

BỘ GIÁO DỤC VÀ ĐÀO TẠO
ĐẠI HỌC PHENIKAA



BÁO CÁO ĐỒ ÁN TỐT NGHIỆP

**Xây dựng và tối ưu hóa website thi trực tuyến
và quản lý học tập**

Sinh viên: Nguyễn Hoàng Dương

Mã số sinh viên: 21013110 Khóa: 15

Ngành: Công Nghệ Thông Tin

Hệ: Đại học chính quy

Giảng viên hướng dẫn: *Ths. Phạm Trung Dũng*

Hà Nội – Năm 2025

MỤC LỤC

DANH MỤC HÌNH ẢNH.....	5
DANH MỤC BẢNG BIỂU.....	7
LỜI CAM ĐOAN.....	10
LỜI MỞ ĐẦU.....	11
CHƯƠNG I: CƠ SỞ LÝ THUYẾT.....	12
1. Giới thiệu về thi trực tuyến.....	12
1.1. Khái niệm về thi trực tuyến.....	12
1.2. Nền tảng và công nghệ sử dụng.....	12
1.2.1. HTML [1].....	12
1.2.2. CSS [2].....	14
1.2.3. Javascript [3].....	16
1.2.4. JavaScript XML.....	17
1.2.5. Bootstrap [4].....	18
1.2.6. ReactJS [5].....	19
1.2.7. Spring Boot [6].....	20
1.2.8. Spring JPA.....	22
1.2.9. MySQL [7].....	23
1.2.10. Docker [8].....	24
1.3. Mô hình phát triển dự án Three Layer [9].....	26
CHƯƠNG II: PHÂN TÍCH ĐẶC TẢ CHỨC NĂNG HỆ THỐNG.....	28
2. Đặc tả hệ thống.....	28
2.1. Kiến trúc hệ thống.....	28
2.2. Yêu cầu hệ thống.....	30

2.2.1.	Yêu cầu chức năng.....	30
2.2.1.	Yêu cầu phi chức năng.....	31
2.3.	Thiết kế hệ thống.....	32
2.3.1.	Use Case tổng quan.....	32
2.3.2.	Use Case phân rã theo chức năng.....	32
2.4.	Bản đặc tả các chức năng.....	38
2.4.1.	Kịch bản usecase đăng nhập.....	38
2.4.2.	Kịch bản usecase đăng kí.....	40
2.4.3.	Kịch bản usecase quản lí tài khoản.....	41
2.4.5.	Kịch bản usecase con UC#03_1 thêm tài khoản mới.....	42
2.4.6.	Kịch bản usecase con UC#03_02: sửa thông tin tài khoản.....	43
2.4.7.	Kịch bản usecase con UC#03_03: xóa tài khoản.....	44
2.4.8.	Kịch bản usecase con UC#03_04: phân quyền tài khoản.....	45
2.4.9.	Kịch bản usecase con UC#03_05: xem lịch sử đăng nhập.....	46
2.4.10.	Kịch bản usecase quản lí lớp học.....	47
2.4.11.	Kịch bản usecase con UC#04_01: thêm lớp học mới.....	48
2.4.12.	Kịch bản usecase con UC#04_02: sửa thông tin lớp học.....	49
2.4.13.	Kịch bản usecase con UC#04_03: xóa lớp học.....	50
2.4.14.	Kịch bản usecase con UC#04_04: thêm sinh viên vào lớp học.....	51
2.4.15.	Kịch bản usecase con UC#04_05: xóa sinh viên khỏi lớp học.....	52
2.4.16.	Kịch bản usecase con UC#04_06: thống kê sinh viên theo lớp học.....	53
2.4.17.	Kịch bản usecase con UC#04_07: tra cứu sinh viên.....	54
2.4.18.	Kịch bản usecase quản lí bài thi.....	55

2.4.19.	Kịch bản usecase con UC#05_01: thêm bài thi mới.....	56
2.4.20.	Kịch bản usecase con UC#05_02: sửa thông tin bài thi.....	58
2.4.21.	Kịch bản usecase con UC#05_03: xóa bài thi.....	59
2.4.22.	Kịch bản usecase con UC#05_04: xem chi tiết bài thi.....	60
2.4.23.	Kịch bản usecase làm bài thi.....	61
CHƯƠNG III: THIẾT KẾ CHỨC NĂNG VÀ CƠ SỞ DỮ LIỆU.....		63
3.	Thiết kế chức năng hệ thống và cơ sở dữ liệu.....	63
3.1.	Sơ đồ tuần tự.....	63
3.1.1.	Sơ đồ tuần tự chức năng đăng nhập.....	63
3.1.2.	Sơ đồ tuần tự chức năng đăng kí.....	64
3.1.3.	Sơ đồ tuần tự chức năng quản lí tài khoản.....	65
3.1.4.	Sơ đồ tuần tự quản lí lớp học (môn học).....	68
3.1.5.	Sơ đồ tuần tự chức năng thống kê.....	70
3.2.	Thiết kế cơ sở dữ liệu.....	71
CHƯƠNG IV: KIỂM THỬ VÀ TRIỂN KHAI HỆ THỐNG.....		79
4.	Kiểm thử, triển khai và thực nghiệm.....	79
4.1.	Kiểm thử [10].....	79
4.1.1.	Phân tích các trường hợp kiểm thử.....	79
4.1.2.	Danh sách các testcase và kết quả thực thi.....	83
4.2.	Yêu cầu về cài đặt.....	83
4.3.	Kết quả thực nghiệm.....	84
CHƯƠNG V: KẾT LUẬN.....		95
5.	Kết quả đạt được và hướng phát triển.....	95
5.1.	Kết quả đạt được.....	95

- Đóng gói docker và deploy thành công lên vps.....	95
5.2. Hạn chế.....	95
5.3. Hướng phát triển.....	96
TÀI LIỆU THAM KHẢO.....	97
PHỤ LỤC.....	98

DANH MỤC HÌNH ẢNH

Hình 1.1: Hypertext Markup Language.....	13
Hình 1.2: Cascading Style Sheets.....	14
Hình 1.3: JavaScript.....	16
Hình 1.4: JSX.....	17
Hình 1.5: Bootstrap.....	18
Hình 1.6: ReactJS.....	19
Hình 1.7: Spring Boot.....	20
Hình 1.8: Cấu trúc của Spring Boot.....	21
Hình 1.9: Spring JPA.....	22
Hình 1.10: MySQL.....	23
Hình 1.11: Cấu trúc Docker.....	24
Hình 1.12: Kiến trúc và thành phần mô hình Three Layer.....	26
Hình 2.1: Kiến trúc mã nguồn Frontend.....	28
Hình 2.2: Kiến trúc mã nguồn Backend.....	29
Hình 2.3: Biểu đồ usecase tổng quát.....	32
Hình 2.4: Biểu đồ usecase phân rã chức năng đăng nhập đăng kí.....	32
Hình 2.5: Biểu đồ usecase phân rã chức năng quản lí tài khoản.....	33
Hình 2.6: Biểu đồ usecase phân rã chức năng quản lí thông tin cá nhân.....	33
Hình 2.7: Biểu đồ usecase phân rã chức năng quản lí lớp học.....	34
Hình 2.8: Biểu đồ usecase phân rã chức năng quản lí câu hỏi.....	34
Hình 2.9: Biểu đồ usecase phân rã chức năng quản lí bài thi.....	35
Hình 2.10: Biểu đồ usecase phân rã chức năng làm bài thi.....	35
Hình 2.11: Biểu đồ usecase phân rã chức năng quản lí thống kê báo cáo.....	36
Hình 2.12: Sơ đồ quan hệ thực thể các chức năng.....	37
Hình 3.1: Sơ đồ tuần tự chức năng đăng nhập.....	63
Hình 3.2: Sơ đồ tuần tự chức năng đăng kí.....	64
Hình 3.3: Sơ đồ tuần tự chức năng quản lí tài khoản (xem tài khoản).....	65
Hình 3.4: Sơ đồ tuần tự chức năng quản lí tài khoản (xóa tài khoản).....	66

Hình 3.5: Sơ đồ tuần tự chức năng quản lý tài khoản (phân quyền).....	67
Hình 3.6: Sơ đồ tuần tự quản lý lớp học (quản lý lớp)	68
Hình 3.7: Sơ đồ tuần tự quản lý lớp học (quản lý sinh viên trong lớp)	69
Hình 3.8: Sơ đồ tuần tự chức năng thống kê.....	70
Hình 3.9: Hình thiết kế cơ sở dữ liệu	71
Hình 4.1: Giao diện Landing Page.....	84
Hình 4.2: Giao diện chức năng đăng kí (không phải admin).....	85
Hình 4.3: Giao diện chức năng đăng nhập.....	85
Hình 4.4: Giao diện chức năng đăng nhập.....	85
Hình 4.5: Giao diện Admin Dashboard.....	86
Hình 4.6: Giao diện Lecture Dashboard.....	86
Hình 4.7: Giao diện Student Dashboard.....	87
Hình 4.8: Giao diện quản lý lớp học của admin và lecture.....	87
Hình 4.9: Giao diện chức năng quản lý sinh viên trong lớp học.....	88
Hình 4.10: Giao diện quản lý bài thi.....	88
Hình 4.11: Giao diện chức năng tạo bài thi.....	89
Hình 4.12: Giao diện chức năng quản lý câu hỏi trong bài thi.....	89
Hình 4.13: Giao diện làm bài thi.....	90
Hình 4.14: Giao diện hệ thống trợ lý học tập AI.....	90
Hình 4.15: Giao diện hiển thị kết quả bài thi.....	91
Hình 4.16: Giao diện chức năng thống kê điểm của admin và lecture.....	91
Hình 4.17: Giao diện chức năng thống kê điểm với student.....	92
Hình 4.18: Giao diện thống kê sinh viên.....	92
Hình 4.19: Giao diện thống kê giảng viên.....	93
Hình 4.20: Giao diện thống kê bài thi.....	93
Hình 4.21: Giao diện quản lý chức năng người dùng của admin.....	94
Hình 4.22: Giao diện chức năng bảo mật 2 lớp.....	94

DANH MỤC BẢNG BIỂU

Bảng 2.1: Kịch bản usecase đăng nhập.....	38
Bảng 2.2: Kịch bản usecase đăng kí.....	40
Bảng 2.3: Kịch bản usecase quản lí tài khoản.....	41
Bảng 2.4: Kịch bản usecase con UC#03_1 thêm tài khoản mới.....	42
Bảng 2.5: Kịch bản usecase con UC#03_02: sửa thông tin tài khoản.....	43
Bảng 2.6: Kịch bản usecase con UC#03_03: xóa tài khoản.....	44
Bảng 2.7: Kịch bản usecase con UC#03_04: phân quyền tài khoản.....	45
Bảng 2.8: Kịch bản usecase con UC#03_05: xem lịch sử đăng nhập.....	46
Bảng 2.9: Kịch bản usecase quản lí lớp học.....	47
Bảng 2.10: Kịch bản usecase con UC#03_01: thêm lớp học mới.....	48
Bảng 2.11: Kịch bản usecase con UC#04_02: sửa thông tin lớp học.....	49
Bảng 2.12: Kịch bản usecase con UC#04_03: xóa lớp học.....	50
Bảng 2.13: Kịch bản usecase con UC#04_04: thêm sinh viên vào lớp học.....	51
Bảng 2.14: Kịch bản usecase con UC#04_05: xóa sinh viên khỏi lớp học.....	52
Bảng 2.15: Kịch bản usecase con UC#04_06: thống kê sinh viên theo lớp học.....	53
Bảng 2.16: Kịch bản usecase con UC#04_07: tra cứu sinh viên.....	54
Bảng 2.17: Kịch bản usecase quản lí bài thi.....	55
Bảng 2.18: Kịch bản usecase con UC#05_01: thêm bài thi mới.....	57
Bảng 2.19: Kịch bản usecase con UC#05_02: sửa thông tin bài thi.....	58
Bảng 2.20: Kịch bản usecase con UC#05_03: xóa bài thi.....	59
Bảng 2.21: Kịch bản usecase con UC#05_04: xem chi tiết bài thi.....	60
Bảng 2.22: Kịch bản usecase làm bài thi.....	62
Bảng 3.1: Danh sách các bảng trong cơ sở dữ liệu.....	72
Bảng 3.2: Bảng User.....	73
Bảng 3.3: Bảng login_history.....	74
Bảng 3.4: Bảng class.....	74
Bảng 3.5: Bảng student_class.....	75
Bảng 3.6: Bảng exam.....	76

Bảng 3.7: Bảng student_exam.....	76
Bảng 3.8: Bảng question	77
Bảng 4.1: Bảng phân tích các trường hợp kiểm thử.....	83

LỜI CẢM ƠN

Lời cảm ơn đầu tiên cho phép tôi được gửi đến các thầy cô đã giảng dạy trong trường Đại học Pheikaa những người đã truyền dạy cho chúng tôi rất nhiều những kiến thức hay và có ích để chúng tôi có thể hoàn thiện bản thân cũng như có thêm nhiều kiến thức hơn về cuộc sống. Tiếp đến cho chúng tôi được gửi lời cảm ơn đến Ths.Phạm Trung Dũng với sự quan tâm, dạy dỗ, chỉ bảo tận tình chu đáo của thầy mà chúng tôi đã có thể xây dựng và hoàn thành báo cáo của mình một cách xuất sắc nhất.

Trong quá trình học tập vừa qua, với thời gian ngắn ngủi cũng như kiến thức còn nhiều thiếu sót của chúng tôi, chúng tôi mong thầy/cô có thể bỏ qua và tạo điều kiện tốt nhất cho chúng tôi để chúng tôi hoàn thành bài báo cáo này.

Chúng tôi xin chân thành cảm ơn!

LỜI CAM ĐOAN

Đề tài này là do tôi tự thực hiện dựa vào một số tài liệu và không sao chép từ tài liệu hay công trình đã có trước đó. Nếu có sao chép chúng tôi xin hoàn toàn chịu trách nhiệm.

Hà Nội, ngày 25 tháng 10 năm 2025

NGƯỜI VIẾT

LỜI MỞ ĐẦU

1. Mục đích

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc ứng dụng công nghệ vào giáo dục đang trở thành xu hướng tất yếu nhằm nâng cao chất lượng giảng dạy, quản lý và kiểm tra đánh giá. Đặc biệt, sau đại dịch COVID-19, nhu cầu tổ chức các kỳ thi trực tuyến ngày càng tăng cao, đặt ra yêu cầu về một hệ thống thi cử hiện đại, linh hoạt và an toàn. Từ thực tiễn đó, tôi lựa chọn xây dựng website thi trực tuyến như một giải pháp hỗ trợ tổ chức thi hiệu quả hơn trong môi trường học đường hiện đại.

2. Mục tiêu

2.1. Mục tiêu kỹ thuật

Xây dựng một giao diện dùng React và Bootstrap trực quan, thân thiện với người dùng và dễ sử dụng, tương thích trên mọi thiết bị.

Sử dụng Java Spring Boot để xây dựng máy chủ (Server) xử lý các yêu cầu từ người dùng.

Lưu trữ dữ liệu trên MySQL, đảm bảo dữ liệu người dùng và kết quả thi được lưu trữ an toàn.

Áp dụng các cơ chế bảo mật như JWT (JSON Web Token) để xác thực người dùng.

2.2. Mục tiêu phát triển

Trong tương lai hi vọng có thể mở rộng hỗ trợ các kì thi có quy mô lớn, với hàng nghìn thí sinh tham gia đồng thời.

Có thể tích hợp AI để phân tích kết quả và đưa ra hướng cải thiện học tập cho thí sinh.

3. Phạm vi

Dự án Website thi trực tuyến được phát triển nhằm đáp ứng nhu cầu trong khuôn khổ các trường đại học. Hệ thống sẽ được hướng tới phù hợp với yêu cầu và mục tiêu của trường cho các kì thi trắc nghiệm hay tự luận nhỏ.

CHƯƠNG I: CƠ SỞ LÝ THUYẾT

1. Giới thiệu về thi trực tuyến

1.1. Khái niệm về thi trực tuyến

Khi xu hướng chuyển dịch công nghệ trong Giáo dục & Đào tạo nói riêng, các lĩnh vực khác nói chung ngày một được chú trọng. Cùng với thời đại công nghệ dần tiến đến 5.0, việc dạy và học trực tuyến, đào tạo và tuyển dụng nhân sự online hay thực hiện công tác tuyên truyền trong quần chúng nhân dân thông qua việc ứng dụng công nghệ thông tin đang được triển khai rầm rộ. Vậy nên để đáp ứng nhu cầu thì hình thức thi trực tuyến ra đời.

Thi trực tuyến là hình thức mà người tham gia dự thi sử dụng các thiết bị điện tử như điện thoại, máy tính bảng, laptop, máy tính bàn có kết nối internet thực hiện các bài thi trên các nền tảng/phần mềm thi trực tuyến. Và bằng việc ứng dụng công nghệ hệ thống sẽ tự cung cấp các bộ đề thi, tự động chấm điểm và đưa ra kết quả. Người dùng có thể tham gia ở bất kì nơi nào, chỉ cần có kết nối mạng.

Ưu điểm :

- Tiết kiệm thời gian và chi phí tổ chức kì thi.
- Tự động hóa quá trình chấm thi, hạn chế được rủi ro gian lận.
- Có sự linh hoạt trong việc sắp xếp và tổ chức các kì thi.

Nhược điểm:

- Phụ thuộc vào kết nối internet và thiết bị phần cứng như máy tính, laptop, điện thoại, ...
- Vấn đề về bảo mật tránh gian lận trong thi cử.

1.2. Nền tảng và công nghệ sử dụng

1.2.1. HTML [1]



Hình 1.1: Hypertext Markup Language

HTML viết tắt cho Hypertext Markup Language, tạm dịch là ngôn ngữ đánh dấu siêu văn bản. HTML được sử dụng để tạo và cấu trúc các thành phần trong trang web hoặc ứng dụng, phân chia các đoạn văn, heading, links, block-quotes,... HTML không phải là ngôn ngữ lập trình. Điều này có nghĩa là nó không thể tạo ra các chức năng "động". Hiểu một cách đơn giản hơn, HTML cũng tương tự như phần mềm Microsoft Word, nó chỉ có tác dụng bố cục và định dạng trang web.

Nhiệm vụ chính của HTML xây dựng cấu trúc siêu văn bản trên một trang web và khai báo các tập tin kỹ thuật số như hình ảnh, video, âm thanh. Ngoài ra, HTML còn có các công dụng quan trọng khác, bao gồm:

- Phân chia và định dạng nội dung: chia đoạn, định dạng và nhấn mạnh văn bản, đặt tiêu đề, ngắt dòng, tạo danh sách, và tạo ký tự đặc biệt.
- Tạo liên kết và chèn hình ảnh: bao gồm liên kết nội bộ và liên kết ngoài web, ảnh minh họa,.. thu hút người dùng
- Tạo các kiểu mẫu: gồm cách hiển thị các đoạn văn bản, hình ảnh, bảng,... giúp cho trang web đồng nhất và dễ dàng quản lý hơn.
- Xác định các thuộc tính của trang web: bao gồm màu sắc, kích thước, vị trí và các thông tin khác.

Ưu điểm:

- Dễ học, dễ sử dụng, có cú pháp đơn giản.
- Hỗ trợ hầu hết các trình duyệt web.
- Là nền tảng cơ bản của mọi trang web.
- Không cần trình biên dịch, có thể chạy trực tiếp trên trình duyệt.

Nhược điểm:

- Không có khả năng xử lý logic (chỉ là ngôn ngữ đánh dấu).
- Phải kết hợp với CSS và JavaScript để tạo giao diện và tương tác động.
- Khó quản lý khi dự án lớn.

1.2.2. CSS [2]



Hình 1.2: Cascading Style Sheets

Cascading Style Sheets là một ngôn ngữ thiết kế được sử dụng trong thiết kế trang web để mô tả cách trình bày, bố cục và phong cách của nội dung trên trang web. CSS giúp tạo ra các trang web chuyên nghiệp và dễ đọc hơn. Lưu ý rằng, CSS không phải ngôn ngữ lập trình.

Nếu ví HTML là bức tường thì CSS giống như lớp sơn trang trí. HTML cung cấp các công cụ cơ bản để cấu trúc nội dung trên trang web còn CSS giúp định dạng (style) nội dung này để nó được hiển thị cho người dùng theo đúng ý định thiết kế. Ba phong cách chính của CSS gồm:

- Internal CSS: Là phong cách được đặt trực tiếp trong thẻ <head> của trang web, cho phép chỉnh sửa các phong cách của trang web nhanh chóng và dễ dàng. Tuy nhiên, nó chỉ ảnh hưởng đến trang web đó, không áp dụng cho các trang khác.
- Inline CSS: Có thể thêm phong cách trực tiếp vào mỗi phần tử HTML trên trang web bằng thuộc tính “style”, cho phép dễ dàng chỉnh sửa phong cách của phần tử mà không cần truy cập trực tiếp vào file CSS.
- External CSS: Cho phép bạn tạo file CSS riêng biệt và áp dụng nó cho nhiều trang web khác nhau. Bạn có thể chỉnh sửa phong cách trong file CSS để thay đổi toàn bộ trang web, qua đó bạn giúp cải thiện tốc độ tải trang web, giảm thiểu thời gian và công sức để thay đổi phong cách của trang web.

Ưu điểm:

- Giúp tách biệt nội dung (HTML) và kiểu dáng (CSS).
- Cải thiện hiệu suất tải trang so với việc sử dụng inline styles.
- Hỗ trợ nhiều framework giúp phát triển nhanh hơn.

Nhược điểm:

- Cấu trúc kế thừa (cascade) đôi khi gây khó khăn trong việc kiểm soát style.
- Không có tính năng lập trình như biến, vòng lặp (trừ khi dùng CSS pre-processors như SASS, LESS).
- Có thể hiển thị khác nhau trên từng trình duyệt.

1.2.3. Javascript [3]



Hình 1.3: JavaScript

JavaScript là một trong những ngôn ngữ lập trình phổ biến nhất hiện nay. Nó được tạo ra bởi Brendan Eich (1995) và trở thành một phần quan trọng của các trang web hiện đại. JavaScript giúp biến những trang web tĩnh trở nên động, tạo ra sự tương tác và cải thiện hiệu suất của máy chủ. Nó làm cho các tính năng như Slideshow, Pop-up quảng cáo, cùng tính năng Autocomplete của Google trở nên dễ dàng và thuận tiện hơn.

Vai trò của JS:

- Tạo bố cục chuyển động với thư viện ảnh và hiệu ứng động.
- Xây dựng các trò chơi, hoạt hình 2D hoặc 3D, và các ứng dụng cơ sở dữ liệu phức tạp để tăng tính tương tác trên trang web.
- Kiểm soát mặc định của trình duyệt bằng cách thay đổi các hành vi mặc định của trình duyệt, chẳng hạn như xử lý các biểu mẫu đăng ký và đăng nhập của người dùng.
- Hỗ trợ việc xử lý dữ liệu phía máy khách (client-side), giảm tải cho máy chủ và tăng tốc độ xử lý trang.

Ưu điểm:

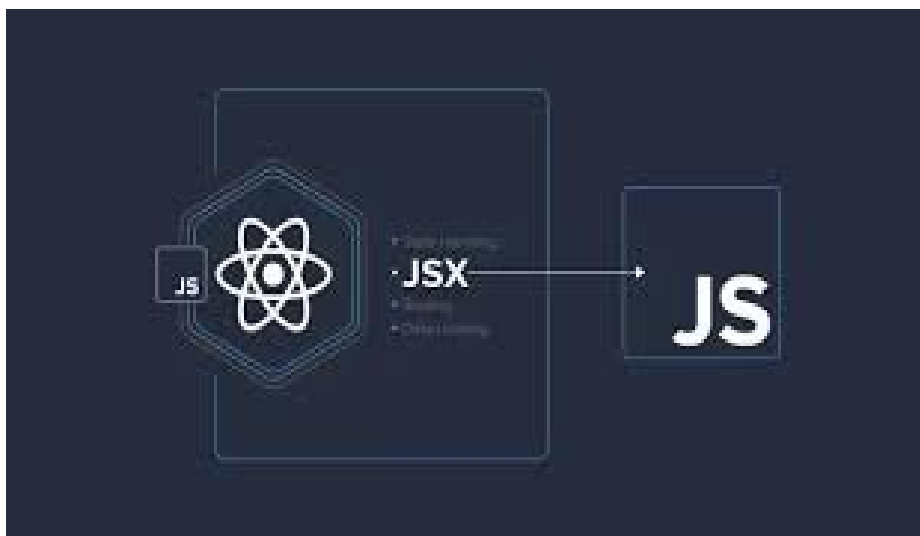
- Ngôn ngữ lập trình phổ biến, chạy được trên cả frontend và backend.

- Hỗ trợ lập trình hướng đối tượng, chức năng, sự kiện.
- Có nhiều thư viện và framework hỗ trợ (React, Angular, Vue, v.v.).

Nhược điểm:

- Vấn đề bảo mật do có thể bị tấn công XSS (Cross-Site Scripting).
- Hiệu suất không cao khi xử lý tác vụ nặng.

1.2.4. JavaScript XML



Hình 1.4: JSX

JavaScript XML (JSX) là cú pháp mở rộng của JavaScript, được sử dụng trong React và nó đi kèm toàn bộ tính năng của JSX. JSX giúp tạo ra cú pháp giống HTML để mô tả cấu trúc và giao diện người dùng trong các ứng dụng web. JSX cho phép việc viết mã JavaScript và HTML trong một file duy nhất, tạo ra mã nguồn dễ đọc và hiểu hơn.

JSX kết hợp tốt với Virtual DOM, giúp tối ưu hóa hiệu suất của ứng dụng React. Thay vì render toàn bộ cây DOM, React sử dụng Virtual DOM để xác định những thay đổi cần được áp dụng, giảm độ trễ và tăng tốc độ render. Có thể thấy JSX kết hợp những điểm mạnh của cả Javascript và HTML, giúp xây dựng UI một cách nhanh chóng và hiệu quả.

Ưu điểm:

- Cung cấp cú pháp giống XML giúp viết React components dễ hiểu hơn.
- Tích hợp JavaScript và HTML vào cùng một file.

Nhược điểm:

- Cần trình biên dịch (Babel) để chạy trên trình duyệt.
- Cú pháp hơi khác so với HTML, có thể gây nhầm lẫn cho người mới.

1.2.5. Bootstrap [4]



Hình 1.5: Bootstrap

Bootstrap là một framework mã nguồn mở được sáng tạo bởi Mark Otto và Jacob Thornton tại Twitter và được xuất bản lần đầu vào 19/8/2011 trên GitHub. Bootstrap bao gồm các HTML, CSS và JavaScript template cho các Web Developer dùng để phát triển giao diện website trong môi trường đa nền tảng một cách nhanh chóng theo một tiêu chuẩn nhất định. Những website được thiết kế bằng công nghệ Bootstrap có ưu điểm là thân thiện và tương thích với nhiều trình duyệt, nhiều thiết bị khác nhau, hỗ trợ màn hình từ desktop đến mobile.

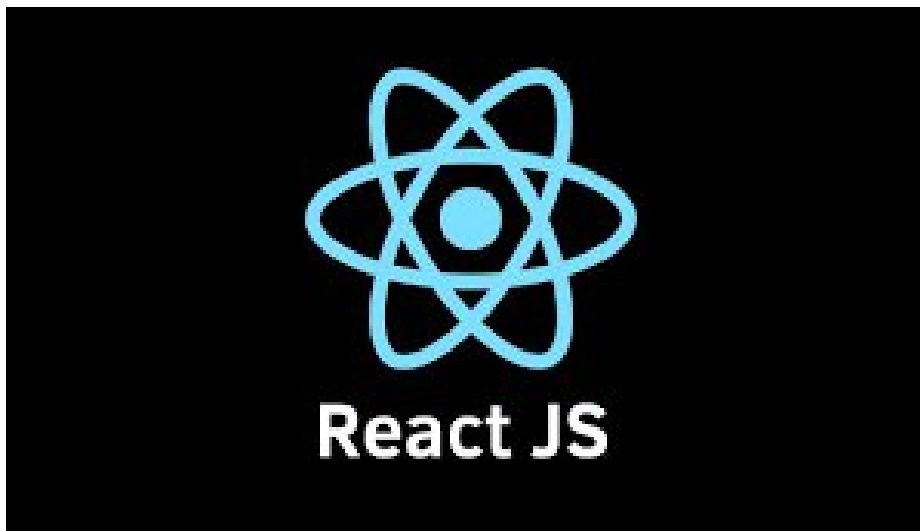
Ưu điểm:

- Rất dễ để sử dụng: Đơn giản vì nó được base trên HTML, CSS và Javascript
- Responsive: Bootstrap xây dựng sẵn responsive css trên các thiết bị Iphones, tablets, và desktops. Tính năng này khiến cho người dùng tiết kiệm được rất nhiều thời gian trong việc tạo ra một website thân thiện với các thiết bị điện tử, thiết bị cầm tay.
- Tương thích với trình duyệt: Nó tương thích với tất cả các trình duyệt như Chrome, Firefox, Internet Explorer, Safari, ...

Nhược điểm:

- Các trang web dùng Bootstrap dễ bị trùng lặp giao diện.
- CSS của Bootstrap có thể làm trang web nặng hơn nếu không tối ưu.

1.2.6. ReactJS [5]



Hình 1.6: ReactJS

Thực tế thì tên chính thức của thư viện là React, không phải ReactJS. Thuật ngữ ReactJS được sử dụng không chính thức trong cộng đồng lập trình viên như một cách để phân biệt với React Native (phiên bản phát triển cho mobile).

React (ReactJS) là một thư viện JavaScript mã nguồn mở, được dùng để xây dựng giao diện người dùng (frontend) cho web. React chỉ tập trung vào phần hiển thị giao diện, chứ không can thiệp vào cách sắp xếp logic nghiệp vụ hoặc cấu trúc ứng dụng.

Một trong những điểm nổi bật nhất của ReactJS đó là việc render dữ liệu không chỉ thực hiện được trên tầng Server mà còn ở dưới Client.

Ưu điểm:

- Tạo giao diện người dùng nhanh và mượt nhờ Virtual DOM.
- Component-based giúp dễ quản lý và tái sử dụng code.
- Có hệ sinh thái lớn, nhiều thư viện hỗ trợ.
- Hỗ trợ SSR (Server-Side Rendering) với Next.js.

Nhược điểm:

- Cần học JSX và cách tổ chức component hợp lý.
- Yêu cầu cài đặt nhiều thư viện phụ trợ.

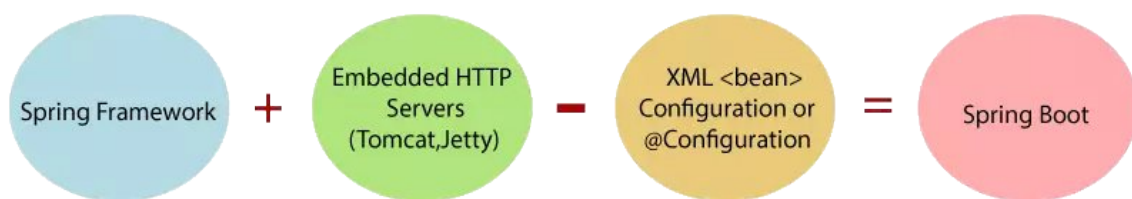
1.2.7. Spring Boot [6]



Hình 1.7: Spring Boot

Spring Boot được xây dựng dựa trên nền của Spring Framework, phát triển bởi ngôn ngữ Java. Nó là một phiên bản có tính tự động hoá cao hơn và đơn giản hơn của Spring.

Spring Boot được phát triển bởi Pivotal Software, với phiên bản đầu tiên (Spring Boot 1.0) được phát hành vào tháng 4 năm 2014. Mục tiêu ban đầu của Spring Boot là giải quyết những hạn chế và phức tạp trong việc cấu hình Spring Framework truyền thống. Spring Framework, mặc dù rất mạnh mẽ và linh hoạt, nhưng yêu cầu nhiều cấu hình thủ công và có thể trở nên cồng kềnh khi xây dựng các ứng dụng phức tạp.



Hình 1.8: Cấu trúc của Spring Boot

Trước khi Spring Boot ra đời, việc cấu hình một ứng dụng Spring thường đòi hỏi nhiều file XML hoặc các class Java cấu hình, gây khó khăn cho việc bảo trì và phát triển. Spring Boot được giới thiệu như một cách tiếp cận đơn giản và hiệu quả hơn, với các cấu hình tự động và các công cụ hỗ trợ như Spring Initializr, giúp nhà phát triển bắt đầu dự án mới nhanh chóng.

Ưu điểm:

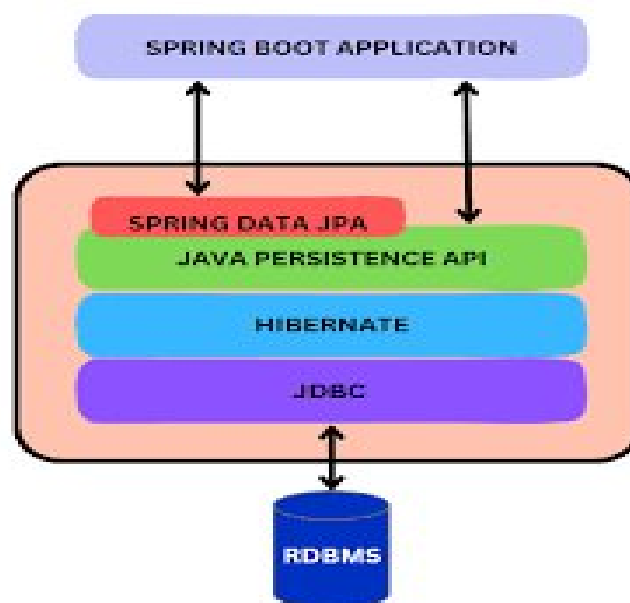
- Cung cấp cấu hình tự động, giúp tiết kiệm thời gian setup.
- Tích hợp dễ dàng với các công nghệ khác như JPA, Hibernate, Security.
- Có hệ sinh thái mạnh và cộng đồng hỗ trợ lớn.

Nhược điểm:

- Cấu hình phức tạp khi mở rộng dự án lớn.
- Tốn nhiều tài nguyên hơn so với các framework nhẹ khác.

1.2.8. Spring JPA

Spring Data JPA là một phần của dự án Spring Data, giúp đơn giản hóa việc truy cập và thao tác với cơ sở dữ liệu thông qua JPA (Java Persistence API). Nó cung cấp một lớp trừu tượng cao hơn cho việc quản lý và truy vấn dữ liệu, giảm bớt công việc viết mã boilerplate



Hình 1.9: Spring JPA

Ưu điểm của việc sử dụng Spring Data JPA:

- Đơn giản hóa mã nguồn: Giảm thiểu mã boilerplate cần viết khi thao tác với cơ sở dữ liệu.
- Tích hợp tốt với Spring Framework: Hỗ trợ tích hợp liền mạch với các tính năng khác của Spring như Dependency Injection, Transaction Management, và các mô-đun khác.
- Hỗ trợ các tính năng nâng cao: Hỗ trợ các tính năng như paging, sorting, và custom query methods một cách dễ dàng.

Nhược điểm:

- Khó tối ưu hiệu suất
- Giới hạn với truy vấn phức tạp

1.2.9. MySQL [7]

Một hệ quản trị cơ sở dữ liệu (Database Management System, viết tắt DBMS) là một chương trình máy tính (một bộ các chương trình) được thiết kế để quản lý một cơ sở dữ liệu, một tập hợp dữ liệu lớn có cấu trúc, phục vụ cho các yêu cầu về dữ liệu của một số lượng lớn người sử dụng.



Hình 1.10: MySQL

MySQL là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là hệ quản trị cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet.

Ưu điểm:

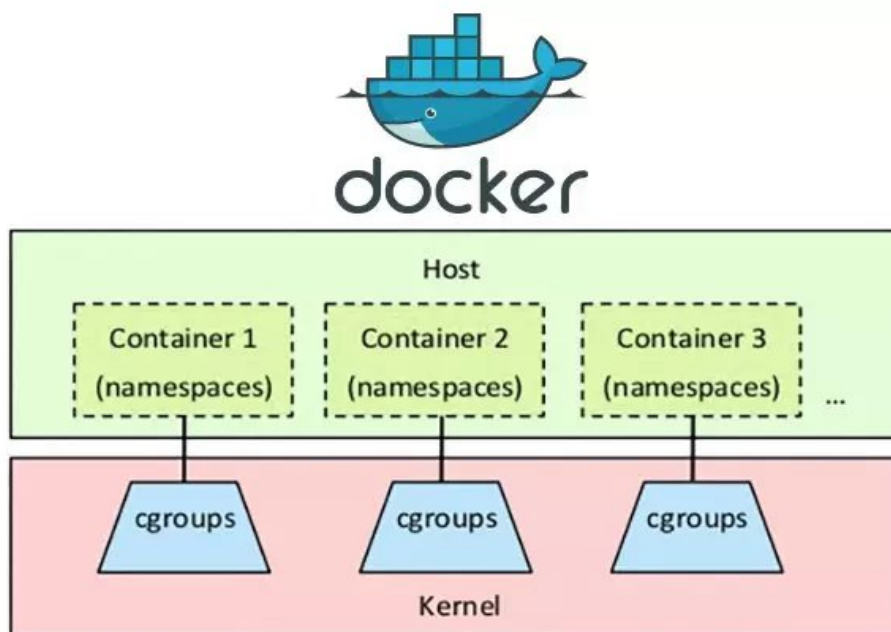
- Giúp truy vấn, thao tác dữ liệu nhanh chóng và chính xác.
- Được hỗ trợ bởi hầu hết các hệ quản trị CSDL (MySQL, PostgreSQL, SQL Server, v.v.).

- Hỗ trợ ACID đảm bảo tính toàn vẹn dữ liệu.

Nhược điểm:

- Không phù hợp với dữ liệu phi cấu trúc.
- Hiệu suất có thể giảm nếu không tối ưu chỉ mục và truy vấn.
- SQL Injection có thể xảy ra nếu không xử lý input cẩn thận.

1.2.10. Docker [8]



Hình 1.11: Cấu trúc Docker

Hiện nay việc cài đặt và triển khai ứng dụng lên một hoặc nhiều server rất phức tạp. Ví dụ từ việc cài các công cụ, môi trường cần cho ứng dụng, hơn nữa là việc đồng nhất môi trường trên các server khác nhau. Vậy nên Docker đã được phát triển để giải quyết vấn đề này.

Thực chất Docker là một nền tảng phát triển, cài đặt và chạy ứng dụng qua các container. Nó cho phép tạo các môi trường độc lập và tách biệt để khởi chạy và phát triển ứng dụng và môi trường này được gọi là container. Khi cần deploy lên bất kỳ server nào chỉ cần run container của Docker thì application của bạn sẽ được khởi chạy ngay lập tức.

Một số khai niệm về Docker cần biết:

- Docker Client: là cách mà bạn tương tác với docker thông qua command trong terminal. Docker Client sẽ sử dụng API gửi lệnh tới Docker Daemon.
- Docker Daemon: là server Docker cho yêu cầu từ Docker API. Nó quản lý images, containers, networks và volume.
- Docker Volumes: là cách tốt nhất để lưu trữ dữ liệu liên tục cho việc sử dụng và tạo apps.
- Docker Hub: là Registry lớn nhất của Docker Images. Có thể tìm thấy images và lưu trữ images của riêng bạn trên Docker Hub.
- Docker Compose: là công cụ cho phép run app với nhiều Docker containers 1 cách dễ dàng hơn. Docker Compose cho phép bạn config các command trong file docker-compose.yml để sử dụng lại. Có sẵn khi cài Docker.
- DockerFile: đây là 1 file config dùng để build ra image. Nó dùng một image cơ bản để xây dựng lớp image ban đầu.

Ưu điểm:

- Có thể khởi chạy container trên mỗi hệ thống mà bạn muốn.
- Thời gian start và stop nhanh (chỉ trong vài giây khi đã build)
- Dễ dàng thiết lập môi trường làm việc. Chỉ cần config 1 lần duy nhất và không bao giờ phải cài đặt lại các dependencies.

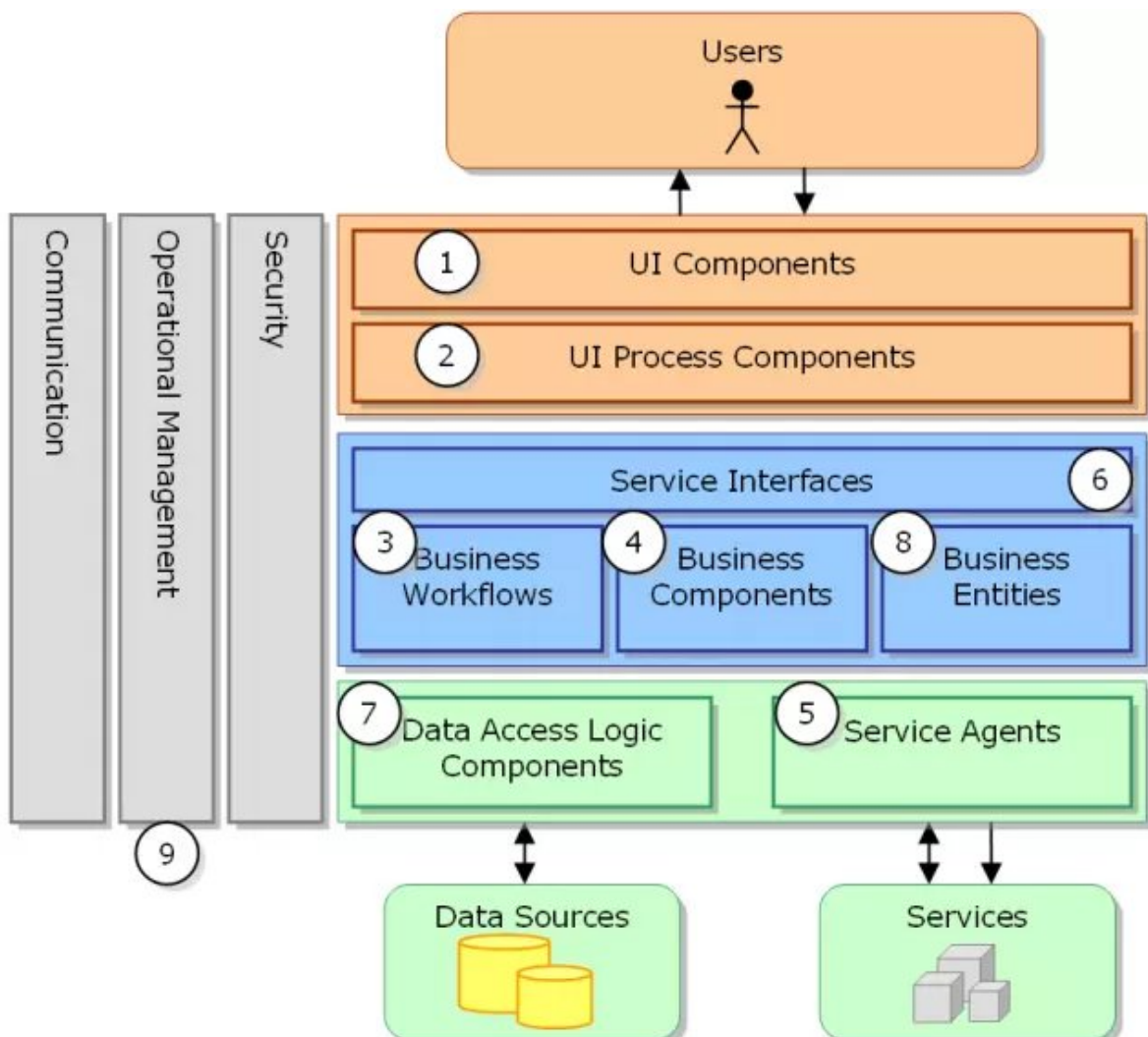
Nhược điểm:

- Tài liệu của Docker có thể không cập nhật kịp theo các nền tảng.
- Vấn đề bảo mật không đảm bảo, các container dễ bị tấn công qua hệ thống server.
- Hạn chế trong việc quản lý nhiều container một lúc.

1.3. Mô hình phát triển dự án Three Layer [9]

Mô hình 3 lớp hay còn được gọi là mô hình Three Layer(3-Layer) ra đời nhằm phân chia các thành phần trong hệ thống, các thành phần cùng chức năng sẽ được nhóm lại với nhau và phân chia công việc cho từng nhóm để dữ liệu không bị chồng chéo và chạy lộn xộn. Mô hình này phát huy hiệu quả nhất khi xây dựng một hệ thống lớn, việc quản lý code và xử lý dữ liệu lỗi dễ dàng hơn.

Mô hình 3-layer gồm có 3 phần chính:



Hình 1.12: Kiến trúc và thành phần mô hình Three Layer

- Presentation Layer (GUI): Lớp này có nhiệm vụ chính là giao tiếp với người dùng. Nó gồm các thành phần giao diện và thực hiện các công việc như nhập liệu, hiển thị dữ liệu, kiểm tra tính đúng đắn dữ liệu trước khi gọi lớp Business Logic Layer (BLL).

- Business Logic Layer (BLL) Layer phân ra 2 thành nhiệm vụ: Đáp ứng các yêu cầu thao tác dữ liệu của GUI layer, xử lý chính nguồn dữ liệu từ Presentation Layer trước khi truyền xuống Data Access Layer và lưu xuống hệ quản trị CSDL. Kiểm tra các ràng buộc, tính toàn vẹn và hợp lệ dữ liệu, thực hiện tính toán và xử lý các yêu cầu nghiệp vụ, trước khi trả kết quả về Presentation Layer.
- Data Access Layer (DAL): Lớp này có chức năng giao tiếp với hệ quản trị CSDL như thực hiện các công việc liên quan đến lưu trữ và truy vấn dữ liệu (tìm kiếm, thêm, xóa, sửa,...).

CHƯƠNG II: PHÂN TÍCH ĐẶC TẢ CHỨC NĂNG HỆ THỐNG

2. Đặc tả hệ thống

2.1. Kiến trúc hệ thống

Dự án sử dụng kiến trúc 3-layers được phân chia rõ ràng như sau:

- Presentation Layer: ở tầng trên cùng này chứa giao diện người dùng, web server (Nginx) phục vụ các static files cũng như các routing điều hướng phía người dùng. Công việc chính của tầng này là hiển thị và xử lý tương tác với người dùng, quản lý trạng thái client cũng như gửi các requests đến backend để xử lý.
- Business Layer: ở tầng này sẽ tiếp nhận các requests và xử lý các yêu cầu logic nghiệp vụ như tính điểm, xác định trạng thái bài thi, kiểm tra JWT để phân quyền theo vai trò cuối cùng trả dữ liệu về cho database.
- Data Layer: tầng này có các chức năng chính như lưu trữ, truy vấn dữ liệu với database.

Kiến trúc giao diện người dùng sử dụng ReactJS:

```
fe/src/
├── components/      # Các thành phần giao diện có thể tái sử dụng
│   ├── auth/       # Thành phần liên quan đến đăng nhập/dăng ký
│   ├── common/     # Thành phần dùng chung cho nhiều trang
│   ├── layout/     # Bố cục trang (thanh điều hướng, chân trang, thanh bên)
│   ├── sections/   # Các phần riêng biệt cho từng trang cụ thể
│   └── ui/         # Thành phần giao diện thuần túy
├── pages/          # Các trang chính của ứng dụng
├── services/       # Lớp giao tiếp với API từ backend
├── hooks/          # Các hook tự tạo để chia sẻ logic
├── contexts/       # Quản lý trạng thái toàn cục
├── routes/         # Cấu hình đường dẫn và điều hướng
├── utils/          # Các hàm tiện ích
├── styles/         # CSS toàn cục
├── theme/          # Cấu hình giao diện (màu sắc, font chữ)
└── assets/         # Tài nguyên tĩnh (hình ảnh, icon)
```

Hình 2.1: Kiến trúc mã nguồn Frontend

- Kiến trúc dựa trên thành phần: Chia giao diện thành các phần nhỏ, độc lập và có thể tái sử dụng
- Hook tùy chỉnh: Chia sẻ logic giữa các thành phần thông qua custom hooks
- Context API: Quản lý dữ liệu toàn cục như thông tin đăng nhập, giao diện
- Lớp dịch vụ: Tách biệt việc gọi API khỏi các thành phần giao diện
- Kiểm soát truy cập dựa trên vai trò: Phân quyền theo vai trò (Admin, Giảng viên, Sinh viên)
- Mẫu bảo vệ đường dẫn: Chỉ cho phép người đã đăng nhập truy cập các trang nhất định

Kiến trúc Backend sử dụng Spring Boot:

```

be/src/main/java/vn/com/example/exam/online/
├── controller/      # Lớp điều khiển - tiếp nhận yêu cầu từ frontend
├── service/         # Lớp xử lý logic nghiệp vụ
├── repository/      # Lớp truy cập dữ liệu từ cơ sở dữ liệu
├── model/           # Các mô hình dữ liệu
│   ├── entity/      # Thực thể JPA (tương ứng với bảng database)
│   ├── request/     # Đối tượng nhận dữ liệu từ frontend
│   └── response/    # Đối tượng trả về cho frontend
├── config/          # Các class cấu hình hệ thống
├── exception/       # Xử lý lỗi và ngoại lệ
├── mapper/          # Chuyển đổi giữa các đối tượng
├── util/            # Các class tiện ích
└── mail/           # Xử lý gửi email

```

Hình 2.2: Kiến trúc mã nguồn Backend

- Kiến trúc phân lớp: Controller → Service → Repository → Entity
- Dependency Injection: Sử dụng Spring IoC Container
- Repository Pattern: Trừu tượng hóa việc truy cập dữ liệu
- DTO Pattern: Đối tượng chuyển đổi dữ liệu cho API
- Builder Pattern: Sử dụng Lombok để sinh mã khởi tạo đối tượng
- Exception Handling Pattern: xử lý ngoại lệ

- Configuration Pattern: Cấu hình được tách ra ngoài

Đóng gói triển khai và Devops:

Sử dụng Docker đóng gói ứng dụng thành các container độc lập. Ở dự án này được đóng gói thành 3 container chính là: frontend, backend và database. Tất cả container được quản lý thông qua Docker Compose.

Bên cạnh đó Nginx được sử dụng làm web server cho frontend.

2.2. Yêu cầu hệ thống

2.2.1. Yêu cầu chức năng

Chức năng chung:

- Thực hiện các chức năng đăng nhập, đăng kí tài khoản.
- Yêu cầu người dùng đăng nhập để sử dụng các chức năng khác.
- Sau khi đăng nhập người dùng có thể tự quản lý thông tin các nhân, quản lý chức năng xác thực 2 yếu tố.
- Sử dụng hệ thống chatbot để hướng dẫn sử dụng cũng như giải đáp.

Admin:

- Quản lý tất cả tài khoản người dùng bao gồm sinh viên và giảng viên với các chức năng như thêm, xóa hay sửa.
- Xem thống kê tất cả lượng người dùng đăng kí, số lớp học và bài thi.
- Quản lý chức năng chỉnh sửa thông tin các khoa, các lớp. môn thi.

Giảng viên (lecture):

- Quản lý các chức năng chỉnh sửa lớp học mà giảng viên đó đứng lớp.
- Quản lý sinh viên mà giảng viên đứng lớp.
- Quản lý bộ câu hỏi và đề thi
- Xem chi tiết bài thi sinh viên
- Thống kê điểm, xuất điểm dạng file excel, csv

Sinh viên (student):

- Làm bài thi với bộ câu hỏi random
- Hiện thị kết quả sau khi hoàn thành bài thi
- Tìm kiếm thông tin lớp học, bài thi

2.2.1. Yêu cầu phi chức năng

Yêu cầu phi chức năng đưa ra tiêu chí nhằm đánh giá hoạt động của hệ thống, đảm bảo hệ thống vận hành ổn định.

Hệ thống quản lý dự án cần đáp ứng được các yêu cầu phi chức năng sau:

- Yêu cầu bảo mật: Bảo vệ thông tin khỏi việc truy cập không được phép, rò rỉ thông tin và các cuộc tấn công mạng. Xác thực và cấp quyền truy cập vào hệ thống cho người dùng dựa trên vai trò và quyền hạn tương ứng. Thực hiện các biện pháp bảo mật dữ liệu theo các tiêu chuẩn quốc tế.
- Yêu cầu mặt giao diện chức năng: Hệ thống đơn giản, dễ sử dụng, đảm bảo người dùng có thể sử dụng thành thạo thông qua hướng dẫn đơn giản. Ngoài ra hệ thống hỗ trợ các trình duyệt phổ biến như Chrome, CocCoc.
- Yêu cầu hiệu năng: Hệ thống đáp ứng các tác vụ và phản hồi trong thời gian ngắn, đảm bảo cân bằng tải phục vụ số lượng người truy cập lớn (khoảng 1000 người tại một thời điểm).

2.3. Thiết kế hệ thống

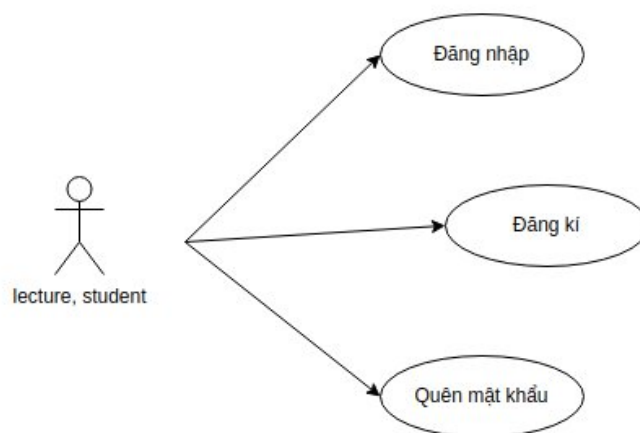
2.3.1. Use Case tổng quan



Hình 2.3: Biểu đồ usecase tổng quát

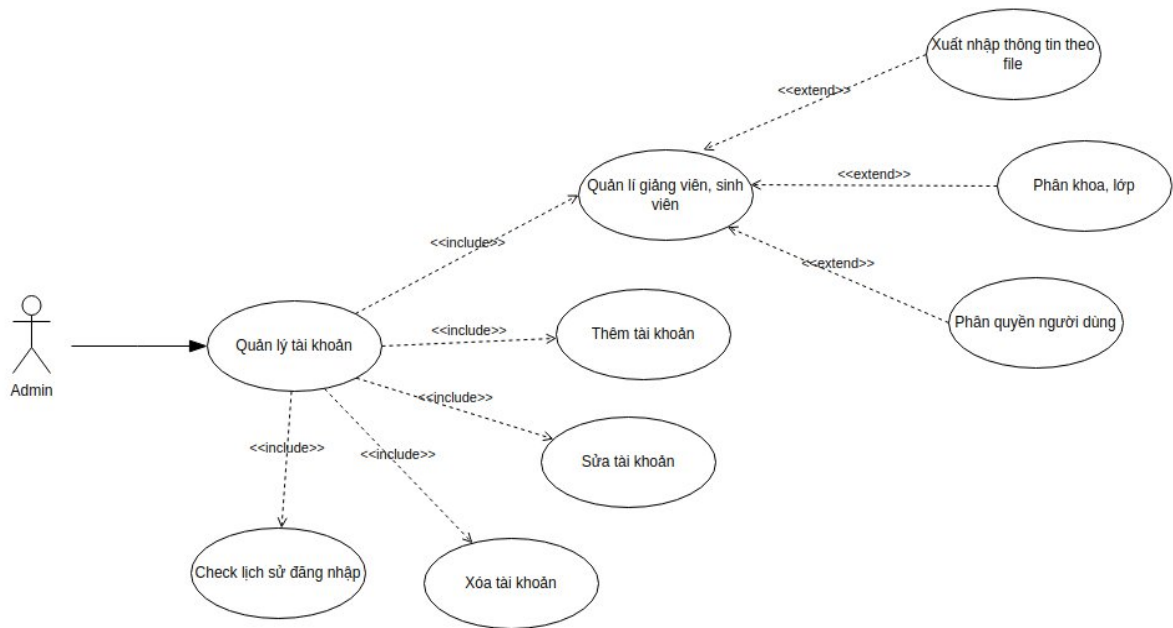
2.3.2. Use Case phân rã theo chức năng

2.3.2.1. Use Case phân rã chức năng đăng nhập đăng kí



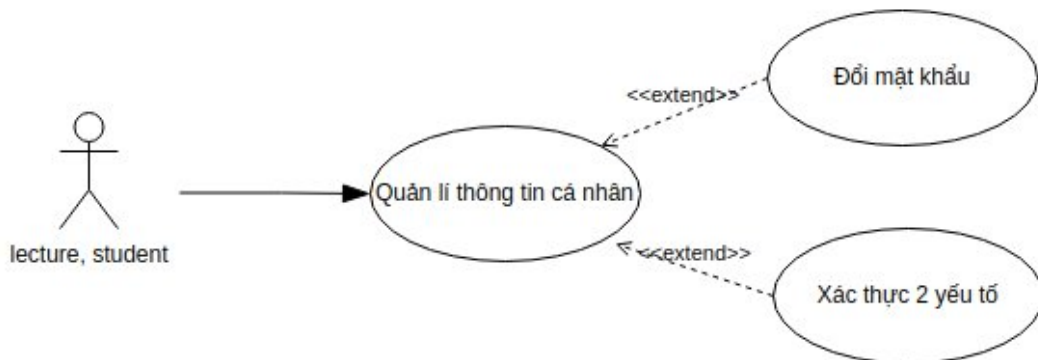
Hình 2.4: Biểu đồ usecase phân rã chức năng đăng nhập đăng kí

2.3.2.2. Use Case phân rã chức năng quản lý tài tài khoản



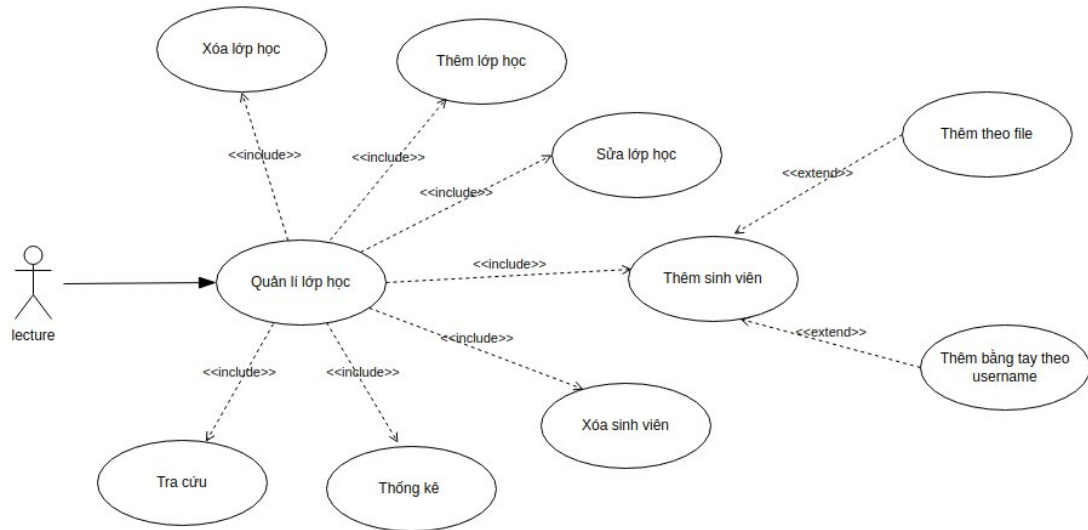
Hình 2.5: Biểu đồ usecase phân rã chức năng quản lý tài tài khoản

2.3.2.3. Use Case phân rã chức năng quản lý thông tin cá nhân



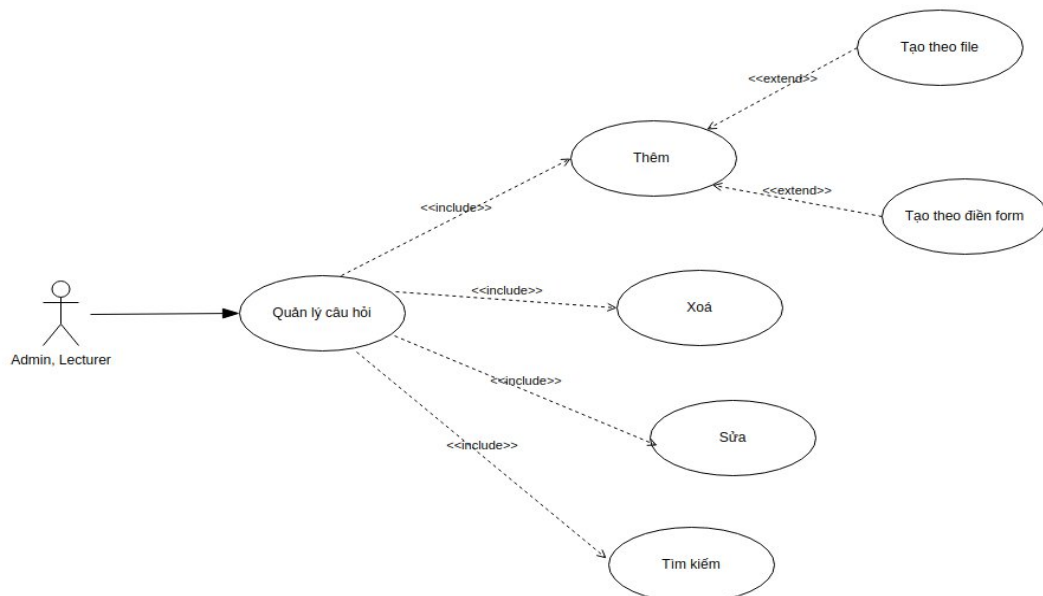
Hình 2.6: Biểu đồ usecase phân rã chức năng quản lý thông tin cá nhân

2.3.2.4. Usecase phân rã chức năng quản lí lớp học



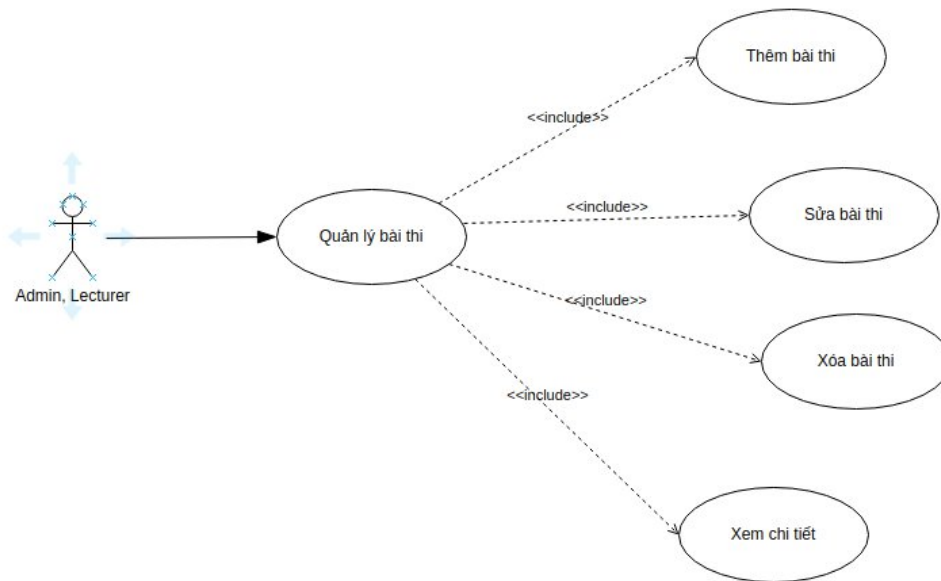
Hình 2.7: Biểu đồ usecase phân rã chức năng quản lí lớp học

2.3.2.5. Usecase phân rã chức năng quản lí câu hỏi



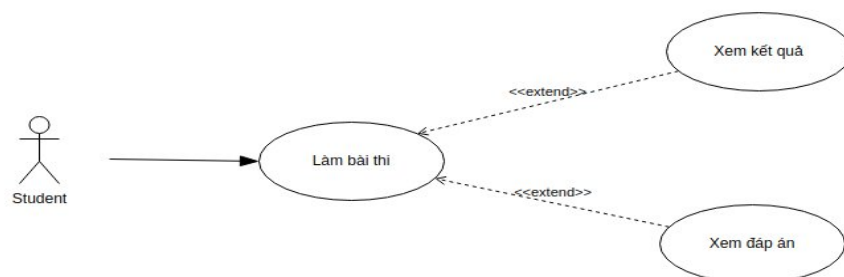
Hình 2.8: Biểu đồ usecase phân rã chức năng quản lí câu hỏi

2.3.2.6. Usecase phân rã chức năng quản lý bài thi



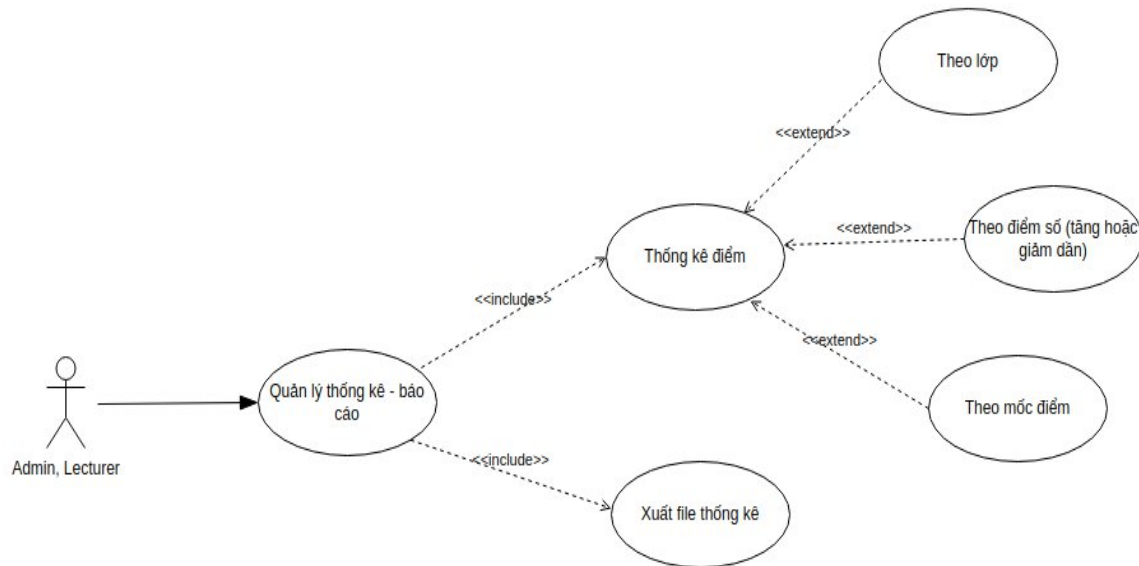
Hình 2.9: Biểu đồ usecase phân rã chức năng quản lý bài thi

2.3.2.7. Use Case phân rã chức năng làm bài thi



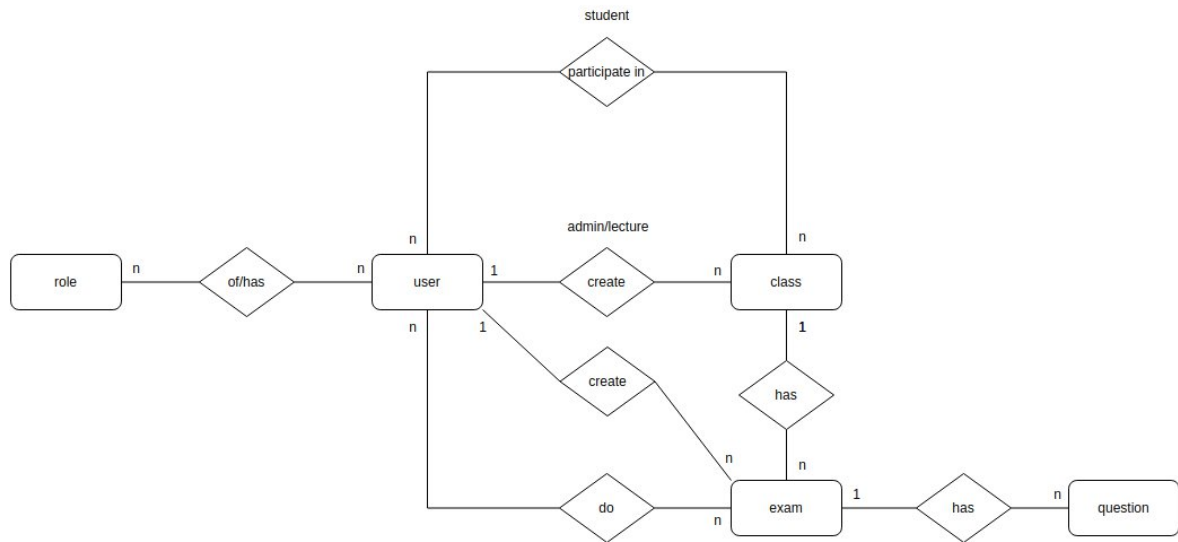
Hình 2.10: Biểu đồ usecase phân rã chức năng làm bài thi

2.3.2.8. Usecase phân rã chức năng quản lý thống kê báo cáo



Hình 2.11: Biểu đồ usecase phân rã chức năng quản lý thống kê báo cáo

2.3.2.9. Sơ đồ quan hệ thực thể các chức năng



Hình 2.12: Sơ đồ quan hệ thực thể các chức năng

2.4. Bản đặc tả các chức năng

2.4.1. Kịch bản usecase đăng nhập

UC#01	Đăng nhập
Mô tả	Người dùng yêu cầu chức năng đăng nhập.
Tác nhân	Admin, giảng viên, sinh viên
Tiền điều kiện	Người dùng có tài khoản trên hệ thống. Tài khoản được phân quyền
Hậu điều kiện	Thành công: đăng nhập thành công vào hệ thống. Lỗi: đăng nhập không thành công
Hành động tác nhân	Phản ứng hệ thống
1. Người dùng truy cập chức năng đăng nhập	1.1. Hệ thống hiển thị popup đăng nhập yêu cầu nhập tài khoản và mật khẩu vào giao diện.
2. Người dùng nhập tài khoản, mật khẩu vào các trường rồi đăng nhập	2.1. Hệ thống kiểm tra và xác thực thông tin, gán thông tin phiên đăng nhập, thông báo đăng nhập thành công và chuyển giao diện.
Ngoại lệ	Phản ứng hệ thống
1. Người dùng nhập sai tài khoản mật khẩu	1.1. Hệ thống thông báo lỗi "Tài khoản hoặc mật khẩu không chính xác" và yêu cầu đăng nhập lại.
2. Người dùng bỏ trống trường bắt buộc	2.1. Hệ thống thông báo lỗi "Vui lòng điền đầy đủ thông tin đăng nhập" và yêu cầu điền đầy đủ thông tin.

Bảng 2.1: Kịch bản usecase đăng nhập

UC#02	Đăng kí
Mô tả	Người dùng yêu cầu chức năng đăng kí tài khoản mới.
Tác nhân	Sinh viên, giảng viên
Tiền điều kiện	Người dùng chưa có tài khoản trên hệ thống.
Hậu điều kiện	Thành công: tạo tài khoản thành công.
Hành động tác nhân	Phản ứng hệ thống
1. Người dùng truy cập chức năng đăng ký	1.1. Hệ thống hiển thị popup đăng ký với các trường thông tin cần thiết.
2. Người dùng nhập thông tin cá nhân và thông tin đăng nhập	2.1. Hệ thống hiển thị các trường thông tin cần thiết: - Bắt buộc: họ tên, email, mật khẩu, vai trò - Tùy chọn: ảnh đại diện
3. Người dùng chọn đăng ký	3.1. Hệ thống kiểm tra tính hợp lệ của thông tin. - Email: đúng định dạng email - Mật khẩu: ít nhất 6 kí tự
	3.2. Hệ thống tạo tài khoản mới.
	3.3. Hệ thống gửi email xác nhận.
	3.4. Hệ thống thông báo đăng ký thành công.
Ngoại lệ	Phản ứng hệ thống
1. Email đã được sử dụng	1.1. Hệ thống thông báo lỗi "Email này đã được đăng ký" và yêu cầu sử dụng email khác.

2. Mật khẩu không đáp ứng yêu cầu	2.1. Hệ thống thông báo lỗi "Mật khẩu phải có ít nhất 8 ký tự" và yêu cầu nhập lại.
3. Mật khẩu xác nhận không khớp	3.1. Hệ thống thông báo lỗi "Mật khẩu xác nhận không khớp" và yêu cầu nhập lại.
4. Người dùng bỏ trống trường bắt buộc	4.1. Hệ thống thông báo lỗi "Vui lòng điền đầy đủ thông tin" và yêu cầu điền đầy đủ thông tin.

2.4.2. Kịch bản usecase đăng kí

Bảng 2.2: Kịch bản usecase đăng kí

UC#03	Quản lí tài khoản
Mô tả	Người dùng sử dụng các chức năng của quản lí tài khoản
Tác nhân	Admin
Tiền điều kiện	Người dùng có tài khoản trên hệ thống với quyền admin
Hành động tác nhân	Phản ứng hệ thống
1. Người dùng truy cập chức năng đăng nhập	1.1. Hệ thống hiển thị popup đăng nhập
2. Người dùng nhập tài khoản đăng nhập admin	2.1. Hệ thống xác thực thông tin và chuyển đến trang quản lí

Bảng 2.3: Kịch bản usecase quản lí tài khoản

2.4.3. Kịch bản usecase quản lí tài khoản

2.4.5. Kịch bản usecase con UC#03_1 thêm tài khoản mới

Hành động tác nhân	Phản ứng hệ thống
1. Admin chọn chức năng "Quản lý tài khoản"	1.1. Hệ thống hiển thị giao diện quản lý tài khoản với danh sách các tài khoản hiện có.
2. Admin chọn chức năng "Thêm tài khoản mới"	2.1. Hệ thống hiển thị form nhập thông tin tài khoản mới.
3. Admin nhập các thông tin cho tài khoản mới (họ tên, email, mã số, vai trò...)	3.1. Hệ thống hiển thị form với các trường thông tin cần thiết.
4. Admin xác nhận tạo tài khoản	4.1. Hệ thống kiểm tra tính hợp lệ của thông tin.
	4.2. Hệ thống tạo tài khoản mới.
	4.3. Hệ thống thông báo tạo tài khoản thành công.
	4.4. Hệ thống gửi email thông báo kèm mật khẩu tạm thời cho người dùng mới.
Ngoại lệ	Phản ứng hệ thống
1. Email đã tồn tại trong hệ thống	1.1. Hệ thống thông báo lỗi "Email này đã được sử dụng" và yêu cầu nhập email khác.
2. Mã số người dùng đã tồn tại	2.1. Hệ thống thông báo lỗi "Mã số này đã tồn tại" và yêu cầu nhập mã số khác.
3. Admin bỏ trống trường bắt buộc	3.1. Hệ thống thông báo lỗi "Vui lòng điền đầy đủ thông tin bắt buộc" và yêu cầu điền đầy đủ.

Bảng 2.4: Kịch bản usecase con UC#03_1 thêm tài khoản mới

2.4.6. Kịch bản usecase con UC#03_02: sửa thông tin tài khoản

Hành động tác nhân	Phản ứng hệ thống
1. Admin tìm kiếm và chọn tài khoản cần chỉnh sửa từ danh sách	1.1. Hệ thống hiển thị danh sách tài khoản với chức năng tìm kiếm và lọc.
2. Admin chọn chức năng "Chỉnh sửa"	2.1. Hệ thống hiển thị form với thông tin hiện tại của tài khoản được chọn.
3. Admin thay đổi thông tin cần chỉnh sửa	3.1. Hệ thống cho phép chỉnh sửa các trường thông tin.
4. Admin xác nhận lưu thay đổi	4.1. Hệ thống kiểm tra tính hợp lệ của thông tin.
	4.2. Hệ thống cập nhật thông tin tài khoản.
	4.3. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	Phản ứng hệ thống
1. Email đã tồn tại trong hệ thống	1.1. Hệ thống thông báo lỗi "Email này đã được sử dụng" và yêu cầu nhập email khác.
2. Admin bỏ trống trường bắt buộc	2.1. Hệ thống thông báo lỗi "Vui lòng điền đầy đủ thông tin bắt buộc" và yêu cầu điền đầy đủ.

Bảng 2.5: Kịch bản usecase con UC#03_02: sửa thông tin tài khoản

2.4.7. Kịch bản usecase con UC#03_03: xóa tài khoản

Hành động tác nhân	Phản ứng hệ thống
1. Admin tìm kiếm và chọn tài khoản cần xóa từ danh sách.	1.1. Hệ thống hiển thị danh sách tài khoản với chức năng tìm kiếm và lọc.
2. Admin chọn chức năng "Xóa tài khoản"	2.1. Hệ thống hiển thị hộp thoại xác nhận xóa tài khoản
3. Admin xác nhận xóa tài khoản	3.1. Hệ thống xóa tài khoản khỏi cơ sở dữ liệu hoặc đánh dấu là không hoạt động.
	3.2. Hệ thống thông báo xóa tài khoản thành công.
Ngoại lệ	Phản ứng hệ thống
1. Tài khoản đang được sử dụng hoặc có dữ liệu liên quan	1.1. Hệ thống thông báo cảnh báo về các dữ liệu liên quan sẽ bị ảnh hưởng.
	1.2. Hệ thống yêu cầu xác nhận lại việc xóa hoặc đề xuất vô hiệu hóa tài khoản thay vì xóa.

Bảng 2.6: Kịch bản usecase con UC#03_03: xóa tài khoản

2.4.8. Kịch bản usecase con UC#03_04: phân quyền tài khoản

Hành động tác nhân	Phản ứng hệ thống
1. Admin tìm kiếm và chọn tài khoản cần phân quyền từ danh sách	1.1. Hệ thống hiển thị danh sách tài khoản với chức năng tìm kiếm và lọc.
2. Admin chọn chức năng "Phân quyền"	2.1. Hệ thống hiển thị giao diện phân quyền với các vai trò hiện có (admin, giảng viên, sinh viên).
3. Admin chọn vai trò cho tài khoản	3.1. Hệ thống hiển thị các quyền tương ứng với vai trò được chọn.
4. Admin xác nhận thay đổi quyền	4.1. Hệ thống cập nhật vai trò và quyền của tài khoản.
	4.2. Hệ thống thông báo phân quyền thành công.
Ngoại lệ	Phản ứng hệ thống
1. Tài khoản đang được sử dụng hoặc có dữ liệu liên quan	1.1. Hệ thống thông báo cảnh báo về các dữ liệu liên quan sẽ bị ảnh hưởng.
	1.2. Hệ thống yêu cầu xác nhận lại việc xóa hoặc đề xuất vô hiệu hóa tài khoản thay vì xóa.

Bảng 2.7: Kịch bản usecase con UC#03_04: phân quyền tài khoản

2.4.9. Kịch bản usecase con UC#03_05: xem lịch sử đăng nhập

1. Admin tìm kiếm và chọn tài khoản cần xem lịch sử từ danh sách	1.1. Hệ thống hiển thị danh sách tài khoản với chức năng tìm kiếm và lọc.
2. Admin chọn chức năng "Xem lịch sử đăng nhập"	2.1. Hệ thống hiển thị danh sách lịch sử đăng nhập của tài khoản được chọn.
	2.2. Thông tin hiển thị gồm: thời gian đăng nhập, trạng thái (thành công/thất bại), thời gian hoạt động.
3. Admin tìm kiếm hoặc lọc lịch sử theo thời gian hoặc trạng thái	3.1. Hệ thống hiển thị kết quả lọc phù hợp với điều kiện tìm kiếm.
4. Admin xuất báo cáo lịch sử đăng nhập	4.1. Hệ thống xuất báo cáo dưới dạng file Excel hoặc PDF.
Ngoại lệ	Phản ứng hệ thống
1. Không có dữ liệu lịch sử	1.1. Hệ thống thông báo "Không có dữ liệu lịch sử đăng nhập cho tài khoản này".

Bảng 2.8: Kịch bản usecase con UC#03_05: xem lịch sử đăng nhập

2.4.10. Kịch bản usecase quản lí lớp học

UC#04	Quản lí lớp học
Mô tả	Người dùng thực hiện các chức năng quản lí lớp học trên hệ thống
Tác nhân	Admin, giảng viên
Tiền điều kiện	Admin hoặc Giảng viên đã đăng nhập vào hệ thống với quyền tương ứng
Hậu điều kiện	Thay đổi thông tin lớp học được cập nhật trong hệ thống

Bảng 2.9: Kịch bản usecase quản lí lớp học

2.4.11. Kịch bản usecase con UC#04_01: thêm lớp học mới

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên chọn chức năng "Quản lý lớp học"	1.1. Hệ thống hiển thị giao diện quản lý lớp học với danh sách các lớp học hiện có.
2. Admin/Giảng viên chọn chức năng "Thêm lớp học mới"	2.1. Hệ thống hiển thị form nhập thông tin lớp học mới.
3. Admin/Giảng viên nhập các thông tin lớp học	3.1. Hệ thống hiển thị form với các trường thông tin cần thiết.
4. Admin/Giảng viên xác nhận tạo môn học	4.1. Hệ thống kiểm tra tính hợp lệ của thông tin.
	4.2. Hệ thống tạo lớp học mới.
	4.3. Hệ thống thông báo tạo môn học thành công.
Ngoại lệ	Phản ứng hệ thống
1. Mã lớp học đã tồn tại	1.1. Hệ thống thông báo lỗi "Mã lớp học này đã tồn tại" và yêu cầu nhập mã khác.
2. Admin/Giảng viên bỏ trống trường bắt buộc	3.1. Hệ thống thông báo lỗi "Vui lòng điền đầy đủ thông tin bắt buộc" và yêu cầu điền đầy đủ.

Bảng 2.10: Kịch bản usecase con UC#03_01: thêm lớp học mới

2.4.12. Kịch bản usecase con UC#04_02: sửa thông tin lớp học

Hành động tác nhân	Phản ứng hệ thống
1. Admin tìm kiếm và chọn lớp học cần chỉnh sửa từ danh sách	1.1. Hệ thống hiển thị danh sách lớp học với chức năng tìm kiếm và lọc.
2. Admin/Giảng viên chọn chức năng "Chỉnh sửa"	2.1. Hệ thống hiển thị form với thông tin hiện tại của lớp học được chọn.
3. Admin/Giảng viên thay đổi thông tin cần chỉnh sửa	3.1. Hệ thống cho phép chỉnh sửa các trường thông tin (trừ mã lớp học)
4. Admin/Giảng viên xác nhận lưu thay đổi	4.1. Hệ thống kiểm tra tính hợp lệ của thông tin.
	4.2. Hệ thống cập nhật thông tin lớp học.
	4.3. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	Phản ứng hệ thống
1. Giảng viên không có quyền chỉnh sửa lớp học	1.1. Hệ thống thông báo lỗi "Không có quyền chỉnh sửa lớp học."
2. Admin/Giảng viên bỏ trống trường bắt buộc	2.1. Hệ thống thông báo lỗi "Vui lòng điền đầy đủ thông tin bắt buộc" và yêu cầu điền đầy đủ.

Bảng 2.11: Kịch bản usecase con UC#04_02: sửa thông tin lớp học

2.4.13. Kịch bản usecase con UC#04_03: xóa lớp học

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên tìm kiếm và chọn lớp học cần xóa từ danh sách.	1.1. Hệ thống hiển thị danh sách lớp học với chức năng tìm kiếm và lọc.
2. Admin/Giảng viên chọn chức năng "Xóa lớp học"	2.1. Hệ thống hiển thị hộp thoại xác nhận xóa lớp học
3. Admin/Giảng viên xác nhận xóa lớp học	3.1. Hệ thống xóa lớp học khỏi cơ sở dữ liệu
	3.2. Hệ thống thông báo xóa lớp học thành công.
Ngoại lệ	Phản ứng hệ thống
1. Giảng viên không có quyền xóa lớp học này	1.1. Hệ thống thông báo lỗi "Bạn không có quyền xóa lớp học này".

Bảng 2.12: Kịch bản usecase con UC#04_03: xóa lớp học

2.4.14. Kịch bản usecase con UC#04_04: thêm sinh viên vào lớp học

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên chọn lớp học cần thêm sinh viên	1.1. Hệ thống hiển thị thông tin chi tiết của lớp học.
2. Admin/Giảng viên chọn chức năng "Thêm sinh viên"	2.1. Hệ thống hiển thị giao diện thêm sinh viên vào lớp học.
3. Admin/Giảng viên tìm kiếm và chọn sinh viên cần thêm	3.1. Hệ thống hiển thị danh sách sinh viên để lựa chọn.
4. Admin/Giảng viên xác nhận thêm sinh viên	4.1. Hệ thống thêm sinh viên vào lớp học.
	4.2. Hệ thống thông báo thêm sinh viên thành công.
Ngoại lệ	Phản ứng hệ thống
1. Sinh viên đã đăng ký lớp học này	1.1. Hệ thống thông báo lỗi "Sinh viên này đã đăng ký lớp học".
2. Môn học đã đạt số lượng sinh viên tối đa	2.1. Hệ thống thông báo lỗi "Lớp học đã đạt số lượng sinh viên tối đa".

Bảng 2.13: Kịch bản usecase con UC#04_04: thêm sinh viên vào lớp học

2.4.15. Kịch bản usecase con UC#04_05: xóa sinh viên khỏi lớp học

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên chọn lớp học cần xóa sinh viên	1.1. Hệ thống hiển thị thông tin chi tiết của lớp học.
2. Admin/Giảng viên tìm kiếm và chọn sinh viên cần xóa	2.1. Hệ thống hiển thị danh sách sinh viên đã đăng ký.
3. Admin/Giảng viên chọn chức năng "Xóa sinh viên"	3.1. Hệ thống hiển thị hộp thoại xác nhận xóa sinh viên khỏi lớp học.
4. Admin/Giảng viên xác nhận xóa sinh viên	4.1. Hệ thống xóa sinh viên khỏi danh sách đăng ký lớp học.
	4.2. Hệ thống thông báo xóa sinh viên thành công.
Ngoại lệ	Phản ứng hệ thống
1. Sinh viên đã có điểm/bài thi trong môn học	1.1. Hệ thống thông báo cảnh báo về dữ liệu điểm/bài thi sẽ bị ảnh hưởng.
	1.2. Hệ thống yêu cầu xác nhận lại việc xóa.

Bảng 2.14: Kịch bản usecase con UC#04_05: xóa sinh viên khỏi lớp học

2.4.16. Kịch bản usecase con UC#04_06:thống kê sinh viên theo lớp học

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên chọn chức năng "Thống kê"	1.1. Hệ thống hiển thị giao diện thống kê sinh viên theo lớp học.
2. Admin/Giảng viên chọn lớp học cần thống kê	2.1. Hệ thống hiển thị các tiêu chí thống kê (theo điểm, theo tình trạng học tập, ...).
3. Admin/Giảng viên chọn tiêu chí thống kê	3.1. Hệ thống hiển thị kết quả thống kê theo tiêu chí được chọn.
	3.2. Hệ thống hiển thị biểu đồ thống kê (nếu có).
4. Admin/Giảng viên xuất báo cáo thống kê (nếu cần)	4.1. Hệ thống xuất báo cáo dưới dạng file Excel hoặc PDF.
Ngoại lệ	Phản ứng hệ thống
1. Không có dữ liệu thống kê	1.1. Hệ thống thông báo "Không có dữ liệu thống kê cho tiêu chí này".

Bảng 2.15: Kịch bản usecase con UC#04_06: thống kê sinh viên theo lớp học

2.4.17. Kịch bản usecase con UC#04_07: tra cứu sinh viên

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên chọn chức năng "Tra cứu sinh viên"	1.1. Hệ thống hiển thị giao diện tra cứu với các tiêu chí tìm kiếm.
2. Admin/Giảng viên nhập thông tin tìm kiếm (mã SV, tên, lớp...)	2.1. Hệ thống hiển thị các trường thông tin để nhập tiêu chí tìm kiếm.
3. Admin/Giảng viên chọn tìm kiếm	3.1. Hệ thống tìm kiếm và hiển thị kết quả phù hợp với tiêu chí.
	3.2. Kết quả bao gồm: thông tin cá nhân, danh sách môn học đã đăng ký, điểm số (nếu có).
4. Admin/Giảng viên chọn xem chi tiết sinh viên	4.1. Hệ thống hiển thị thông tin chi tiết của sinh viên được chọn.
Ngoại lệ	Phản ứng hệ thống
1. Không tìm thấy sinh viên phù hợp	1.1. Hệ thống thông báo "Không tìm thấy sinh viên phù hợp với tiêu chí tìm kiếm".

Bảng 2.16: Kịch bản usecase con UC#04_07: tra cứu sinh viên

2.4.18. Kịch bản usecase quản lý bài thi

UC#05	Quản lý bài thi
Mô tả	Người dùng thực hiện các thao tác quản lý bài thi trong hệ thống.
Tác nhân	Admin, giảng viên
Tiền điều kiện	Admin hoặc Giảng viên đã đăng nhập vào hệ thống với quyền tương ứng
Hậu điều kiện	Thay đổi thông tin bài thi được cập nhật trong hệ thống

Bảng 2.17: Kịch bản usecase quản lý bài thi

2.4.19. Kịch bản usecase con UC#05_01: thêm bài thi mới

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên chọn môn học để tạo bài thi	1.1. Hệ thống hiển thị thông tin chi tiết của môn học.
2. Admin/Giảng viên chọn chức năng "Thêm bài thi mới"	2.1. Hệ thống hiển thị form nhập thông tin bài thi mới.
3. Admin/Giảng viên nhập thông tin bài thi (tên bài thi, thời gian làm bài, thời gian mở/đóng, loại bài thi, điểm tối đa...)	3.1. Hệ thống hiển thị form với các trường thông tin cần thiết.
4. Admin/Giảng viên chọn phương thức tạo câu hỏi (nhập thủ công, nhập từ file)	4.1. Hệ thống hiển thị giao diện tương ứng với phương thức được chọn.
5. Admin/Giảng viên thêm câu hỏi vào bài thi	5.1. Hệ thống cho phép nhập nội dung, đáp án, điểm số cho từng câu hỏi.
6. Admin/Giảng viên xác nhận tạo bài thi	6.1. Hệ thống kiểm tra tính hợp lệ của thông tin.
	6.2. Hệ thống tạo bài thi mới.
	Hệ thống thông báo tạo bài thi thành công.
Ngoại lệ	Phản ứng hệ thống
1. Không có câu hỏi nào	1.1. Hệ thống thông báo lỗi "Bài thi phải có ít nhất một câu

trong bài thi	hỏi" và yêu cầu thêm câu hỏi.
2. Thời gian làm bài không hợp lệ	2.1. Hệ thống thông báo lỗi "Thời gian làm bài không hợp lệ" và yêu cầu nhập lại.
3. Thời gian mở/đóng bài thi không hợp lệ	3.1. Hệ thống thông báo lỗi "Thời gian mở/đóng bài thi không hợp lệ" và yêu cầu nhập lại.

Bảng 2.18: Kịch bản usecase con UC#05_01: thêm bài thi mới

2.4.20. Kịch bản usecase con UC#05_02: sửa thông tin bài thi

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên tìm kiếm và chọn bài thi cần chỉnh sửa	1.1. Hệ thống hiển thị danh sách bài thi với chức năng tìm kiếm và lọc.
2. Admin/Giảng viên chọn chức năng "Chỉnh sửa bài thi"	2.1. Hệ thống hiển thị form với thông tin hiện tại của bài thi được chọn.
3. Admin/Giảng viên thay đổi thông tin cần chỉnh sửa	3.1. Hệ thống cho phép chỉnh sửa các trường thông tin bài thi.
4. Admin/Giảng viên chỉnh sửa câu hỏi (thêm, sửa, xóa câu hỏi)	4.1. Hệ thống hiển thị giao diện chỉnh sửa câu hỏi.
5. Admin/Giảng viên xác nhận lưu thay đổi	5.1. Hệ thống kiểm tra tính hợp lệ của thông tin.
	5.2. Hệ thống cập nhật thông tin bài thi.
	5.3. Hệ thống thông báo cập nhật thành công.
Ngoại lệ	Phản ứng hệ thống
1. Admin/Giảng viên bỏ trống trường bắt buộc	1.1. Hệ thống thông báo lỗi "Vui lòng điền đầy đủ thông tin bắt buộc" và yêu cầu điền đầy đủ.
2. Bài thi đang diễn ra hoặc đã kết thúc	2.1. Hệ thống thông báo cảnh báo "Bài thi đang diễn ra hoặc đã kết thúc, việc chỉnh sửa có thể ảnh hưởng đến dữ liệu hiện tại".

Bảng 2.19: Kịch bản usecase con UC#05_02: sửa thông tin bài thi

2.4.21. Kịch bản usecase con UC#05_03: xóa bài thi

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên tìm kiếm và chọn bài thi cần xóa	1.1. Hệ thống hiển thị danh sách bài thi với chức năng tìm kiếm và lọc.
2. Admin/Giảng viên chọn chức năng "Xóa bài thi"	2.1. Hệ thống hiển thị hộp thoại xác nhận xóa bài thi.
3. Admin/Giảng viên xác nhận xóa bài thi	3.1. Hệ thống xóa bài thi khỏi cơ sở dữ liệu hoặc đánh dấu là không hoạt động.
	3.2. Hệ thống thông báo xóa bài thi thành công.
Ngoại lệ	Phản ứng hệ thống
1. Bài thi đang diễn ra	1.1. Hệ thống thông báo cảnh báo "Bài thi đang diễn ra hoặc đã có sinh viên làm bài, việc xóa sẽ ảnh hưởng đến dữ liệu hiện tại".
	1.2. Hệ thống yêu cầu xác nhận lại việc xóa.

Bảng 2.20: Kịch bản usecase con UC#05_03: xóa bài thi

2.4.22. Kịch bản usecase con UC#05_04: xem chi tiết bài thi

Hành động tác nhân	Phản ứng hệ thống
1. Admin/Giảng viên tìm kiếm và chọn bài thi cần xem	1.1. Hệ thống hiển thị danh sách bài thi với chức năng tìm kiếm và lọc.
2. Admin/Giảng viên chọn chức năng "Xem chi tiết"	2.1. Hệ thống hiển thị thông tin chi tiết của bài thi được chọn.
	2.2. Thông tin bao gồm: tên bài thi, thời gian làm bài, thời gian mở/đóng, loại bài thi, danh sách câu hỏi và đáp án, điểm số, trạng thái bài thi.
3. Admin/Giảng viên xem danh sách sinh viên đã làm bài	3.1. Hệ thống hiển thị danh sách sinh viên đã làm bài thi kèm theo thông tin: thời gian làm bài, điểm số, trạng thái nộp bài.
4. Admin/Giảng viên chọn xem chi tiết bài làm của sinh viên	4.1. Hệ thống hiển thị chi tiết bài làm của sinh viên được chọn, bao gồm: câu trả lời, điểm từng câu, tổng điểm.
5. Admin/Giảng viên xuất báo cáo kết quả bài thi(nếu cần)	5.1. Hệ thống xuất báo cáo dưới dạng file Excel hoặc PDF.
Ngoại lệ	Phản ứng hệ thống
1. Bài thi chưa diễn ra hoặc chưa có sinh viên làm bài	1.1. Hệ thống thông báo "Chưa có sinh viên nào làm bài thi này".

Bảng 2.21: Kịch bản usecase con UC#05_04: xem chi tiết bài thi

2.4.23. Kịch bản usecase làm bài thi

UC#06	Làm bài thi
Mô tả	Người dùng thực hiện các chức năng liên quan đến làm bài thi
Tác nhân	Sinh viên
Tiền điều kiện	<ul style="list-style-type: none"> - Sinh viên đã đăng nhập vào hệ thống. - Bài thi đã được giảng viên viên tạo và mở - Đang trong thời gian cho phép làm bài thi.
Hậu điều kiện	<p>Thành công: Sinh viên hoàn thành bài thi và xem được kết quả.</p> <p>Lỗi: Không thể hoàn thành bài thi.</p>
Hành động tác nhân	Phản ứng hệ thống
1. Sinh viên truy cập chức năng làm bài thi	1.1. Hệ thống hiển thị trang nhập mã bài thi.
2. Sinh viên nhập mã bài thi và xác nhận	2.1. Hệ thống kiểm tra tính hợp lệ của mã bài thi.
	2.2. Hệ thống hiển thị thông tin bài thi (tên bài thi, thời gian làm bài, số lượng câu hỏi).
	2.3. Hệ thống hiển thị nút "Bắt đầu làm bài".

3. Sinh viên chọn "Bắt đầu làm bài"	3.1. Hệ thống bắt đầu tính thời gian và hiển thị các câu hỏi.
	3.2. Hệ thống lưu tạm thời các câu trả lời của sinh viên.
4. Sinh viên làm bài thi	4.1. Hệ thống ghi nhận các câu trả lời được chọn.
5. Sinh viên hoàn thành và nộp bài	5.1. Hệ thống hiển thị thông báo xác nhận nộp bài.
	5.2. Hệ thống chấm điểm và hiển thị kết quả bài thi.
	5.3. Hệ thống lưu kết quả vào cơ sở dữ liệu.
6. Sinh viên chọn xem đáp án (nếu được phép)	6.1. Hệ thống hiển thị danh sách câu hỏi kèm đáp án đúng.
Ngoại lệ	Phản ứng hệ thống
1. Mã bài thi không hợp lệ	1.1. Hệ thống thông báo lỗi "Mã bài thi không hợp lệ" và yêu cầu nhập lại.
2. Mất kết nối internet khi đang làm bài	2.1. Hệ thống lưu tạm thời câu trả lời đã chọn.
	2.2. Khi kết nối được khôi phục, hệ thống đồng bộ và cho phép sinh viên tiếp tục làm bài.
3. Hết thời gian làm bài	4.1. Hệ thống thông báo "Hết thời gian làm bài" và lưu tự động các câu trả lời
	4.2. Hệ thống chuyển sang hiển thị kết quả.
4. Đáp án không được phép xem	4.1. Hệ thống không hiển thị tùy chọn xem đáp án.

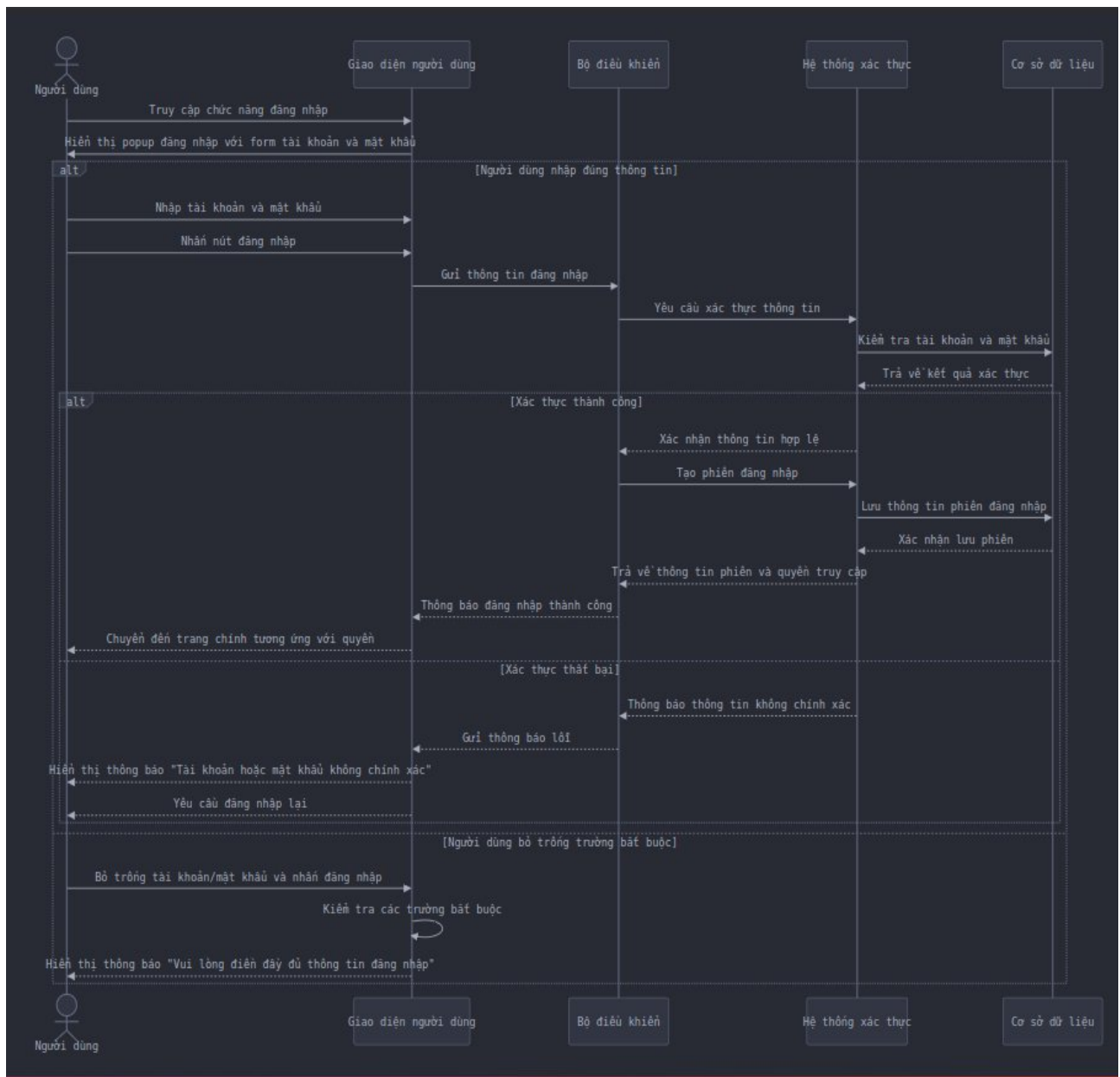
Bảng 2.22: Kịch bản usecase làm bài thi

CHƯƠNG III: THIẾT KẾ CHỨC NĂNG VÀ CƠ SỞ DỮ LIỆU

3. Thiết kế chức năng hệ thống và cơ sở dữ liệu

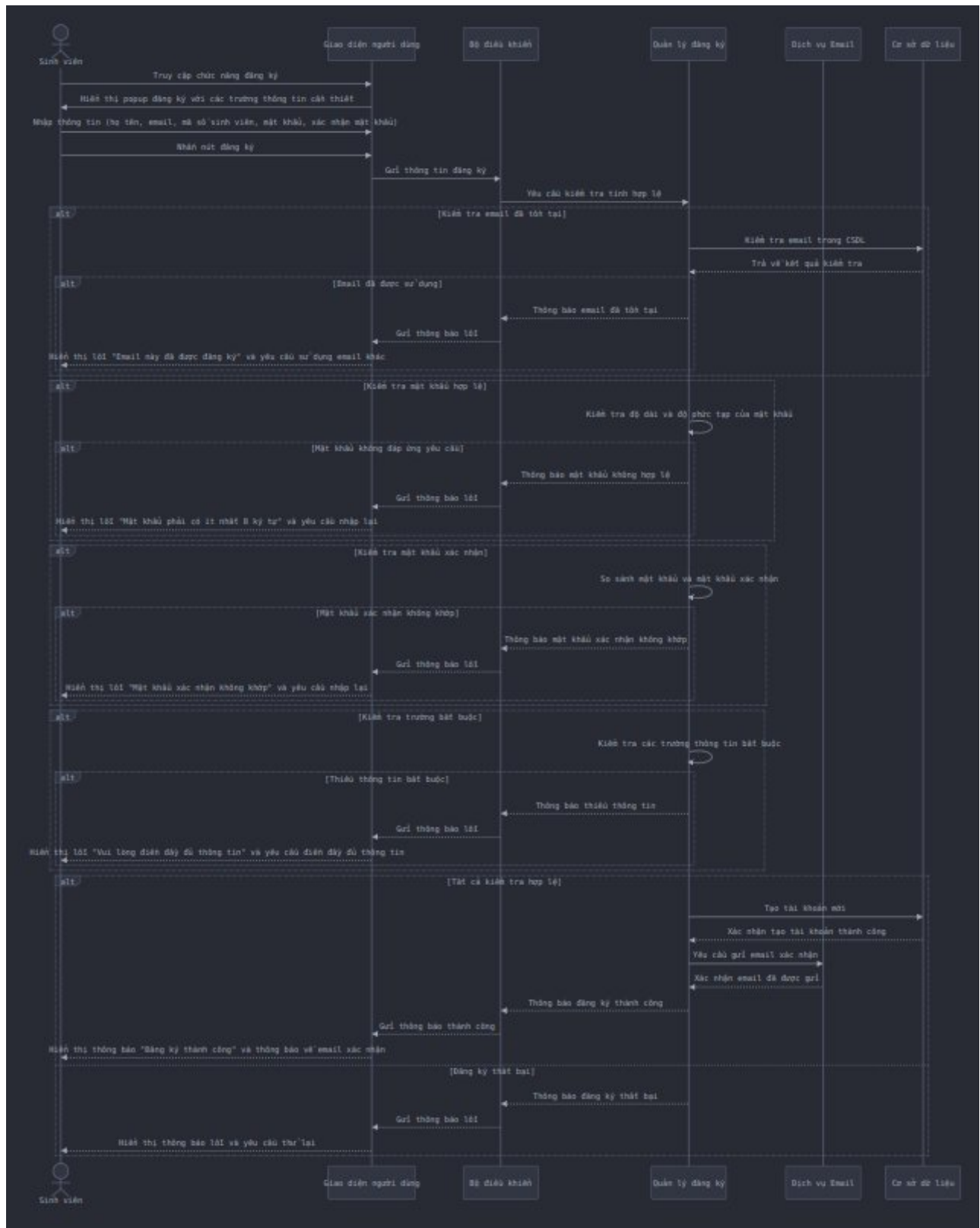
3.1. Sơ đồ tuần tự

3.1.1. Sơ đồ tuần tự chức năng đăng nhập



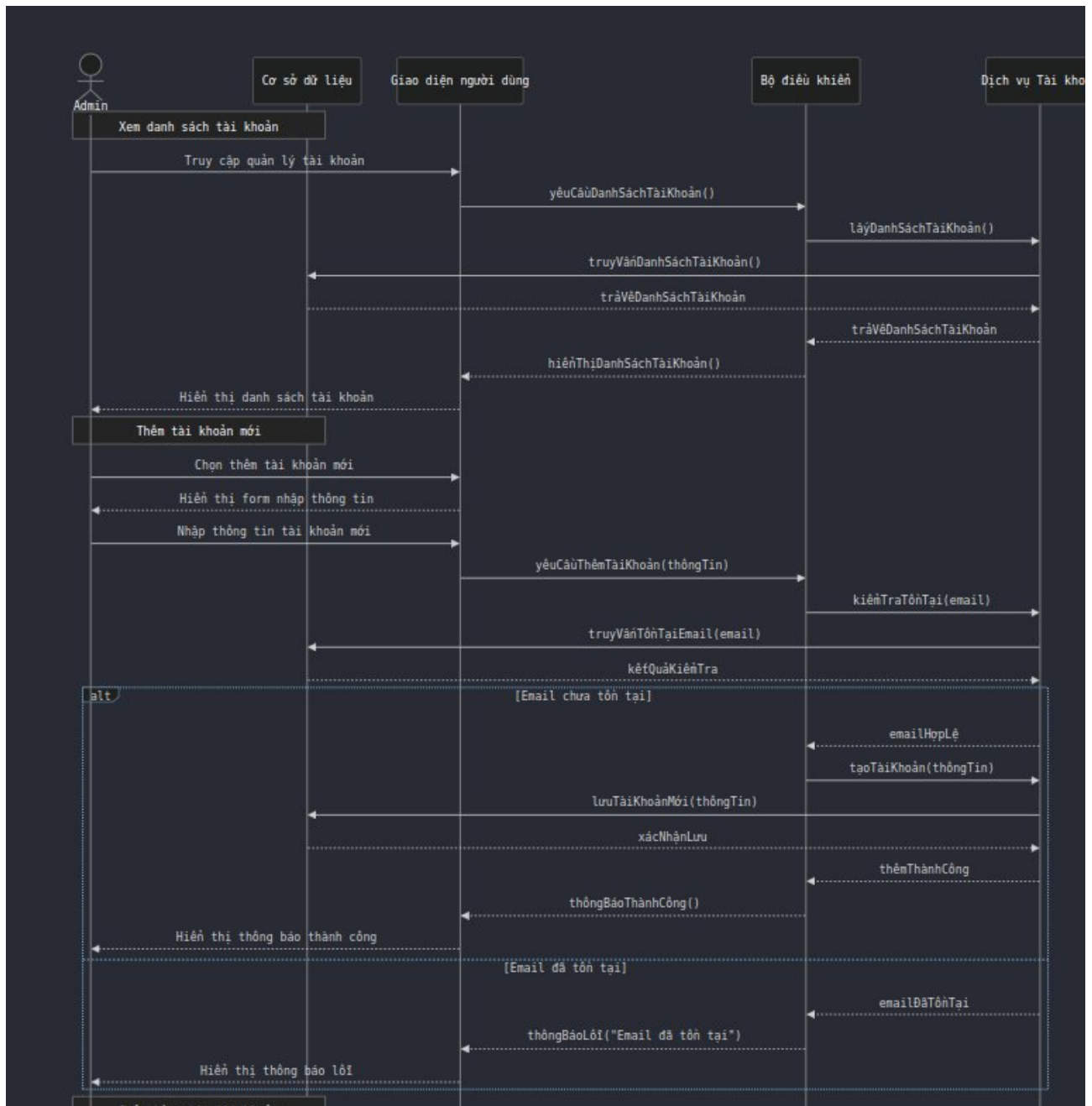
Hình 3.1: Sơ đồ tuần tự chức năng đăng nhập

3.1.2. Sơ đồ tuần tự chức năng đăng kí

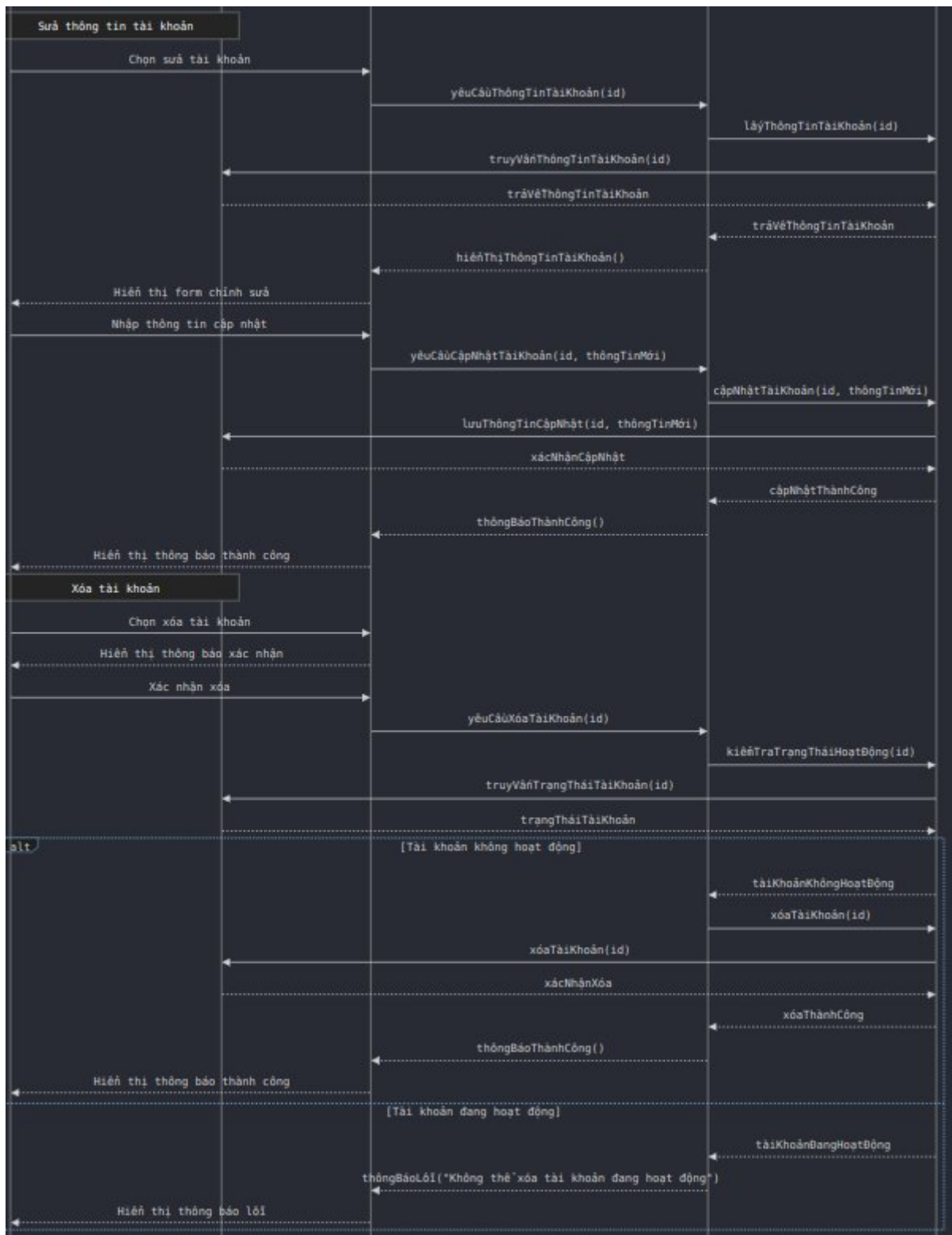


Hình 3.2: Sơ đồ tuần tự chức năng đăng kí

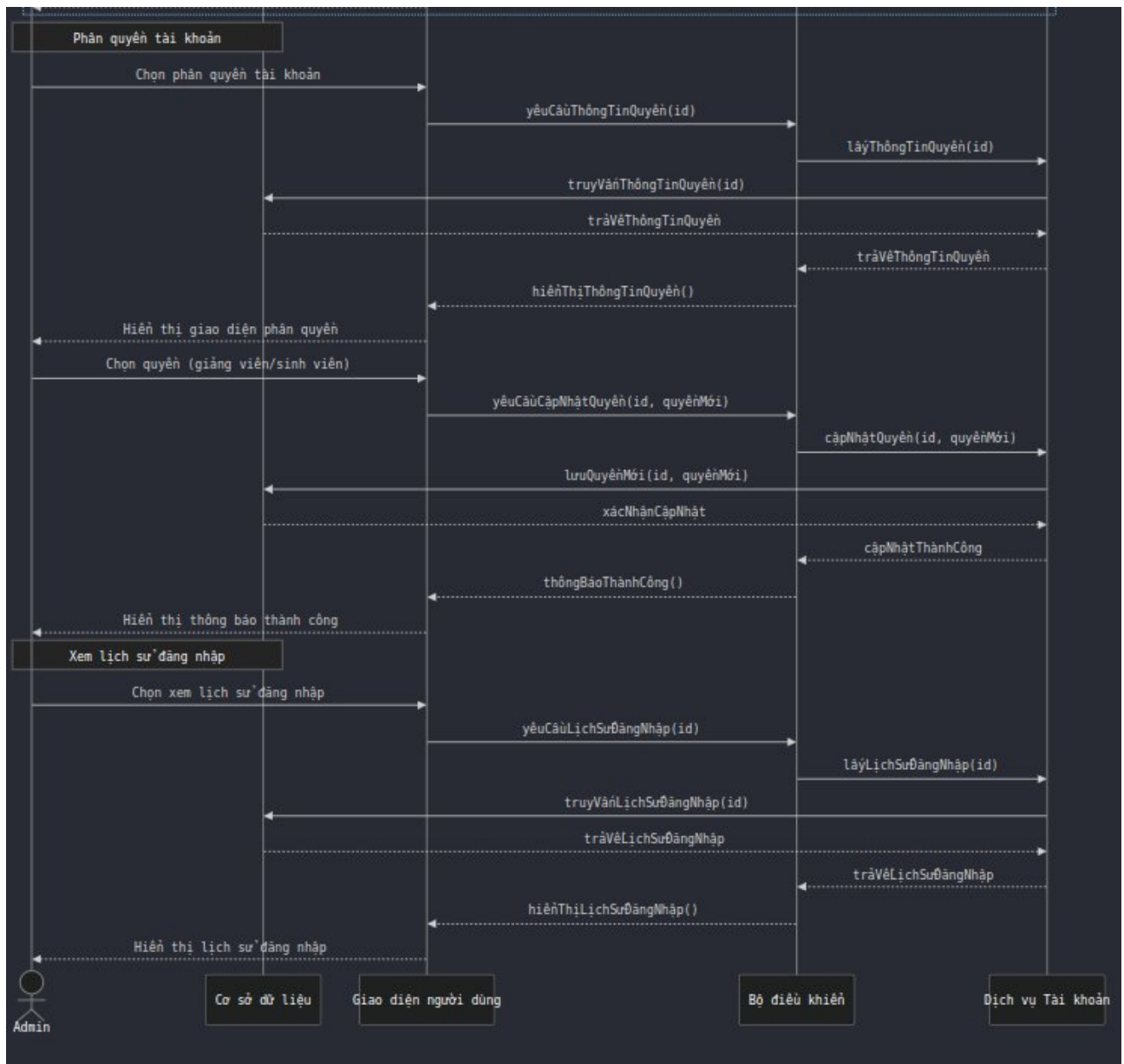
3.1.3. Sơ đồ tuần tự chức năng quản lý tài khoản



Hình 3.3: Sơ đồ tuần tự chức năng quản lý tài khoản (xem tài khoản)

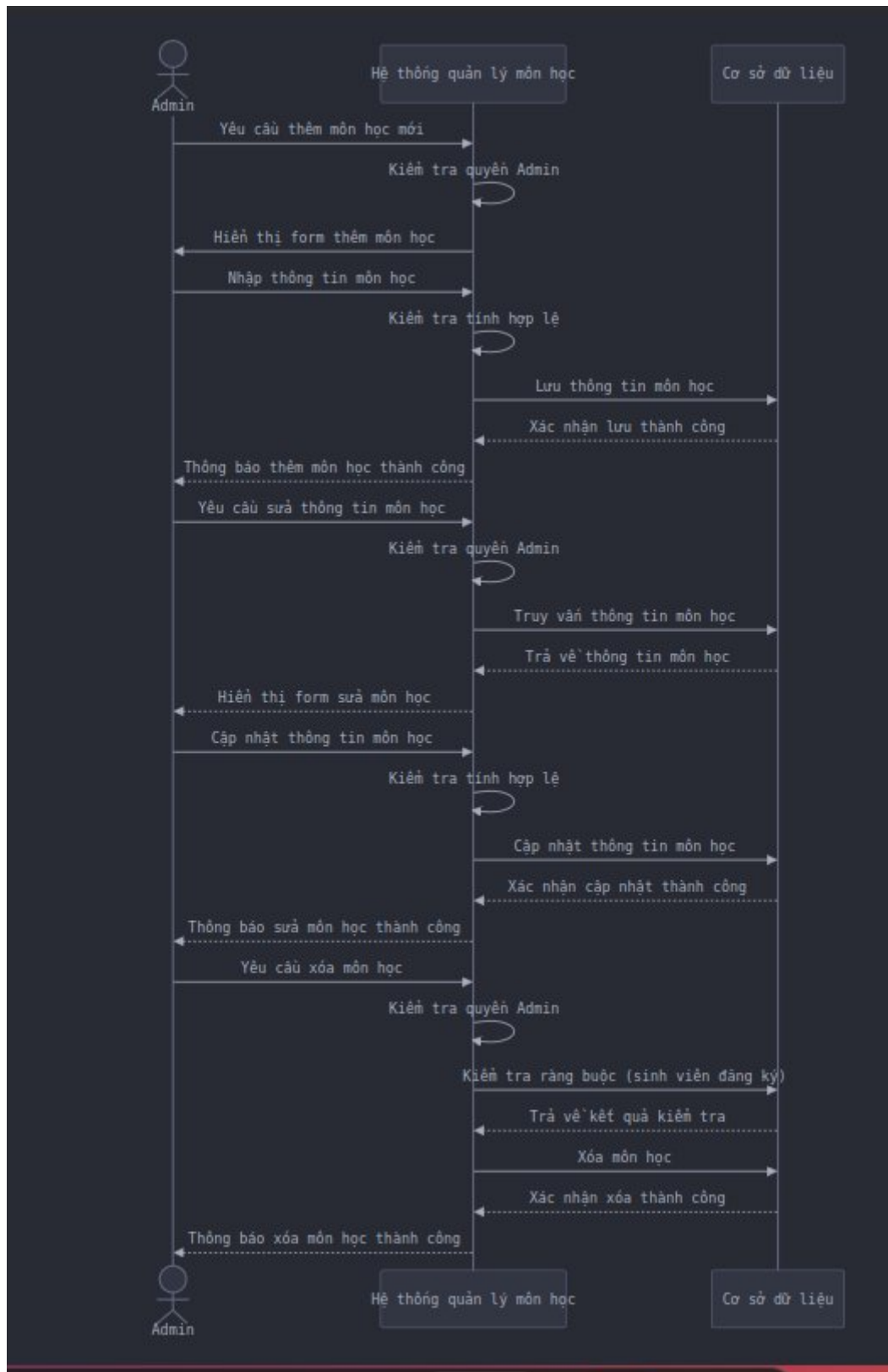


Hình 3.4: Sơ đồ tuần tự chức năng quản lý tài khoản (xóa tài khoản)

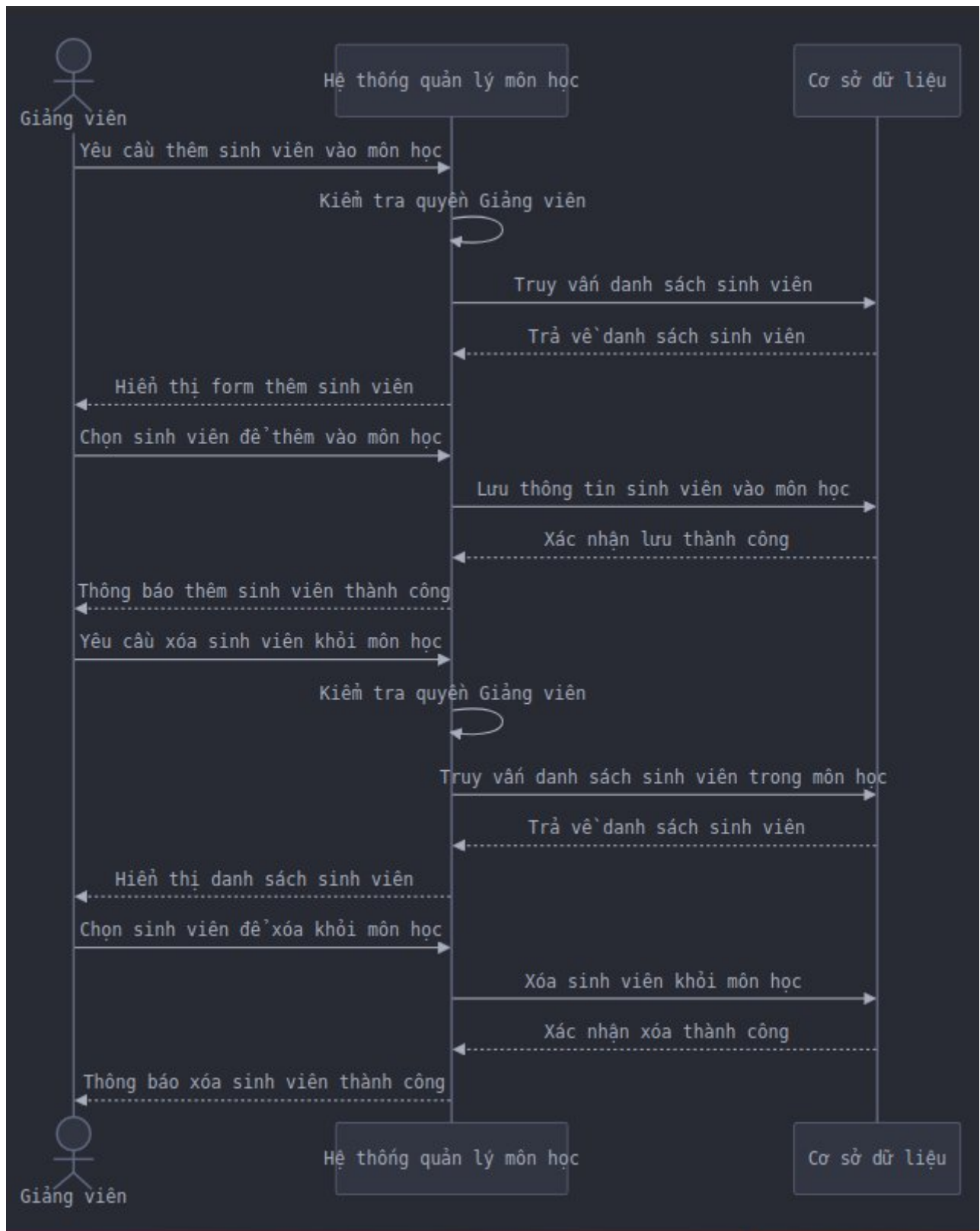


Hình 3.5: Sơ đồ tuần tự chức năng quản lý tài khoản (phân quyền)

3.1.4. Sơ đồ tuần tự quản lý lớp học (môn học)

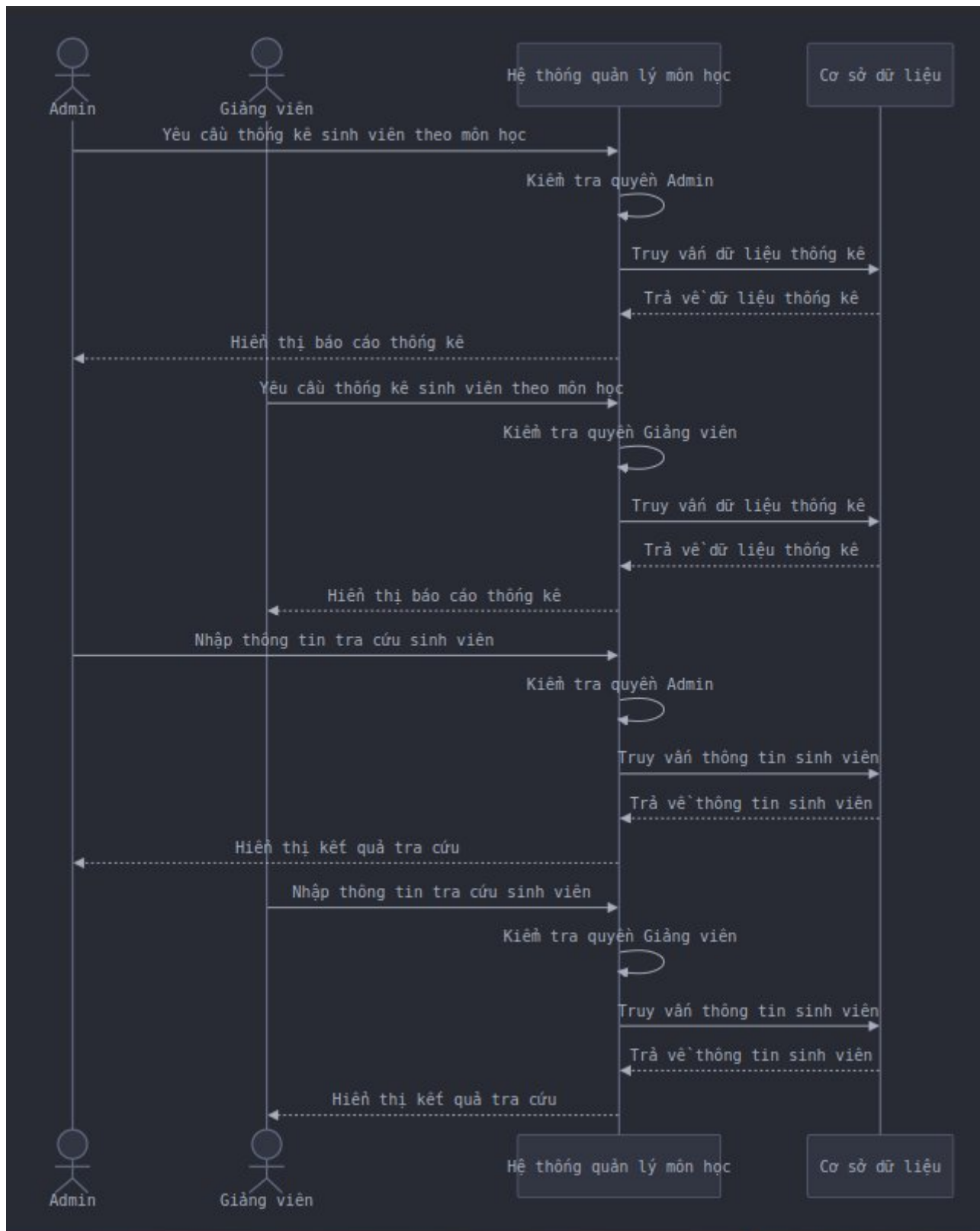


Hình 3.6: Sơ đồ tuần tự quản lý lớp học (quản lý lớp)



Hình 3.7: Sơ đồ tuần tự quản lý lớp học (quản lý sinh viên trong lớp)

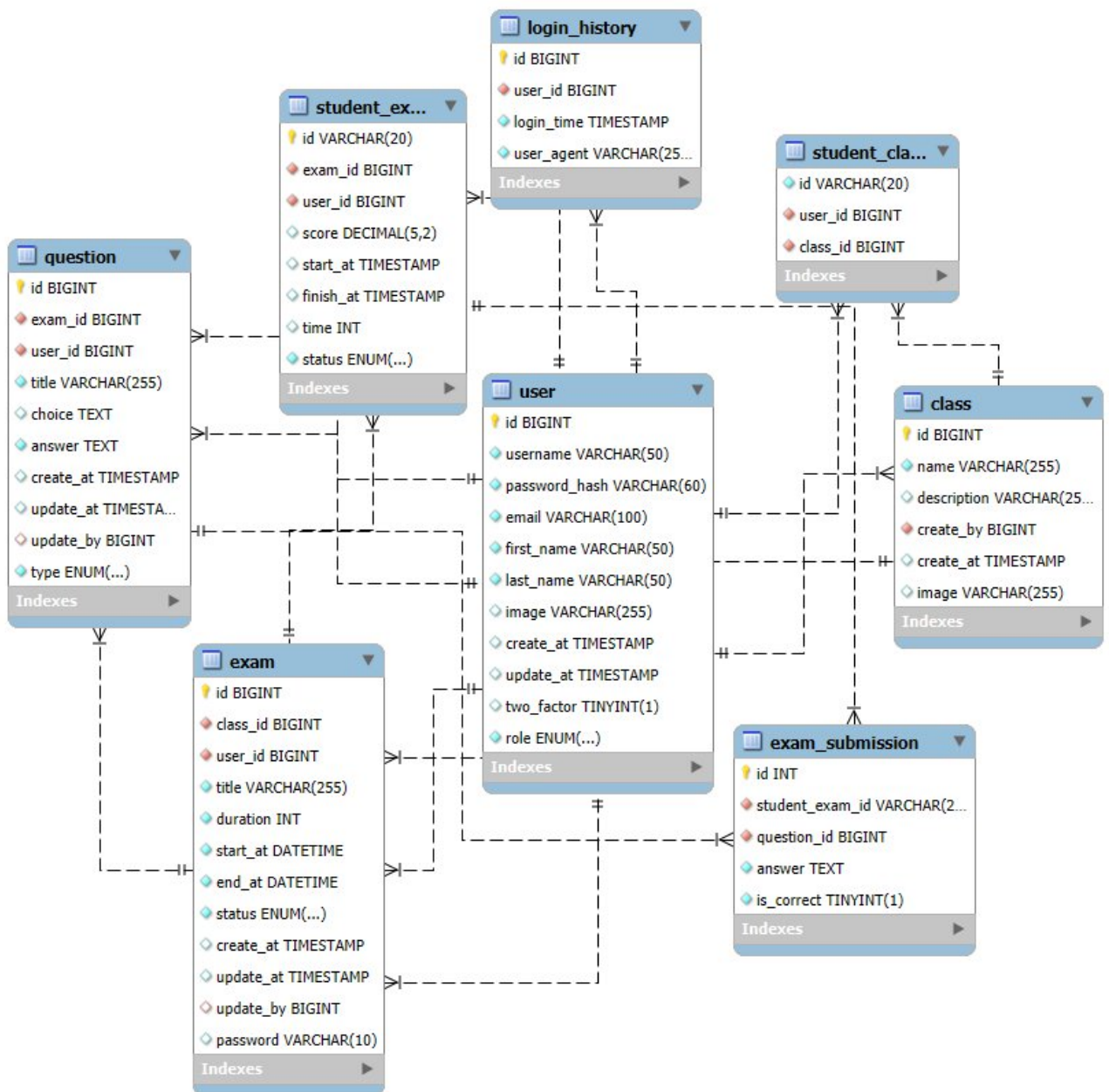
3.1.5. Sơ đồ tuần tự chức năng thống kê



Hình 3.8: Sơ đồ tuần tự chức năng thống kê

3.2. Thiết kế cơ sở dữ liệu

Lược đồ cơ sở dữ liệu



Hình 3.9: Hình thiết kế cơ sở dữ liệu

Danh sách các bảng

STT	Tên bảng	Mô tả
1	user	Bảng lưu thông tin người dùng
2	login_history	Bảng lưu lịch sử đăng nhập
3	class	Bảng lưu thông tin lớp học
4	student_class	Bảng lưu thông tin sinh viên tham gia lớp học
5	exam	Bảng lưu thông tin bài kiểm tra của lớp học
6	student_exam	Bảng lưu thông tin bài kiểm tra của sinh viên
7	question	Bảng lưu câu hỏi của bài kiểm tra
8	exam_submission	Bảng lưu thông tin câu trả lời của sinh viên

Bảng 3.1: Danh sách các bảng trong cơ sở dữ liệu

Chi tiết các bảng

Bảng user

ST T	Trường	Kiểu dữ liệu	Mô tả
1	id	bigint	Id của người dùng
2	username	varchar(50)	Tên đăng nhập
3	pass- word_hash	varchar(60)	Mật khẩu hash
4	email	varchar(100)	Email của người dùng
5	first_name	varchar(50)	Tên người dùng
6	last_name	varchar(50)	Họ người dùng
7	image	varchar(255)	Ảnh đại diện
8	create_at	timestamp	Thời gian tạo tài khoản
9	update_at	timestamp	Thời gian cập nhật thông tin tài khoản
10	two_factor	tinyint(1)	Dấu hiệu nhận biết có bật bảo mật hai lớp không
11	role	enum('ADMIN', 'LECTURER', 'STUDENT')	Chức vụ của người dùng

Bảng 3.2: Bảng User

Bảng login_history

STT	Trường	Kiểu dữ liệu	Mô tả
1	id	bigint	Id của bảng
2	user_id	bigint	Mã người dùng
3	login_time	timestamp	Thời gian đăng nhập
4	user_agent	varchar(255)	Trình duyệt người dùng sử dụng

Bảng 3.3: Bảng login_history**Bảng class**

STT	Trường	Kiểu dữ liệu	Mô tả
1	id	bigint	Id của bảng
2	name	varchar(255)	Tên lớp
3	description	varchar(255)	Mô tả
4	create_by	bigint	Mã người dùng (là người tạo ra lớp)
5	create_at	timestamp	Thời gian tạo
6	image	varchar(255)	Ảnh

Bảng 3.4: Bảng class

Bảng student_class

STT	Trường	Kiểu dữ liệu	Mô tả
1	id	varchar(20)	Id của bảng
2	user_id	bigint	Mã người dùng
3	class_id	bigint	Mã lớp học

Bảng 3.5: Bảng student_class**Bảng exam**

ST T	Trường	Kiểu dữ liệu	Mô tả
1	id	bigint	Id của bảng
2	class_id	bigint	Id của lớp học
3	user_id	bigint	Id của người dùng (người tạo)
4	title	varchar(255)	Tiêu đề bài kiểm tra
5	duration	int	Thời gian làm bài
6	start_at	datetime	Ngày bắt đầu
7	end_at	datetime	Ngày kết thúc
8	status	enum('PENDING', 'ONGOING', 'COMPLETED', 'CANCELED')	Trạng thái
9	create_at	timestamp	Thời gian tạo

10	update_at	timestamp	Thời gian cập nhật
11	update_by	bigint	Mã người dùng đã cập nhật
12	password	varchar(10)	Mật khẩu để làm bài

Bảng 3.6: Bảng exam

Bảng student_exam

ST T	Trường	Kiểu dữ liệu	Mô tả
1	id	varchar(20)	Id của bảng
2	exam_id	bigint	Id bài kiểm tra
3	user_id	bigint	Id của sinh viên
4	score	decimal(5,2)	Điểm
5	start_at	timestamp	Thời gian làm bài
6	finish_at	timestamp	Thời gian kết thúc
7	time	int	Tổng thời gian làm
8	status	enum('NOT_STARTED', 'IN_PROGRESS', 'COMPLETED', CANCELED')	Trạng thái

Bảng 3.7: Bảng student_exam

Bảng question

ST T	Trường	Kiểu dữ liệu	Mô tả
1	id	bigint	Id của bảng
2	exam_id	bigint	Id bài kiểm tra
3	user_id	bigint	Id của người tạo
4	title	varchar(255)	Tiêu đề
5	type	enum('essay', 'multiple_choice', 'single_choice')	Loại câu hỏi
6	choice	text	Đáp án lựa chọn
7	answer	text	Câu trả lời đúng
8	create_at	timestamp	Thời gian tạo
9	update_at	timestamp	Thời gian sửa
10	update_by	bigint	Người sửa

Bảng 3.8: Bảng question

Bảng exam_submission

ST T	Trường	Kiểu dữ liệu	Mô tả
1	id	bigint	Id của bảng
2	stu- dent_exam_id	varchar(20)	Id bảng stu- dent_exam_id
3	question_id	bigint	Id của câu hỏi
4	answer	text	Câu trả lời
5	is_correct	tinyint(1)	Đúng ko?

CHƯƠNG IV: KIỂM THỬ VÀ TRIỂN KHAI HỆ THỐNG

4. Kiểm thử, triển khai và thực nghiệm

4.1. Kiểm thử [10]

4.1.1. Phân tích các trường hợp kiểm thử

STT	Chức năng	Mục tiêu
1	Đăng nhập	<ul style="list-style-type: none">- Kiểm tra giao diện chung màn hình, hiển thị đầy đủ và chính xác các button, textbox tương ứng.- Kiểm tra tính hợp lệ của các giá trị đầu vào và hiển thị các thông báo lỗi trên form đăng nhập như: username, email, mật khẩu.- Kiểm tra chức năng đăng nhập vào hệ thống
2	Đăng kí	<ul style="list-style-type: none">- Kiểm tra giao diện chung màn hình, hiển thị đầy đủ và chính xác các button, textbox tương ứng.- Kiểm tra tính hợp lệ của các giá trị đầu vào và hiển thị các thông báo lỗi trên form đăng kí như:<ul style="list-style-type: none">+ Tên+ Email+ Username+ Password+ Role- Kiểm tra chức năng đăng kí vào hệ thống
3	Cập nhập tài khoản	<ul style="list-style-type: none">- Kiểm tra giao diện chung màn hình, hiển thị đầy đủ và chính xác các button, textbox tương ứng.

		<ul style="list-style-type: none"> - Kiểm tra tính hợp lệ của các giá trị đầu vào và hiển thị các thông báo lỗi trên form cập nhập tài khoản như: <ul style="list-style-type: none"> + First name + Last name + Password - Kiểm tra thao tác hiển thị, cập nhập và lưu thông tin tài khoản.
4	Quản lí người dùng	<ul style="list-style-type: none"> - Kiểm tra giao diện hiển thị danh sách người dùng. - Kiểm tra chức năng xem thông tin chi tiết tài khoản: <ul style="list-style-type: none"> + Thông tin cá nhân (tên, email) + Quyền truy cập - Kiểm tra chức năng xem lịch sử đăng nhập (thời gian, trạng thái phiên) - Kiểm tra chức năng cập nhật thông tin người dùng
5	Quản lí lớp học	<ul style="list-style-type: none"> - Kiểm tra giao diện danh sách lớp học. - Kiểm tra chức năng thêm lớp học mới: <ul style="list-style-type: none"> Tên lớp học. Mô tả lớp học. Giảng viên quản lí lớp.
 - Kiểm tra chức năng sửa/xóa thông tin lớp học.

		<ul style="list-style-type: none"> - Kiểm tra chức năng thêm sinh viên vào lớp học (bằng file, thủ công). - Kiểm tra chức năng xem danh sách sinh viên trong lớp. - Kiểm tra chức năng xóa sinh viên khỏi lớp.
6	Quản lí bài thi	<ul style="list-style-type: none"> - Kiểm tra giao diện danh sách bài thi trong lớp. - Kiểm tra chức năng tạo bài thi mới: <ul style="list-style-type: none"> + Tên bài thi. + Thời gian làm bài. + Số câu hỏi (thêm thủ công/bằng file) + Thời gian bắt đầu/kết thúc. + Mật khẩu bài thi. - Kiểm tra chức năng sửa/xóa thông tin bài thi. - Kiểm tra chức năng xem lịch sử làm bài của sinh viên (thời gian bắt đầu/kết thúc, điểm số, trạng thái hoàn thành). - Kiểm tra chức năng giám sát chống gian lận: <ul style="list-style-type: none"> + Phát hiện chuyển tab. + Ghi nhận hành vi bất thường.
7	Làm bài thi	<ul style="list-style-type: none"> - Kiểm tra mật khẩu trước khi làm bài thi. - Kiểm tra giao diện làm bài thi. - Kiểm tra chức năng hiển thị câu hỏi: <ul style="list-style-type: none"> + Nội dung câu hỏi.

		<ul style="list-style-type: none"> + Các lựa chọn đáp án. + Hình ảnh. - Kiểm tra bộ đếm thời gian làm bài. - Kiểm tra chức năng điều hướng giữa các câu hỏi. - Kiểm tra cảnh báo khi chuyển tab/cửa sổ. - Kiểm tra chức năng tự động lưu tiến độ.
8	Nộp bài thi	<ul style="list-style-type: none"> - Kiểm tra giao diện xác nhận nộp bài. - Kiểm tra chức năng nộp bài thủ công. - Kiểm tra chức năng tự động nộp bài khi hết thời gian. - Kiểm tra xác nhận hoàn thành bài thi. - Kiểm tra tính toán điểm số tự động.
9	Thống kê điểm thi	<ul style="list-style-type: none"> - Kiểm tra giao diện bảng thống kê điểm. - Kiểm tra hiển thị danh sách điểm theo lớp học: <ul style="list-style-type: none"> + Tên sinh viên. + Điểm số từng bài thi. + Điểm trung bình. - Kiểm tra chức năng lọc và sắp xếp kết quả. - Kiểm tra biểu đồ thống kê: <ul style="list-style-type: none"> + Điểm trung bình. + Điểm cao nhất/thấp nhất.
10	Xem kết quả điểm	<ul style="list-style-type: none"> - Kiểm tra giao diện xem điểm của sinh viên.

	thi (sinh viên)	<ul style="list-style-type: none"> - Kiểm tra hiển thị danh sách bài thi đã làm <ul style="list-style-type: none"> + Tên bài thi. + Điểm số đạt được. + Trạng thái (đạt/không đạt). - Kiểm tra quyền truy cập (chỉ xem điểm của bản thân).
--	-----------------	--

Bảng 4.1: Bảng phân tích các trường hợp kiểm thử

4.1.2. Danh sách các testcase và kết quả thực thi

Phụ lục 1

4.2. Yêu cầu về cài đặt

Yêu cầu phần mềm:

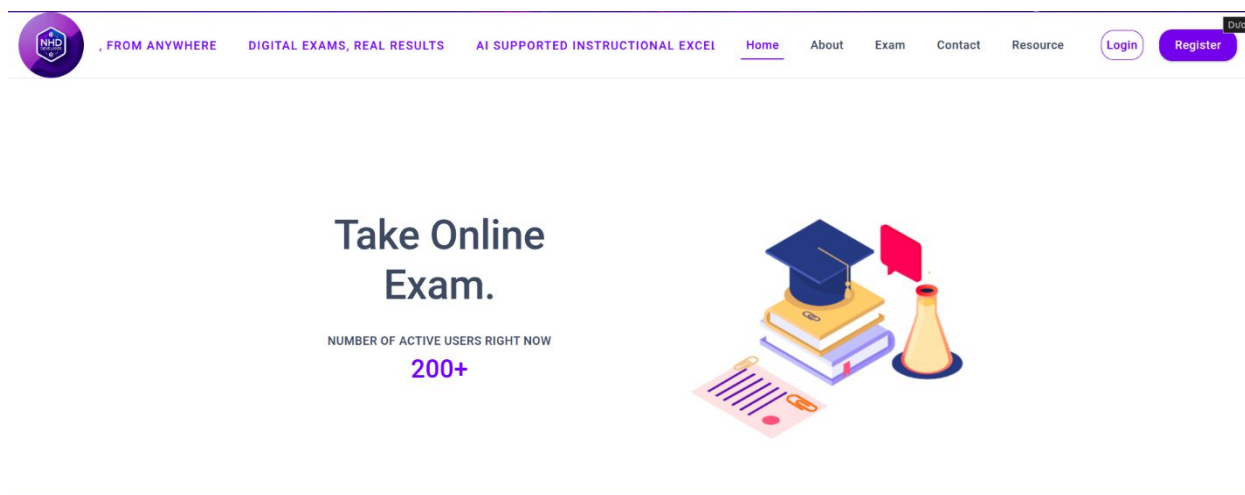
- Hệ điều hành: có thể sử dụng nhiều hệ điều hành từ Window (window 10 trở lên), macOS hoặc đề xuất ưu tiên sử dụng Ubuntu.
- IntelliJ IDEA: IntelliJ IDEA là một môi trường phát triển tích hợp (IDE) được phát triển bởi JetBrains. Nó hỗ trợ nhiều ngôn ngữ lập trình như Java, Scala, Kotlin và nhiều ngôn ngữ khác, cung cấp tính năng phân tích mã thông minh, công cụ quản lý dự án, và tính năng hỗ trợ cho việc phát triển web và ứng dụng doanh nghiệp. IDE này đặc biệt được ưa chuộng trong các nhà phát triển Java vì khả năng hỗ trợ mã nguồn sâu, tự động hoàn thiện mã và phân tích mã nguồn mạnh mẽ.
- Visual Studio Code (VSC): VSC là một IDE phát triển mạnh mẽ tích hợp nhiều extension hữu dụng, phù hợp cho việc triển khai giao diện người dùng bằng ReactJS.

Yêu cầu phần cứng: Có laptop cá nhân hoặc máy tính để bàn với bộ chip xử lý, SSD hoặc HDD, main, card đủ khỏe để chạy và phát triển ứng dụng.

Yêu cầu Internet: cần có đường trường mạng ổn định và khỏe, ưu tiên sử dụng wifi hoặc cắm cap trực tiếp.

4.3. Kết quả thực nghiệm

Giao diện Landing Page



Hình 4.1: Giao diện Landing Page

Giao diện đăng kí

The screenshot shows a registration form titled "Lets Register Account" overlaid on a website background. The background features a navigation bar with links like Home, About, Exam, Contact, and Resource, along with "Login" and "Get started" buttons. The main content area has the text "Take Only Exam." and "NUMBER OF ACTIVE USERS R 200+". The registration form includes the following fields: Name, last name, Email, Username, Password (with an eye icon for visibility), and a Student dropdown menu. A purple "Create Account" button is at the bottom of the form, followed by a link "Already have an account? Login". The background also features an illustration of a person wearing a red headscarf working on a laptop.

Hình 4.2: Giao diện chức năng đăng kí (không phải admin)

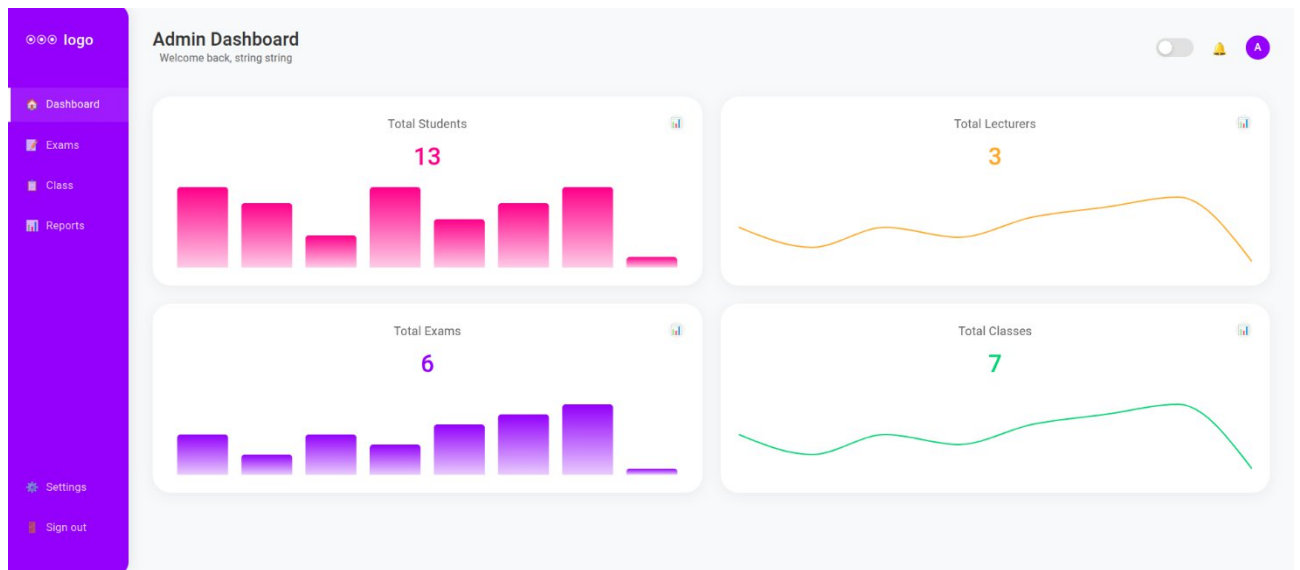
Giao diện đăng nhập hệ thống

The screenshot shows a login form titled "Lets Sign you in" overlaid on the same website background as the registration form. The login form includes the following fields: "Email or Username" and "Password" (with an eye icon for visibility). A link "Forgot Password?" is located below the password field. A purple "Sign In" button is at the bottom of the form. Below the button, there is an "or" separator and three social media login icons: Google, Facebook, and Apple. At the very bottom, there is a link "Don't have an account? Register Now". The background elements, including the navigation bar, main content area, and illustration, are identical to the registration form screenshot.

Hình 4.4: Giao diện chức năng đăng nhập

Giao diện trang Dashboard phân quyền với từng user

Admin dashboard:



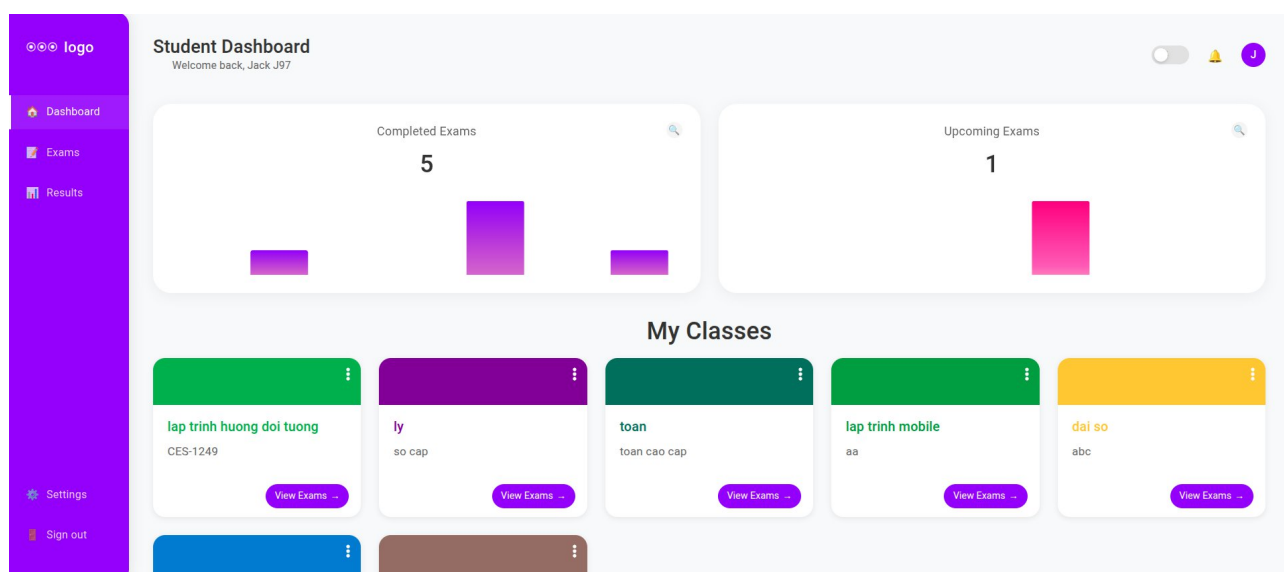
Hình 4.5: Giao diện Admin Dashboard

Lecture dashboard:



Hình 4.6: Giao diện Lecture Dashboard

Student dashboard:



Hình 4.7: Giao diện Student Dashboard

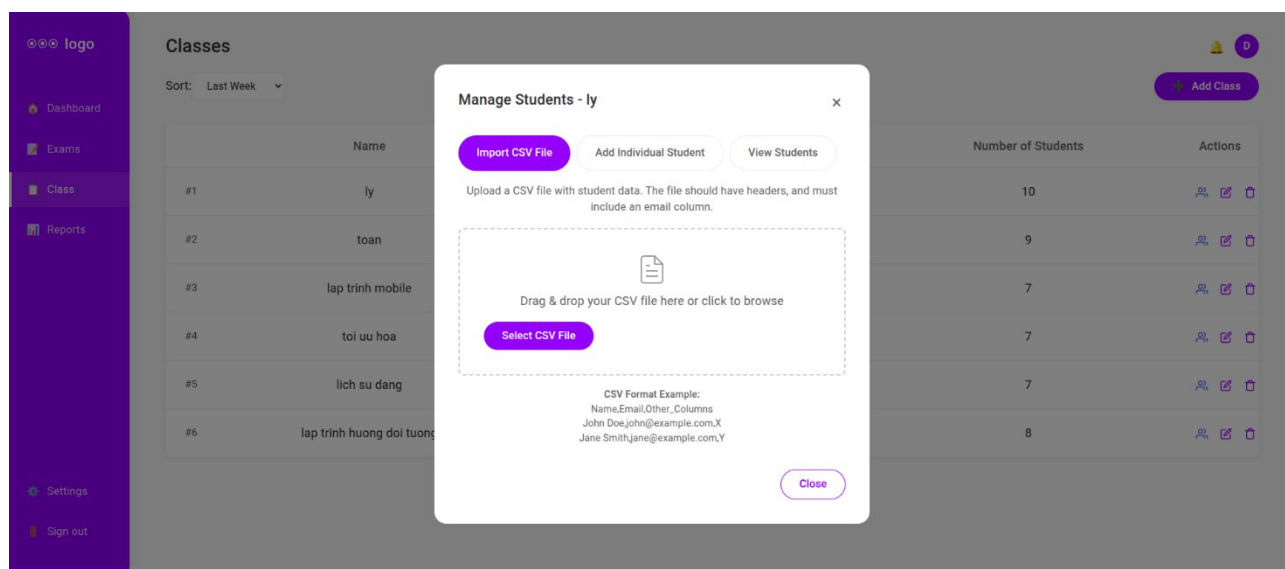
Giao diện hiển thị lớp học của admin và lecture

The Classes management interface features a purple sidebar on the left with navigation links: logo, Dashboard, Exams, Class, Reports, Settings, and Sign out. The main content area is titled "Classes" and includes a "Sort: Last Week" dropdown and an "Add Class" button. It displays a table with the following data:

	Name	Description	Number of Students	Actions
#1	ly	so cap	10	View Edit Delete
#2	toan	toan cao cap	9	View Edit Delete
#3	lap trinh mobile	aa	7	View Edit Delete
#4	dai so	abc	8	View Edit Delete
#5	toi uu hoa	cacas	7	View Edit Delete
#6	lich su dang	aacs	7	View Edit Delete
#7	lap trinh huong doi tuong	CES-1249	8	View Edit Delete

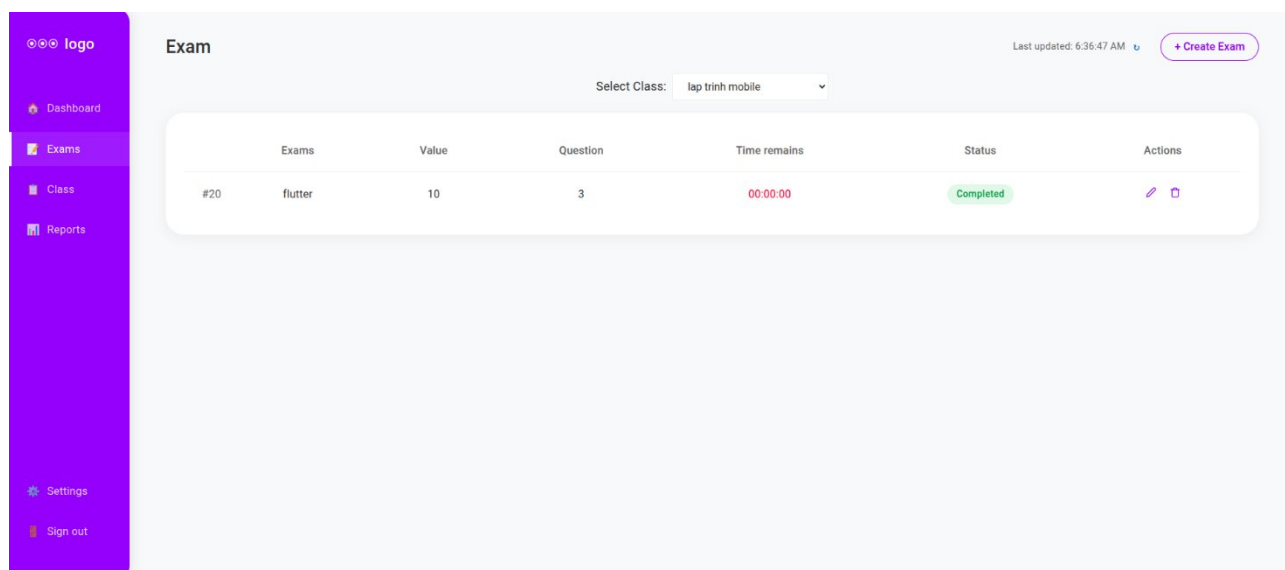
Hình 4.8: Giao diện quản lý lớp học của admin và lecture

Giao diện chức năng quản lý sinh viên trong lớp học



Hình 4.9: Giao diện chức năng quản lý sinh viên trong lớp học

Giao diện quản lý bài thi



Hình 4.10: Giao diện quản lý bài thi

Giao diện chức năng tạo bài thi

Create New Exam

Exam Title
Enter exam title

Class
Select a class

Duration (minutes)
60

Start Date & Time
mm/dd/yyyy, --:--

End Date & Time
mm/dd/yyyy, --:--

Status
Scheduled

Note: Statuses will automatically update based on time:
• Before start time: Scheduled
• Between start and end time: Ongoing
• After end time: Completed

Exam Password
Enter exam password

Students will need to enter this password to access the exam.

Hình 4.11: Giao diện chức năng tạo bài thi

Giao diện chức năng quản lý câu hỏi trong bài thi

Add Question | Import Questions | View Questions (6)

Question Text
Enter your question

Question Image (optional)
Upload Image

Question Type
Single Choice

Answer Choices

A. Option A

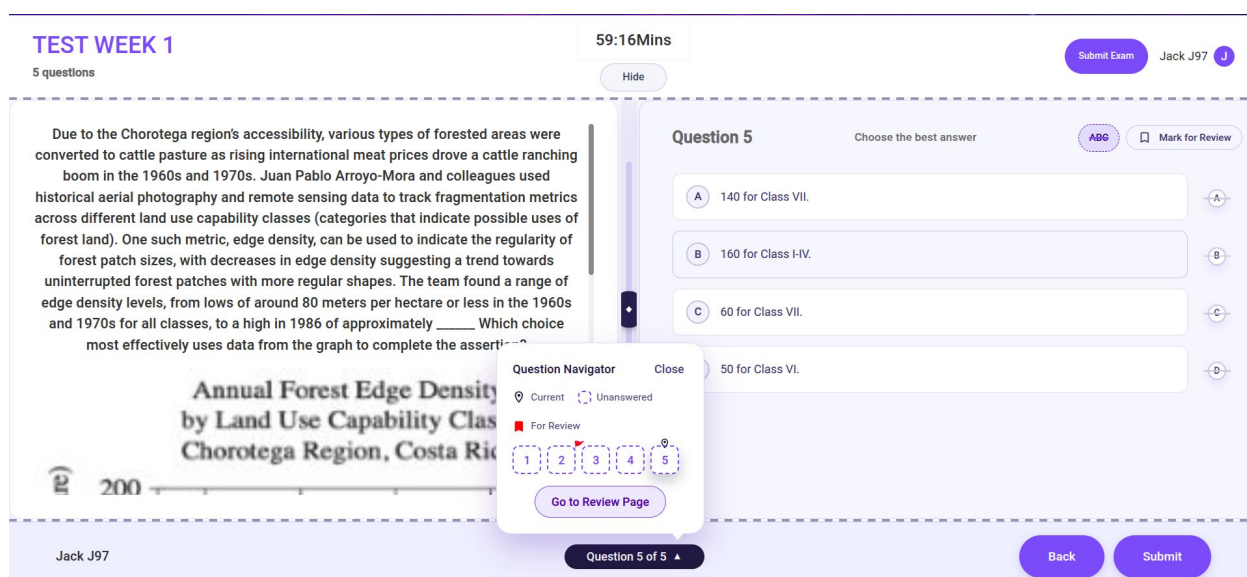
B. Option B

C. Option C

D. Option D

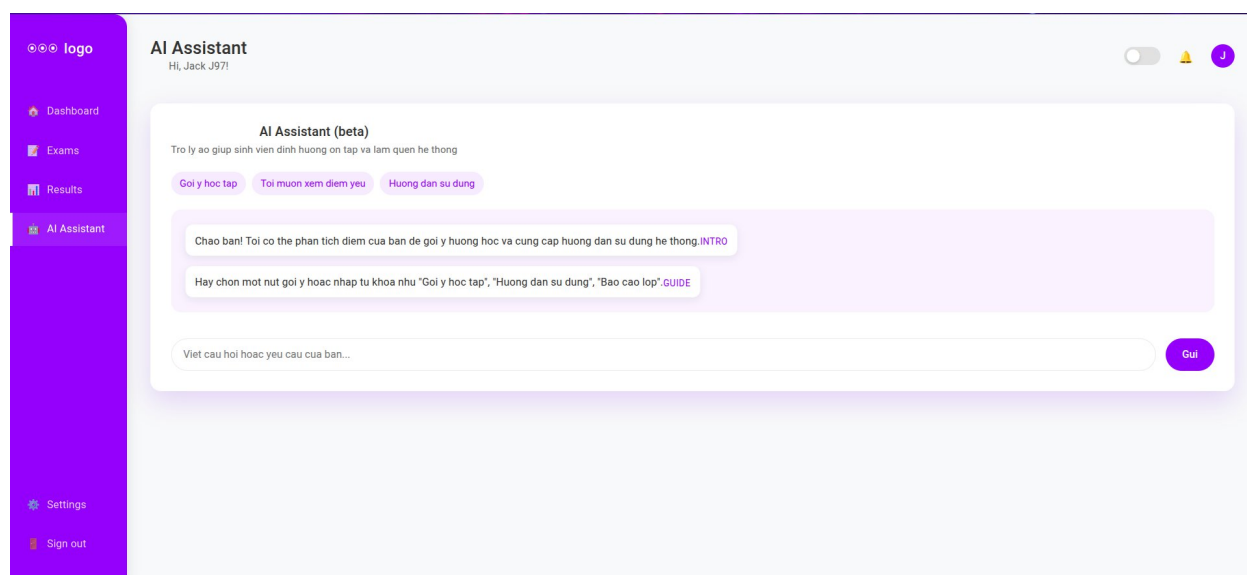
Hình 4.12: Giao diện chức năng quản lý câu hỏi trong bài thi

Giao diện làm bài thi



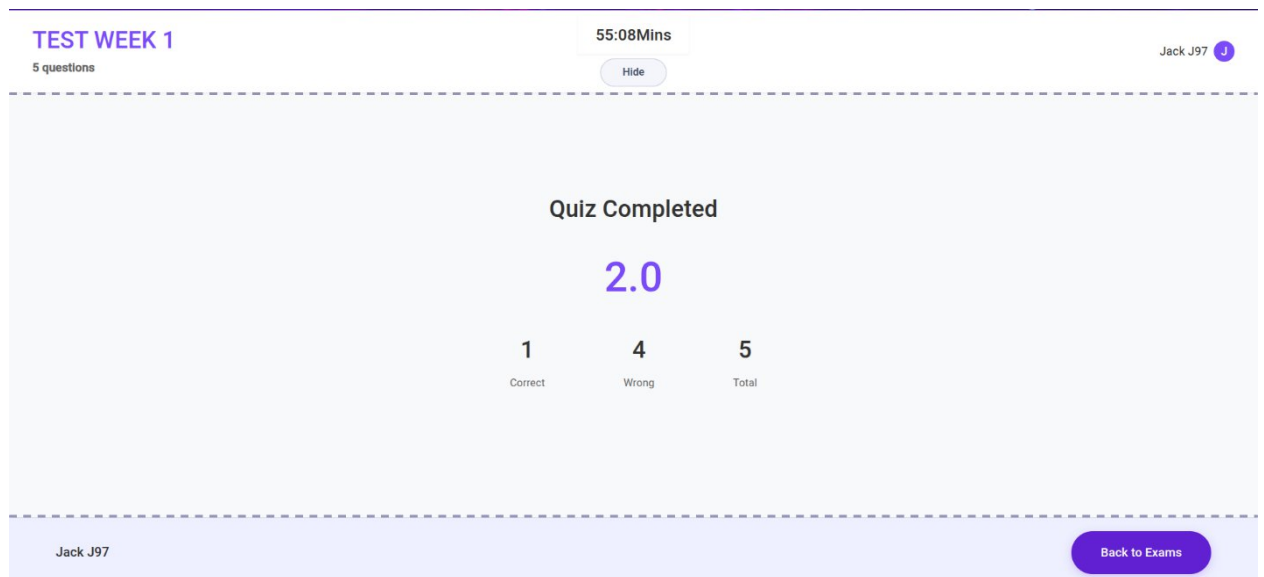
Hình 4.13: Giao diện làm bài thi

Giao diện hệ thống AI assistant



Hình 4.14: Giao diện hệ thống trợ lý học tập AI

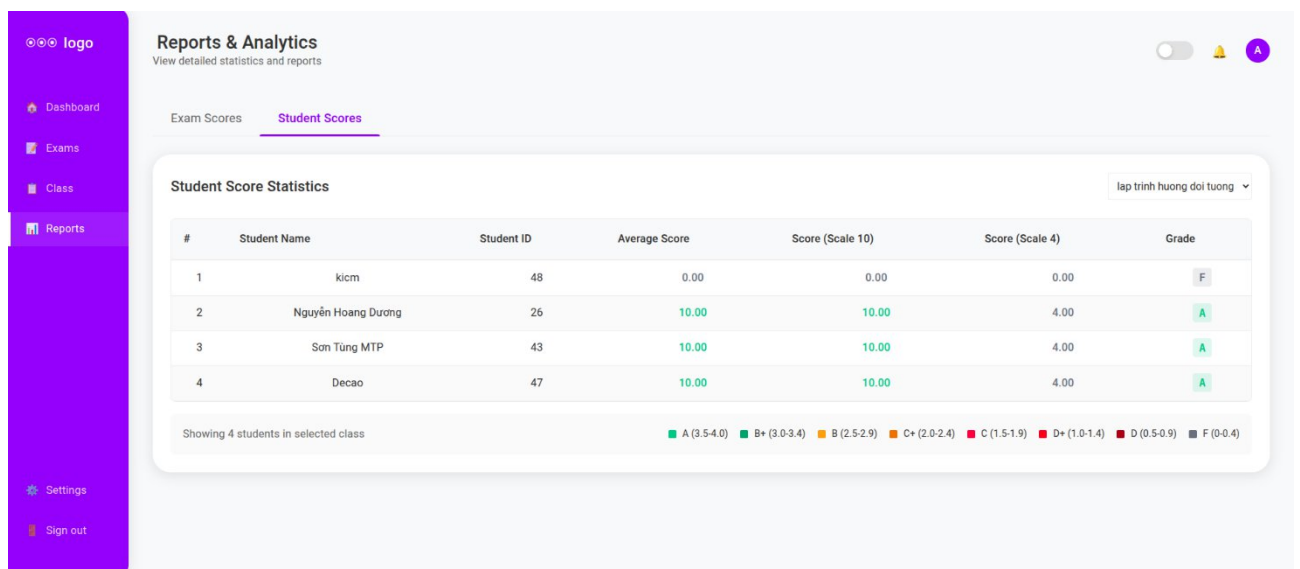
Giao diện hiển thị kết quả bài thi



Hình 4.15: Giao diện hiển thị kết quả bài thi

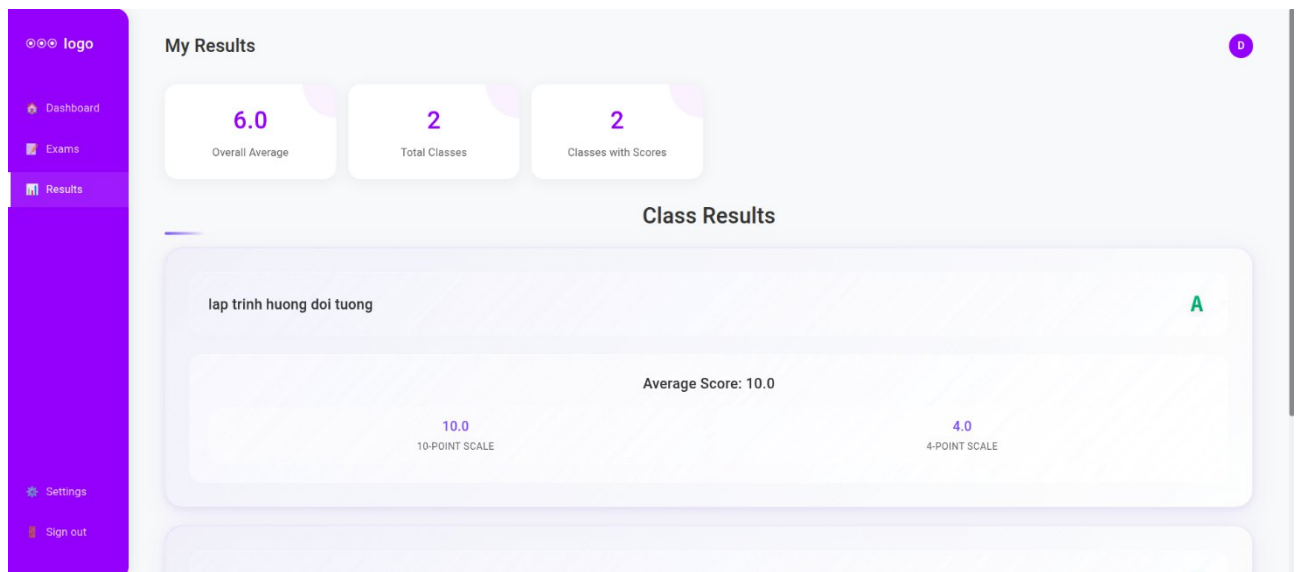
Giao diện chức năng thống kê, hiển thị kết quả

Thống kê kết quả student đối với admin và lecture



Hình 4.16: Giao diện chức năng thống kê điểm của admin và lecture

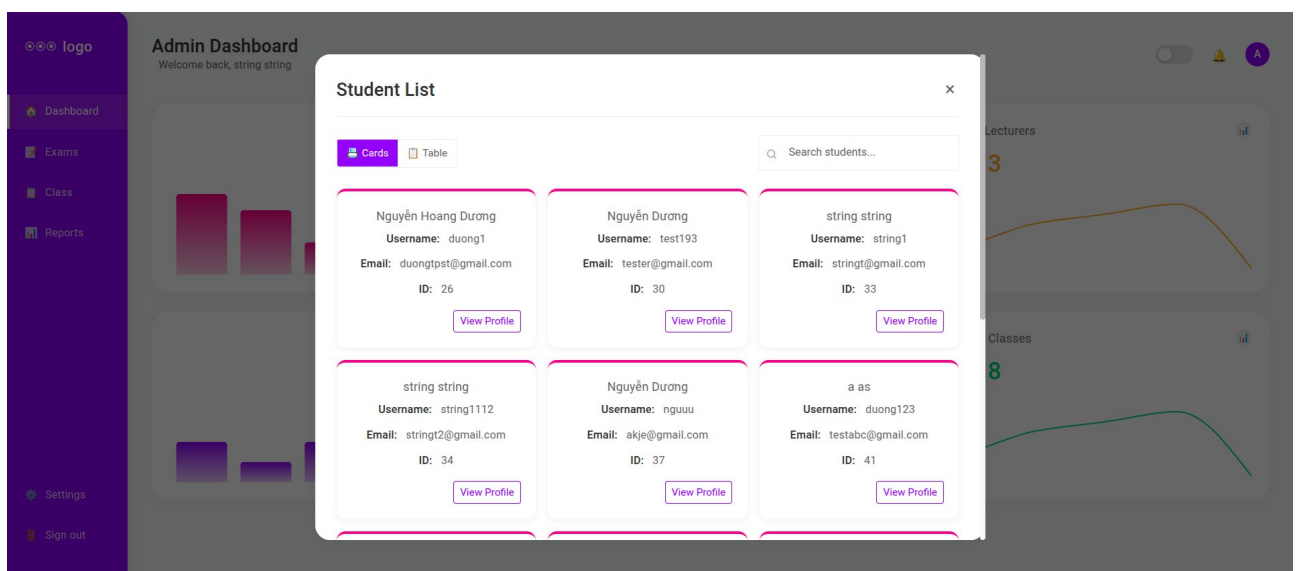
Thống kê kết quả đối với student:



Hình 4.17: Giao diện chức năng thống kê điểm với student

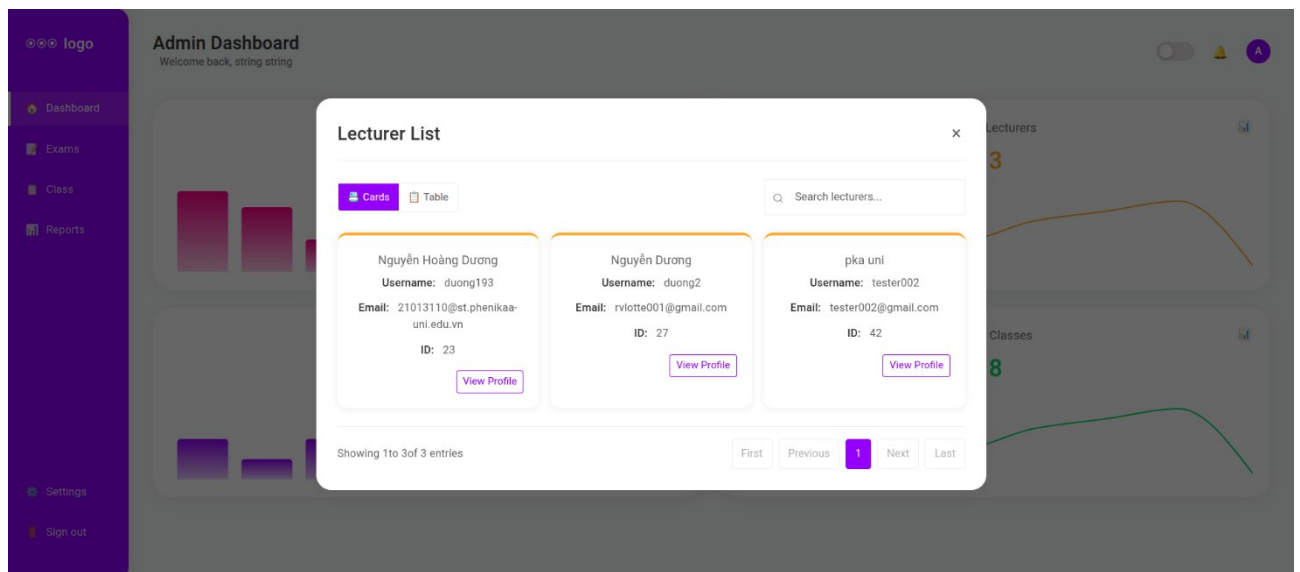
Giao diện hiển thị thống kê người dùng, lớp, bài thi

Thống kê sinh viên:



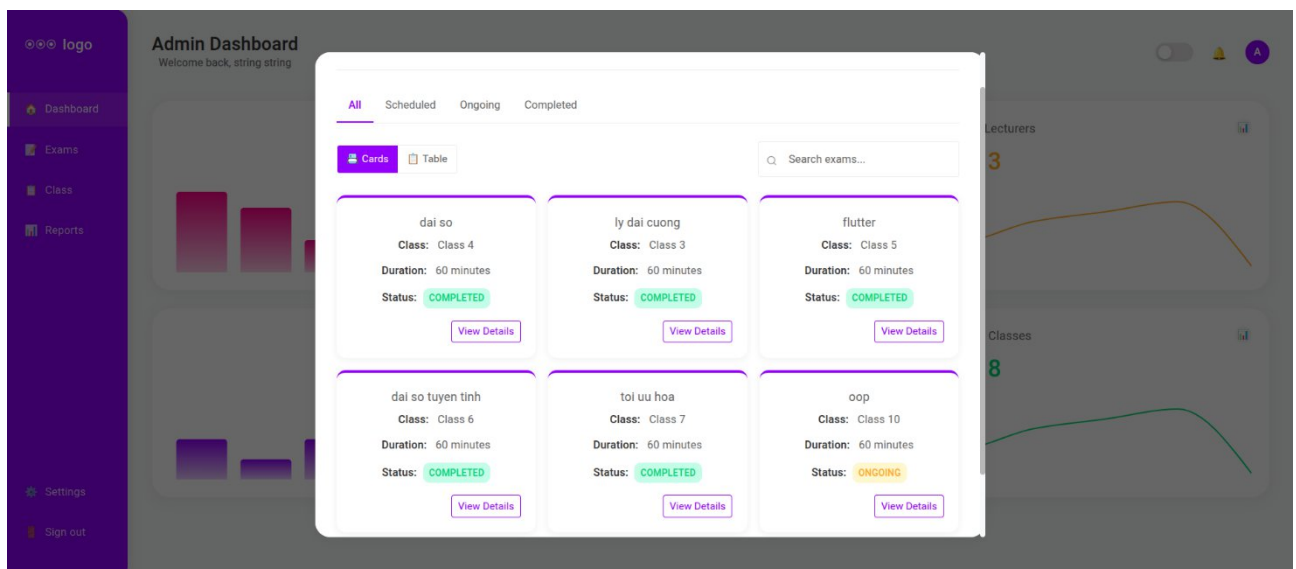
Hình 4.18: Giao diện thống kê sinh viên

Thống kê giảng viên:



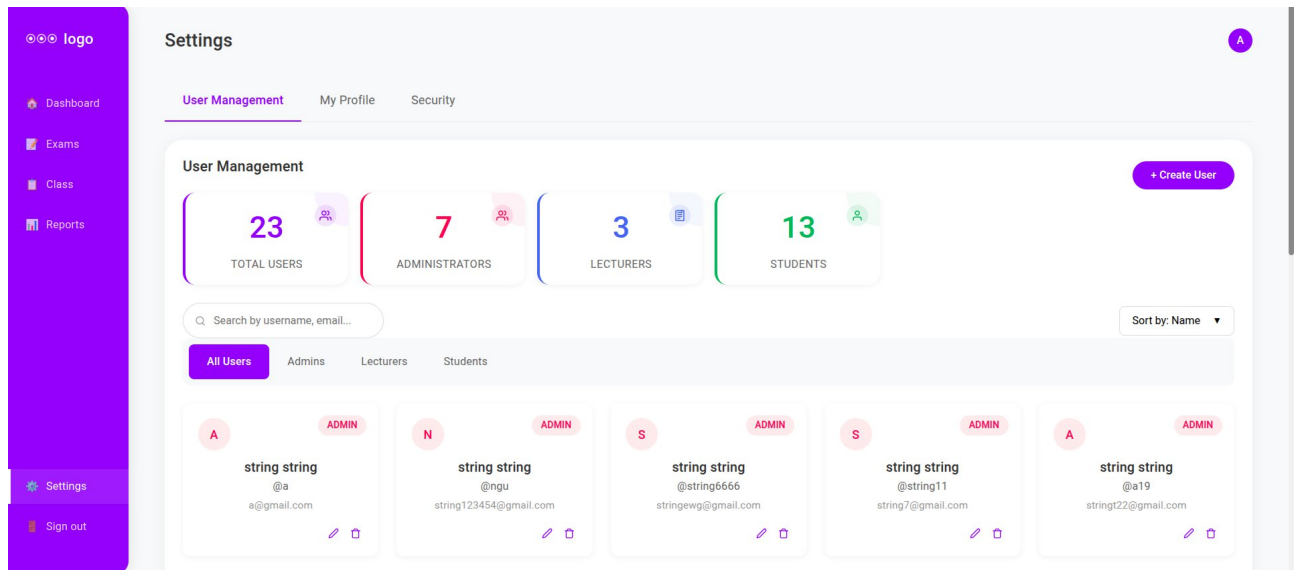
Hình 4.19: Giao diện thống kê giảng viên

Thống kê bài thi:



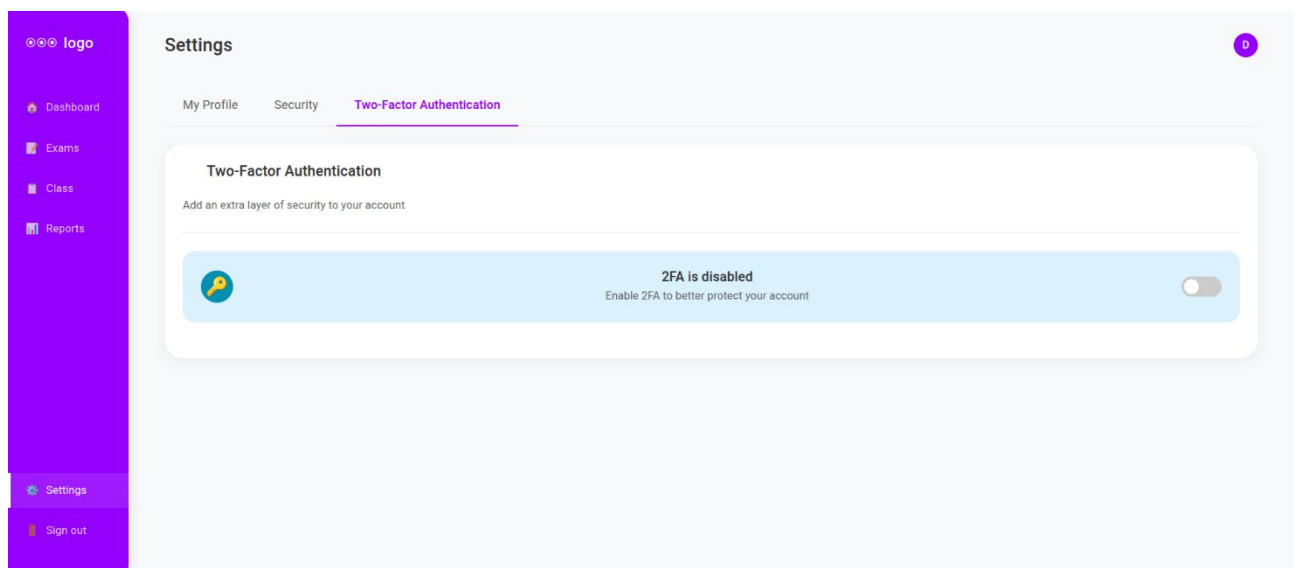
Hình 4.20: Giao diện thống kê bài thi

Giao diện quản lý chức năng người dùng của admin



Hình 4.21: Giao diện quản lý chức năng người dùng của admin

Giao diện chức năng bảo mật 2 lớp



Hình 4.22: Giao diện chức năng bảo mật 2 lớp

CHƯƠNG V: KẾT LUẬN

5. Kết quả đạt được và hướng phát triển

5.1. Kết quả đạt được

Trong quá trình triển khai và phát triển hệ thống web thi online, tôi đã đạt được nhiều kết quả triển vọng như sau:

- Hoàn thiện tài liệu phân tích và đặc tả các yêu cầu hệ thống một cách chi tiết.
- Thiết kế cơ sở dữ liệu bằng MySQL phù hợp với hệ thống, đảm bảo lưu trữ và quản lý dữ liệu hiệu quả.
- Xây dựng giao diện thân thiện dễ sử dụng bằng ReactJs cho web thi online.
- Xây dựng các trang quản lí, dashboard phân quyền đáp ứng đủ yêu cầu chức năng cơ bản.
- Xây dựng các chức năng lọc, tìm kiếm, thống kê một cách logic, rõ ràng.
- Hệ thống thi được phân chia và bảo mật rõ ràng theo từng lớp học, từng mốc thời gian.
- Kiểm thử thành công các chức năng cơ bản của hệ thống.
- Đóng gói docker và deploy thành công lên vps.

5.2. Hạn chế

Bên cạnh đó trong quá trình phát triển hệ thống tôi còn gặp một số khó khăn :

- Giao diện Landing Page còn thô, chưa thật sự hoàn chỉnh cho phần liên hệ và giới thiệu.
- Hiệu suất giữa việc gọi API với các services còn chưa tối ưu, đặc biệt ở phần lấy mã OTP.
- Còn 1 số lỗi liên quan đến bài thi chưa được kiểm thử triệt để.
- Gặp khó khăn trong việc quản lí limit rate của chat bot.

5.3. Hướng phát triển

Dựa trên kết quả đạt được và những khó khăn gặp phải, tôi định hướng phát triển hệ thống trong tương lai như sau:

- Nâng cấp giao diện Landing Page cho hoàn chỉnh, bổ sung một vài animation để tăng trải nghiệm người dùng.
- Sử dụng nhiều component hơn để dễ dàng bảo trì và mở rộng hệ thống.
- Tối ưu các services xử lý ở backend, sử dụng Redis để giảm tải cho cơ sở dữ liệu.
- Tuning để tối ưu hóa bot chat hướng dẫn sử dụng cũng như định hướng học tập cho người dùng.
- Kiểm tra lại quy trình đóng gói Docker tránh lỗi.

Với những gì đã triển khai, tôi tin dự án này còn có tiềm năng phát triển thành một sản phẩm hoàn chỉnh và thương mại hóa trong tương lai. Thời gian tới tôi sẽ cố gắng tìm hiểu thêm về nhu cầu người dùng cũng như yêu cầu của bên sử dụng để tiếp tục hoàn thiện tạo ra một sản phẩm tốt cho nền giáo dục.

TÀI LIỆU THAM KHẢO

- [1] Vilbo, "HTML là gì?," 2021. [Online]. Available: <https://viblo.asia/p/html-la-gi-aWj53o3Q56m>.
- [2] TopDev, "CSS là gì?," [Online]. Available: <https://topdev.vn/blog/css-la-gi/>.
- [3] 200lab, "Javascript là gì? Ngôn ngữ lập trình phổ biến nhất hiện nay," 08 2024. [Online]. Available: <https://200lab.io/blog/javascript-la-gi>.
- [4] Vilbo, "Bootstrap là gì? Giới thiệu về Bootstrap," 11 2016. [Online]. Available: <https://viblo.asia/p/bai-1-bootstrap-la-gi-gioi-thieu-ve-bootstrap-DzVk-pLbDknW>.
- [5] Vilbo, "ReactJS là gì? Những điều có thể bạn chưa biết về ReactJS," 06 2021. [Online]. Available: <https://viblo.asia/p/reactjs-la-gi-nhung-dieu-co-the-ban-chua-biet-ve-reactjs-eW65G4RRKDO>.
- [6] TopDev, "Spring Boot là gì?," [Online]. Available: <https://topdev.vn/blog/gioi-thieu-ve-spring-boot-spring-boot-la-gi/>.
- [7] J. Erickson, "MySQL: Understanding What It Is and How It's Used," 29 08 2024. [Online]. Available: <https://www.oracle.com/mysql/what-is-mysql/>.
- [8] Amazon, "What is Docker?," [Online]. Available: https://aws.amazon.com/docker/?nc1=h_ls.
- [9] M. Chiaramonte, "The benefits of a three-layered application architecture," 09 06 2024. [Online]. Available: <https://vfunction.com/blog/the-benefits-of-a-three-layered-application-architecture/>.
- [10] Vilbo, "Kiểm thử phần mềm là gì? Quy trình kiểm thử phần mềm," 16 04 2019. [Trực tuyến]. Available: <https://viblo.asia/p/kiem-thu-phan-mem-la-gi-quy-trinh-kiem-thu-phan-mem-1VgZvayYKAw>.

PHỤ LỤC

Phụ lục 1. Danh sách các testcase chi tiết: Tài liệu đính kèm

Phụ lục 2. [Link figma thiết kế UI/UX](#)

Phụ lục 3. [Link web](#)

DANH SÁCH CÁC TESTCASE CHI TIẾT

Mã testcase	Mục tiêu kiểm thử	Tiền điều kiện	Các bước thực hiện	Kết quả mong muốn	Kết quả hiện tại	Trạng thái	ID Bug
1. ĐĂNG KÝ - UI/UX TESTING							
TCDK_UI_01	Kiểm tra tổng thể giao diện màn hình đăng ký	<ul style="list-style-type: none"> - Truy cập vào trang đăng ký - Màn hình hiển thị bình thường 	<ol style="list-style-type: none"> 1. Kiểm tra bố cục tổng thể 2. Kiểm tra logo hiển thị 3. Kiểm tra font chữ 4. Kiểm tra các trường thông tin hiển thị đầy đủ 	<ul style="list-style-type: none"> - Logo hiển thị rõ ràng - Font chữ đồng nhất - Bố cục gọn gàng, chuyên nghiệp - Các trường: FirstName, LastName, Username, Email, Password hiển thị đầy đủ 	<ul style="list-style-type: none"> - Logo hiển thị rõ ràng - Font chữ đồng nhất - Bố cục gọn gàng, chuyên nghiệp - Các trường hiển thị đầy đủ 	Pass	
TCDK_UI_02	Kiểm tra giao diện	Truy cập vào trang đăng ký	1. Thu nhỏ cửa sổ trình duyệt	- Giao diện hiển thị đầy đủ	- Giao diện responsive hoàn	Pass	

	responsive		<p>2. Kiểm tra hiển thị trên mobile</p> <p>3. Kiểm tra bố cục không bị xô lệch</p>	<p>- Không bị xô lệch bố cục</p> <p>- Tất cả element vẫn sử dụng được</p>	<p>hảo</p> <p>- Không có lỗi xô lệch</p> <p>- Tất cả element hoạt động tốt</p>		
TCDK_UI_03	Kiểm tra navigation bằng phím Tab	Trang đăng ký đã load hoàn tất	<p>1. Click vào trường đầu tiên</p> <p>2. Nhấn phím Tab liên tục</p> <p>3. Quan sát thứ tự di chuyển</p>	<p>- Di chuyển từ trái sang phải</p> <p>- Di chuyển từ trên xuống dưới</p> <p>- Thứ tự logic và hợp lý</p>	<p>- Thứ tự tab đúng logic</p> <p>- Di chuyển mượt mà</p> <p>- Focus hiển thị rõ ràng</p>	Pass	
TCDK_UI_04	Kiểm tra chức năng phím Enter	Trang đăng ký hiển thị bình thường	<p>1. Focus vào nút đăng ký</p> <p>2. Nhấn Enter</p> <p>3. Điền đầy đủ thông tin hợp lệ</p>	<p>- Khi focus vào button và nhấn Enter: thực hiện chức năng</p> <p>- Khi điền đủ thông tin và nhấn Enter: thực hiện đăng ký</p>	<p>- Phím Enter hoạt động chính xác</p> <p>- Submit form khi nhấn Enter</p> <p>- Không có lỗi xử lý</p>	Pass	

			4. Nhấn Enter				
TCDK_UI_05	Kiểm tra thông báo trường bắt buộc	Trang đăng ký hiển thị	1. Để trống các trường bắt buộc 2. Click nút đăng ký 3. Kiểm tra thông báo lỗi	- Hiện thị thông báo "Trường này là bắt buộc" - Thông báo hiển thị rõ ràng - Không submit form	- Validation hoạt động đúng - Thông báo lỗi hiển thị - Form không được submit	Pass	
TCDK_UI_06	Kiểm tra nút hide/unhide mật khẩu	Trường password hiển thị	1. Nhập mật khẩu vào trường password 2. Click nút show/hide 3. Kiểm tra hiển thị mật khẩu	- Mặc định password bị ẩn (***) - Click show: hiển thị mật khẩu rõ ràng - Click hide: ẩn lại mật khẩu	- Nút toggle hoạt động tốt - Show/hide mật khẩu chính xác - Icon thay đổi phù hợp	Pass	
1. ĐĂNG KÝ – FUNCTIONAL TESTING							
TCDK_F_01	Đăng ký thành công	- Trang đăng ký hiển thị bình thường	1. Nhập FirstName hợp lệ 2. Nhập LastName	- Hiện thị thông báo "Đăng ký thành công" - Tài khoản được	- Đăng ký thành công - Tài khoản được	Pass	

		- Kết nối internet ổn định	hợp lệ 3. Nhập Username chưa tồn tại 4. Nhập Email hợp lệ chưa tồn tại 5. Nhập Password hợp lệ 6. Click nút Đăng ký	- Tạo tài khoản mới trong hệ thống - Chuyển hướng đến trang đăng nhập	tạo - Redirect đến login page		
TCDK_F_02	Kiểm tra validation FirstName	Trang đăng ký hiển thị	1. Nhập FirstName có ký tự đặc biệt (!@#\$%) 2. Nhập FirstName > 50 ký tự 3. Nhập FirstName tiếng Việt có dấu 4. Click đăng ký	- Ký tự đặc biệt: Báo lỗi "Không được chứa ký tự đặc biệt" - > 50 ký tự: Báo lỗi "Tối đa 50 ký tự" - Tiếng Việt có dấu: Chấp nhận	- Validation ký tự đặc biệt - Validation độ dài - Chấp nhận tiếng Việt có dấu	Pass	

TCDK_F_03	Kiểm tra validation Username	Trang đăng ký hiển thị	1. Nhập Username đã tồn tại 2. Nhập Username có ký tự đặc biệt 3. Nhập Username có dấu 4. Nhập Username > 50 ký tự	- Username tồn tại: "Username đã được sử dụng" - Ký tự đặc biệt: "Không được chứa ký tự đặc biệt" - Có dấu: "Không được chứa ký tự có dấu" - > 50 ký tự: "Tối đa 50 ký tự"	- Username tồn tại: "Username đã được sử dụng" - Ký tự đặc biệt: "Không được chứa ký tự đặc biệt" - Có dấu: "Không được chứa ký tự có dấu" - > 50 ký tự: "Tối đa 50 ký tự"	Pass	
TCDK_F_04	Kiểm tra validation Email	Trang đăng ký hiển thị	1. Nhập email sai định dạng (test@) 2. Nhập email đã tồn tại 3. Nhập email hợp lệ	- Sai định dạng: "Email không hợp lệ" - Đã tồn tại: "Email đã được sử dụng" - Hợp lệ: Chấp nhận	- Sai định dạng: "Email không hợp lệ" - Đã tồn tại: "Email đã được sử dụng" - Hợp lệ: Chấp	Pass	

					nhận		
2. ĐĂNG NHẬP – UI/UX TESTING							
TCDN_UI_1	Kiểm tra tổng thể giao diện màn hình		1. Kiểm tra vị trí bố cục, font chữ, chính tả, màu chữ 2. Kiểm tra title của màn hình 3. Kiểm tra header, foote	- Kiểm tra logo trên cùng, bên trái màn hình pop up - Hiện thị title Đăng nhập dưới logo - Hiện thị đầy đủ các trường: Email và Mật khẩu - Hiện thị các giá trị mặc định của trường Email và Mật khẩu + Email: null + Mật khẩu: null - Header, footer theo design có sẵn	- Kiểm tra logo trên cùng, bên trái màn hình pop up - Hiện thị title Đăng nhập dưới logo - Hiện thị đầy đủ các trường: Email và Mật khẩu - Hiện thị các giá trị mặc định của trường Email và Mật khẩu + Email: null + Mật khẩu: null	Pass	

					- Header, footer theo design cố sẵn		
TCDN_UI_2	Thực hiện chức năng khi ấn Enter	Nhấn nút Enter	<p>1. Nếu chuột không focus vào button nào thì không thực hiện chức năng nào</p> <p>2. Sau khi điền các thông tin hết thì ấn enter sẽ thực hiện chức năng đăng nhập</p>	<p>- Nếu chuột không focus vào button nào thì không thực hiện chức năng nào</p> <p>- Sau khi điền các thông tin hết thì ấn enter sẽ thực hiện chức năng đăng nhập</p>	<p>- Nếu chuột không focus vào button nào thì không thực hiện chức năng nào</p> <p>- Sau khi điền các thông tin hết thì ấn enter sẽ thực hiện chức năng đăng nhập</p>	Pass	
TCDN_UI_3	Kiểm tra giao diện khi thu nhỏ màn hình		Thu nhỏ màn hình và kiểm tra hiển thị	Hiển thị đầy đủ, không bị xô lệch bố cục	Hiển thị đầy đủ, không bị xô lệch bố cục	Pass	
TCDN_UI_4	Kiểm tra thứ tự di chuyển khi		Nhấn phím Tab liên tục và quan sát thứ tự di chuyển	Di chuyển từ trái sang phải, trên xuống dưới	Di chuyển từ trái sang phải, trên xuống dưới	Pass	

	nhấn Tab						
TCDN_UI_5	Kiểm tra thông báo trường bắt buộc		Không nhập thông tin và thực hiện đăng nhập	Có thông báo trường bắt buộc nhập	Có thông báo trường bắt buộc nhập	Pass	
TCDN_UI_6	Kiểm tra nút hide/unhide mật khẩu		Click vào icon mắt bên cạnh ô mật khẩu	Mật khẩu được hiển thị/ẩn khi click	Mật khẩu được hiển thị/ẩn khi click	Pass	
2. ĐĂNG NHẬP – FUNCTIONAL TESTING							
TCDN_F_7	Đăng nhập thành công		Nhập đầy đủ thông tin hợp lệ và đăng nhập	Đăng nhập thành công, chuyển đến trang chủ. Nếu có 2FA thì có bước xác thực qua email	Đăng nhập thành công, chuyển đến trang chủ. Nếu có 2FA thì có bước xác thực qua email	Pass	
TCDN_F_8	Kiểm tra trường bắt		Bỏ trống các trường và thực hiện đăng	Hiện thị thông báo bắt buộc nhập	Hiện thị thông báo bắt buộc nhập	Pass	

	buộc nhập		nhập				
TCDN_F_9	Kiểm tra nhập sai tài khoản		Nhập tài khoản không tồn tại	Hiển thị thông báo lỗi tài khoản không đúng	Hiển thị thông báo lỗi tài khoản không đúng	Pass	
TCDN_F_10	Kiểm tra nhập sai mật khẩu	Có tài khoản hợp lệ	Nhập mật khẩu sai	Hiển thị thông báo lỗi mật khẩu không đúng	Hiển thị thông báo lỗi mật khẩu không đúng	Pass	
3. CẬP NHẬP MẬT KHẨU – UI/UX TESTING							
TCMK_UI_1	Kiểm tra tổng thể giao diện màn hình		Kiểm tra bố cục, logo, font chữ, hiển thị đầy đủ thông tin	Giao diện hiển thị đúng thiết kế	Giao diện hiển thị đúng thiết kế	Pass	
TCMK_UI_2	Kiểm tra giao diện khi thu nhỏ		Thu nhỏ màn hình	Hiển thị đầy đủ, không bị xô lệch bố cục	Hiển thị đầy đủ, không bị xô lệch bố cục	Pass	
TCMK_UI_3	Kiểm tra thứ tự di		Nhấn Tab liên tục	Di chuyển từ trái sang phải, trên xuống	Di chuyển từ trái sang phải, trên	Pass	

	chuyển Tab			dưới	xuống dưới		
TCMK_UI_4	Kiểm tra chức năng khi nhấn Enter		Nhấn Enter khi focus và khi điền đủ thông tin	Thực hiện chức năng khi focus, cập nhật khi điền đủ thông tin	Thực hiện chức năng khi focus, cập nhật khi điền đủ thông tin	Pass	
TCMK_UI_5	Kiểm tra thông báo trường bắt buộc		Bỏ trống trường bắt buộc	Có thông báo trường bắt buộc nhập	Có thông báo trường bắt buộc nhập	Pass	
TCMK_UI_6	Kiểm tra nút hide/unhide mật khẩu		Click icon mắt ở ô mật khẩu	Mật khẩu được hiển thị/ẩn	Mật khẩu được hiển thị/ẩn	Pass	
3. CẬP NHẬP MẬT KHẨU – FUNCTIONAL TESTING							
TCMK_F_7	Cập nhật thành công	Đã đăng nhập	Nhập thông tin hợp lệ và cập nhật	Cập nhật mật khẩu thành công	Cập nhật mật khẩu thành công	Pass	
TCMK_F_8	Kiểm tra	Đã đăng nhập	Nhập mật khẩu cũ	Thông báo mật khẩu	Thông báo mật	Pass	

	mật khẩu cũ hợp lệ		sai	cũ không đúng	khẩu cũ không đúng		
TCMK_F_9	Kiểm tra 2 trường mật khẩu mới	Đã đăng nhập	Nhập 2 mật khẩu mới khác nhau	Thông báo 2 mật khẩu phải trùng nhau	Thông báo 2 mật khẩu phải trùng nhau	Pass	
4. QUẢN LÝ LỚP HỌC – UI/UX TESTING							
TCLH_UI_1	Kiểm tra tổng thể giao diện màn hình	Đã đăng nhập	Kiểm tra bố cục, logo, font chữ, hiển thị đầy đủ thông tin	Giao diện hiển thị đúng thiết kế	Giao diện hiển thị đúng thiết kế	Pass	
TCLH_UI_2	Kiểm tra giao diện khi thu nhỏ	Đã đăng nhập	Thu nhỏ màn hình	Hiển thị đầy đủ, không bị xô lệch bố cục	Hiển thị đầy đủ, không bị xô lệch bố cục	Pass	
TCLH_UI_3	Kiểm tra thứ tự di chuyển Tab	Đã đăng nhập	Nhấn Tab liên tục	Di chuyển từ trái sang phải, trên xuống dưới	Di chuyển từ trái sang phải, trên xuống dưới	Pass	
TCLH_UI_4	Kiểm tra	Đã đăng nhập	Nhấn Enter khi focus	Thực hiện chức năng	Thực hiện chức	Pass	

	chức năng khi nhấn Enter		và khi điền đủ thông tin	khi focus, thực hiện action khi điền đủ thông tin	năng khi focus, thực hiện action khi điền đủ thông tin		
TCLH_UI_5	Kiểm tra thông báo trường bắt buộc	Đã đăng nhập	Bỏ trống trường bắt buộc	Có thông báo trường bắt buộc nhập	Có thông báo trường bắt buộc nhập	Pass	
TCLH_UI_6	Kiểm tra alert xác nhận xóa lớp	Đã đăng nhập	Click vào action xóa lớp	Có alert xác nhận xóa lớp học	Có alert xác nhận xóa lớp học	Pass	
4. QUẢN LÝ LỚP HỌC – FUNCTIONAL TESTING							
TCLH_F_7	Tạo lớp học thành công	Đã đăng nhập	Nhập đầy đủ thông tin hợp lệ và tạo lớp	Tạo lớp học thành công	Tạo lớp học thành công	Pass	
TCLH_F_8	Cập nhật thông tin	Đã đăng nhập, có	Sửa thông tin lớp học	Cập nhật thông tin	Cập nhật thông tin	Pass	

	thành công	lớp học	và lưu	thành công	thành công		
TCLH_F_9	Kiểm tra trường tên lớp bắt buộc	Đã đăng nhập	Bỏ trống tên lớp và tạo/cập nhật	Thông báo trường tên lớp bắt buộc nhập	Thông báo trường tên lớp bắt buộc nhập	Pass	
TCLH_F_10	Kiểm tra max size tên lớp	Đã đăng nhập	Nhập tên lớp quá 255 ký tự	Không cho nhập quá 255 ký tự	Không cho nhập quá 255 ký tự	Pass	
TCLH_F_11	Kiểm tra tên lớp ký tự đặc biệt	Đã đăng nhập	Nhập tên lớp có tiếng Việt và ký tự đặc biệt	Cho nhập tiếng Việt có dấu, không cho ký tự đặc biệt	Cho nhập tiếng Việt có dấu, không cho ký tự đặc biệt	Pass	
TCLH_F_12	Kiểm tra max size mô tả	Đã đăng nhập	Nhập mô tả quá 255 ký tự	Không cho nhập quá 255 ký tự	Không cho nhập quá 255 ký tự	Pass	
TCLH_F_13	Kiểm tra mô tả ký tự đặc biệt	Đã đăng nhập	Nhập mô tả có tiếng Việt và ký tự đặc biệt	Cho nhập tiếng Việt có dấu, không cho ký tự đặc biệt	Cho nhập tiếng Việt có dấu, không cho ký tự đặc biệt	Pass	

THÊM SINH VIÊN VÀO LỚP – FUNCTIONAL TESTING							
TCLH_F_14	Thêm bằng file thành công	Đã đăng nhập, có lớp học	Upload file danh sách sinh viên đúng định dạng	Thêm sinh viên thành công	Thêm sinh viên thành công	Pass	
TCLH_F_15	Thêm thủ công thành công	Đã đăng nhập, có lớp học	Nhập thông tin sinh viên và thêm	Thêm sinh viên thành công	Thêm sinh viên thành công	Pass	
TCLH_F_16	Kiểm tra sinh viên đã có tài khoản	Đã đăng nhập, có lớp học	Thêm sinh viên chưa có tài khoản	Thông báo sinh viên chưa có tài khoản	Thông báo sinh viên chưa có tài khoản	Pass	
TCLH_F_17	Kiểm tra file đúng định dạng	Đã đăng nhập, có lớp học	Upload file sai định dạng (số cột, tên cột)	Thông báo file không đúng định dạng	Thông báo file không đúng định dạng	Pass	
5. QUẢN LÝ BÀI THI – UI/UX TESTING							
TCBT_UI_1	Kiểm tra tổng thể	Đã đăng nhập	Kiểm tra bố cục, logo, font chữ, hiển	Giao diện hiển thị đúng thiết kế	Giao diện hiển thị đúng thiết kế	Pass	

	giao diện màn hình		thị đầy đủ thông tin				
TCBT_UI_2	Kiểm tra giao diện khi thu nhỏ	Đã đăng nhập	Thu nhỏ màn hình	Hiển thị đầy đủ, không bị xô lệch bố cục	Hiển thị đầy đủ, không bị xô lệch bố cục	Pass	
TCBT_UI_3	Kiểm tra thứ tự di chuyển Tab	Đã đăng nhập	Nhấn Tab liên tục	Di chuyển từ trái sang phải, trên xuống dưới	Di chuyển từ trái sang phải, trên xuống dưới	Pass	
TCBT_UI_4	Kiểm tra chức năng khi nhấn Enter	Đã đăng nhập	Nhấn Enter khi focus và khi điền đủ thông tin	Thực hiện chức năng khi focus, thực hiện action khi điền đủ thông tin	Thực hiện chức năng khi focus, thực hiện action khi điền đủ thông tin	Pass	
TCBT_UI_5	Kiểm tra thông báo trường bắt buộc	Đã đăng nhập	Bỏ trống trường bắt buộc	Có thông báo trường bắt buộc nhập	Có thông báo trường bắt buộc nhập	Pass	

TCBT_UI_6	Kiểm tra alert xác nhận xóa bài thi	Đã đăng nhập, có bài thi	Click vào action xóa bài thi	Có alert xác nhận xóa bài thi	Có alert xác nhận xóa bài thi	Pass	
5. QUẢN LÝ BÀI THI – FUNCTIONAL TESTING							
TẠO/CẬP NHẬP THÔNG TIN BÀI THI							
TCBT_F_7	Tạo/cập nhật thông tin thành công	Đã đăng nhập	Nhập đầy đủ thông tin hợp lệ	Tạo/cập nhật bài thi thành công	Tạo/cập nhật bài thi thành công	Pass	
TCBT_F_8	Kiểm tra các trường bắt buộc nhập	Đã đăng nhập	Bỏ trống các trường bắt buộc	Thông báo trường bắt buộc nhập	Thông báo trường bắt buộc nhập	Pass	
TCBT_F_9	Kiểm tra title ký tự đặc biệt	Đã đăng nhập	Nhập title có tiếng Việt và ký tự đặc biệt	Cho nhập tiếng Việt có dấu, không cho ký tự đặc biệt	Cho nhập tiếng Việt có dấu, không cho ký tự đặc biệt	Pass	

TCBT_F_10	Kiểm tra duration	Đã đăng nhập	Nhập duration không phải số nguyên dương	Thông báo phải là số nguyên lớn hơn 0	Thông báo phải là số nguyên lớn hơn 0	Pass	
TCBT_F_11	Kiểm tra startAt	Đã đăng nhập	Nhập startAt nhỏ hơn thời gian hiện tại	Thông báo phải lớn hơn hoặc bằng thời gian hiện tại	Thông báo phải lớn hơn hoặc bằng thời gian hiện tại	Pass	
TCBT_F_12	Kiểm tra endAt	Đã đăng nhập	Nhập endAt nhỏ hơn thời gian hiện tại	Thông báo phải lớn hơn hoặc bằng thời gian hiện tại	Thông báo phải lớn hơn hoặc bằng thời gian hiện tại	Pass	
TCBT_F_13	Kiểm tra password	Đã đăng nhập	Nhập password có dấu và khoảng trắng	Phải là chữ không dấu, không chứa khoảng trắng	Phải là chữ không dấu, không chứa khoảng trắng	Pass	
THÊM CÂU HỎI VÀO BÀI THI - FUNCTIONAL TESTING							
TCBT_F_14	Thêm câu hỏi thủ công thành công	Đã đăng nhập, có bài thi	Nhập thông tin câu hỏi và thêm	Thêm câu hỏi thành công	Thêm câu hỏi thành công	Pass	

TCBT_F_15	Thêm câu hỏi bằng file thành công	Đã đăng nhập, có bài thi	Upload file câu hỏi đúng định dạng	Thêm câu hỏi thành công	Thêm câu hỏi thành công	Pass	
TCBT_F_16	Kiểm tra file đúng định dạng	Đã đăng nhập, có bài thi	Upload file câu hỏi sai định dạng	Thông báo file không đúng định dạng (số cột, tên cột, thiếu trường)	Thông báo file không đúng định dạng (số cột, tên cột, thiếu trường)	Pass	
TCBT_F_17	Kiểm tra câu hỏi type khác essay	Đã đăng nhập, có bài thi	Tạo câu hỏi thủ công type khác essay thiếu đáp án	Thông báo phải đủ 4 đáp án	Thông báo phải đủ 4 đáp án	Pass	
6. LÀM BÀI THI – UI/UX TESTING							
TCLB_UI_1	Kiểm tra tổng thể giao diện màn hình	Đã đăng nhập, có bài thi	Kiểm tra bố cục, logo, font chữ, hiển thị đầy đủ thông tin	Giao diện hiển thị đúng thiết kế	Giao diện hiển thị đúng thiết kế	Pass	

TCLB_UI_2	Kiểm tra giao diện khi thu nhỏ	Đã đăng nhập, có bài thi	Thu nhỏ màn hình	Hiển thị đầy đủ, không bị xô lệch bố cục	Hiển thị đầy đủ, không bị xô lệch bố cục	Pass	
TCLB_UI_3	Kiểm tra thứ tự di chuyển Tab	Đã đăng nhập, có bài thi	Nhấn Tab liên tục	Di chuyển từ trái sang phải, trên xuống dưới	Di chuyển từ trái sang phải, trên xuống dưới	Pass	
TCLB_UI_4	Kiểm tra chức năng khi nhấn Enter	Đã đăng nhập, có bài thi	Nhấn Enter khi focus và khi điền đủ thông tin	Thực hiện chức năng khi focus, thực hiện action khi điền đủ thông tin	Thực hiện chức năng khi focus, thực hiện action khi điền đủ thông tin	Pass	
TCLB_UI_5	Kiểm tra thông báo trường bắt buộc	Đã đăng nhập, có bài thi	Bỏ trống trường bắt buộc	Có thông báo trường bắt buộc nhập	Có thông báo trường bắt buộc nhập	Pass	
6. LÀM BÀI THI – FUNCTIONAL TESTING							
TCLB_F_6	Tham gia bài thi	Đã đăng nhập, có bài thi đang diễn	Nhập mật khẩu đúng	Tham gia bài thi	Tham gia bài thi	Pass	

	thành công	ra	và tham gia	thành công	thành công		
TCLB_F_7	Kiểm tra mật khẩu bắt buộc	Đã đăng nhập, có bài thi	Bỏ trống mật khẩu và tham gia	Thông báo mật khẩu bắt buộc điền	Thông báo mật khẩu bắt buộc điền	Pass	
TCLB_F_8	Kiểm tra mật khẩu đúng	Đã đăng nhập, có bài thi	Nhập mật khẩu sai	Thông báo mật khẩu không đúng	Thông báo mật khẩu không đúng	Pass	
TCLB_F_9	Kiểm tra thời gian bắt đầu	Đã đăng nhập, có bài thi chưa bắt đầu	Tham gia bài thi trước thời gian bắt đầu	Thông báo bài thi chưa bắt đầu	Thông báo bài thi chưa bắt đầu	Pass	
TCLB_F_10	Kiểm tra thời gian kết thúc	Đã đăng nhập, có bài thi đã kết thúc	Tham gia bài thi sau thời gian kết thúc	Thông báo bài thi đã kết thúc	Thông báo bài thi đã kết thúc	Pass	
TCLB_F_11	Kiểm tra thời gian đếm ngược	Đã tham gia bài thi	Quan sát thời gian trong quá trình làm bài	Thời gian đếm ngược chính xác	Thời gian đếm ngược chính xác	Pass	
TCLB_F_12	Kiểm tra	Đã bắt đầu làm	Thoát ra và vào lại	- Thời gian còn lại	- Thời gian còn lại	Pass	

	thoát ra vào lại	bài, đã trả lời một số câu	bài thi	chính xác - Câu hỏi hiển thị giống trước khi thoát ra	chính xác - Câu hỏi hiển thị giống trước khi thoát ra		
--	---------------------	-------------------------------	---------	--	--	--	--