo tuần	63

LỜI GIỚI THIỆU

Học phần đồ án tốt nghiệp có ý nghĩa vô cùng quan trọng trong chặng đường học tập của sinh viên khoa Công nghệ thông tin nói riêng và sinh viên của các trường đại học nói chung, là cơ hội để sinh viên thể hiện tất cả những kiến thức và kỹ năng mà họ đã tích lũy suốt thời gian học tập. Đồng thời, quản lý hiệu quả quá trình thực hiện đồ án tốt nghiệp là một thách thức đối với cả sinh viên và nhóm giảng viên hướng dẫn. Với mục tiêu giải quyết những vấn đề còn tồn đọng, hướng tới tiết kiệm tối đa thời gian và công sức của những đối tượng tham gia học kì đồ án tốt nghiệp. Đối với sinh viên, các bạn sẽ có thể theo dõi, cập nhật được thông tin mới nhất về kì đồ án mà mình bảo vệ và tiết kiệm được thời gian nộp báo cáo cho giảng viên hướng dẫn. Về phía giảng viên, các thầy cô sẽ luôn theo dõi được quá trình thực tập của mỗi sinh viên mà mình đang hướng dẫn một cách sát sao nhất, tránh trường hợp bỏ sót thông tin. Với Khoa Công nghệ thông tin, các thầy cô quản lý thực tập sẽ tiết kiệm được rất nhiều thời gian khi thực hiện các công việc cần thiết ở giải đoạn quản lý đồ án của các sinh viên.

Chính từ nhận thức sâu sắc về vấn đề này, em đã lựa chọn đề tài "Quản lý đồ án tốt nghiệp cho khoa Công nghệ thông tin." Đồ án không chỉ hướng đến việc xây dựng một hệ thống quản lý hiệu quả mà còn tập trung vào việc tối ưu hóa quá trình giao tiếp và tương tác giữa sinh viên và giảng viên, từ đó tạo ra một môi trường làm việc tích cực và chủ động.

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Khảo sát

1.1.1. Khảo sát hiện trạng

Trong thời đại hiện nay, việc chuyển đổi số đã trở thành một xu hướng không thể phủ nhận, đặc biệt là trong lĩnh vực Công nghệ thông tin. Sự chuyển đổi này không chỉ áp dụng cho các quy trình sản xuất mà còn mở ra những cơ hội mới trong việc quản lý và tổ chức thông tin, đặc biệt là trong việc bảo vệ đồ án tốt nghiệp tại các khoa Công nghệ thông tin.

Trong quá trình học phần đồ án, sinh viên sẽ trải qua nhiều giai đoạn quản lý và đánh giá từ phía giảng viên. Thông thường, ở năm thứ 4, sinh viên có thể đăng ký học phần này. Sau khi đăng kí, sinh viên sẽ phải khai báo hướng học tập, định hướng cũng như lựa chọn giảng viên để hướng dẫn. Từ đó, nhà trường có thể dễ dàng gán giảng viên hướng dẫn theo đúng nguyện vọng và định hướng của sinh viên.

Trong quá trình thực hiện đồ án, sinh viên sẽ phải tuân thủ một kế hoạch làm đồ án được tổ chức bởi khoa. Sinh viên sẽ cần phải tiến hành để xác nhận ý kiến và nhận xét của giảng viên để có thể thống nhất đề tài của đồ án. Việc trao đổi thông tin giữa sinh viên và giảng viên là một phần quan trọng của quá trình này và có thể đòi hỏi một khoảng thời gian đáng kể.

Tuy nhiên, mỗi giảng viên phải hướng dẫn một lượng sinh viên không nhỏ có thể là một thách thức đối với giảng viên nói riêng và các trường đại học nói chung. Đồng thời, đối với khoa, việc theo dõi tiến trình của các sinh viên cũng có thể trở nên khó khăn khi phải xử lý một lượng lớn thông tin.

Với những thách thức này, nhu cầu về một hệ thống quản lý thông tin cho đồ án tốt nghiệp trở nên cấp bách hơn bao giờ hết. Một hệ thống như vậy không chỉ giúp tối ưu hóa quy trình mà còn tăng cường tính minh bạch và khả năng theo dõi tiến trình của các sinh viên, từ đó giúp các bên liên quan đưa ra các quyết định và hỗ trợ đúng đắn hơn.

1.1.2. Khảo sát thực tế

- Địa điểm khảo sát: Tầng 3, tòa A9 trường Đại học Giao Thông Vận Tải
- Quá trình khảo sát:
 - o Khảo sát về thực trạng quản lý sinh viên đồ án hiện tại của trường Giao Thông Vận Tải trong khoa Công nghệ thông tin
 - o Những khó khăn mà khoa gặp phải
- Kết quả khảo sát thực tế:

Bảng 1.1. Bảng kết quả phỏng vấn

	PHIẾU PHỎNG VẤN	
	<p>Người hỏi: Trần Minh Đức</p> <p>Ngày: 19/01/2024</p>	<p>Dự án: Khảo sát về quy trình quản lý đồ án của khoa Công nghệ thông tin</p> <p>Người trả lời: TS. Hoàng Văn Thông</p> <p>Vị trí: Trưởng khoa công nghệ thông tin trường đại học Giao Thông Vận Tải</p>
1	- Hiện nay nhà trường đã có phần mềm quản lý sinh viên bảo vệ đồ án chưa ạ ?	- Hiện nay, trường ta đang chưa có phần mềm quản lý sinh viên cho học phần đồ án tốt nghiệp
2	- Có phải sinh viên sẽ được phân công giảng viên hướng dẫn phụ thuộc chuyên ngành đó hay không ? Nếu đúng thì nếu mà có quá nhiều sinh viên chọn cùng 1 chuyên ngành mà chỉ có ít giảng viên dạy chuyên ngành đó thì sao ?	- Nhà trường hiện đang dựa vào nguyện vọng đăng ký của sinh viên qua phiếu khảo sát để có thể sắp xếp giảng viên hướng dẫn, bên cạnh đó nhà trường cũng sẽ dựa theo hệ số bậc hàm của giảng viên đó để chia ví dụ: GS, ThS,... Trường hợp mà quá nhiều sinh viên chọn một giảng viên hay ngược lại giảng viên có ít sinh viên chọn sẽ phải dựa vào tiêu chí phụ như điểm GPA, thành tích,... để ưu tiên để đảm bảo rằng các giảng viên sẽ được phân bổ đều cho các giảng viên hướng dẫn
4	- Hiện tại, quy trình duyệt đề cương đồ án tốt nghiệp của trường như thế nào ạ ?	- Sinh viên sẽ đăng ký đề cương đồ án và gửi lên khoa.

		Khoa sẽ thành lập nhóm xét duyệt đồ án và phân các giảng viên đó vào các nhóm, đảm bảo rằng không có đề cương đồ án của sinh viên mà giảng viên đó hướng dẫn ở trong nhóm đó. Sau đó nhóm sẽ tiến hành đánh giá và xem xét, thư kí nhóm sẽ giữ vai trò tổng hợp lại nhận xét của các thành viên trong nhóm để gửi lại với giảng viên hướng dẫn của sinh viên đó. Từ đó giảng viên hướng dẫn và sinh viên sẽ điều chỉnh theo nhận xét của nhóm xét duyệt
5	- Hiện tại giảng viên hướng dẫn có thể quản lý sinh viên qua đâu ạ ?	- Hiện tại giảng viên hướng dẫn chưa có hệ thống để quản lý báo cáo, đồ án của sinh viên mà đều qua sự thống nhất lập hội nhóm qua zalo, facebook,... các mạng xã hội bên ngoài để trao đổi thông tin
6	- Việc phân chia hội đồng chấm thi và giảng viên phản biện hiện tại có theo quy tắc nào không ạ ?	- Việc phân chia hội đồng thi vẫn đang được chia ngẫu nhiên và theo nguyên tắc là hội đồng đó không có giảng viên hướng dẫn sinh viên có trong hội đồng. Còn đối với giảng viên phản biện sẽ là giảng viên có trong hội đồng bảo vệ đó
7	- Việc cho điểm thì sẽ có những đầu điểm nào và được tính như nào ạ ?	- Đối với điểm Quá trình sẽ được tính bằng trung bình

		<p>của 2 đầu điểm sau:</p> <ul style="list-style-type: none"> + Điểm giảng viên hướng dẫn + Điểm giảng viên phản biện - Điểm thi là điểm trung bình hội đồng bảo vệ: Điểm này sẽ dựa trên kết quả chấm của thành viên trong hội đồng từ đó sẽ công vào và chia điểm lấy trung bình - Từ hai điểm trên ta sẽ tính trung bình để ra được điểm tổng kết của học phần bảo vệ đồ án tốt nghiệp
8	- Nếu mà sinh viên muốn đổi đề tài thì có được đổi ko, nếu dc đổi thì thời hạn đổi là đến khi nào ?	- Sinh viên sẽ không được đổi đề tài sau khi đã nộp bản đề cương đồ án.
9	- Với quy trình hiện tại có đang gặp khó khăn nào ạ ?	- Việc bảo vệ đồ án đang chưa có nơi quản lý tập trung và mọi thứ đang được xử lý thủ công như chia giảng viên hướng dẫn, hội đồng chấm thi, hội đồng xét duyệt,...

1.2. Mục tiêu và yêu cầu của hệ thống

Từ những khó khăn đó và trải nghiệm của bản thân, em quyết định lựa chọn đề tài **“Quản lý đồ án tốt nghiệp cho sinh viên khoa công nghệ thông tin”** với mục tiêu là tạo ra một môi trường trực tuyến giúp sinh viên dễ dàng theo dõi thông tin về đồ án, đăng ký bảo vệ, và tương tác với giảng viên hướng dẫn. Đồng thời, hệ thống cũng sẽ hỗ trợ ban bảo vệ và quản lý trong việc tổ chức lịch trình, đánh giá đồ án, và quản lý kết quả, giúp sinh viên tiết kiệm thời gian trong việc làm đồ án và dễ dàng theo dõi quá trình và điểm làm đồ án. Giảng viên sẽ dễ dàng hơn trong việc quản lý các giấy tờ và đầu điểm của các học sinh mà giảng viên đó hướng dẫn. Khoa sẽ có thể dễ dàng quản lý và phân chia giảng viên, sinh viên, hội đồng phòng thi dễ dàng hơn.

Đối với Khoa công nghệ thông tin, có thể thêm được sinh viên và giảng viên cũng như quản lý được danh sách sinh viên và giảng viên hướng dẫn làm đồ án theo từng học kỳ. Ngoài ra hệ thống sẽ tự động chia giảng viên hướng dẫn, giảng viên phản biện, nhóm đánh giá và hội đồng phòng thi. Khoa có thể lập lịch lên kế hoạch để cho giảng viên và sinh viên thực hiện theo tiến trình. Khoa có thể xem được tiến độ làm đồ án của từng sinh viên. Khoa cũng có thể up mẫu báo cáo lên cho sinh viên và giảng viên tải về để in ra và điền vào. Sau khi kết thúc kì làm đồ án Khoa có thể xuất báo cáo tổng kết điểm của toàn khóa.

Đối với giảng viên có thể xem được danh sách sinh viên mà người đó quản lý qua các kỳ học, lập lịch báo cáo hàng tuần để đưa ra nhận xét và đánh giá cho sinh viên và cho điểm sinh viên, xem được hội đồng phòng thi, xem được danh sách sinh viên mà giảng viên đó đảm nhiệm dưới vai trò là giảng viên phản biện, xem được vai trò và danh sách sinh viên trong hội đồng xét duyệt và hội đồng chấm thi.

Đối với sinh viên có thể xem được điểm, thông tin cá nhân và giảng viên hướng dẫn, giảng viên phản biện và hội đồng phòng bảo vệ. Sinh viên báo cáo hàng tuần với giảng viên hướng dẫn, xem được nhận xét từ hội đồng xét duyệt và giảng viên phản biện. Theo dõi được từng đầu điểm, sinh viên có thể xuất đề cương báo cáo.

1.3. Phạm vi đề tài

Sau khi khảo sát quá trình làm đồ án của các trường đại học, em đã rút ra được quy trình để tối ưu bước quản lý, phục vụ dễ dàng hơn cho việc quản lý. Đối tượng hướng tới trong đề tài của em sẽ là các bạn bảo vệ đồ án của khoa công nghệ thông tin, các thầy cô sẽ tham gia vào quá trình làm đồ án của các sinh viên và các thầy cô quản lý khoa Công nghệ thông tin của trường đại học Giao thông vận tải

CHƯƠNG 2: CÁC CÔNG NGHỆ SỬ DỤNG

2.1. Nhận định nền tảng sử dụng

Trước khi triển khai hệ thống quản lý bảo vệ đồ án tốt nghiệp, việc lựa chọn nền tảng là quan trọng để đảm bảo tính ổn định, linh hoạt, và khả năng mở rộng của hệ thống. Dựa trên nghiên cứu và đánh giá, đối tượng sử dụng chính là các giảng viên và sinh viên cũng như các thầy cô quản lý quản lý của khoa Công nghệ thông tin. Chính vậy em quyết định chọn nền tảng Web để phát triển sản phẩm của mình vì Web có thể truy cập vào đa nền tảng, từ máy tính cho đến điện thoại chỉ cần có truy cập Internet là có thể truy cập vào ứng dụng.

2.2. Công nghệ hệ thống

Để đảm bảo hệ thống được vận hành tốt em sẽ phân chia hệ thống thành hai bộ phận đó là:

- Frontend: Phần này sẽ đảm nhiệm xử lý giao diện và những logic xử lý để hiển thị nên giao diện người dùng, giúp người dùng dễ dàng tương tác với hệ thống
- Backend: Phần này sẽ đảm nhiệm xử lý logic chính của nền tảng sau khi nhận được yêu cầu từ phía Frontend để xử lý rồi trả lại phản hồi sang bên phía Frontend. Dữ liệu trả về sẽ dưới dạng API, để giúp cho việc xử lý bên Frontend trở lên dễ dàng hơn

2.2.1. Các công nghệ sử dụng ở phía Frontend

a) HTML (Hypertext Markup Language)

HTML là viết tắt của "HyperText Markup Language" (Ngôn ngữ Đánh dấu Siêu văn bản). Đây là một ngôn ngữ lập trình dùng để tạo và định dạng nội dung trên các trang web. HTML sử dụng các thẻ (tags) để mô tả cấu trúc và định dạng của các phần tử trên trang web, như tiêu đề, đoạn văn bản, hình ảnh, liên kết và bảng. HTML không phải là ngôn ngữ lập trình. Điều này có nghĩa là nó không thể thực hiện các chức năng "động". Ta có thể hình dung cũng tương tự như phần mềm Microsoft Word, HTML chỉ có tác dụng bố cục và định dạng trang web. HTML khi kết hợp với CSS và JavaScript sẽ thực sự tạo ra một trang web hoàn chỉnh.

Mặc dù HTML được coi là ngôn ngữ lập trình, nhưng nó không phải là một ngôn ngữ lập trình theo nghĩa truyền thống. HTML không thể thực hiện các chức năng "động" như các ngôn ngữ lập trình khác như JavaScript. Thay vào đó, HTML đóng vai trò quan trọng trong việc bố cục và định dạng nội dung trên trang web. Bạn có thể tưởng tượng HTML giống như việc sử dụng Microsoft Word để tạo ra và định dạng tài liệu văn bản. Nó chỉ cung cấp các công cụ để tổ chức và hiển thị nội dung, chứ không thể tự mình xử lý logic hay tính toán.

Với HTML, các nhà phát triển web có thể tạo ra cấu trúc cơ bản của trang web và đảm bảo rằng nội dung được trình bày một cách có tổ chức và hợp lý. Khi kết hợp với CSS và JavaScript, HTML giúp tạo nên trải nghiệm người dùng phong phú và linh hoạt, đáp ứng được nhiều yêu cầu và mong đợi khác nhau từ người dùng. Việc hiểu rõ và sử dụng thành thạo HTML là bước đầu tiên quan trọng để trở thành một nhà phát triển web chuyên nghiệp.

b) SCSS

Viết tắt của "Sassy CSS", là một ngôn ngữ dấu đầu tương thích ngược với CSS (Cascading Style Sheets). SCSS cung cấp các tính năng và cú pháp mở rộng hơn so với CSS truyền thống, giúp cho việc viết và quản lý mã CSS trở nên dễ dàng hơn.

c) Tailwind CSS

Là một framework CSS hướng utility, được thiết kế để tăng cường tốc độ phát triển và linh hoạt trong việc xây dựng giao diện người dùng. Nó cung cấp một bộ sưu tập các class utility đã được định sẵn, cho phép bạn áp dụng các thuộc tính và phong cách vào các phần tử trong trang web một cách nhanh chóng.

SCSS cung cấp khả năng sử dụng biến để lưu trữ các giá trị CSS, cho phép lồng ghép các quy tắc CSS giúp mã nguồn dễ đọc và duy trì hơn. Nó hỗ trợ các phép toán học trên các giá trị CSS, cho phép thực hiện các phép cộng, trừ, nhân và chia, giúp tối ưu hóa việc tính toán giá trị. Tính năng kế thừa cho phép một selector kế thừa các quy tắc của selector khác, giúp giảm thiểu sự lặp lại mã. SCSS còn cung cấp khả năng tạo các khối mã có thể tái sử dụng thông qua mixin, giúp đơn giản hóa việc áp dụng các quy tắc CSS phức tạp ở nhiều nơi trong mã nguồn.

Với các tính năng nâng cao như vậy, SCSS giúp các nhà phát triển web tiết kiệm thời gian và công sức trong việc viết và duy trì mã CSS, đồng thời cải thiện hiệu suất và tính nhất quán của trang web. Việc sử dụng SCSS cho phép tổ chức mã nguồn một cách logic và hiệu quả, từ đó tạo ra các dự án web chất lượng cao hơn.

d) Typescript

Là một ngôn ngữ lập trình mã nguồn mở dựa trên JavaScript, được phát triển bởi Microsoft. Nó là một "superset" của JavaScript, có nghĩa là mã TypeScript có thể chạy trực tiếp trên trình duyệt web hoặc trình thực thi JavaScript mà không cần qua bất kỳ bước biên dịch nào. Một trong những đặc điểm nổi bật của TypeScript là hỗ trợ kiểu dữ liệu tĩnh. Điều này có nghĩa là bạn có thể định rõ các kiểu dữ liệu cho biến, tham số hàm và giá trị trả về, giúp phát hiện lỗi và cung cấp gợi ý thông qua công cụ hỗ trợ phát triển (IDE) trong quá trình viết mã. Kiểu dữ liệu tĩnh giúp tăng tính tin cậy và khả năng bảo trì của mã nguồn, đồng thời giúp đội phát triển làm việc cùng nhau hiệu quả hơn.

e) ReactJs

Là một framework JavaScript phổ biến được sử dụng rộng rãi trong phát triển ứng dụng web hiện đại. Với cách tiếp cận khái niệm "Single-Page Application" (SPA), ReactJS cho phép xây dựng giao diện người dùng linh hoạt, tương tác và hiệu quả. Một điểm mạnh của ReactJS là khả năng tái sử dụng thành phần (component-based). Thay vì xây dựng toàn bộ giao diện từ đầu, ReactJS cho phép chia nhỏ các thành phần giao diện thành các phần nhỏ hơn, có thể được sử dụng lại trong nhiều vị trí khác nhau. Điều này giúp tăng tính module, giảm sự lặp lại và dễ dàng bảo trì mã nguồn. ReactJS cũng hỗ trợ việc quản lý trạng thái ứng dụng một cách hiệu quả thông qua khái niệm "Virtual DOM" (Document Object Model). ReactJS cũng rất linh hoạt trong việc tích hợp với các thư viện và framework khác. Với cộng đồng lớn và sự hỗ trợ mở rộng, có rất nhiều gói mở rộng (packages) và công cụ hữu ích cho ReactJS như Redux, React Router, Axios, và nhiều hơn nữa. Điều này giúp tăng tốc quá trình phát triển và mở rộng khả năng của ứng dụng.

Một điểm mạnh của ReactJS là khả năng tái sử dụng thành phần (component-based). Thay vì xây dựng toàn bộ giao diện từ đầu, ReactJS cho phép chia nhỏ các thành phần giao diện thành các phần nhỏ hơn, có thể được sử dụng lại trong nhiều vị trí khác nhau. Điều này giúp tăng tính module, giảm sự lặp lại và dễ dàng bảo trì mã nguồn. ReactJS sử dụng cú pháp JSX, một phần mở rộng của JavaScript, để viết các mẫu giao diện người dùng. JSX kết hợp cú pháp HTML và JavaScript trong cùng một file, giúp người phát triển tạo ra mã nguồn giao diện người dùng một cách rõ ràng và dễ đọc. JSX cũng cho phép viết các thành phần React và kết hợp chúng lại với nhau để tạo thành các ứng dụng phức tạp.

ReactJS cũng hỗ trợ việc quản lý trạng thái ứng dụng một cách hiệu quả thông qua khái niệm "Virtual DOM" (Document Object Model). ReactJS rất linh hoạt trong việc tích hợp với các thư viện và framework khác. Với cộng đồng lớn và sự hỗ trợ mở rộng, có rất nhiều gói mở rộng (packages) và công cụ hữu ích cho ReactJS như Redux, React Router, Axios, và nhiều hơn nữa. Điều này giúp tăng tốc quá trình phát triển và mở rộng khả năng của ứng dụng.

ReactJS không chỉ hỗ trợ phát triển trên môi trường web, mà còn có thể được sử dụng để xây dựng các ứng dụng di động bằng React Native. React Native cho phép người phát triển xây dựng ứng dụng di động đa nền tảng (cross-platform) với hiệu suất gần như tương đương với ứng dụng native. Điều này giúp tiết kiệm thời gian và công sức trong việc phát triển và duy trì ứng dụng trên nhiều nền tảng khác nhau.

Ngoài ra, ReactJS được phát triển và duy trì bởi Facebook, một trong những công ty công nghệ hàng đầu thế giới. Điều này đảm bảo rằng ReactJS được cập nhật và phát

triển liên tục, đồng thời có được sự hỗ trợ và tài liệu phong phú từ cộng đồng phát triển.

f) MUI (Material-UI)

Material-UI (MUI) là một thư viện giao diện người dùng (UI) mã nguồn mở dành cho React, phát triển dựa trên các nguyên tắc thiết kế Material Design của Google. MUI mang lại sự tiện lợi và hiệu quả cho các nhà phát triển web bằng cách cung cấp một loạt các thành phần giao diện người dùng phong phú và dễ sử dụng, giúp tạo ra các ứng dụng web đẹp mắt, hiện đại và nhất quán.

MUI tuân thủ nghiêm ngặt các nguyên tắc thiết kế Material Design, mang đến trải nghiệm người dùng trực quan, nhất quán và có tính thẩm mỹ cao. Các nguyên tắc này giúp người dùng dễ dàng tương tác và cảm thấy thoải mái khi sử dụng các ứng dụng web được phát triển với MUI. Một trong những ưu điểm nổi bật của MUI là nó cung cấp một bộ sưu tập đa dạng các thành phần giao diện người dùng như nút bấm, hộp thoại, biểu mẫu, thanh công cụ, bảng, thẻ và nhiều thành phần khác. Những thành phần này không chỉ được thiết kế sẵn và có thể dễ dàng sử dụng mà còn được tối ưu hóa để hoạt động tốt trên nhiều loại thiết bị và trình duyệt khác nhau, giúp tiết kiệm thời gian và công sức cho các nhà phát triển.

Khả năng tùy biến mạnh mẽ là một điểm mạnh khác của MUI. Thư viện này cho phép tùy chỉnh giao diện một cách linh hoạt thông qua hệ thống chủ đề (theme), cho phép các nhà phát triển dễ dàng thay đổi màu sắc, kiểu dáng và các thuộc tính khác của các thành phần để phù hợp với thương hiệu và phong cách của ứng dụng. Hệ thống chủ đề này không chỉ giúp duy trì sự nhất quán trong thiết kế mà còn giúp tạo ra những trải nghiệm người dùng độc đáo và hấp dẫn.

MUI được thiết kế đặc biệt để tích hợp tốt với React, một trong những thư viện JavaScript phổ biến nhất hiện nay để xây dựng giao diện người dùng. Sự tích hợp này giúp các nhà phát triển dễ dàng sử dụng và kết hợp MUI với các tính năng khác của React, từ đó tăng cường khả năng phát triển ứng dụng web phức tạp và giàu tính năng. Khả năng tương tác mượt mà giữa MUI và React giúp đơn giản hóa quy trình phát triển, cho phép các nhà phát triển tập trung vào việc tạo ra các tính năng và cải tiến mới mà không cần lo lắng quá nhiều về các vấn đề tương thích.

Ngoài ra, MUI còn có tài liệu hướng dẫn chi tiết và một cộng đồng hỗ trợ tích cực. Tài liệu hướng dẫn cung cấp các ví dụ, hướng dẫn sử dụng và các phương pháp tốt nhất để tận dụng tối đa các tính năng của MUI. Cộng đồng hỗ trợ tích cực gồm các nhà phát triển từ khắp nơi trên thế giới, sẵn sàng chia sẻ kinh nghiệm, giải đáp thắc mắc và cung cấp giải pháp cho các vấn đề gặp phải khi sử dụng thư viện này. Điều này giúp các nhà phát triển nhanh chóng giải quyết các khó khăn và tối ưu hóa quá trình phát triển ứng dụng.

Với tất cả những đặc điểm và ưu điểm trên, Material-UI không chỉ giúp tăng tốc quá trình phát triển mà còn đảm bảo rằng các ứng dụng web được xây dựng theo tiêu chuẩn cao, mang lại trải nghiệm người dùng tốt nhất và đáp ứng được yêu cầu ngày càng cao của thị trường. Material-UI đã trở thành một công cụ không thể thiếu cho nhiều nhà phát triển web, giúp họ tạo ra những sản phẩm chất lượng, đẹp mắt và dễ sử dụng.

2.2.2. Các công nghệ sử dụng ở phía Backend

a, ASP.NET Core

ASP.NET là một nền tảng phát triển ứng dụng web mạnh mẽ và linh hoạt, được hoàn thiện bởi Microsoft vào 2012 và là một giải pháp dùng để hỗ trợ quá trình sản xuất các loại website và nội dung động. Sự ra đời của ASP Net cho phép các nhà phát triển công nghệ, lập trình viên dễ dàng hơn trong quá trình xây dựng các trang web, ứng dụng hay các dịch vụ có sử dụng nội dung động.

Với ASP.NET Core, phiên bản open-source mới nhất của nền tảng ASP.NET hỗ trợ phát triển ứng dụng web chạy trên đa nền tảng, được phát hành lần đầu tiên vào năm 2016 và là thiết kế lại của các phiên bản ASP.NET chỉ dành cho window trước đó. ASP.NET Core cung cấp các công cụ và thư viện mạnh mẽ để phát triển ứng dụng web nhanh chóng và an toàn. Điều này mang lại sự linh hoạt lớn cho việc triển khai ứng dụng trên các môi trường máy chủ khác nhau. ASP.NET Core được thiết kế để cho phép các thành phần thời gian chạy, API, trình biên dịch và ngôn ngữ phát triển nhanh chóng trong khi vẫn cung cấp nền tảng ổn định và được hỗ trợ để duy trì hoạt động của ứng dụng. Nhiều phiên bản ASP.NET Core có thể tồn tại cạnh nhau trên cùng một máy chủ. Có nghĩa là một ứng dụng có thể sử dụng phiên bản mới nhất, trong khi các ứng dụng khác vẫn tiếp tục chạy trên phiên bản mà chúng đã được thử nghiệm.

ASP.NET Core có thể xây dựng ứng dụng web đa tầng, từ ứng dụng web phía máy chủ (server-side rendering) đến ứng dụng web phía khách hàng (client-side rendering) hoặc phát triển các restful API để cung cấp dữ liệu cho các ứng dụng di động và trình duyệt.

Không chỉ vậy, ASP.NET Core còn cung cấp hiệu suất và khả năng mở rộng cao, cho phép các nhà phát triển có thể xây dựng mọi ứng dụng từ đơn giản đến phức tạp với hiệu năng mạnh mẽ và khả năng linh hoạt tối ưu. ASP.NET core sở hữu các tính năng thân thiện với nhà phát triển như khả năng triển khai tự động, khả năng điều khiển kéo/thả trên máy chủ, khả năng giám sát, quản lý chặt chẽ mọi quy trình để kịp thời sử dụng các ứng dụng mới để thay thế những yêu cầu chết. Bên cạnh đó, ASP.NET Core được tối ưu hóa để đạt hiệu suất cao. Nó sử dụng mô hình xử lý yêu cầu và phản hồi không đồng bộ, giúp ứng dụng xử lý đồng thời nhiều yêu cầu mà

không bị chặn. Ngoài ra, ASP.NET Core cũng hỗ trợ bộ đệm, nén dữ liệu và các kỹ thuật tối ưu hóa khác để cải thiện tốc độ tải trang. Với việc hỗ trợ các giao thức xác thực tiêu chuẩn trong ngành và tính năng bảo mật tích hợp, ASP.NET Core có thể bảo vệ ứng dụng chống lại các cuộc tấn công như cross-site scripting (XSS) và cross-site request forgery (CSRF).

Với sự kết hợp của tính linh hoạt, hiệu suất, bảo mật và hỗ trợ lớn từ cộng đồng, ASP.NET đã trở thành một công cụ ưu việt cho việc phát triển ứng dụng web.

b, My SQL

MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở phổ biến và mạnh mẽ, được sử dụng rộng rãi trên toàn thế giới. Được phát triển, hỗ trợ và duy trì bởi Oracle Corporation, MySQL đã trở thành một phần quan trọng của nền tảng công nghệ thông tin hiện đại.

Mã nguồn mở của MySQL không chỉ mang lại sự tiết kiệm chi phí mà còn cho phép các nhà phát triển tùy chỉnh và mở rộng hệ thống một cách linh hoạt. Điều này làm cho MySQL trở thành một lựa chọn phổ biến cho cả các dự án mã nguồn mở nhỏ và các doanh nghiệp lớn đòi hỏi hiệu suất và tính ổn định cao.

MySQL cung cấp một loạt các tính năng mạnh mẽ như truy vấn SQL mạnh mẽ, hỗ trợ giao thức ACID (Atomicity, Consistency, Isolation, Durability) để đảm bảo tính nhất quán và an toàn của dữ liệu, cùng với các công cụ quản trị dữ liệu linh hoạt và dễ sử dụng. Điểm mạnh của MySQL nằm ở khả năng mở rộng, từ các dự án nhỏ với lượng dữ liệu nhỏ đến các hệ thống phân tán lớn với hàng triệu hàng tỷ bản ghi.

Một trong những ưu điểm quan trọng của MySQL là khả năng tương thích cao. Nó hoạt động trên nhiều nền tảng hệ điều hành, bao gồm Linux, Windows và macOS, và hỗ trợ nhiều ngôn ngữ lập trình phổ biến như PHP, Python, Java và nhiều ngôn ngữ khác.

Không chỉ là một công cụ lưu trữ dữ liệu, MySQL còn là một nền tảng cung cấp các tính năng linh hoạt cho các ứng dụng web và doanh nghiệp. Từ các trang web thương mại điện tử đến hệ thống quản lý quan hệ khách hàng (CRM) và hệ thống quản lý nội dung (CMS), MySQL là một lựa chọn ưu việt cho việc lưu trữ và quản lý dữ liệu trong môi trường web và doanh nghiệp ngày nay.

CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

3.1. Tóm tắt yêu cầu của hệ thống

Nhiệm vụ của hệ thống sẽ nhằm mục đích giảm thiểu thời gian, quy trình giữa sinh viên và giảng viên trong quá trình làm đồ án, bên cạnh cũng giúp cho khoa có thể dễ dàng quản lý quá trình bảo vệ đồ án diễn ra dễ dàng hơn.

Dựa vào quá trình làm đồ án thực tế hiện tại, em đã rút ra được các yêu cầu hệ thống phải đáp ứng được như sau:

- Trước khi đăng ký đồ án, sinh viên sẽ đăng ký hướng chọn đề tài và giảng viên hướng dẫn, khoa sẽ dựa vào đó để phân chia giảng viên hướng dẫn cho sinh viên => *Cần một chức năng cho sinh viên có thể đăng ký giảng viên hướng dẫn và chia giảng viên hướng dẫn cho sinh viên*
- Sau khi thống nhất được tên đề tài, phạm vi, công nghệ... với giảng viên hướng dẫn, sinh viên sẽ phải viết đề cương đồ án để gửi lên trên Khoa, từ đó Khoa sẽ thành lập nhóm xét duyệt đề cương và gửi nhận xét về cho sinh viên => *Cần chức năng đăng ký đề cương đề tài để gửi lên trên khoa, khoa thành lập nhóm hội đồng để xét duyệt đề cương và gửi về nhận xét cho từng sinh viên đó.*
- Giảng viên hướng dẫn sẽ tiến hành hướng dẫn sinh viên làm đồ án => *Yêu cần một hệ thống có thể giúp giảng viên hướng dẫn theo dõi được quá trình làm đồ án của sinh viên mà mình hướng dẫn*
- Sau khi kết thúc quá trình hướng dẫn, khoa sẽ thành lập hội đồng và phân giảng viên phản biện cho từng sinh viên, giảng viên phản biện sau khi duyệt xong sẽ cho điểm và nhận xét => *Yêu cầu hệ thống sẽ phải phân được giảng viên phản biện và hội đồng cho sinh viên, giảng viên phản biện có thể đưa ra nhận xét và điểm cho sinh viên.*
- Mỗi giảng viên trong hội đồng có thể xem được danh sách và cho điểm, nhận xét của sinh viên trong hội đồng đó, đồng thời cũng xem được chức vụ của mình trong hội đồng đó => *Yêu cầu giảng viên trong hội đồng có thể cho điểm và cho nhận xét*
- Khoa có thể in ra bảng điểm đồ án sau khi sinh viên bảo vệ hoàn tất => *Hệ thống phải quản lý điểm và xuất báo cáo điểm của tất cả sinh viên trong trường*
- Khoa có thể tạo sinh viên, giảng viên học kỳ và lập kết hoạch cho học kỳ đó => *Chức năng quản trị viên*
- Có thể nhiều sinh viên và giảng viên có thể truy cập vào hệ thống => *Chức năng đăng ký/đăng nhập, phân quyền*

3.2. Thiết kế hệ thống

3.2.1 Mô tả tác nhân

Bảng 3.1. Bảng mô tả tác nhân

Tên tác nhân	Công việc/Vai trò
Admin	<ul style="list-style-type: none"> - Đăng nhập - Đăng xuất - Quản lý kỳ học - Quản lý và lập kế hoạch làm đồ án - Quản lý sinh viên - Quản lý giảng viên - Quản lý được nhóm xét duyệt - Quản lý hội đồng phòng thi - Quản lý biểu mẫu đăng ký của đề cương, biểu mẫu cho điểm của giảng viên hướng dẫn và phản biện
Sinh viên	<ul style="list-style-type: none"> - Đăng nhập - Nộp đồ án hàng tuần - Đăng ký đề cương đồ án - Xem được nhận xét của giảng viên hướng dẫn, phản biện, nhóm xét duyệt và hội đồng thi
Giảng viên	<ul style="list-style-type: none"> - Đăng nhập - Xem được danh sách sinh viên hướng dẫn, sinh viên phản biện, danh sách sinh viên trong hội đồng phòng thi. - Xem được vai trò trong nhóm xét duyệt. - Xem được vai trò trong hội đồng phòng thi và cho điểm. - Cho điểm và đánh giá sinh viên hàng tuần nếu là giảng viên hướng dẫn. - Nếu là thư ký trong nhóm xét duyệt thì sẽ gửi thông báo về cho giảng viên và sinh viên.

3.2.2 Mô tả chức năng hệ thống

- **Đăng ký người dùng:** Người có quyền quản trị sẽ được quyền tạo người dùng và gán cho người dùng đó vai trò là sinh viên hoặc giảng viên.
- **Đăng nhập:** Giảng viên và sinh viên đều có quyền đăng nhập vào hệ thống
- **Quản lý người dùng:** Quản trị viên có thể xem, thêm, sửa, xóa và cập nhật trạng thái người dùng là giảng viên và sinh viên
- **Quản lý học kỳ:** Chỉ có người có quyền quản trị mới có thể thêm, sửa học kỳ
- **Quản lý chuyên ngành:** Chỉ có người có quyền quản trị mới có thể thêm, sửa, xóa chuyên ngành
- **Quản lý học vấn:** Chỉ có người có quyền quản trị mới có thể thêm, sửa, xóa học vấn
- **Lập kế hoạch đồ án học kỳ:** Quản trị viên sau khi lập học kỳ có thể lập kế hoạch làm đồ án cho học kỳ đó để gửi đến sinh viên và giảng viên. Trong lập lịch có hai loại lịch đó là lịch *phi chức năng* và *lịch chức năng*, lịch chức năng gồm 5 loại lịch sau:
 - Lịch nộp đề cương: Sau khoảng thời gian này sinh viên sẽ không thể chỉnh sửa thông tin đề cương đồ án của kỳ học năm đó nữa.
 - Lịch nhận xét điểm GVHD: Khi đến khoảng thời gian này sẽ hiển thị mẫu nhập cho giảng viên hướng dẫn đánh giá sinh viên đó và sau khoảng thời gian này thì không thể thay đổi thông tin của sinh viên đó nữa
 - Lịch nhận xét điểm GVPB: Khi đến khoảng thời gian này sẽ hiển thị mẫu nhập cho giảng viên phản biện đánh giá sinh viên đó và sau khoảng thời gian này thì không thể thay đổi thông tin của sinh viên đó nữa
 - Lịch nộp báo cáo tổng kết: Khi đến khoảng thời gian này thì sinh viên có thể tải lên bản báo cáo cuối cùng và giảng viên có thể tải bản báo cáo cuối cùng này về để đánh giá
 - Lịch chấm điểm tổng kết: Khi đến khoảng thời gian này giảng viên trong hội đồng có thể cho điểm, sau khoảng thời gian này thì giảng viên không thể sửa điểm cho sinh viên
- **Quản trị biểu mẫu:** Quản trị viên có thể quản lý biểu mẫu bao gồm:
 - Biểu mẫu đánh giá của giảng viên hướng dẫn
 - Biểu mẫu đánh giá của giảng viên phản biện
 - Biểu mẫu đề cương đồ án
- **Gán giảng viên hướng dẫn:** Quản trị viên có thể gán giảng viên hướng dẫn cho từ sinh viên bằng tay hoặc chia tự động có sẵn của hệ thống
- **Phân hội đồng phòng thi**
 - Quản trị viên có thể thêm, sửa, xóa hội đồng phòng thi

- Quản trị viên có thể phân sinh viên vào các hội đồng bằng tay hoặc dùng thuật toán của hệ thống
- Xuất được bảng điểm giảng sinh viên trong hội đồng
- **Gán giảng viên phản biện:** Sau khi phân hội đồng, quản trị viên có thể gán giảng viên phản biện cho sinh viên trong hội đồng đó bằng tay hoặc dùng thuật toán của hệ thống
- **Quản lý đề cương đồ án**
 - Đối với sinh viên có thể cập nhật đề cương đồ án sau khi đến lịch nộp đề cương, sau khi hết thời hạn sinh viên có thể in đồ án thành bản cứng
 - Sau khi kết thúc lịch làm đồ án giảng viên hướng dẫn có thể in đề cương của từng sinh viên thành bản cứng
 - Đối với quản trị thì có thể cập nhật trạng thái của đồ án đó và sau khi kết thúc thời gian nộp đề cương cũng có thể in bản cứng
- **Lập lịch đánh giá hàng tuần:** Giảng viên hướng dẫn có thể lập lịch báo cáo hàng tuần cho sinh viên mình hướng dẫn hàng tuần

3.2.3. Mô tả các chức năng tự động của hệ thống

a, Chức năng tự động gán giảng viên

- Hệ thống sẽ lọc ra những sinh viên có trong học kỳ với điều kiện sinh viên đó phải còn hoạt động, chưa bị xóa, đã có chuyên ngành và chưa có giảng viên hướng dẫn và sắp xếp theo điểm GPA giảm dần
- Hệ thống cũng lấy ra giảng viên có trong học kỳ và giảng viên đó phải còn hoạt động, chưa bị xóa, đã có chuyên ngành và đã có học vấn
- Kiểm tra sinh viên đó đăng kí giảng viên nào nếu giảng viên còn có thể nhận thêm sinh viên thì gán sinh viên vào giảng viên đó còn nếu không thì sẽ phải lấy danh sách giảng viên của chuyên ngành đó để gán ngẫu nhiên.
- Trường hợp cuối cùng là nếu giảng viên mà sinh viên đăng kí hết suất và giảng viên của chuyên ngành đó thì sẽ gán ngẫu nhiên trong tất cả các giảng viên

b, Chức năng tự động chia nhóm xét duyệt

- Hệ thống sẽ lấy ra những đề cương đã có giáo viên hướng dẫn, chưa có nhóm xét duyệt, trạng thái của đề cương phải ở trạng thái đang làm đồ án.
- Hệ thống cũng lấy ra các giảng viên hợp lệ trong học kì đó đồng thời cũng lấy ra danh sách các nhóm xét duyệt hợp lệ
- Hệ thống sẽ chia các giảng viên vào trong từng nhóm xét duyệt một cách đồng đều nhất.
- Hệ thống sẽ lặp qua từng đề cương đồ án và kiểm tra xem giảng viên hướng dẫn của đề cương đồ án đó đang nằm trong nhóm xét duyệt nào thì loại ra. Sau đó sẽ ngẫu nhiên gán sinh viên đó vào trong nhóm hội đồng xét duyệt còn lại

c, Chức năng tự động chia hội đồng và giảng viên phản biện

- Hệ thống sẽ gán ngẫu nhiên giảng viên hợp lệ vào trong hội đồng chấm thi với vai trò ngẫu nhiên (Sẽ ưu tiên vai trò Chủ tịch và Thư ký trước). Trong trường hợp mà giảng viên nhiều hơn số hội đồng phòng thi cho phép thì hệ thống sẽ tự tạo ra hội đồng để thêm giảng viên đó vào trong hội đồng.
- Hệ thống sẽ lấy danh sách các đồ án có trạng thái là được chấp nhận bảo vệ để xét. Từ đó hệ thống sẽ tính toán được trung bình một hội đồng sẽ cần phải có bao nhiêu sinh viên.
- Hệ thống sẽ lặp qua từng hội đồng. Mỗi hội đồng sẽ lấy ra được danh sách các sinh viên không có giảng viên nằm trong hội đồng ấy và sẽ gán lần lượt vào trong hội đồng đó cho đến khi đủ chỉ tiêu của hội đồng đó thì dừng.
- Hệ thống cũng sẽ lặp qua danh sách các sinh viên thêm một lần nữa để kiểm tra xem còn xét sinh viên nào không. Nếu có thì cũng sẽ loại bỏ hội đồng chứa giảng viên hướng dẫn của sinh viên đó và gán ngẫu nhiên vào các hội đồng còn lại.

d, Chức năng tự động chia giảng viên phản biện

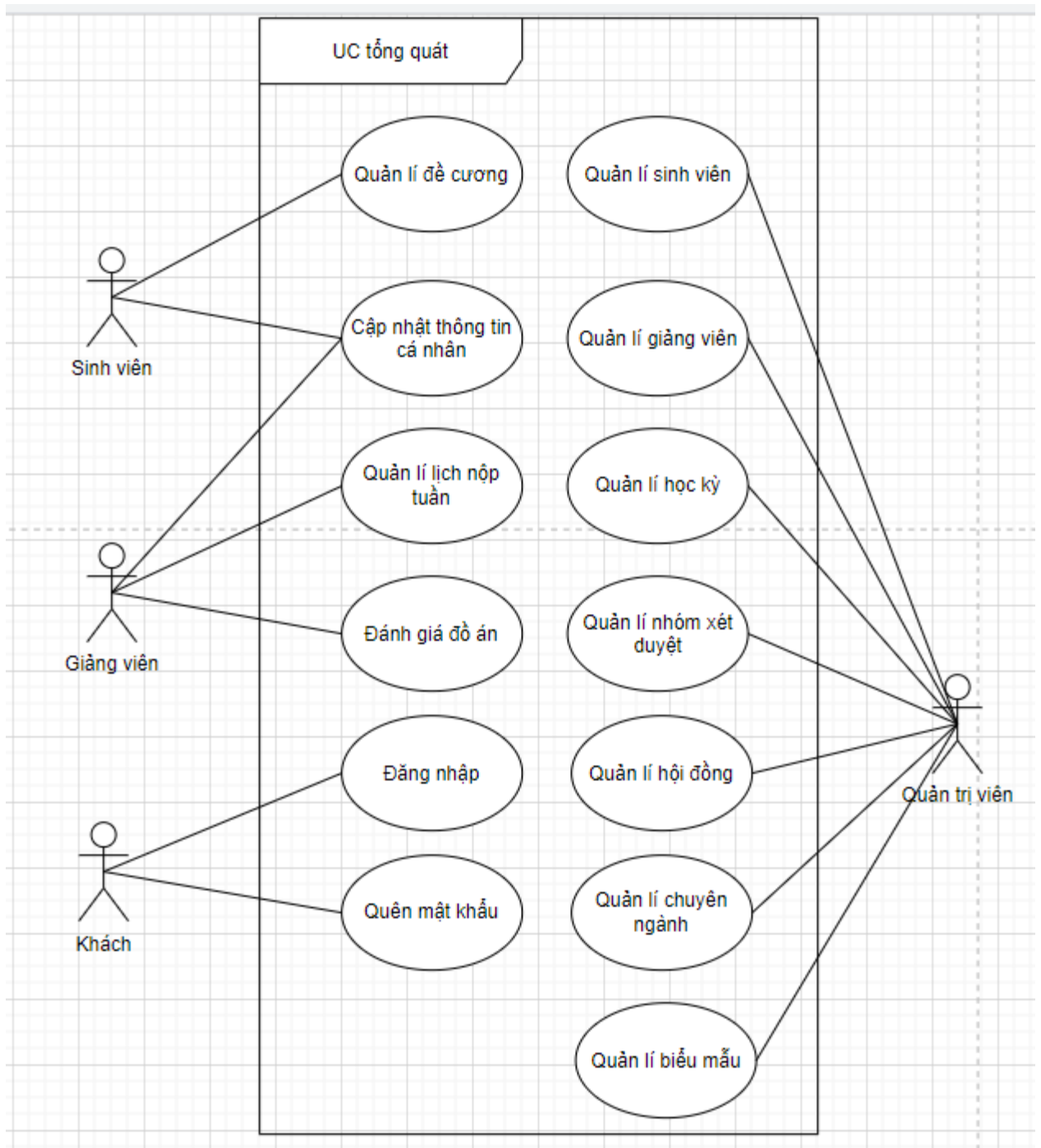
- Sau khi sinh viên đã có hội đồng, hệ thống sẽ bốc ngẫu nhiên giảng viên ở trong hội đồng để làm giảng viên phản biện cho sinh viên đó.

3.2.4. Yêu cầu phi chức năng

- Website có trải nghiệm người dùng tốt, tốc độ xử lý nhanh.
- Các chức năng dễ dàng sử dụng, giao diện đẹp mắt.
- Website phù hợp trên máy tính có thể truy cập bất cứ đâu.
- Dễ dàng tìm kiếm các thông tin.
- Đảm bảo tính bảo mật: phân quyền hệ thống cho từng đối tượng sử dụng tránh tính trạng thông tin, dữ liệu bị mất hoặc bị sửa đổi.
- Đảm bảo an toàn thông tin: phải có giải pháp bảo đảm an toàn dữ liệu: có khả năng backup dữ liệu và phục hồi dữ liệu khi có sự cố.

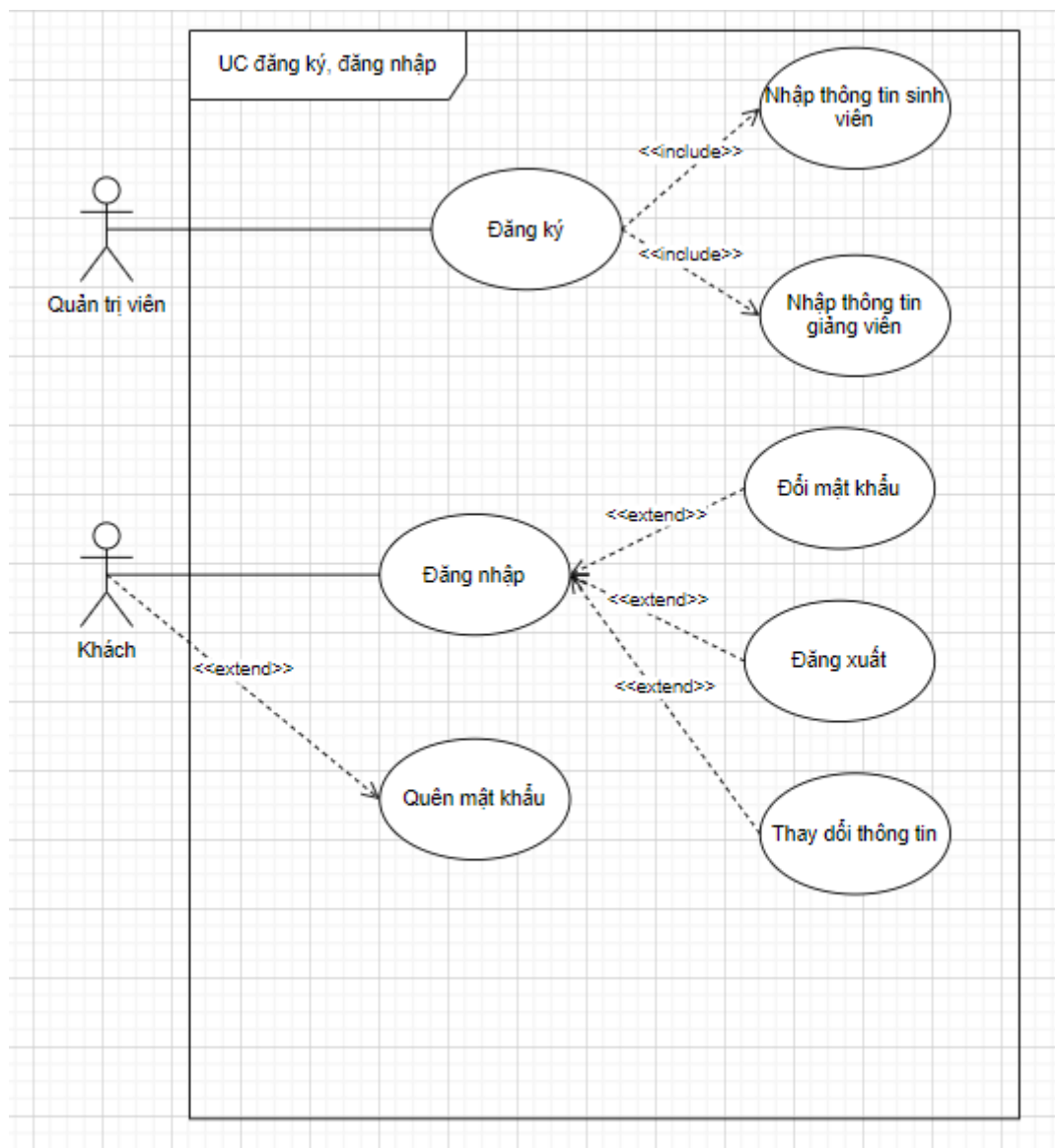
3.3. Sơ đồ UseCase

3.3.1. Sơ đồ use case Tổng quát



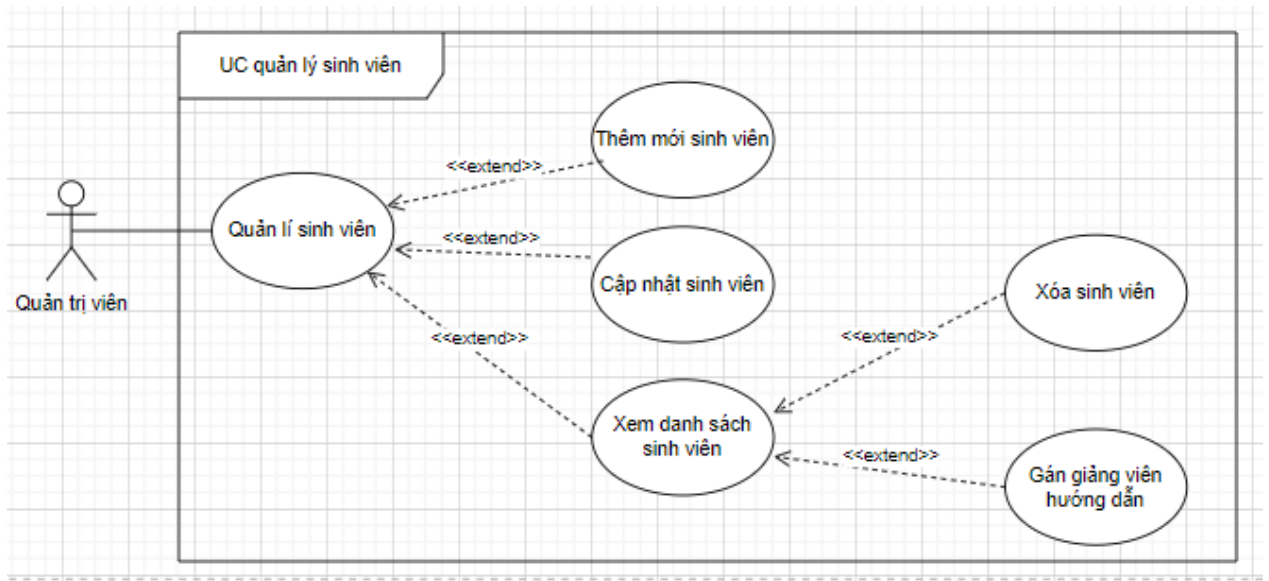
Hình 3.1. Sơ đồ usecase tổng quát

3.3.2. Sơ đồ usecase Đăng nhập, đăng ký



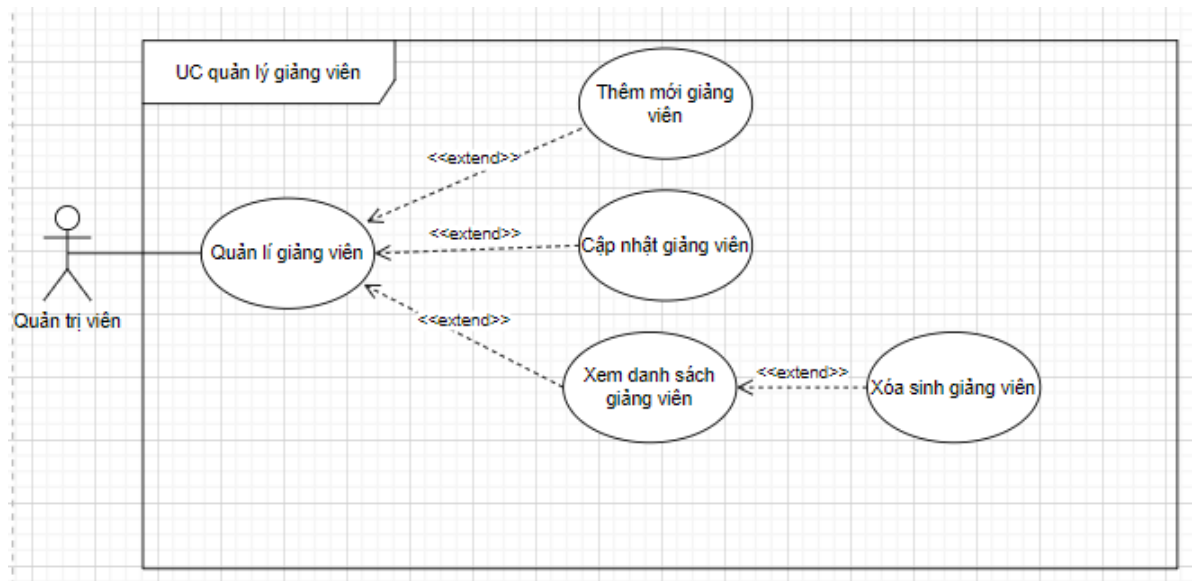
Hình 3.2. Sơ đồ usecase đăng nhập, đăng ký

3.3.3. Sơ đồ usecase quản lý sinh viên



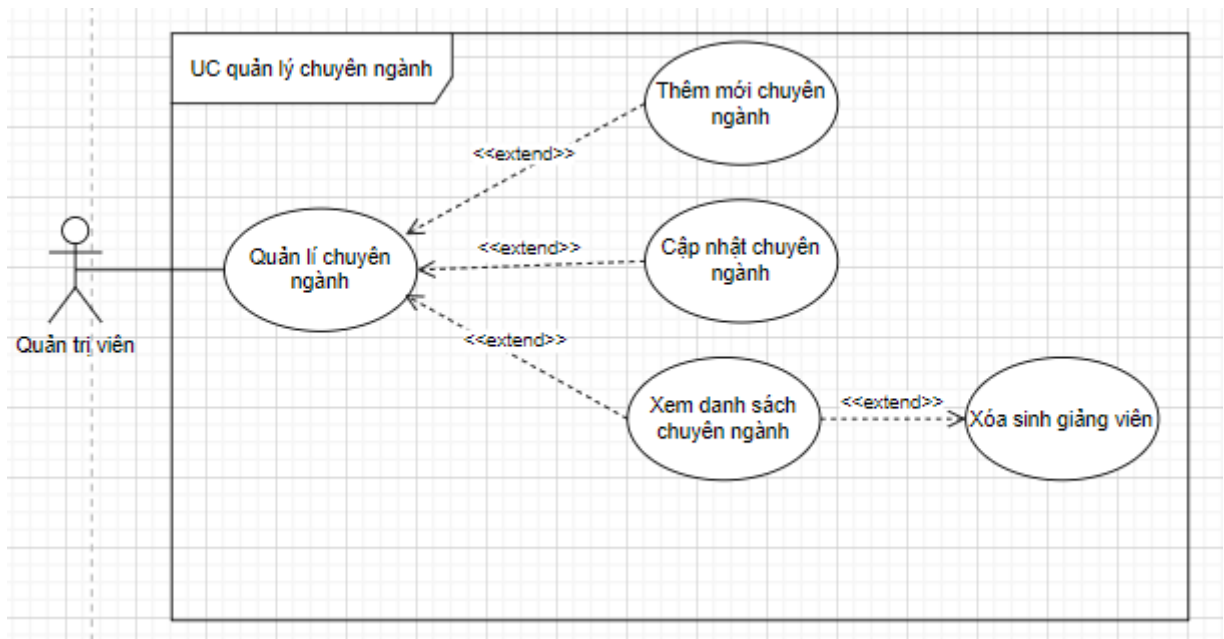
Hình 3.3. Sơ đồ usecase quản lý sinh viên

3.3.4. Sơ đồ usecase quản lý giảng viên



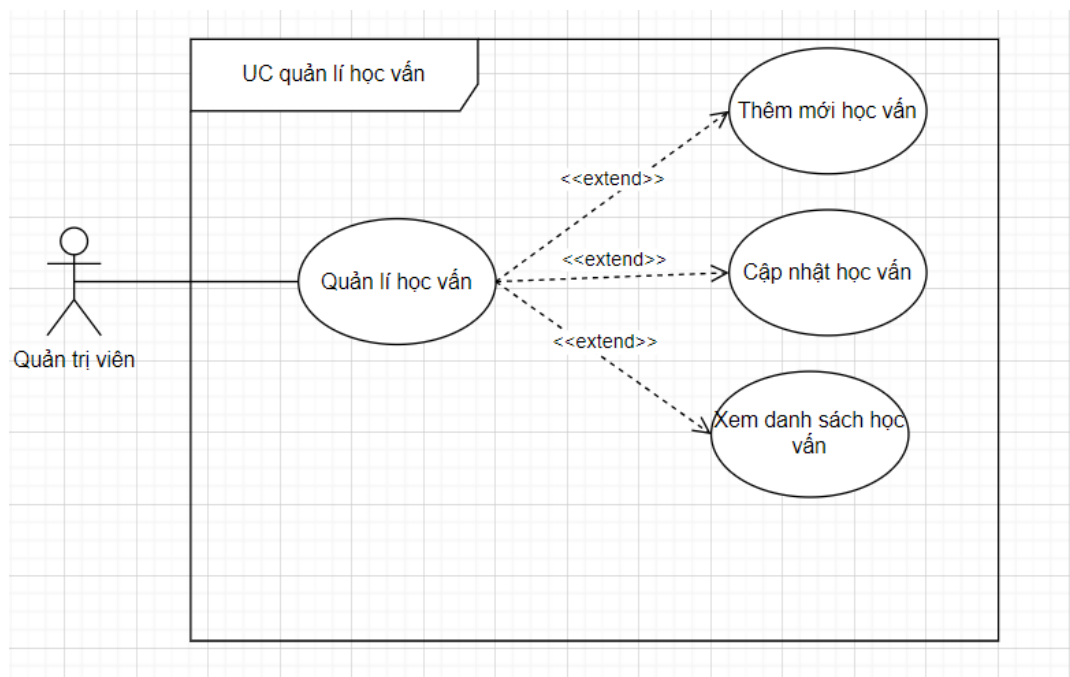
Hình 3.4. Sơ đồ usecase quản lý giảng viên

3.3.5. Sơ đồ usecase quản lý chuyên ngành



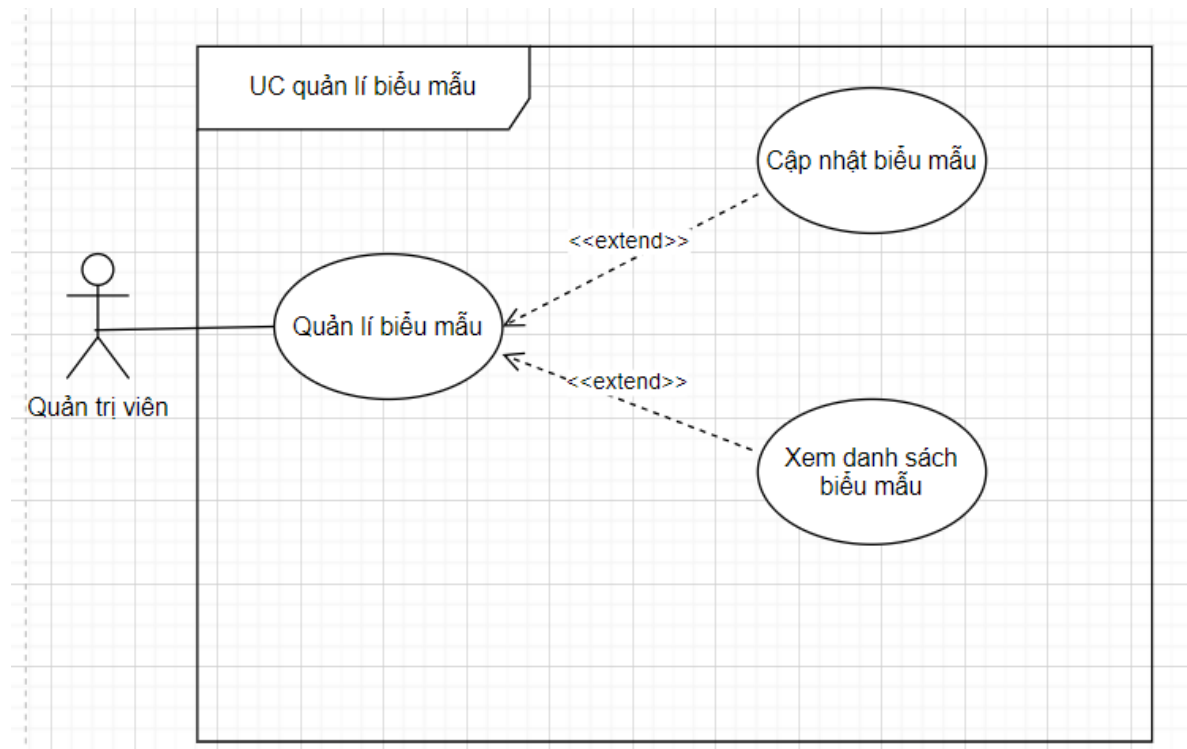
Hình 3.5. Sơ đồ usecase quản lý chuyên ngành

3.3.6. Sơ đồ usecase quản lý học vấn



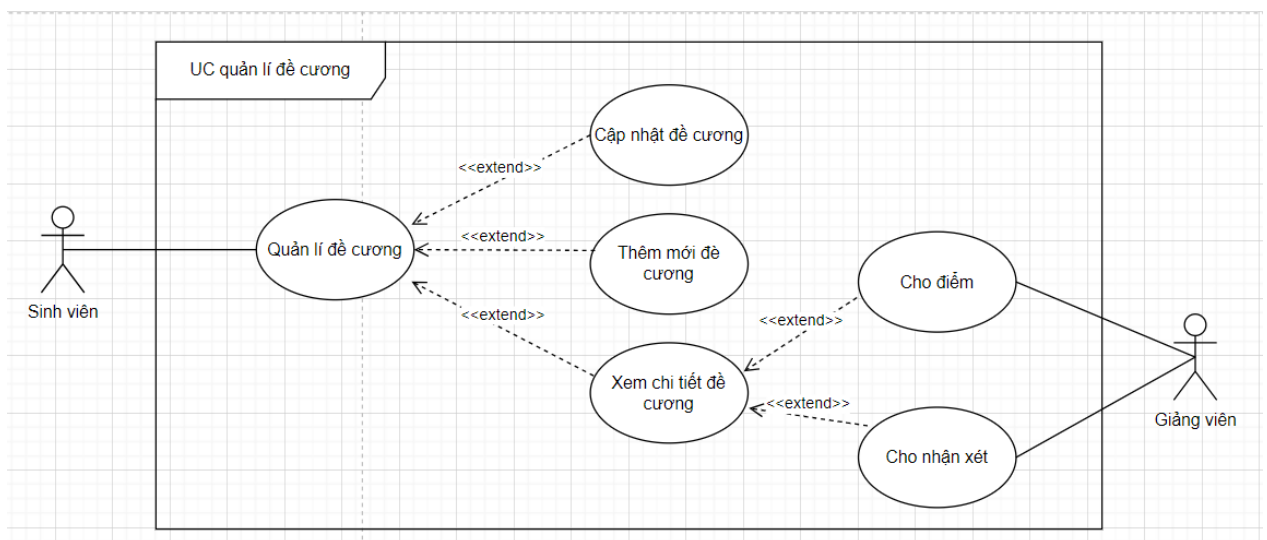
Hình 3.6. Sơ đồ usecase quản lý học vấn

3.3.7. Sơ đồ usecase quản lý biểu mẫu



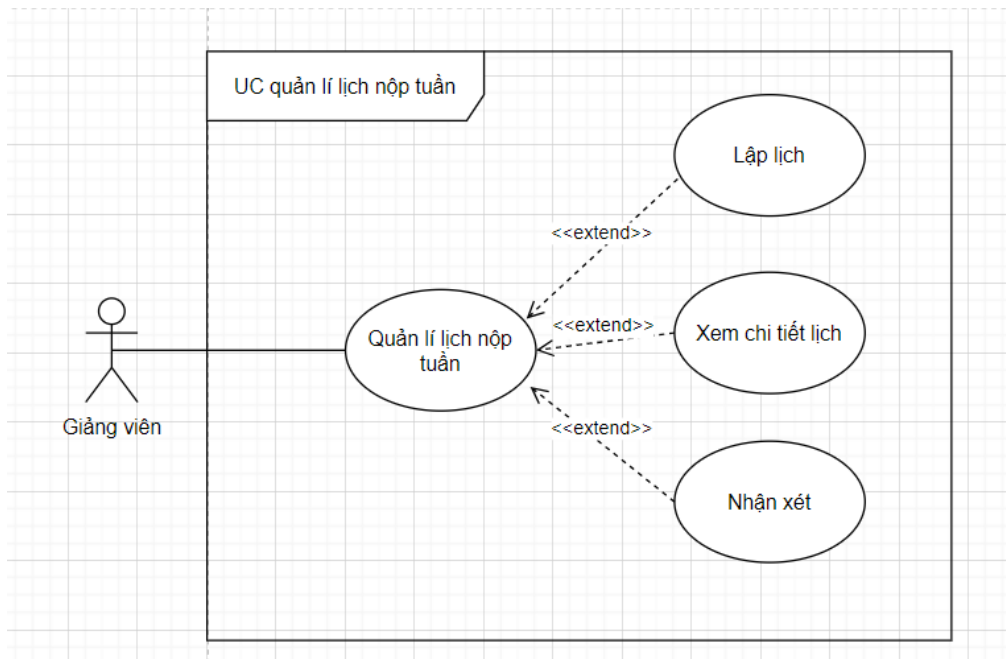
Hình 3.7. Sơ đồ usecase quản lý biểu mẫu

3.3.8. Sơ đồ usecase quản lý đề cương



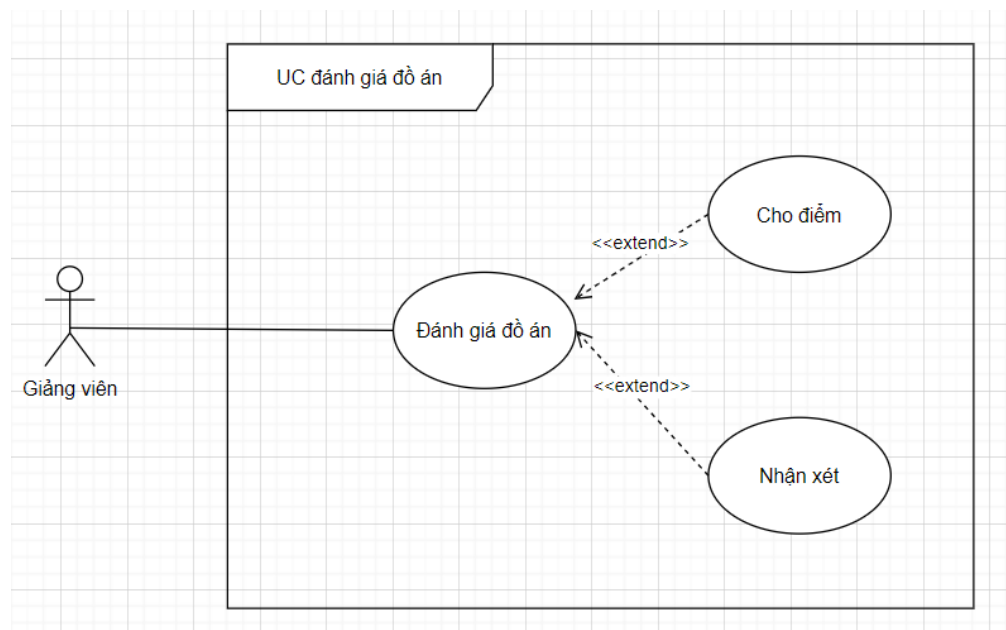
Hình 3.8. Sơ đồ usecase quản lý đề cương

3.3.9. Sơ đồ usecase quản lý lịch nộp tuần



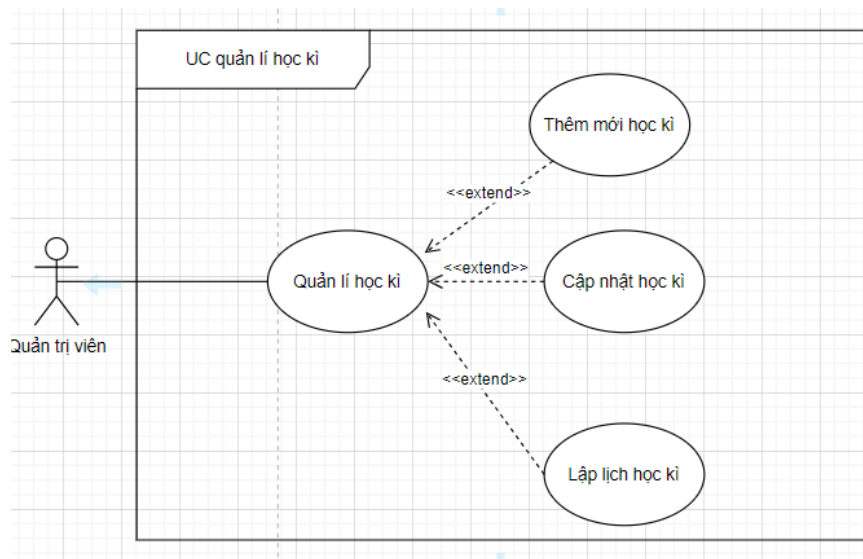
Hình 3.9. Sơ đồ usecase quản lý lịch nộp tuần

3.3.10. Sơ đồ usecase đánh giá đồ án



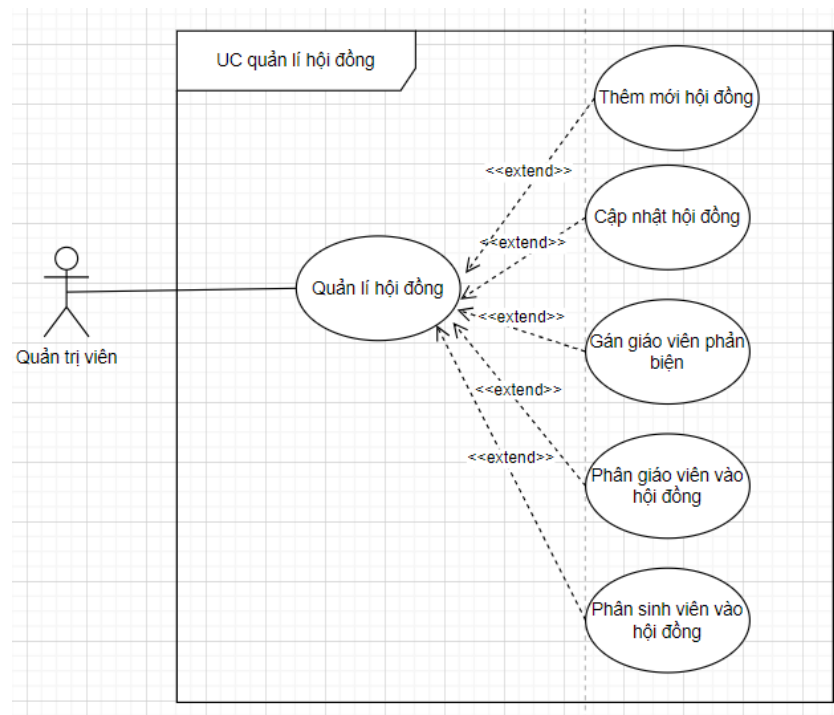
Hình 3.10. Sơ đồ usecase đánh giá đồ án

3.3.11. Sơ đồ usecase quản lý học kì



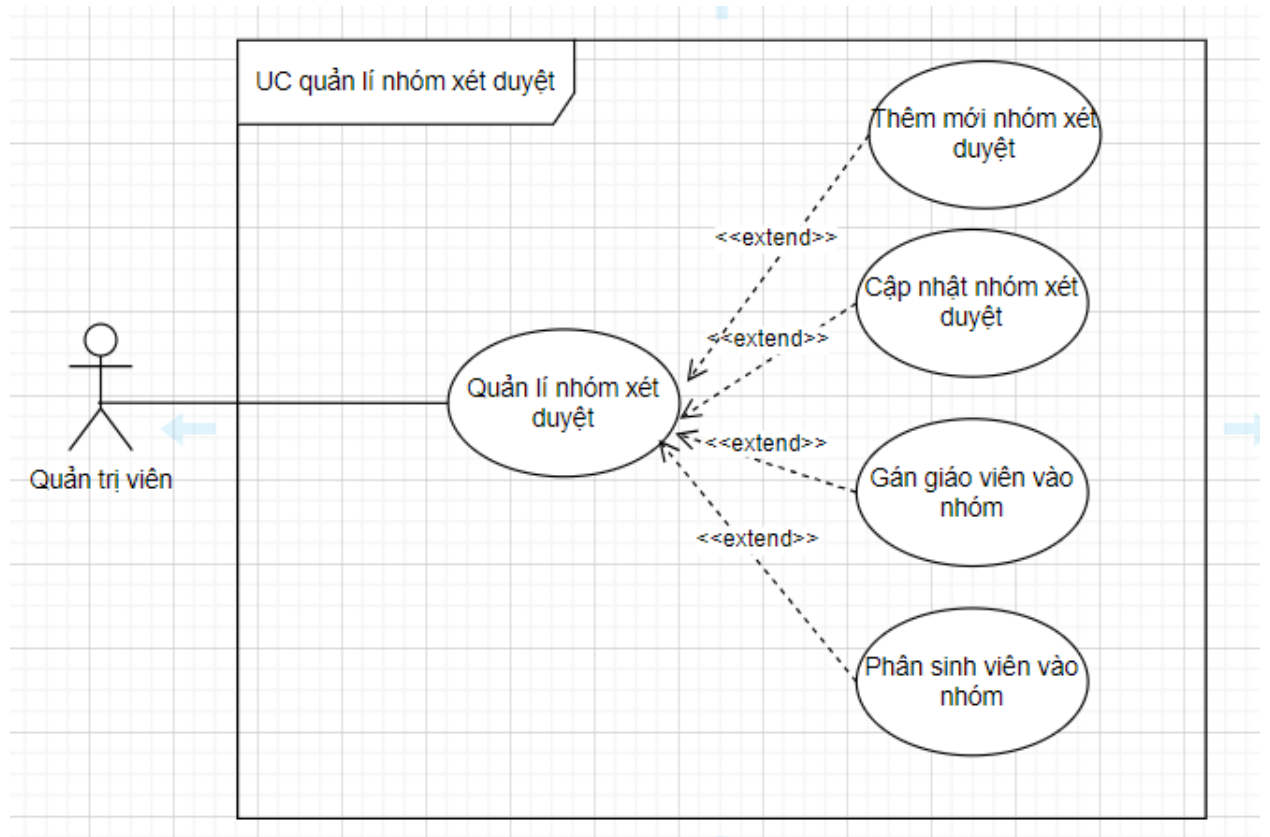
Hình 3.11. Sơ đồ usecase quản lý học kì

3.3.12. Sơ đồ usecase quản lý hội đồng



Hình 3.12. Sơ đồ usecase quản lý hội đồng

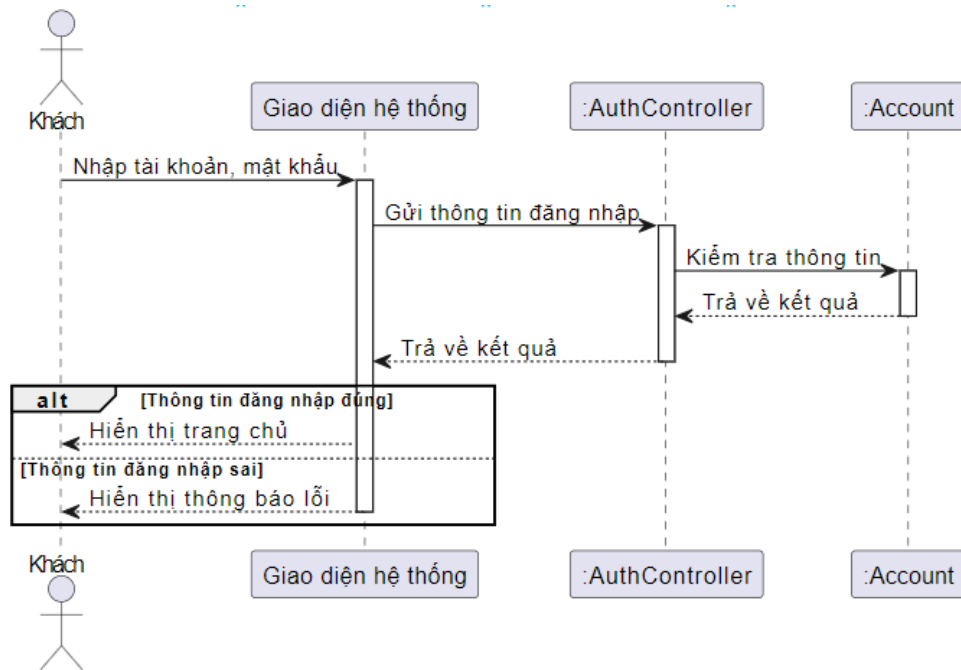
3.3.13. Sơ đồ usecase quản lý nhóm xét duyệt



Hình 3.13. Sơ đồ usecase quản lý nhóm xét duyệt

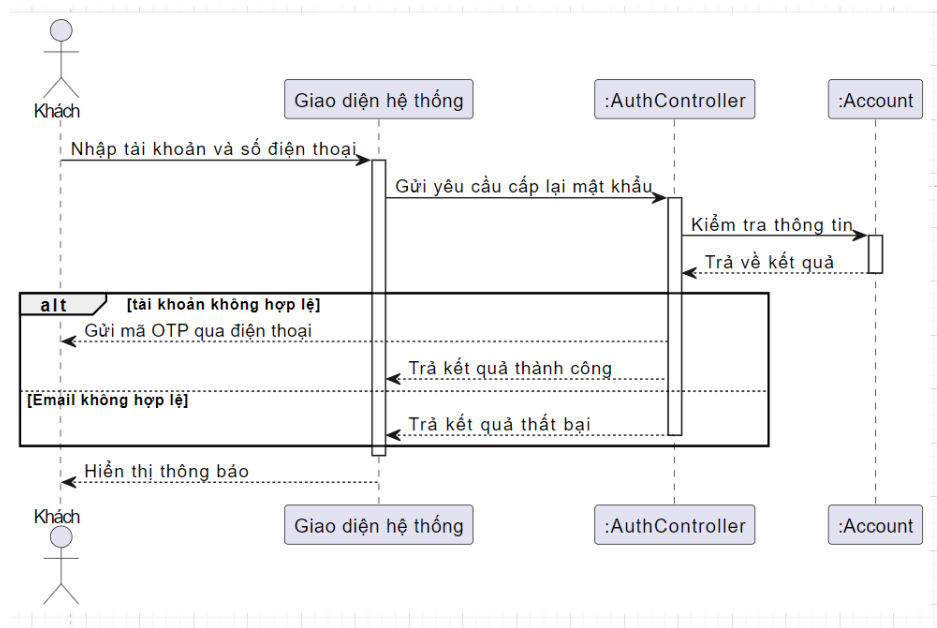
3.4. Sơ đồ tuần tự

3.4.1. Sơ đồ tuần tự đăng nhập



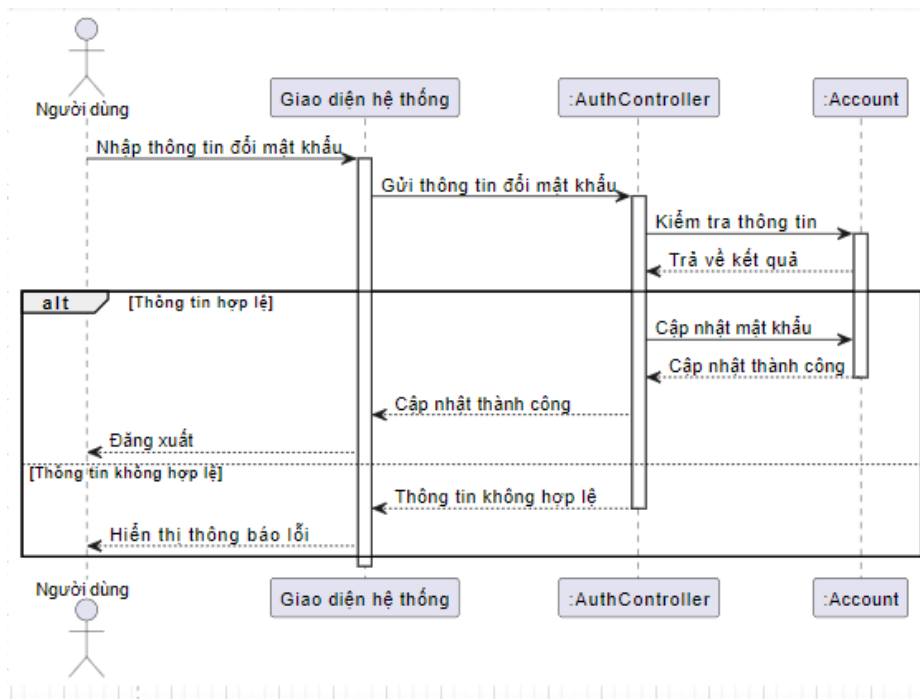
Hình 3.14. Sơ đồ tuần tự đăng nhập

3.4.2. Sơ đồ tuần tự cấp lại mật khẩu



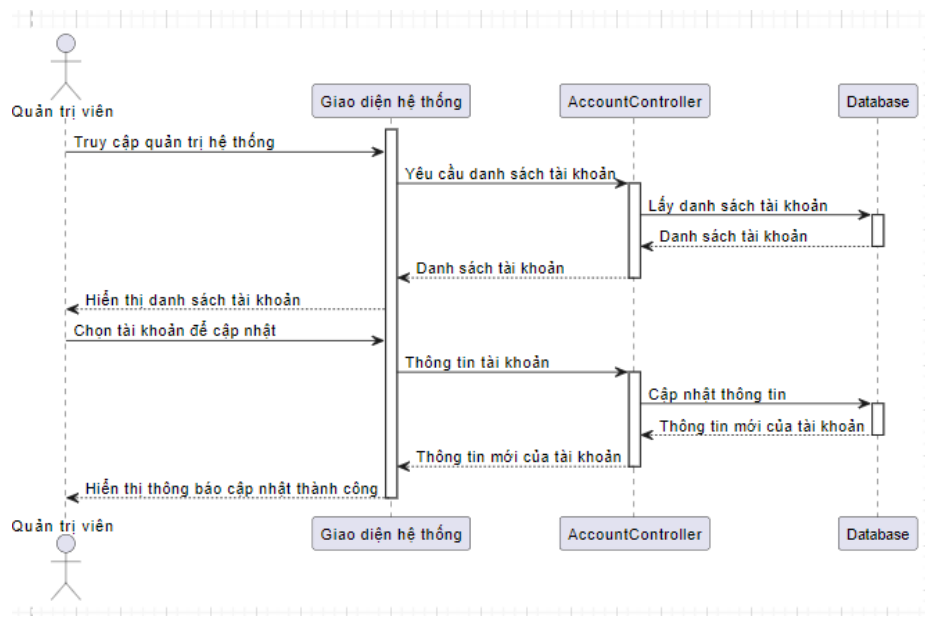
Hình 3.15. Sơ đồ tuần tự cấp lại mật khẩu

3.4.3. Sơ đồ tuần tự đổi mật khẩu



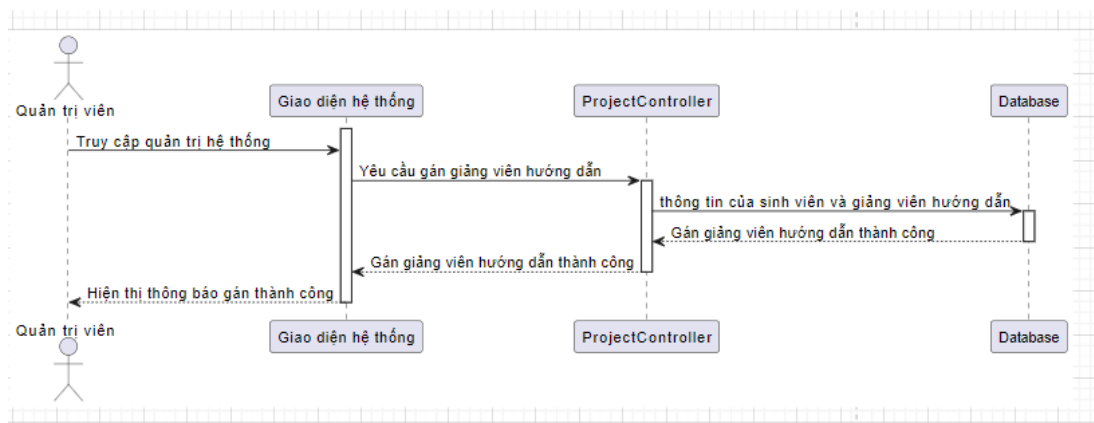
Hình 3.16. Sơ đồ tuần tự đổi mật khẩu

3.4.4. Sơ đồ tuần tự quản lý sinh viên, giảng viên



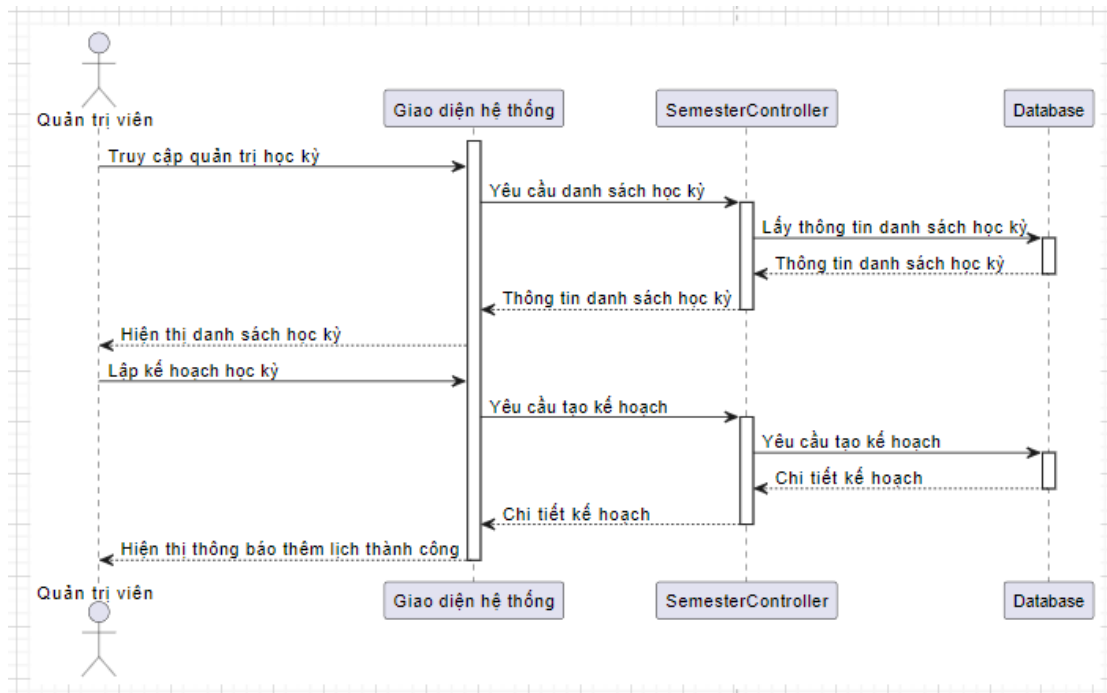
Hình 3.17. Sơ đồ tuần tự quản lý sinh viên, giảng viên

3.4.5. Sơ đồ tuần tự gán giảng viên hướng dẫn



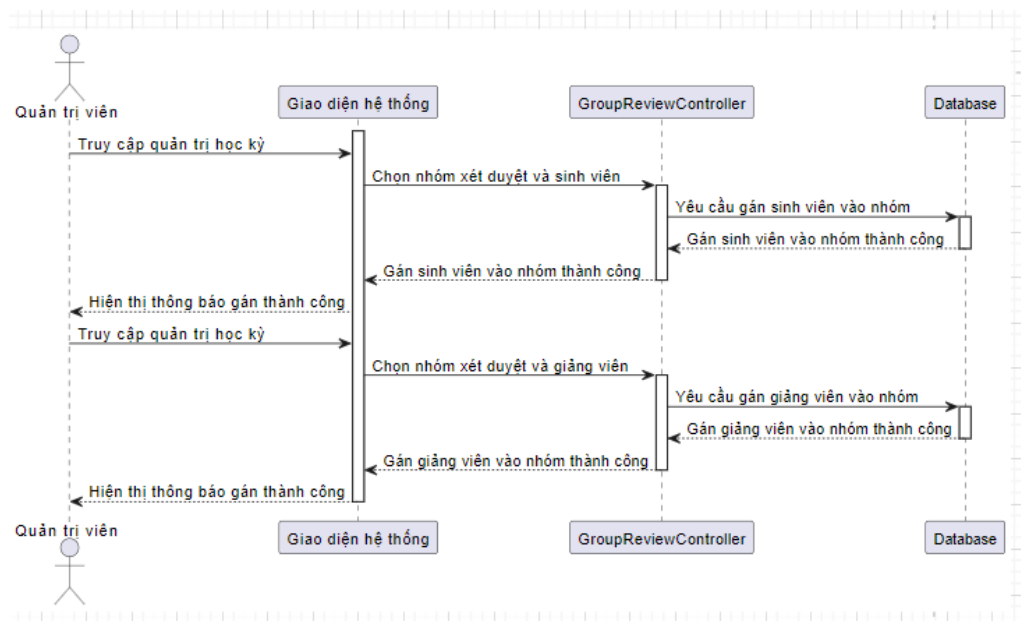
Hình 3.18. Sơ đồ tuần tự gán giảng viên hướng dẫn

3.4.6. Sơ đồ tuần tự quản lý học kỳ



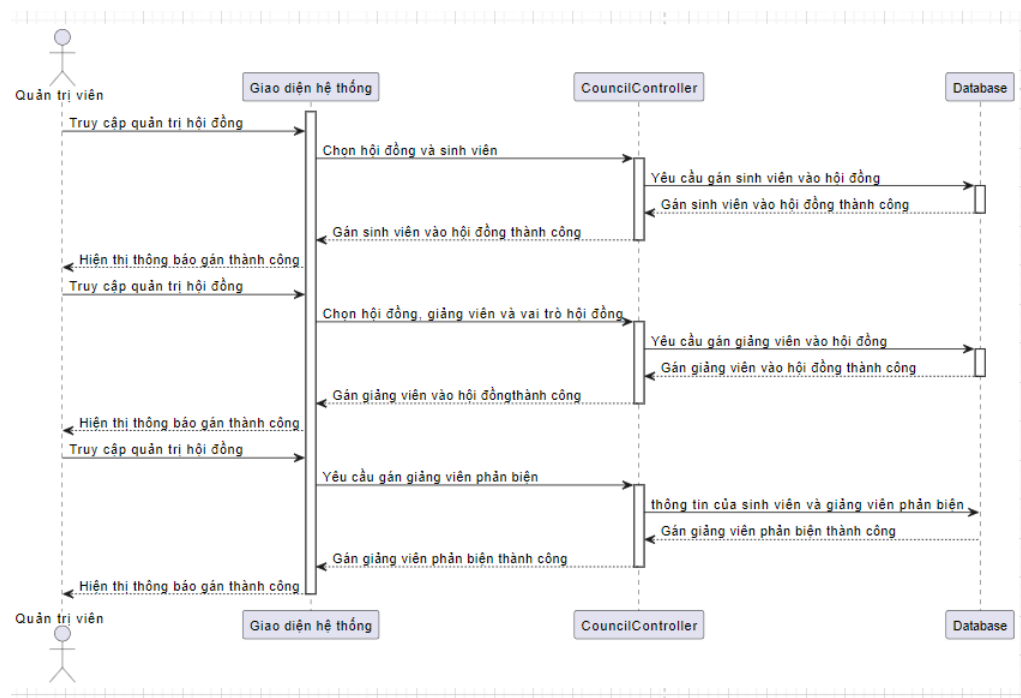
Hình 3.19. Sơ đồ tuần tự quản lý học kỳ

3.4.7. Sơ đồ tuần tự quản lý nhóm xét duyệt



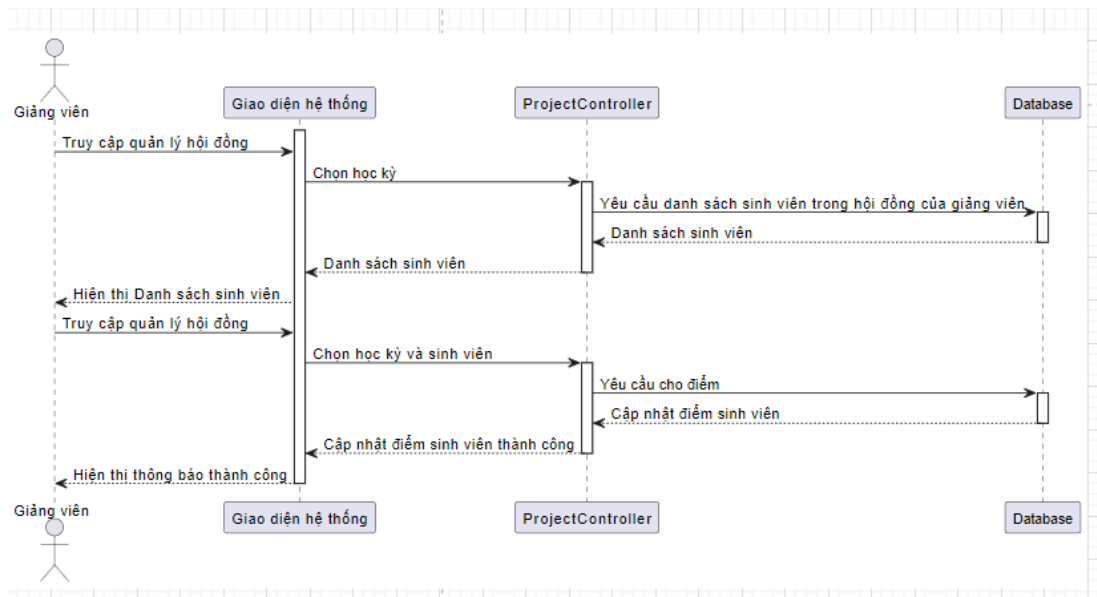
Hình 3.20. Sơ đồ tuần tự quản lý nhóm xét duyệt

3.4.8. Sơ đồ tuần tự quản lý hội đồng



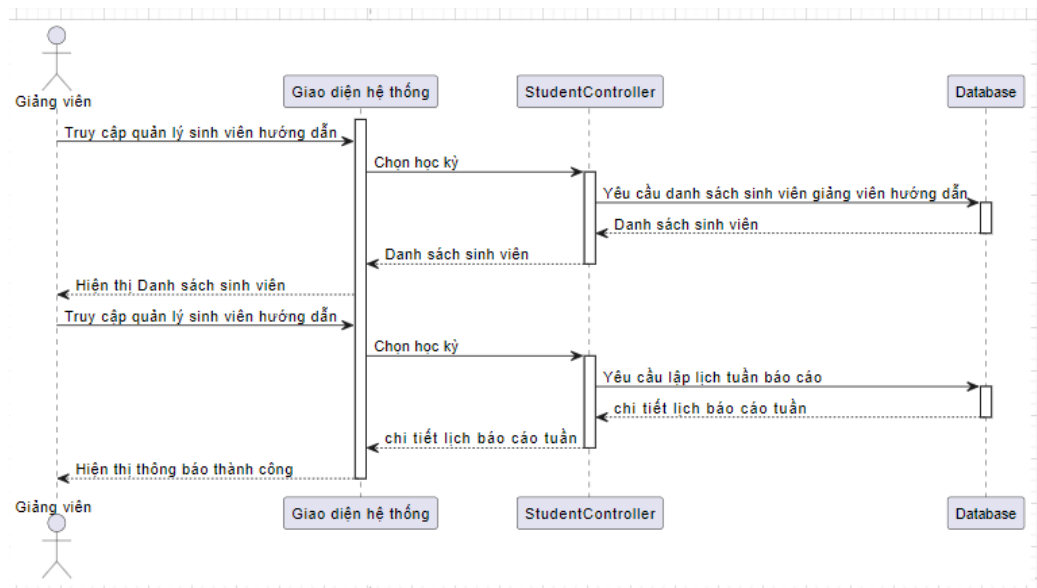
Hình 3.21. Sơ đồ tuần tự quản lý hội đồng

3.4.9. Sơ đồ tuần tự giảng viên cho điểm sinh viên



Hình 3.22. Sơ đồ tuần tự cấp lại mật khẩu

3.4.10. Sơ đồ tuần tự quản lý sinh viên cho giảng viên hướng dẫn



Hình 3.23. Sơ đồ tuần tự cấp lại mật khẩu

3.5 Thiết kế cơ sở dữ liệu

3.5.1. Tổng quan

Hệ thống sẽ quản lý các đợt bảo vệ theo học kỳ nên ta sẽ có *bảng Semester*. Mỗi học kỳ sẽ có những kế hoạch khác nhau nên ta phải quản lý thông qua *bảng ScheduleSemester*.

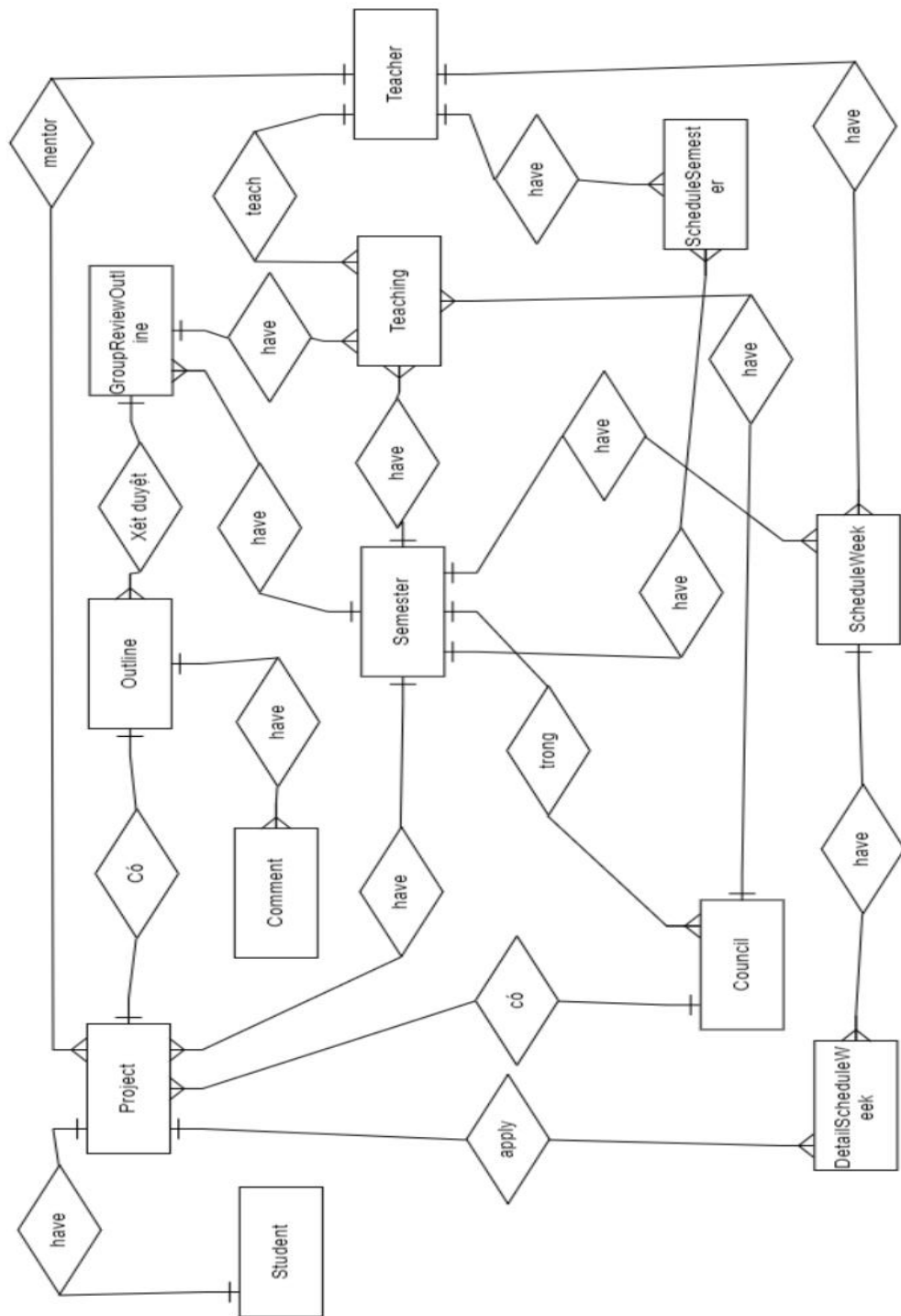
Hệ thống sẽ quản lý sinh viên theo học kì đã tạo của quản trị viên và mỗi học kì sẽ chỉ tồn tại một tài khoản của sinh viên đó và mỗi sinh viên cũng chỉ có thể có một đề án bảo vệ trong một học kỳ nên ta sẽ có ba *bảng Student*, *bảng Project* và *bảng Semester* được liên kết 1-1 với nhau

Mỗi sinh viên có thể có thể viết đề cương đề án cho kì đề án nên ta sẽ có *bảng ProjectOutline* và sẽ được một nhóm xét duyệt cho đánh giá và nhận xét nên ta sẽ có liên kết 1-1 với *bảng GroupReviewOutline*. Trong mỗi nhóm xét duyệt theo từng kỳ sẽ có nhiều giảng viên trong một nhóm nên ta sẽ có liên kết 1-n với *bảng Teaching*. Khi đó giảng viên trong nhóm xét duyệt có thể được nhận xét và đánh giá qua *bảng Comment*

Để có thể dễ dàng quản lý giảng viên theo học kỳ ta sẽ tạo ra *bảng Teaching*. Mỗi một học kỳ giảng viên có thể đóng vai trò khác nhau trong hội đồng (thư ký, ủy viên, chủ tịch) nên ta sẽ lưu vai trò của giảng viên đó qua bảng này

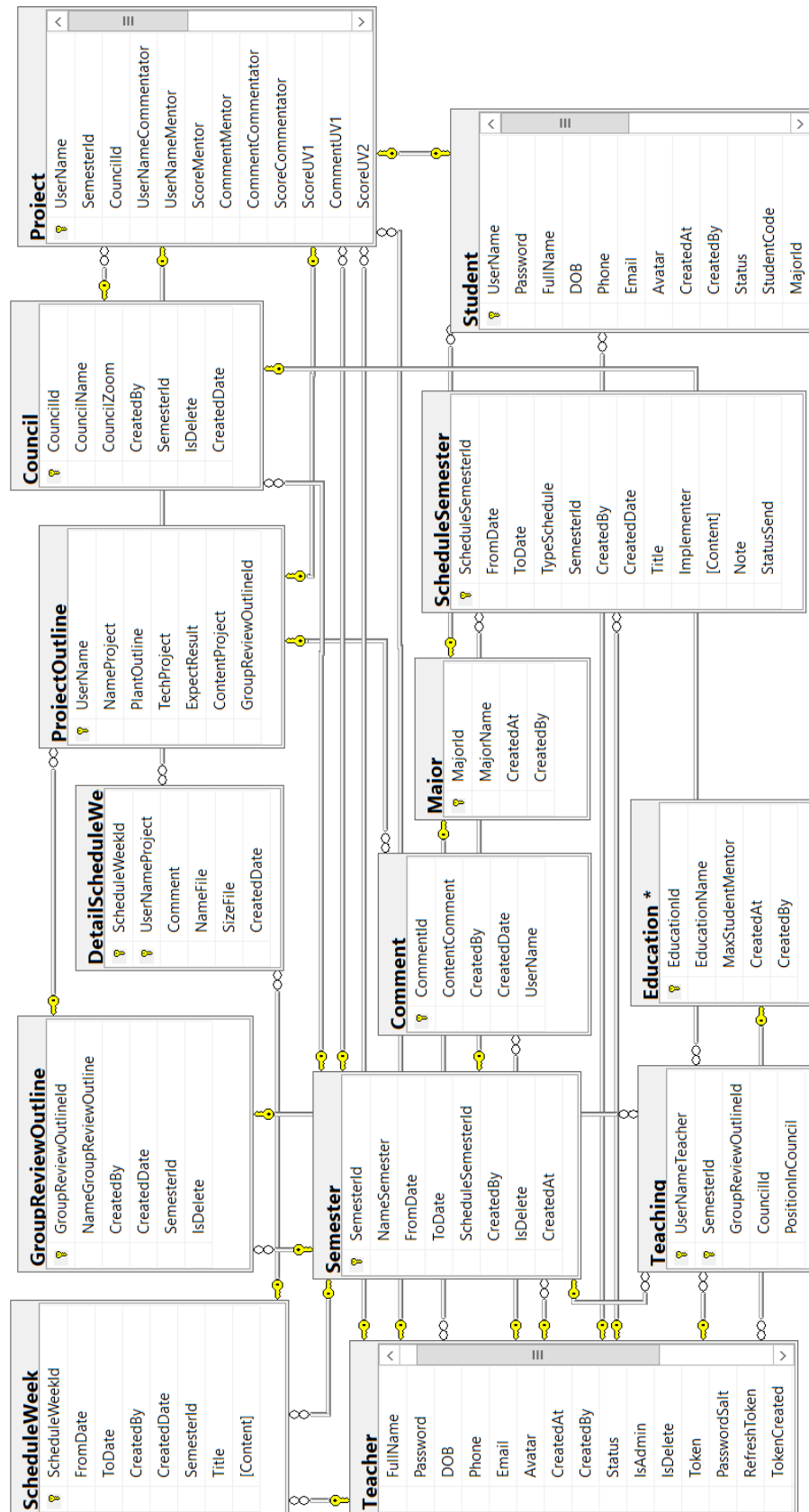
Để có thể quản lý đề cương đề án, giảng viên hướng dẫn có thể lập lịch kế hoạch tuần nên ta sẽ tạo *bảng ScheduleWeek*, mỗi một lịch này sinh viên chỉ có thể gửi một tệp tin báo cáo tuần đó và giảng viên có thể ghi nhận xét trên hệ thống

3.5.2 Mô hình thực thể liên kết



Hình 3. 24. Mô hình thực thể liên kết

3.5.3 Mô hình quan hệ



Hình 3.25. Mô hình quan hệ

3.5.4 Mô tả dữ liệu

Bảng 3.2. Mô tả bảng Student

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>UserName</u>	NVARCHAR(100)	được tạo ra từ mã sinh viên và mã học kỳ	lưu tài khoản đăng nhập của sinh viên
Email	NVARCHAR(100)	đúng định dạng	
Password	NVARCHAR(100)		
Fullname	NVARCHAR(MAX)		
Avatar	NVARCHAR(MAX)		
Phone	NVARCHAR(20)		
DOB	DATE		
StudentCode	NVARCHAR(30)		
MajorId	NVARCHAR(10)		
ClassName	NVARCHAR(50)		
SchoolYearName	NVARCHAR(50)		
IsDelete	INT		
Status	NVARCHAR(50)		lưu trạng thái tài khoản
Gender	INT		
PasswordSalt	NVARCHAR(100)		
RefreshToken	NVARCHAR(100)		
TokenCreated	DATETIME		
TokenExpires	DATETIME		
CreatedAt	DATETIME		
CreatedBy	NVARCHAR(MAX)		

Bảng 3.3. Mô tả bảng Teacher

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>UserName</u>	NVARCHAR(50)	Không chứa kí tự đặc biệt và dấu cách	Lưu
Email	NVARCHAR(100)	đúng định dạng	
Password	NVARCHAR(100)		
Fullname	NVARCHAR(MAX)		
Avatar	NVARCHAR(MAX)		
Phone	NVARCHAR(20)		
DOB	DATE		
MajorId	NVARCHAR(10)		
IsDelete	INT		
Status	NVARCHAR(50)		lưu trạng thái tài khoản
IsAdmin	INT		
Education	NVARCHAR(50)		
Gender	INT		
PasswordSalt	NVARCHAR(100)		
RefreshToken	NVARCHAR(100)		
TokenCreated	DATETIME		
TokenExpires	DATETIME		
CreatedAt	DATETIME		
CreatedBy	NVARCHAR(50)		

Bảng 3.4. Mô tả bảng Council

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>CouncilId</u>	NVARCHAR(50)		
CouncilName	NVARCHAR(100)		
CouncilZoom	NVARCHAR(100)		
CreatedBy	NVARCHAR(50)		
SemesterId	NVARCHAR(50)		
IsDelete	INT		
CreatedDate	DATETIME		

Bảng 3.5. Mô tả bảng Comment

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>CommentId</u>	NVARCHAR(50)		
ContentComment	NVARCHAR(100)		
CreatedBy	NVARCHAR(100)		
CreatedDate	DATETIME		
UserName	NVARCHAR(50)		

Bảng 3.6. Mô tả bảng Teaching

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>UserNameTeacher</u>	NVARCHAR(50)		
SemesterId	NVARCHAR(50)		
GroupReviewOutlineId	NVARCHAR(50)		
CouncilId	NVARCHAR(50)		

Bảng 3.7. Mô tả bảng Project

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>UserName</u>	NVARCHAR(50)		
SemesterId	NVARCHAR(50)		
CouncilId	NVARCHAR(50)		
UserNameCommentator	NVARCHAR(50)		
UserNameMentor	NVARCHAR(50)		
ScoreMentor	FLOAT		
CommentMentor	NVARCHAR(MAX)		
ScoreCommentator	FLOAT		
CommentCommentator	NVARCHAR(MAX)		
ScoreUV1	FLOAT		
Comment UV1	NVARCHAR(MAX)		
ScoreUV2	FLOAT		
CommentUV2	NVARCHAR(MAX)		
ScoreUV3	FLOAT		
CommentUV3	NVARCHAR(MAX)		
ScoreCT	FLOAT		
CommentCT	NVARCHAR(MAX)		
ScoreTK	FLOAT		
CommentTK	NVARCHAR(MAX)		
ScoreFinal	FLOAT		
Hash	NVARCHAR(MAX)		

Bảng 3.8. Mô tả bảng ProjectOutline

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>UserName</u>	NVARCHAR(50)		
NameProject	NVARCHAR(100)		
PlantOutline	NVARCHAR(MAX)		
TechProject	NVARCHAR(200)		
ExpectResult	NVARCHAR(MAX)		
ContentProject	NVARCHAR(MAX)		
GroupReviewOutlineId	NVARCHAR(50)		

Bảng 3.9. Mô tả bảng GroupReviewOutline

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>GroupReviewOutlineId</u>	NVARCHAR(50)		
NameGroupReviewOutline	NVARCHAR(100)		
CreatedBy	NVARCHAR(50)		
CreatedDate	DATETIME		
SemesterId	NVARCHAR(50)		
IsDelete	INT		

Bảng 3.10. Mô tả bảng ScheduleSemester

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>ScheduleSemester</u>	NVARCHAR(50)		
FromDate	DATETIME		
ToDate	DATETIME		
TypeSchedule	NVARCHAR(50)		
Content	NVARCHAR(MAX)		
SemesterId	NVARCHAR(50)		
Implementer	NVARCHAR(MAX)		
NOTE	NVARCHAR(MAX)		
IsDelete	INT		
CreatedBy	NVARCHAR(50)		
CreatedDate	DATETIME		

Bảng 3.11. Mô tả bảng Major

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>MajorId</u>	NVARCHAR(50)		
MajorName	NVARCHAR(100)		

Bảng 3.12. Mô tả bảng ScheduleWeek

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>ScheduleWeekId</u>	NVARCHAR(50)		
FromDate	DATETIME		
ToDate	DATETIME		
Title	NVARCHAR(MAX)		
Content	NVARCHAR(MAX)		
IsDelete	INT		
CreatedBy	NVARCHAR(50)		
CreatedDate	DATETIME		

Bảng 3.13. Mô tả bảng DetailScheduleWeek

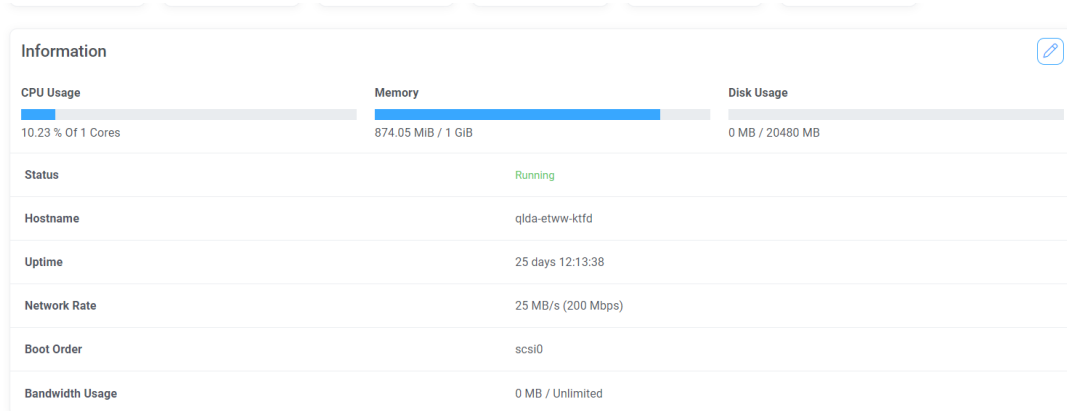
Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>ScheduleWeekId</u>	NVARCHAR(50)		
UserNameProject	NVARCHAR(50)		
Comment	NVARCHAR(MAX)		
NameFile	NVARCHAR(MAX)		
SizeFile	NVARCHAR(MAX)		
CreatedDate	DATE		

Bảng 3.14. Mô tả bảng Semester

Tên thuộc tính	Kiểu dữ liệu	Ràng buộc	Mô tả
<u>SemesterId</u>	NVARCHAR(50)		
NameSemester	NVARCHAR(50)		
FromDate	DATETIME		
ToDate	DATETIME		
CreatedAt	DATE		
CreatedBy	NVARCHAR(50)		
IsDelete	INT		

3.6. Triển khai hệ thống

Để triển khai hệ thống lên môi trường, chúng ta sẽ thuê một máy chủ VPS với hệ điều Ubuntu. để có thể triển khai phần Backend và FrontEnd lên môi trường.



Hình 3.26. Thông tin cấu hình máy chủ VPS

Chúng ta sẽ sử dụng 1 máy chủ với cùng cấu hình gồm 1 vCPU, bộ nhớ RAM 2GB, dung lượng ổ cứng SSD 20GB, sử dụng hệ điều hành Ubuntu 18.04 (LTS). Cấu hình này sẽ đủ khả năng để phục vụ quy mô nhỏ như hệ thống đang xây dựng.

Chúng ta sẽ sử dụng nginx để cấu hình máy chủ


```
server {
    listen 80;
    listen [::]:80;

    root /var/www/qlda/html;
    index index.html index.htm index.nginx-debian.html;

    server_name qlda-utc.site 14.225.211.124;

    location / {
        try_files $uri $uri/ =404;
    }
}
```

Hình 3.27. Cấu hình máy chủ nginx

```
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Drop-In: /etc/systemd/system/nginx.service.d
            └─override.conf
   Active: active (running) since Tue 2024-05-21 17:33:18 +07; 1 weeks 1 days ago
     Docs: man:nginx(8)
   Process: 23127 ExecStop=/sbin/start-stop-daemon --quiet --stop --retry QUIT/5 --pidfile /run/nginx.pid (code=exited, status=0/SUCCESS)
   Process: 23206 ExecReload=/usr/sbin/nginx -g daemon on; master_process on; -s reload (code=exited, status=0/SUCCESS)
   Process: 23136 ExecStartPost=/bin/sleep 0.1 (code=exited, status=0/SUCCESS)
   Process: 23134 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 23133 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Main PID: 23135 (nginx)
    Tasks: 2 (limit: 1107)
   CGroup: /system.slice/nginx.service
           └─23135 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             └─23207 nginx: worker process

May 21 17:33:18 qlda-etcw-ktfd systemd[1]: Starting A high performance web server and a reverse proxy server...
May 21 17:33:18 qlda-etcw-ktfd systemd[1]: Started A high performance web server and a reverse proxy server.
May 21 17:33:22 qlda-etcw-ktfd systemd[1]: Reloading A high performance web server and a reverse proxy server.
May 21 17:33:22 qlda-etcw-ktfd systemd[1]: Reloaded A high performance web server and a reverse proxy server.
May 21 17:33:37 qlda-etcw-ktfd systemd[1]: Reloading A high performance web server and a reverse proxy server.
May 21 17:33:37 qlda-etcw-ktfd systemd[1]: Reloaded A high performance web server and a reverse proxy server.
May 21 17:38:08 qlda-etcw-ktfd systemd[1]: Reloading A high performance web server and a reverse proxy server.
May 21 17:38:08 qlda-etcw-ktfd systemd[1]: Reloaded A high performance web server and a reverse proxy server.
```

Hình 3.28. Trạng thái của máy chủ nginx

Tiếp để ta sẽ triển khai ASP.net Core API lên VPS, trước tiên ta sẽ phải cấu hình nginx

```
server {
    listen 80;
    listen [::]:80;
    server_name utc.qlda-utc.site;

    location / {
        proxy_pass http://localhost:5000;
    }
}
```

Hình 3.29. Cấu hình Nginx

Sau khi cấu hình thành công. Tiếp đến ta sẽ cấu hình service để có thể chạy runtime trên môi trường để luôn giữ server hoạt động.

```
[Unit]
Description=API APP

[Service]
WorkingDirectory=/var/www/app
ExecStart=/usr/bin/dotnet /var/www/app/GraduateProject.dll
Restart=always
RestartSec=10
SyslogIdentifier=api
User=root
Environment=ASPNETCORE_ENVIRONMENT=Development

[Install]
WantedBy=multi-user.target
```

Hình 3.30. Cấu hình service

```

Last login: Thu May 30 00:55:57 2024 from 58.186.91.73
root@qlda-etww-ktfd:~# sudo systemctl status DemoReporter.service
● DemoReporter.service - API APP
   Loaded: loaded (/etc/systemd/system/DemoReporter.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2024-05-10 00:10:06 +07; 2 weeks 6 days ago
     Main PID: 22838 (dotnet)
        Tasks: 30 (limit: 1107)
    CGroup: /system.slice/DemoReporter.service
            └─22838 /usr/bin/dotnet /var/www/app/GraduateProject.dll

May 30 01:28:09 qlda-etww-ktfd api[22838]: SELECT `s`.`ScheduleSemesterId`, `s`.`Content`, `s`.`CreatedBy`, `s`.`CreatedDate`,
May 30 01:28:09 qlda-etww-ktfd api[22838]: FROM `ScheduleSemester` AS `s`
May 30 01:29:39 qlda-etww-ktfd api[22838]: info: Microsoft.EntityFrameworkCore.Database.Command[20101]
May 30 01:29:39 qlda-etww-ktfd api[22838]: Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
May 30 01:29:39 qlda-etww-ktfd api[22838]: SELECT `s`.`ScheduleSemesterId`, `s`.`Content`, `s`.`CreatedBy`, `s`.`CreatedDate`,
May 30 01:29:39 qlda-etww-ktfd api[22838]: FROM `ScheduleSemester` AS `s`
May 30 01:30:09 qlda-etww-ktfd api[22838]: info: Microsoft.EntityFrameworkCore.Database.Command[20101]
May 30 01:30:09 qlda-etww-ktfd api[22838]: Executed DbCommand (0ms) [Parameters=[], CommandType='Text', CommandTimeout='30']
May 30 01:30:09 qlda-etww-ktfd api[22838]: SELECT `s`.`ScheduleSemesterId`, `s`.`Content`, `s`.`CreatedBy`, `s`.`CreatedDate`,
May 30 01:30:09 qlda-etww-ktfd api[22838]: FROM `ScheduleSemester` AS `s`
lines 1-18/18 (END)

```

Hình 3.31. Kiểm tra trạng thái của service

Để triển khai cơ sở dữ liệu MySQL, chúng ta lựa chọn hosting để có thể làm nơi lưu trữ dữ liệu. Với CPU 1 nhân, 1.5 GB ram và 4 GB ổ cứng thì có thể thoải mái để triển khai một ứng dụng nhỏ đáp ứng được nhu cầu sử dụng

HOSTING CHEAP 3

CPU: 1 Core

RAM: 1.5 GB

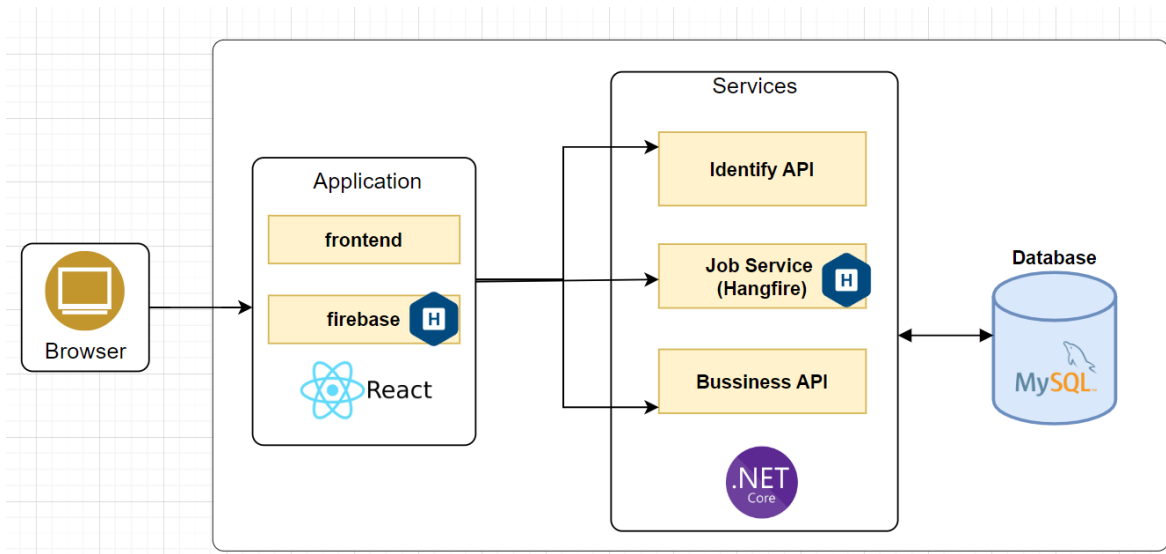
Dung lượng SSD: 4 GB

Domain: 1

Hình 3.32. Thông tin cấu hình Hosting

CHƯƠNG 4: KẾT QUẢ SẢN PHẨM

4.1. Kiến trúc hệ thống



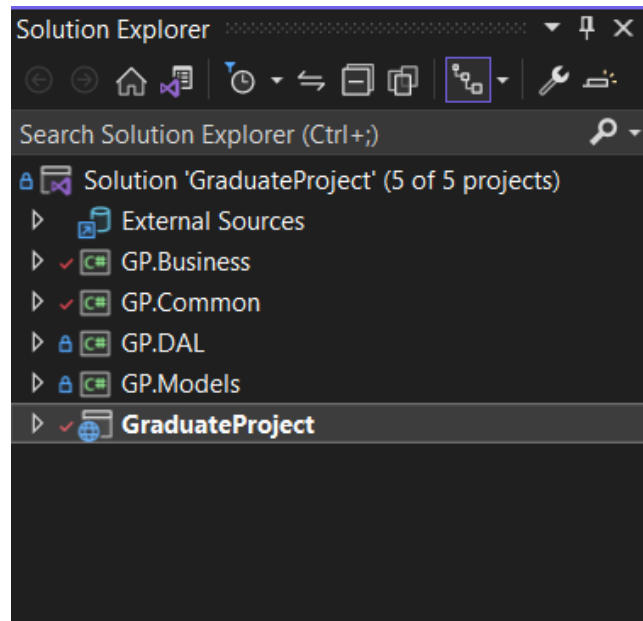
Hình 4.1. Kiến trúc hệ thống

Hệ thống quản lý đồ án bao gồm 3 thành phần chính:

- Application: là phần front-end sử dụng framework ReactJS, gồm có dịch vụ firebase.
- Service: là phần back-end cung cấp các API để đáp ứng yêu cầu từ front-end và kết nối với cơ sở dữ liệu, xử lý các logic nghiệp vụ của hệ thống.
- Database: cơ sở dữ liệu My SQL lưu trữ toàn bộ dữ liệu của hệ thống.

4.2. Cấu trúc dự án

Dự án được xây dựng theo kiến trúc Clean Architecture, là một thiết kế phần mềm chia cấu trúc dự án thành các lớp độc lập, giúp tách biệt logic ứng dụng và các thành phần phụ thuộc. Điều này giúp giảm sự phức tạp của ứng dụng và dễ dàng bảo trì, mở rộng và thay đổi.



Hình 4.2. Cấu trúc dự án

Trong đó:

GraduateProject: Đại diện cho tầng Presentation. Nó chịu trách nhiệm nhận các yêu cầu HTTP từ phía client dưới dạng đối tượng JSON và điều hướng yêu cầu đó đến các tầng xử lý logic.

GP.Business: Đại diện cho tầng Business Layer. Nó chứa các logic nghiệp vụ và luồng xử lý của ứng dụng. Tầng này được triển khai độc lập, không phụ thuộc vào các thành phần cụ thể của cơ sở dữ liệu hay giao diện người dùng.

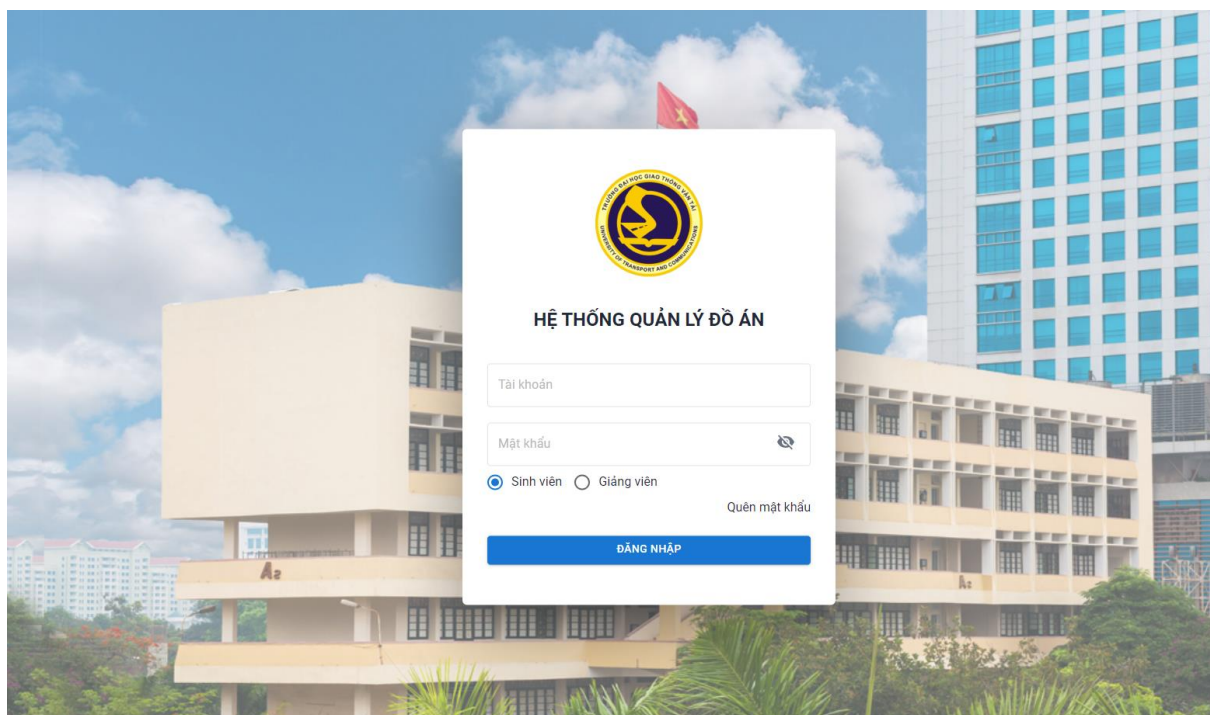
GP.Common: Đại diện cho các thành phần dùng chung và các lớp làm nhiệm vụ hỗ trợ xử lý logic trong dự án. Tầng này cung cấp các hàm tiện ích, đối tượng và các logic xử lý chung được sử dụng bởi các thành phần khác trong dự án.

GP.DAL: Đại diện cho tầng Data Access Layer. Nó chịu trách nhiệm truy xuất và tương tác với cơ sở dữ liệu. Tầng này cung cấp các phương thức và lớp để thực hiện việc truy vấn và xử lý dữ liệu.

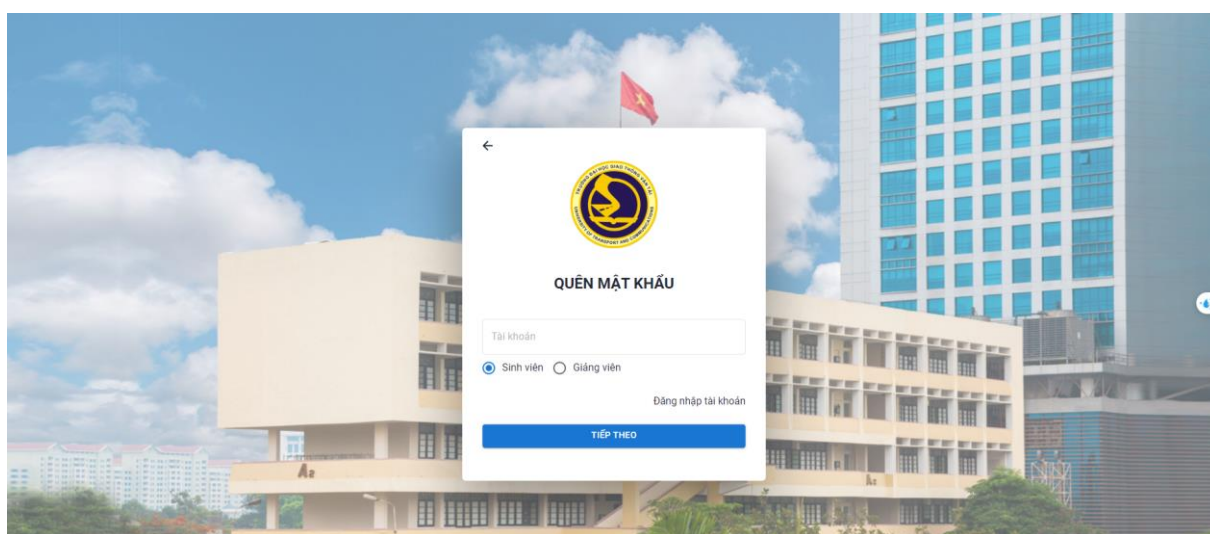
4.3. Giao diện người dùng

Sau khi hoàn thiện, hệ thống sẽ được triển khai trên địa chỉ: <http://qlda-utc.site/>

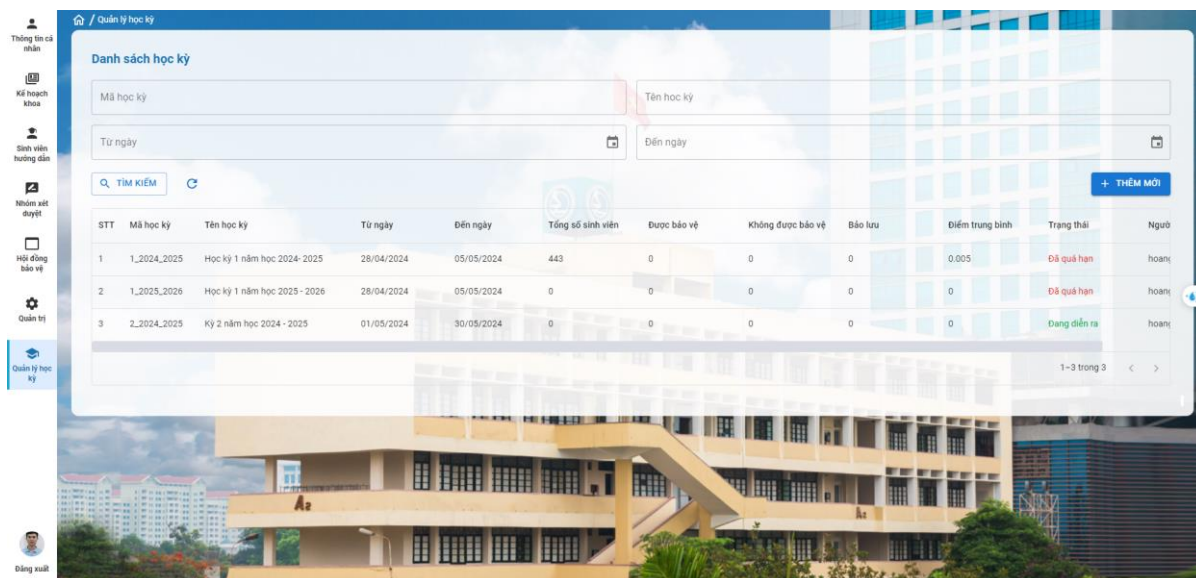
Trong chương này, em sẽ liệt kê kết quả của sản phẩm, để dễ dàng xác định được liệu hệ thống có đáp ứng đủ các chức năng hay không. Em sẽ mô tả các chức năng theo từng vai trò người dùng.



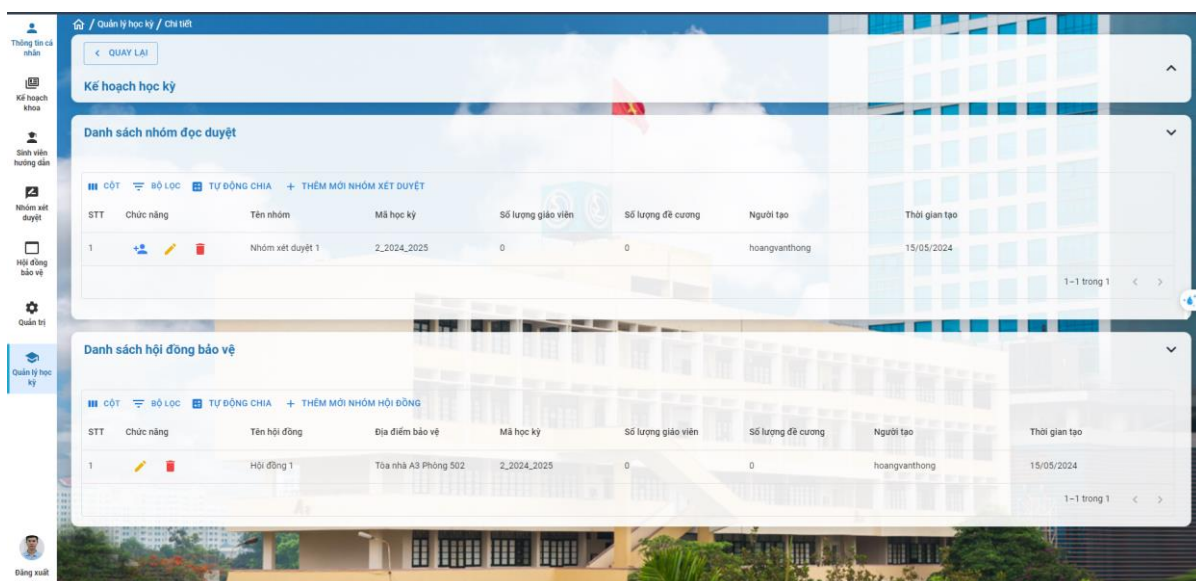
Hình 4.3. Chức năng đăng nhập



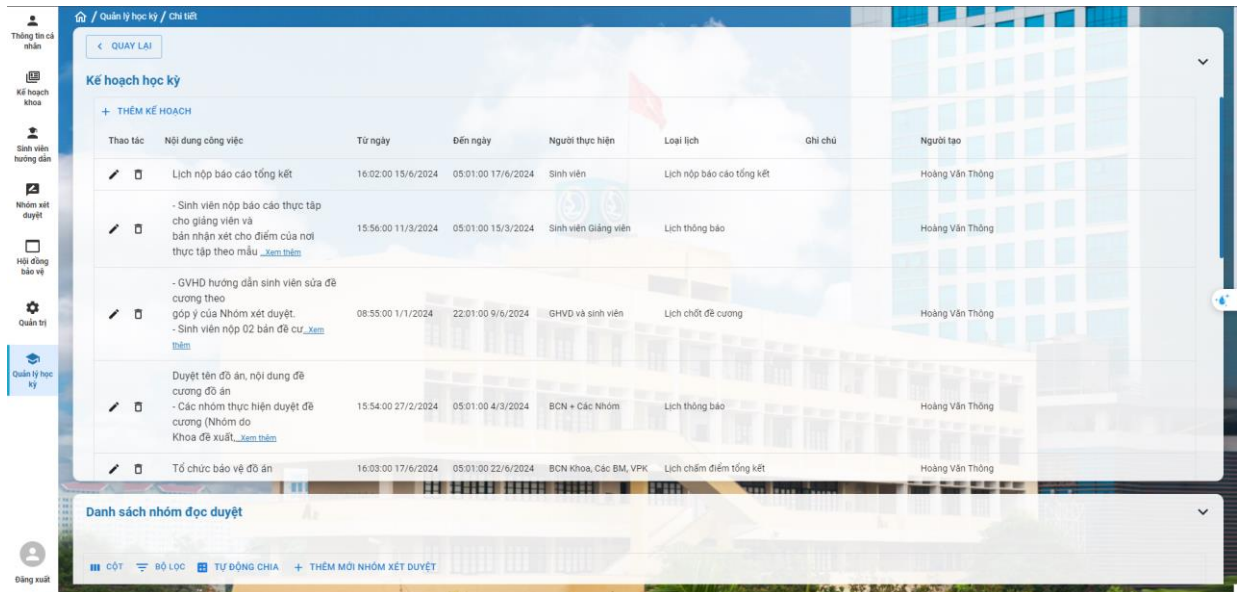
Hình 4.4. Chức năng quên mật khẩu



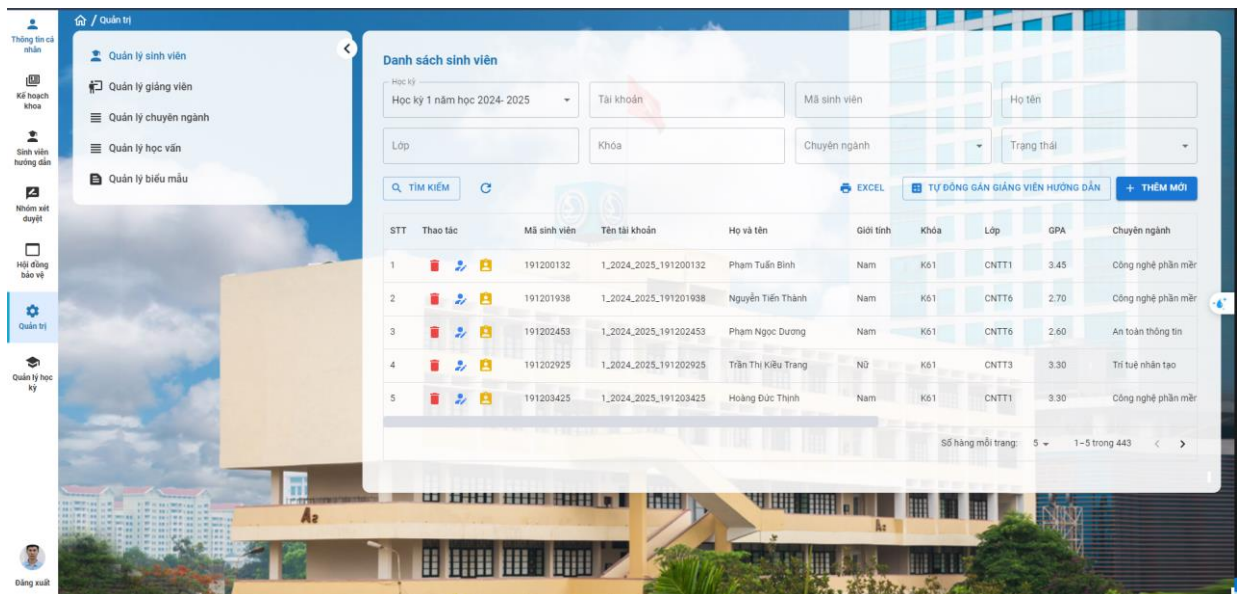
Hình 4.5. Chức năng quản lý học kỳ



Hình 4.6. Chức năng quản lý chi tiết học kỳ



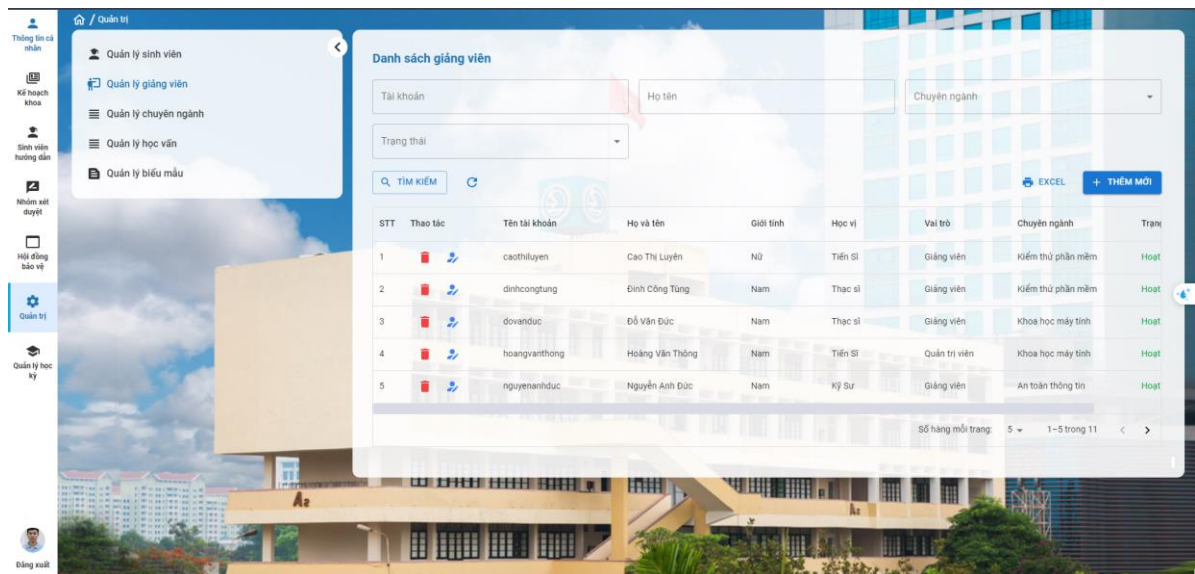
Hình 4.7. Chức năng lập kế hoạch học kỳ



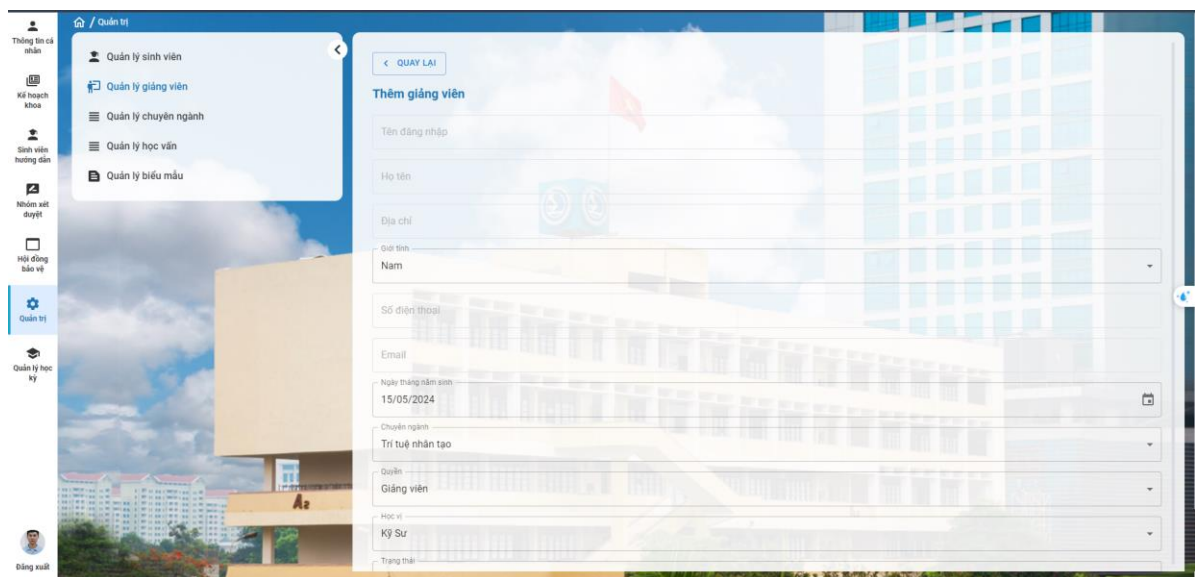
Hình 4.8. Chức năng quản lý sinh viên

Hình 4.9. Chức năng thêm sinh viên

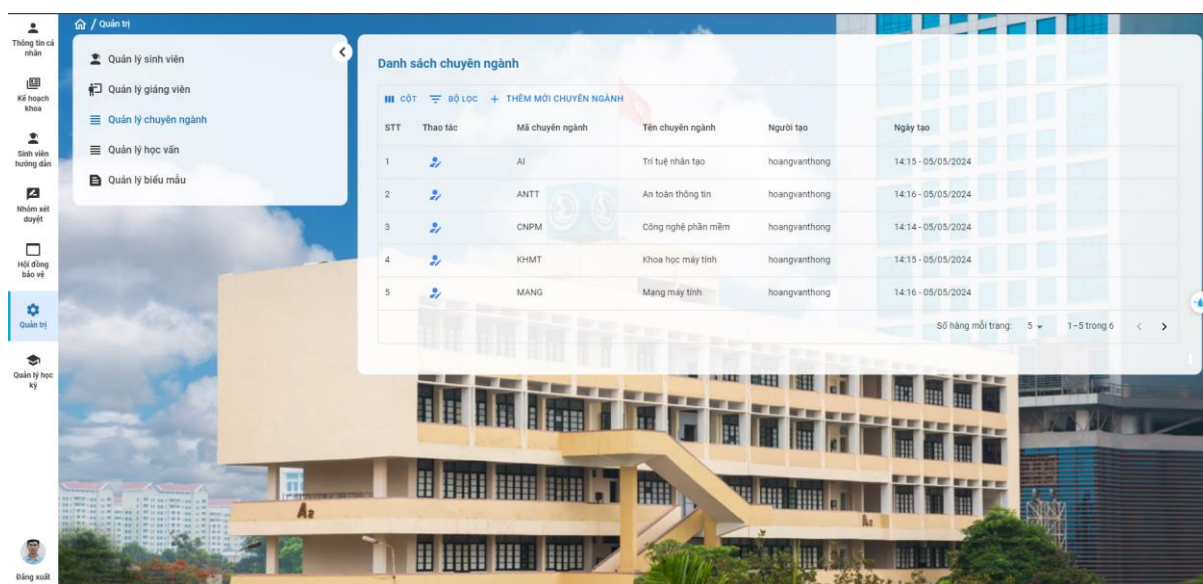
Hình 4.10. Chức năng thêm danh sách sinh viên



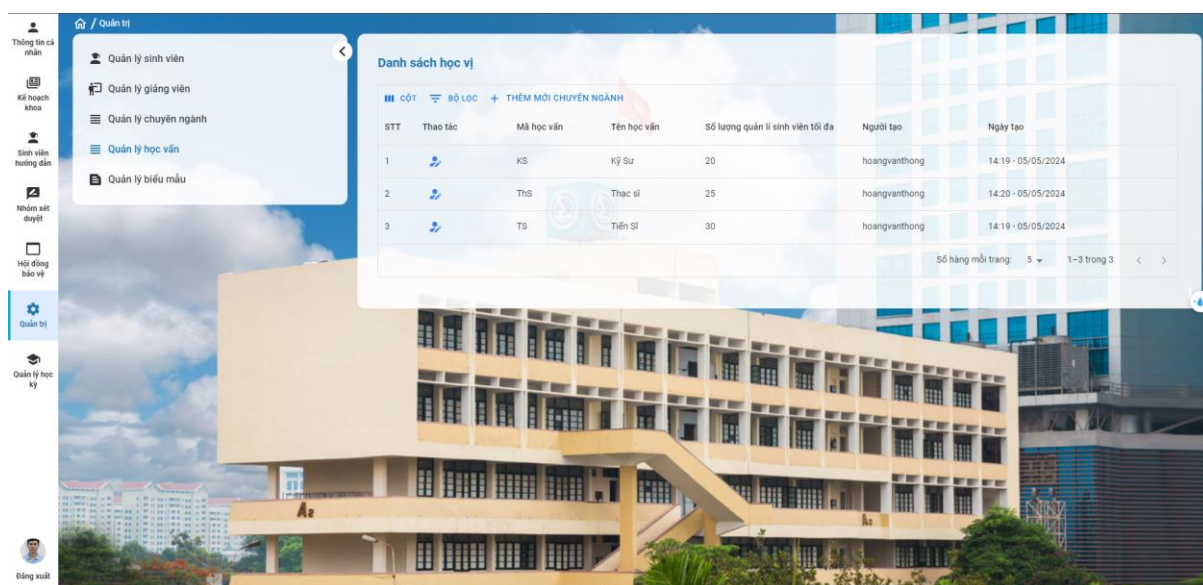
Hình 4.11. Chức năng quản lý giảng viên



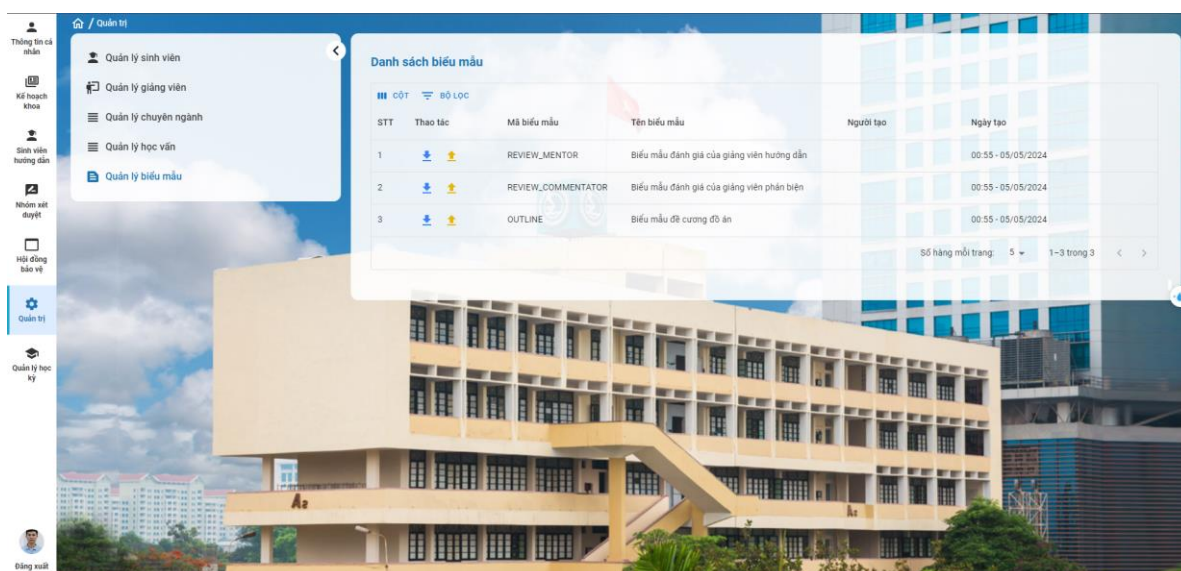
Hình 4.12. Chức năng thêm giảng viên



Hình 4.13. Chức năng quản lý chuyên ngành



Hình 4.14. Chức năng quản lý học vấn



Hình 4.15. Chức năng quản lý biểu mẫu



Hình 4.16. Chức năng quản lý thông tin cá nhân của giảng viên

Trang cá nhân / Chính sửa

QUAY LẠI

Thông tin tài khoản

Tên tài khoản: hoangvanthong

Họ và tên: Hoàng Văn Thông

Email: hoangvanthong@gmail.com

Số điện thoại: 0367218700

Địa chỉ: Hà Nội

Thông tin giảng viên

Chuyên ngành: Khoa học máy tính

CẬP NHẬT

Hình 4.17. Chức năng cập nhật thông tin cá nhân của giảng viên

Kế hoạch khoa

Học kỳ: Học kỳ 1 năm học 2024- 2025

Kế hoạch thực hiện đồ án tốt nghiệp

Mã học kỳ: 1_2024_2025

Bắt đầu: 28/04/2024

Tên học kỳ: Học kỳ 1 năm học 2024- 2025

Kết thúc: 05/05/2024

Kế hoạch đang thực hiện

17 ngày trước (28/04/2024 - 17/05/2024)
Lịch GVHD

17 ngày trước (28/04/2024 - 24/05/2024)
Chốt điểm Hội đồng

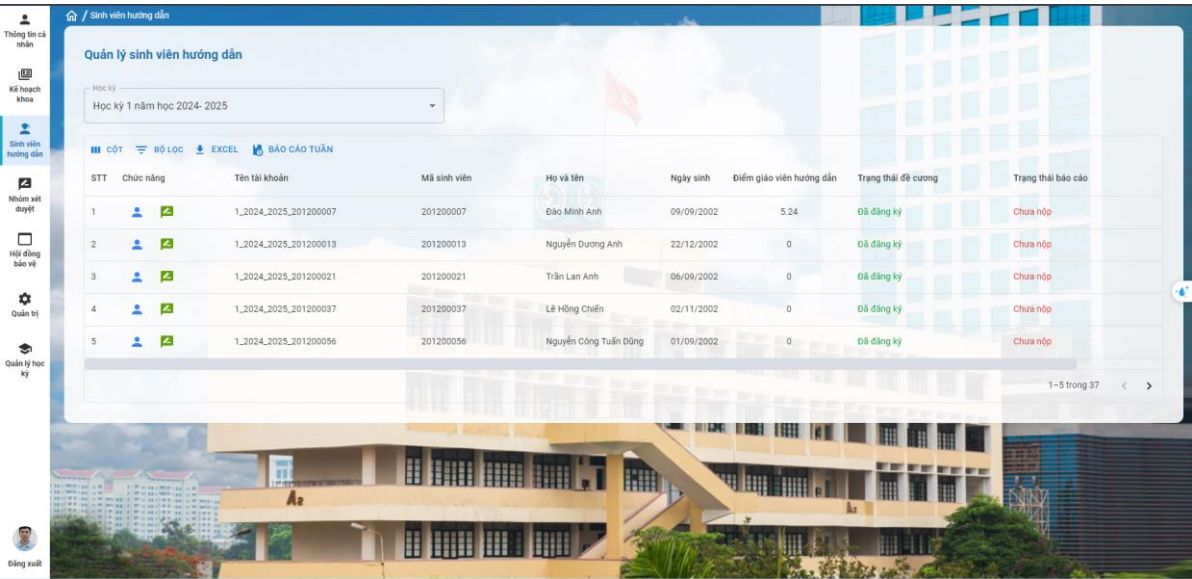
17 ngày trước (28/04/2024 - 25/05/2024)
Chốt đồ án

1 tháng trước (11/04/2024 - 16/05/2024)
Lịch GVPB

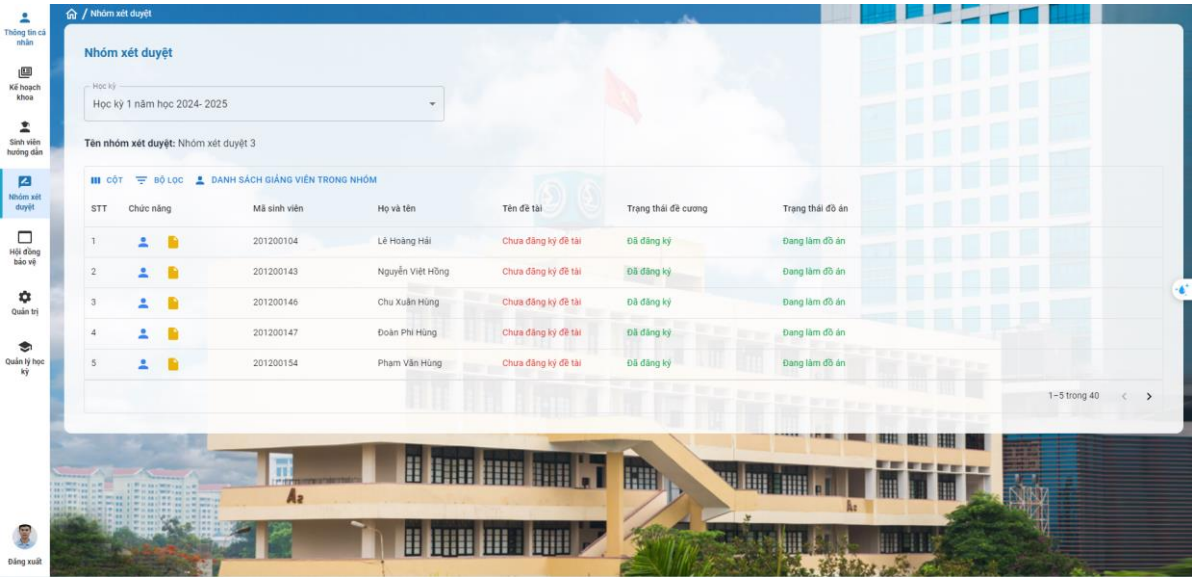
Kế hoạch kết thúc

Kế hoạch sắp tới

Hình 4.18. Chức năng quản lý kế hoạch khoa



Hình 4.19. Chức năng quản lý sinh viên hướng dẫn



Hình 4.20. Chức năng quản lý nhóm xét duyệt



Hình 4.21. Chức năng quản lý đề cương



Hình 4.22. Chức năng cập nhật điểm giảng viên hướng dẫn

BẢN NHẬN XÉT CỦA NGƯỜI ĐỌC DUYỆT ĐỒ ÁN TỐT NGHIỆP

Thông tin giảng viên hướng dẫn
 Mã giảng viên: 202020
 Chuyên ngành: Công nghệ phần mềm
 Học vị: Tiến sĩ

Thông tin sinh viên
 Mã sinh viên: 201200003
 Họ và tên: Nguyễn Hữu An
 Tên đề tài:

Đánh giá đồ án
 Điểm: 2.23

Nhiệm vụ của người đọc duyệt:

- Bố cục và hình thức (2 điểm)
- Nội dung (3 điểm)
- Sản phẩm đồ án (phần mềm, bài lab, ...) (5 điểm)

[LƯU](#)

Hình 4.23. Chức năng cập nhật điểm của giảng viên của đọc duyệt đồ án

Trang cá nhân

Thông tin tài khoản
 Họ và tên: Đào Minh Anh
 Ngày sinh: 09/09/2002
 Giới tính: Nữ
 Email: minhahng@gmail.com
 Số điện thoại: 123456
 Vai trò: Sinh viên
 Trạng thái tài khoản: **Hoạt động**

Thông tin sinh viên
 Mã sinh viên: 201200007
 Lớp: CNTT4
 Khóa: K61
 Mã học kỳ: 1_2024_2025
 Chuyên ngành: Khoa học máy tính

Thông tin đồ án
 Điểm giảng viên hướng dẫn: 5.24
 Điểm giảng viên phản biện: 1.93
 Giảng viên hướng dẫn: Hoàng Văn Thông
 Giảng viên phản biện: Cao Thị Luyện
 Trạng thái làm đồ án: **Mới tạo**

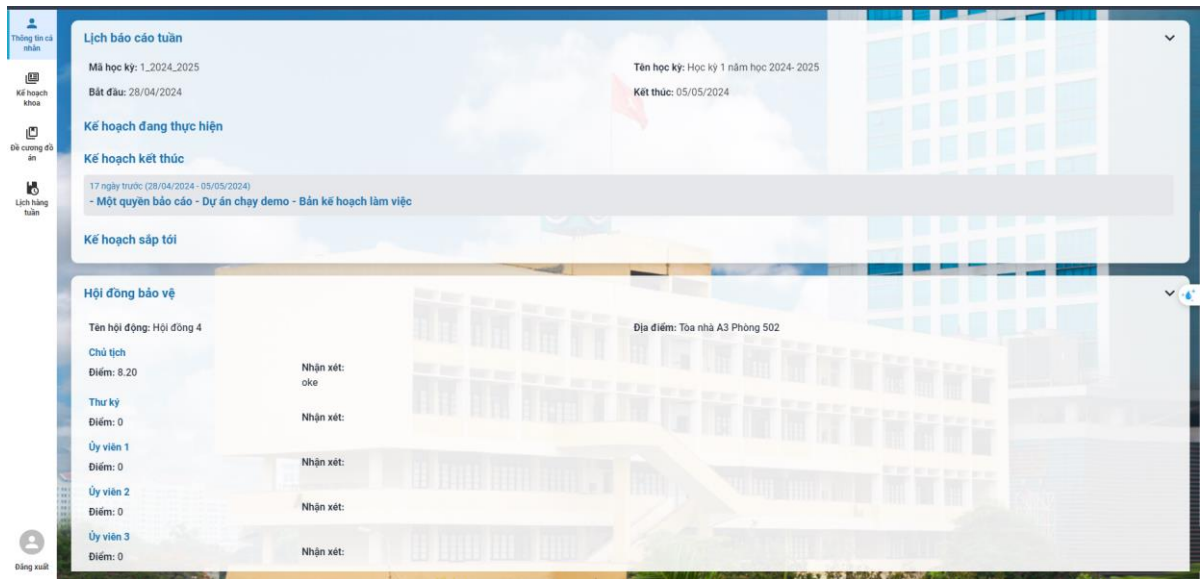
[CHI TIẾT ĐỀ CƯƠNG](#)

Lịch báo cáo tuần

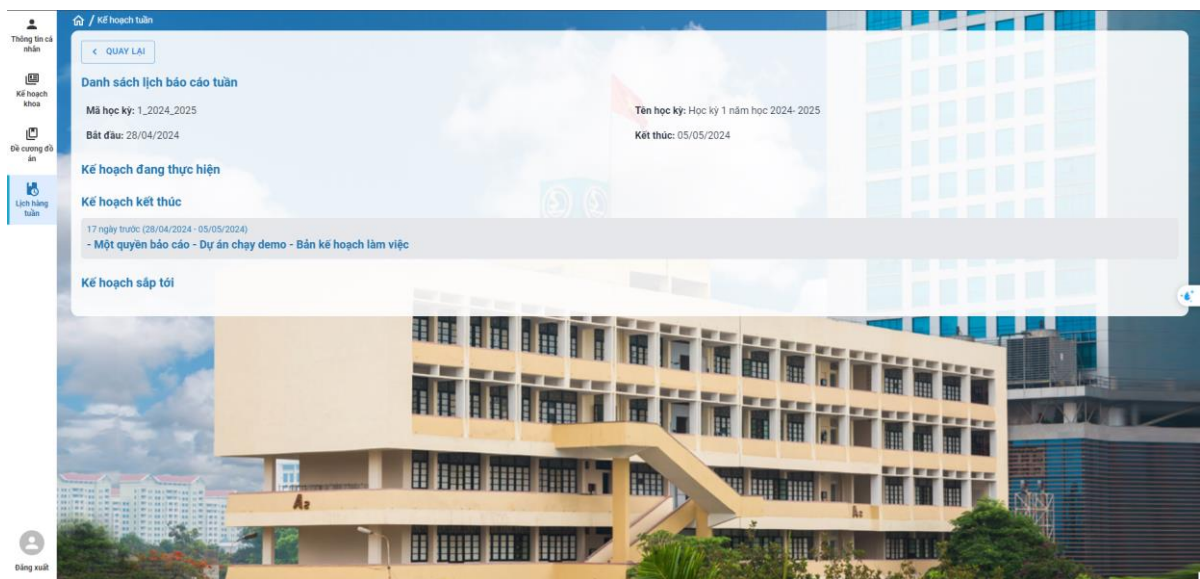
Hội đồng bảo vệ

Đổi mật khẩu

Hình 4.24. Chức năng quản lý trang cá nhân của sinh viên



Hình 4.25. Chức năng quản lý hội đồng bảo vệ và lịch tuần của sinh viên



Hình 4.26. Chức năng theo dõi lịch báo cáo tuần

KẾT LUẬN

Kết quả đạt được

Thông qua quá trình triển khai đề tài “Quản lý đồ án” đã giúp em có cơ hội học tập được thêm rất nhiều kinh nghiệm để trau dồi kiến thức

- Kỹ năng phân tích nghiệp vụ và xử lý vấn đề trong quá trình phát triển đề tài
- Lập lịch lên kế hoạch để giải quyết những vấn đề của dự án
- Vận dụng những kiến thức đã học để áp dụng vào thực tế
- củng cố phát triển tư duy lập trình và cách sử dụng các công nghệ hiện nay

Hạn chế

- Tốc độ của hệ thống còn chậm và có thể ảnh hưởng đến người dùng
- Hệ thống chưa thể ứng dụng được AI vào việc tự sắp xếp và gán giảng viên

Hướng phát triển

Đề tài sẽ phát triển theo hướng tự động hóa các thao tác và quản lý sinh viên, giảng viên trong quá trình làm đồ án cũng như tích hợp những công cụ, thuật toán để có thể quản lý một cách hiệu quả. Điều này có thể đòi hỏi việc nghiên cứu chuyên sâu về thuật toán và sử dụng AI vào việc phân tích

Tiếp đó sẽ triển khai kiểm thử một cách hoàn chỉnh toàn bộ các chức năng của hệ thống và đánh giá về hiệu năng cũng như tốc độ xử lý để tìm ra lỗi, khắc phục và hoàn thiện hệ thống