

ĐẠI HỌC BÁCH KHOA HÀ NỘI
KHOA TOÁN - TIN

ĐỒ ÁN TỐT NGHIỆP

**Phân tích thiết kế hệ thống
website ôn luyện và tổ chức các kỳ thi Toán THPT**

KHỔNG THỊ HOÀI PHƯƠNG

Email: Phuong.KTH216871@sis.hust.edu.vn

Mã sinh viên: 20216871

Chuyên ngành Toán - Tin

Giảng viên hướng dẫn: TS. Ngô Quốc Hoàn

Chữ ký của GVHD

HÀ NỘI, 6/2025

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

1. Mục tiêu và nội dung của đề án

(a) Mục tiêu:

(b) Nội dung:

2. Kết quả đạt được

(a)

(b)

3. Ý thức làm việc của sinh viên:

(a)

(b)

(c)

Hà Nội, tháng ... năm 2025

Giảng viên hướng dẫn

TS. Ngô Quốc Hoàn

Phiếu báo cáo tiến độ đồ án

Lời cảm ơn

Đồ án này được hoàn thành nhờ sự hướng dẫn và chỉ bảo tận tình của TS. Ngô Quốc Hoàn. Em xin gửi lời cảm ơn sâu sắc và lòng kính trọng nhất đến thầy. Thầy không chỉ hướng dẫn em trong việc nghiên cứu và tìm hiểu, mà còn luôn thông cảm và tạo mọi điều kiện thuận lợi, động viên em trong suốt quá trình thực hiện đồ án.

Em cũng xin gửi lời tri ân sâu sắc đến các thầy cô của khoa Toán Tin, Đại học Bách khoa Hà Nội, những người đã trang bị cho em nền tảng kiến thức vững chắc để em có thể hoàn thành đồ án này.

Trong quá trình thực hiện đồ án và xử lý văn bản, chắc chắn em khó tránh khỏi những hạn chế và thiếu sót. Em rất mong nhận được sự góp ý của các thầy cô cũng như các độc giả để đồ án được hoàn thiện hơn.

Tóm tắt nội dung Đồ án

- 1.
- 2.
- 3.

Hà Nội, tháng ... năm 2025

Tác giả đồ án

Khổng Thị Hoài Phương

Mục lục

Bảng ký hiệu và chữ viết tắt	i
Danh sách bảng	ii
Danh sách hình vẽ	iii
Chương 1 Giới thiệu	1
1.1 Đặt vấn đề	1
1.2 Lý do chọn đề tài	2
1.3 Bài toán đặt ra	3
1.4 Mục tiêu và phương pháp nghiên cứu	3
1.5 Cấu trúc đồ án	4
Chương 2 Cơ sở lý thuyết	5
2.1 Công nghệ sử dụng	5
2.1.1 Ngôn ngữ lập trình	5
2.1.2 Công nghệ backend	5
2.1.3 Công nghệ frontend	6
2.1.4 Cơ sở dữ liệu	6
2.1.5 Thư viện và công cụ hỗ trợ	6
2.1.6 Kiến trúc hệ thống	6
2.2 Kiến thức nền tảng liên quan	7
2.2.1 HTTP và REST API	7
2.2.2 Authentication và Authorization	7
2.2.3 CRUD Operation	8
2.2.4 Mô hình Client-Server	8

2.3	Các mô hình tham khảo trong nghiên cứu và triển khai	8
2.3.1	Google Forms	8
2.3.2	Azota	9
2.3.3	Moodle	9
2.3.4	Bài học rút ra	9
Chương 3	Phân tích hệ thống	10
3.1	Mô hình Use Case	10
3.1.1	Mô hình UC của admin	16
3.1.2	Mô hình UC của học sinh	21
3.1.3	Mô hình UC của giáo viên	28
3.2	Mô hình động	34
3.2.1	Hệ thống	34
3.2.2	Học sinh	38
3.2.3	Giáo viên	38
3.3	Sơ đồ lớp	38
Chương 4	Thiết kế hệ thống	39
4.1	Sơ đồ luồng dữ liệu	39
4.2	Mô hình thực thể liên kết	39
4.3	Danh sách các bảng	39
4.4	Sơ đồ dữ liệu quan hệ	39
Chương 5	Triển khai và thử nghiệm hệ thống	40
5.1	Khảo sát và thu thập dữ liệu	40
5.2	Giao diện hệ thống	40
5.2.1	Giao diện Học sinh	40
5.2.2	Giao diện Giáo viên	40
5.2.3	Giao diện Admin	40
Kết luận		41
Chương A	MÃ NGUỒN CHƯƠNG TRÌNH	42
Chương B	HƯỚNG DẪN SỬ DỤNG SẢN PHẨM THI CÔNG	43

Chỉ mục	44
Tài liệu tham khảo	45

Bảng ký hiệu và chữ viết tắt

abc abc

THPT trung học phổ thông

UC Use Case

Danh sách bảng

Danh sách hình vẽ

3.1	Mô hình UC người dùng quản lý tài khoản	10
3.2	Mô hình UC admin quản trị hệ thống	16
3.3	Mô hình UC admin quản lý tài khoản	17
3.4	Mô hình UC admin báo cáo thống kê	19
3.5	Mô hình UC học sinh vào thi	21
3.6	Mô hình UC học sinh luyện tập	23
3.7	Mô hình UC học sinh tra cứu kết quả thi và phúc tra	25
3.8	Mô hình UC giáo viên cập nhật kỳ thi	28
3.9	Mô hình UC giáo viên cập nhật tài liệu	30
3.10	Mô hình UC giáo viên xử lý kết quả thi	32
3.11	Sơ đồ tuần tự người dùng đăng nhập hệ thống	34
3.12	Sơ đồ hoạt động người dùng đăng nhập hệ thống	34
3.13	Sơ đồ tuần tự người dùng thay đổi mật khẩu	35
3.14	Sơ đồ hoạt động người dùng thay đổi mật khẩu	36
3.15	Sơ đồ tuần tự người dùng đăng xuất	36
3.16	Sơ đồ tuần tự người dùng quên mật khẩu	37
3.17	Sơ đồ hoạt động người dùng quên mật khẩu	37
3.18	Sơ đồ tuần tự học sinh vào thi	38
3.19	Sơ đồ tuần tự giáo viên tạo kỳ thi	38

Chương 1

Giới thiệu

1.1 Đặt vấn đề

Trong bối cảnh công nghệ số phát triển mạnh mẽ, giáo dục Việt Nam đang trải qua quá trình chuyển đổi số sâu rộng để đáp ứng yêu cầu của thời đại 4.0. Theo Báo cáo của Cục Công nghệ thông tin, Bộ Giáo dục và Đào tạo (Bộ GD&ĐT) tại Hội thảo Chuyển đổi số trong giáo dục tháng 12/2022, hơn 80% các cơ sở giáo dục phổ thông đã ứng dụng các nền tảng học trực tuyến như Google Classroom, Microsoft Teams, và Zoom để hỗ trợ giảng dạy [1]. Sau đại dịch COVID-19, mô hình học trực tuyến và hybrid (kết hợp trực tuyến và trực tiếp) đã chứng minh tính linh hoạt, giúp duy trì hoạt động giáo dục trong bối cảnh giãn cách xã hội. Tuy nhiên, trong khi giảng dạy đã có những bước tiến trong số hóa, công tác tổ chức và quản lý các kỳ thi, đặc biệt là kỳ thi môn Toán ở cấp trung học phổ thông (THPT), vẫn còn nhiều hạn chế.

Hiện nay, tại nhiều trường THPT ở Việt Nam, việc tổ chức các kỳ thi môn Toán chủ yếu dựa trên phương pháp truyền thống: in ấn đề thi trắc nghiệm trên giấy, thu bài thủ công, và chấm điểm bằng máy quét hoặc tay. Các trường THPT, đặc biệt ở khu vực nông thôn, vẫn gặp khó khăn trong việc áp dụng công nghệ vào quản lý thi cử, dẫn đến tốn kém thời gian và nhân lực [2]. Những quy trình quản lý thi cử hiện nay không chỉ tốn thời gian mà còn tiềm ẩn rủi ro như sai sót trong khâu quét đáp án, thất lạc phiếu trả lời, hoặc thiếu minh bạch trong quản lý điểm số. Với môn Toán trắc nghiệm, vốn đòi hỏi độ chính xác cao trong kiểm tra kiến thức và khả năng tư duy logic, việc tổ

chức thi thủ công gây khó khăn trong việc lưu trữ, tra cứu kết quả, và cung cấp phản hồi nhanh chóng cho học sinh.

Bên cạnh đó, nhu cầu đánh giá năng lực học sinh một cách liên tục và hiệu quả đang ngày càng gia tăng. Theo Chương trình Giáo dục Phổ thông 2018 của Bộ GD&ĐT, việc đánh giá học sinh không chỉ dựa trên điểm số mà còn phải phản ánh được quá trình học tập và năng lực tư duy [3]. Tuy nhiên, các phương pháp quản lý thi truyền thống khó đáp ứng được yêu cầu này do thiếu các công cụ hỗ trợ phân tích dữ liệu kết quả thi, theo dõi tiến bộ học tập, và cung cấp phản hồi kịp thời cho học sinh.

Trên thế giới, nhiều quốc gia đã triển khai các hệ thống thi trực tuyến để giải quyết những vấn đề trên. Chẳng hạn, hệ thống quản lý thi trực tuyến của ExamSoft (Hoa Kỳ) hay Moodle Quiz (Úc) đã được ứng dụng rộng rãi, giúp tự động hóa quy trình ra đề, chấm điểm, và phân tích kết quả [4] [5]. Tại Việt Nam, một số trường đại học lớn như Đại học Quốc gia Hà Nội đã bắt đầu thí điểm các kỳ thi trực tuyến, nhưng ở cấp THPT, các giải pháp tương tự vẫn còn hạn chế [6]. Điều này đặt ra nhu cầu cấp thiết về việc phát triển một website quản lý kỳ thi môn Toán dành riêng cho học sinh THPT, với các tính năng như tổ chức thi trực tuyến, chấm điểm tự động, lưu trữ kết quả, và cung cấp báo cáo phân tích.

Website quản lý kỳ thi môn Toán THPT không chỉ giúp giảm tải cho giáo viên trong các khâu tổ chức và đánh giá, mà còn nâng cao tính minh bạch, chính xác và hiệu quả của quá trình thi cử. Đồng thời, hệ thống này có thể hỗ trợ học sinh theo dõi tiến độ học tập, nhận phản hồi chi tiết về bài thi, từ đó cải thiện kết quả học tập. Với những lợi ích thiết thực, đề tài “Xây dựng website quản lý kỳ thi môn Toán trung học phổ thông” không chỉ đáp ứng nhu cầu thực tiễn mà còn góp phần thúc đẩy quá trình chuyển đổi số trong giáo dục Việt Nam.

1.2 Lý do chọn đề tài

Là sinh viên ngành Toán – Tin ứng dụng, em đặc biệt quan tâm đến việc áp dụng công nghệ web để giải quyết các bài toán thực tiễn trong giáo dục. Trong quá trình học tập và làm việc với các công nghệ như Django, ReactJS và MySQL, em nhận ra rằng

một hệ thống thi trắc nghiệm trực tuyến hoàn toàn có thể được xây dựng hiệu quả từ những công cụ mã nguồn mở hiện đại.

Từ đó, em quyết định lựa chọn đề tài "Xây dựng hệ thống quản lý kỳ thi trực tuyến Zamix" – một nền tảng thi trắc nghiệm trực tuyến hướng đến sự đơn giản trong sử dụng nhưng vẫn đảm bảo đầy đủ các chức năng cần thiết cho giáo viên và học sinh. Đề tài không chỉ giúp em củng cố kiến thức về lập trình web, cơ sở dữ liệu, mà còn là cơ hội để giải quyết một vấn đề mang tính ứng dụng cao trong thực tế.

1.3 Bài toán đặt ra

Đề tài tập trung vào việc xây dựng một hệ thống web hỗ trợ tổ chức và quản lý kỳ thi trực tuyến. Hệ thống này cần đáp ứng các yêu cầu chính sau:

- Quản lý thông tin kỳ thi, môn học, ngân hàng câu hỏi và danh sách học sinh.
- Cho phép học sinh làm bài trắc nghiệm trực tiếp trên nền tảng, chấm điểm tự động sau khi nộp bài.
- Giao diện đơn giản, dễ thao tác, phù hợp với người dùng không chuyên công nghệ.
- Thống kê kết quả, lưu trữ lịch sử bài thi và xuất báo cáo hỗ trợ đánh giá.
- Đảm bảo tính bảo mật, toàn vẹn dữ liệu và có khả năng mở rộng trong tương lai.

1.4 Mục tiêu và phương pháp nghiên cứu

Mục tiêu

- Thiết kế và triển khai một hệ thống web quản lý kỳ thi trực tuyến có khả năng hoạt động ổn định, hiệu quả và dễ bảo trì.
- Hệ thống đáp ứng đầy đủ các chức năng cơ bản như tạo đề thi, làm bài, chấm điểm và thống kê.

Phương pháp thực hiện

- Tìm hiểu và áp dụng mô hình kiến trúc phân tầng (Layered Architecture) trong thiết kế phần mềm.
- Sử dụng Django làm backend để xử lý logic và quản lý dữ liệu.
- Sử dụng ReactJS để xây dựng giao diện người dùng một cách linh hoạt và tương tác cao.
- Thiết kế cơ sở dữ liệu bằng MySQL, đảm bảo dữ liệu được lưu trữ an toàn và có cấu trúc rõ ràng.
- Áp dụng quy trình phát triển phần mềm bao gồm: phân tích yêu cầu, thiết kế, cài đặt, kiểm thử và đánh giá.

1.5 Cấu trúc đồ án

Đồ án được trình bày gồm 4 chương chính sau:

- **Chương 2: Cơ sở lý thuyết và công nghệ** – Trình bày các kiến thức nền tảng và công nghệ được sử dụng trong quá trình xây dựng hệ thống.
- **Chương 3: Phân tích hệ thống** – Tập trung làm rõ các yêu cầu chức năng và phi chức năng của hệ thống Zamix. Từ đó, đưa ra mô hình nghiệp vụ, các use case cũng như phân tích luồng xử lý dữ liệu để làm nền tảng cho giai đoạn thiết kế.
- **Chương 4: Thiết kế hệ thống** – Thiết kế cơ sở dữ liệu và giao diện người dùng.
- **Chương 5: Triển khai và thử nghiệm hệ thống** – Mô tả quá trình xây dựng, cài đặt và triển khai hệ thống Zamix từ backend đến frontend. Đồng thời, đánh giá hiệu quả hoạt động của hệ thống thông qua quá trình kiểm thử chức năng, giao diện và hiệu suất.

Chương 2

Cơ sở lý thuyết

2.1 Công nghệ sử dụng

Trong quá trình xây dựng hệ thống web quản lý kỳ thi Toán THPT, tôi đã sử dụng nhiều công nghệ hiện đại trong cả phía máy chủ (backend) lẫn giao diện người dùng (frontend). Mục tiêu là tạo ra một hệ thống thân thiện với người dùng, dễ mở rộng, dễ bảo trì và có hiệu năng tốt.

2.1.1 Ngôn ngữ lập trình

- **Python:** Được sử dụng cho backend của hệ thống với các framework như Django. Python có cú pháp đơn giản, dễ viết và hệ sinh thái phong phú, đặc biệt phù hợp với các ứng dụng web và xử lý dữ liệu.
- **JavaScript:** Ngôn ngữ chính để xây dựng frontend, kết hợp với thư viện ReactJS để xây dựng giao diện tương tác nhanh, mượt mà.

2.1.2 Công nghệ backend

- **Django:** Framework web Python mạnh mẽ, cung cấp sẵn nhiều tính năng như quản lý người dùng, ORM, phân quyền, phù hợp với các hệ thống phức tạp như quản lý kỳ thi.

2.1.3 Công nghệ frontend

- **ReactJS:** Thư viện JavaScript phổ biến dùng để xây dựng giao diện người dùng động, giúp phản hồi nhanh khi người dùng thao tác. ReactJS dễ mở rộng, dễ tích hợp với các thư viện như Redux, Tailwind CSS, Axios,...
- **Tailwind CSS:** Framework CSS giúp xây dựng giao diện hiện đại, nhất quán và có khả năng tùy biến cao.

2.1.4 Cơ sở dữ liệu

- **MySQL:** Là hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở, đáng tin cậy, hỗ trợ tốt các truy vấn phức tạp. Dữ liệu người dùng, thông tin kỳ thi, câu hỏi trắc nghiệm, kết quả thi,... được lưu trữ trong hệ thống MySQL.

2.1.5 Thư viện và công cụ hỗ trợ

- **Axios:** Dùng để giao tiếp giữa frontend và backend thông qua các API REST.
- **React Router:** Quản lý định tuyến trong frontend, cho phép chuyển trang mà không cần tải lại toàn bộ.
- **pgAdmin:** Công cụ quản trị cơ sở dữ liệu PostgreSQL.
- **VS Code:** Môi trường phát triển chính trong quá trình lập trình.

2.1.6 Kiến trúc hệ thống

Hệ thống được xây dựng theo mô hình kiến trúc ba lớp (Three-tier Architecture):

- **Lớp trình diễn (Presentation Layer):** ReactJS hiển thị giao diện người dùng, tiếp nhận thao tác của học sinh và giám thị (ví dụ: đăng nhập, chọn kỳ thi, làm bài, xem kết quả).
- **Lớp logic nghiệp vụ (Application Layer):** Django xử lý nghiệp vụ như xác thực người dùng, tính điểm, lưu kết quả thi, v.v.

- **Lớp dữ liệu (Data Layer):** PostgreSQL lưu trữ toàn bộ dữ liệu hệ thống: thông tin kỳ thi, đề thi, câu hỏi, học sinh, kết quả,...

2.2 Kiến thức nền tảng liên quan

Hệ thống quản lý kỳ thi trực tuyến là một ứng dụng web điển hình, đòi hỏi sự phối hợp chặt chẽ giữa frontend và backend, giữa người dùng và máy chủ. Một số kiến thức nền tảng cần nắm vững khi phát triển hệ thống này bao gồm:

2.2.1 HTTP và REST API

HTTP (HyperText Transfer Protocol) là giao thức truyền tải dữ liệu chính giữa client (trình duyệt hoặc frontend) và server (backend). Trong hệ thống này, HTTP được sử dụng để gửi các yêu cầu (request) từ phía người dùng tới máy chủ và nhận về dữ liệu (response).

REST API (Representational State Transfer - Application Programming Interface) là phong cách thiết kế kiến trúc cho việc tạo các dịch vụ web, dựa trên giao thức HTTP. REST API sử dụng các phương thức HTTP như GET (đọc dữ liệu), POST (tạo dữ liệu mới), PUT/PATCH (cập nhật dữ liệu) và DELETE (xoá dữ liệu), qua đó cho phép frontend giao tiếp hiệu quả với backend.

2.2.2 Authentication và Authorization

Authentication (xác thực) là quá trình kiểm tra danh tính người dùng, thường thông qua tên đăng nhập và mật khẩu. Authorization (phân quyền) là quá trình xác định người dùng được phép thực hiện những hành động nào trong hệ thống, ví dụ như chỉ quản trị viên mới được tạo kỳ thi mới.

Trong hệ thống này, authentication được triển khai thông qua JSON Web Token (JWT), cho phép người dùng đăng nhập một lần và sử dụng token để xác minh trong các yêu cầu tiếp theo. Authorization đảm bảo người dùng chỉ truy cập các tài nguyên phù hợp với vai trò của họ.

2.2.3 CRUD Operation

CRUD là viết tắt của Create (tạo), Read (đọc), Update (cập nhật) và Delete (xoá) – bốn thao tác cơ bản nhất trong mọi hệ thống quản lý dữ liệu. Hệ thống thi trực tuyến cần hỗ trợ đầy đủ các thao tác CRUD đối với các đối tượng như kỳ thi, câu hỏi, đáp án và kết quả.

2.2.4 Mô hình Client-Server

Hệ thống được xây dựng theo mô hình client-server, trong đó frontend (client) chịu trách nhiệm hiển thị giao diện người dùng và gửi yêu cầu, còn backend (server) xử lý logic, truy xuất cơ sở dữ liệu và trả về phản hồi. Mô hình này giúp phân tách rõ ràng giữa phần hiển thị và phần xử lý, dễ mở rộng và bảo trì.

2.3 Các mô hình tham khảo trong nghiên cứu và triển khai

Để đảm bảo hệ thống thi trực tuyến được thiết kế hiệu quả, việc tham khảo và phân tích các nền tảng hiện có là cần thiết. Một số hệ thống đáng chú ý bao gồm:

2.3.1 Google Forms

Google Forms là công cụ phổ biến được sử dụng trong giáo dục để tạo bài kiểm tra và khảo sát nhanh. Ưu điểm của Google Forms là dễ sử dụng, miễn phí, tích hợp với Google Sheets để quản lý kết quả. Tuy nhiên, hệ thống này có những hạn chế như:

Giao diện đơn giản, không phù hợp cho đề thi nhiều phần hoặc yêu cầu trình bày phức tạp.

Không hỗ trợ phân quyền phức tạp (giáo viên, học sinh, quản trị).

Không có chức năng chấm điểm tự động cho bài tự luận hoặc phân tích thống kê chi tiết.

2.3.2 Azota

Azota là nền tảng hỗ trợ thi và kiểm tra trực tuyến được sử dụng rộng rãi trong trường phổ thông tại Việt Nam. Hệ thống này có giao diện thân thiện, hỗ trợ đề thi trắc nghiệm và tự luận, chấm điểm tự động, thống kê kết quả.

Tuy nhiên, Azota vẫn tồn tại một số hạn chế:

Không hỗ trợ người dùng tự thiết kế cấu trúc kỳ thi phức tạp (nhiều phần, nhiều đề con).

Bảo mật chưa được đánh giá cao.

Không có mã nguồn mở, khó tùy biến theo nhu cầu riêng.

2.3.3 Moodle

Moodle là hệ thống quản lý học tập mã nguồn mở (LMS – Learning Management System), hỗ trợ đầy đủ các chức năng tạo bài kiểm tra, phân quyền, chấm điểm, xuất báo cáo. Moodle có khả năng mở rộng mạnh, nhưng lại gặp khó khăn ở:

Giao diện người dùng còn rườm rà, chưa thân thiện với học sinh phổ thông.

Khó triển khai nhanh đối với những hệ thống đơn giản, yêu cầu cấu hình nhiều bước.

Cài đặt, tùy biến và bảo trì tương đối phức tạp đối với người dùng không chuyên.

2.3.4 Bài học rút ra

Từ việc phân tích các hệ thống trên, có thể rút ra một số định hướng cho hệ thống cần xây dựng:

Cần giao diện thân thiện, tối giản, phù hợp với học sinh THPT.

Hỗ trợ phân quyền rõ ràng giữa các vai trò (học sinh, giáo viên, quản trị viên).

Tích hợp chức năng chấm điểm tự động, thống kê kết quả và theo dõi lịch sử thi.

Có khả năng mở rộng, dễ bảo trì, dễ tích hợp với các hệ thống khác.

Chương 3

Phân tích hệ thống

3.1 Mô hình Use Case



Hình 3.1: Mô hình UC người dùng quản lý tài khoản

Use Case Name	Đăng ký
Description	Người dùng tạo tài khoản mới để truy cập hệ thống.

Actor(s)	Người dùng
Trigger	Người dùng muốn tạo tài khoản cá nhân để sử dụng hệ thống.
Pre-Condition(s)	<ul style="list-style-type: none"> • Người dùng chưa có tài khoản trong hệ thống. • Thiết bị có kết nối Internet.
Post-Condition(s)	Tài khoản mới được tạo và lưu trữ trong cơ sở dữ liệu.
Basic Flow	<ol style="list-style-type: none"> 1. Người dùng chọn chức năng “Đăng ký”. 2. Hệ thống hiển thị biểu mẫu yêu cầu nhập thông tin cá nhân. 3. Người dùng nhập tài khoản cá nhân, mật khẩu và các thông tin cần thiết. 4. Hệ thống kiểm tra thông tin và tạo tài khoản mới. 5. Hệ thống thông báo đăng ký thành công.
Alternative Flow	Không có.
Exception Flow	<p>4a. Tài khoản đã tồn tại trong hệ thống:</p> <p>4a1. Hệ thống thông báo lỗi và yêu cầu người dùng nhập tài khoản khác.</p> <p>4a2. Quay lại bước 3.</p>

Use Case Name	Đăng nhập
Description	Người dùng truy cập vào hệ thống bằng tài khoản đã đăng ký.

Actor(s)	Người dùng
Trigger	Người dùng muốn sử dụng các chức năng của hệ thống.
Pre-Condition(s)	Người dùng đã có tài khoản.
Post-Condition(s)	Người dùng được chuyển đến giao diện chính của hệ thống.
Basic Flow	<ol style="list-style-type: none"> 1. Người dùng chọn chức năng “Đăng nhập”. 2. Người dùng nhập tài khoản và mật khẩu. 3. Hệ thống kiểm tra thông tin. 4. Nếu hợp lệ, người dùng được đăng nhập vào hệ thống.
Alternative Flow	<p>2a. Người dùng quên mật khẩu:</p> <p>2a1. Chuyển sang Use Case “Xử lý quên mật khẩu”.</p>
Exception Flow	<p>3a. Tài khoản hoặc mật khẩu sai:</p> <p>3a1. Hệ thống thông báo lỗi và yêu cầu nhập lại.</p>

Use Case Name	Cập nhật tài khoản
Description	Người dùng chỉnh sửa các thông tin tài khoản cá nhân.
Actor(s)	Người dùng
Trigger	Người dùng muốn thay đổi thông tin tài khoản.
Pre-Condition(s)	Người dùng đã đăng nhập.
Post-Condition(s)	Thông tin cá nhân được cập nhật thành công.

Basic Flow	<ol style="list-style-type: none"> 1. Người dùng chọn chức năng “Cập nhật tài khoản”. 2. Hệ thống hiển thị thông tin hiện tại. 3. Người dùng thay đổi các thông tin cần cập nhật. 4. Hệ thống lưu lại các thay đổi và hiển thị thông báo thành công.
Alternative Flow	Không có.
Exception Flow	<p>4a. Thông tin không hợp lệ:</p> <p>4a1. Hệ thống yêu cầu nhập lại các trường bị lỗi.</p>

Use Case Name	Đăng xuất
Description	Người dùng thoát khỏi hệ thống một cách an toàn.
Actor(s)	Người dùng
Trigger	Người dùng muốn kết thúc phiên làm việc.
Pre-Condition(s)	Người dùng đã đăng nhập vào hệ thống.
Post-Condition(s)	<ul style="list-style-type: none"> • Phiên đăng nhập kết thúc. • Người dùng được đưa về trang đăng nhập.

Basic Flow	<ol style="list-style-type: none"> 1. Người dùng chọn chức năng “Đăng xuất”. 2. Hệ thống xác nhận hành động và kết thúc phiên làm việc. 3. Hệ thống đưa người dùng về trang đăng nhập.
Alternative Flow	Không có.
Exception Flow	Không có.

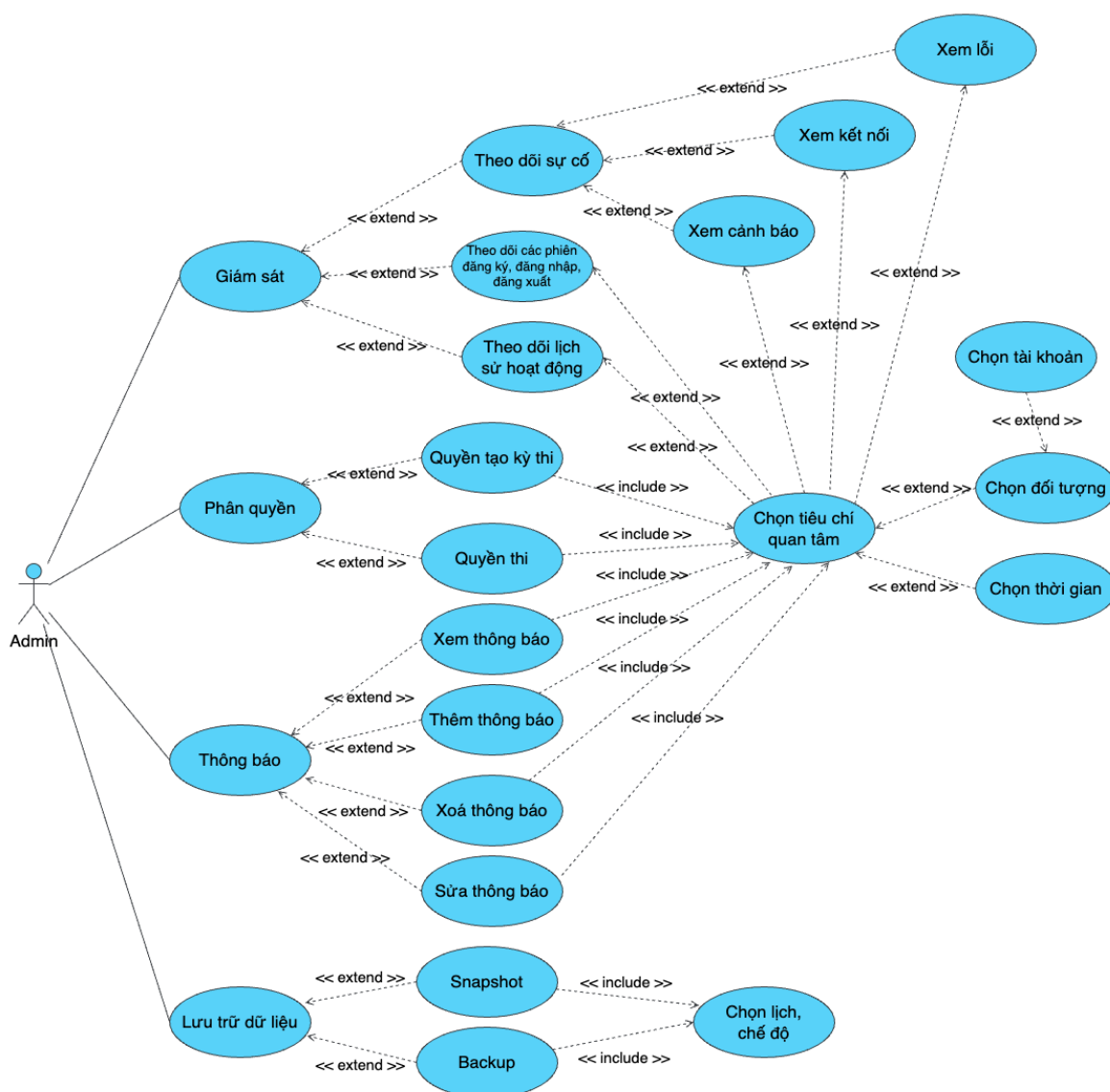
Use Case Name	Đổi mật khẩu
Description	Người dùng thay đổi mật khẩu tài khoản đang sử dụng.
Actor(s)	Người dùng
Trigger	Người dùng muốn nâng cao bảo mật tài khoản bằng cách thay đổi mật khẩu.
Pre-Condition(s)	Người dùng đã đăng nhập hệ thống.
Post-Condition(s)	Mật khẩu mới được cập nhật thành công.
Basic Flow	<ol style="list-style-type: none"> 1. Người dùng chọn chức năng “Đổi mật khẩu”. 2. Hệ thống yêu cầu nhập mật khẩu cũ. 3. Người dùng nhập mật khẩu cũ, mật khẩu mới. 4. Hệ thống kiểm tra và lưu lại mật khẩu mới nếu hợp lệ. 5. Hệ thống hiển thị thông báo đổi mật khẩu thành công.
Alternative Flow	Không có.

Exception Flow	3a. Mật khẩu cũ không đúng: 3a1. Hệ thống hiển thị thông báo lỗi và yêu cầu nhập lại.
-----------------------	---

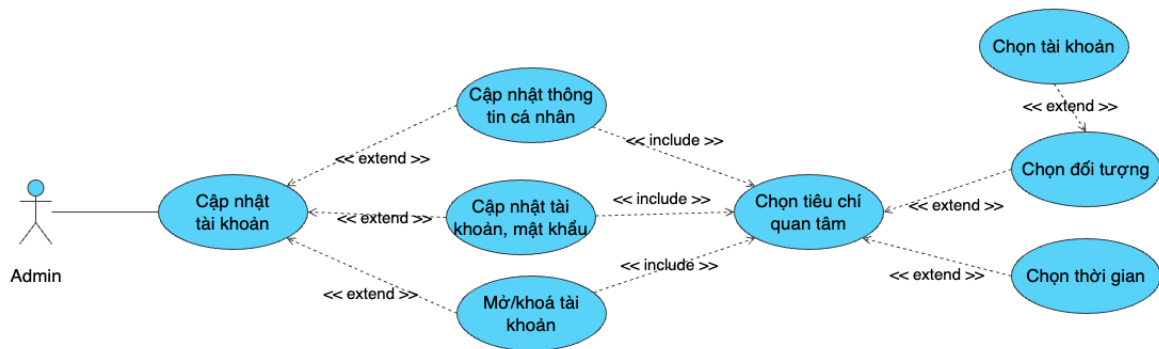
Use Case Name	Xử lý quên mật khẩu
Description	Người dùng khôi phục lại mật khẩu khi quên thông tin đăng nhập.
Actor(s)	Người dùng
Trigger	Người dùng không nhớ mật khẩu và muốn khôi phục tài khoản.
Pre-Condition(s)	Người dùng có tài khoản hợp lệ trong hệ thống.
Post-Condition(s)	Mật khẩu mới được cập nhật và người dùng có thể sử dụng lại tài khoản.
Basic Flow	<ol style="list-style-type: none"> 1. Người dùng chọn “Quên mật khẩu”. 2. Hệ thống yêu cầu nhập tài khoản đăng ký. 3. Hệ thống gửi mã OTP qua email. 4. Người dùng nhập mã OTP nhận được. 5. Hệ thống yêu cầu nhập mật khẩu mới. 6. Người dùng thiết lập mật khẩu mới và hoàn tất quá trình khôi phục.
Alternative Flow	4a. Mã OTP hết hạn hoặc sai: 4a1. Hệ thống thông báo lỗi và yêu cầu gửi lại mã mới.

Exception Flow	<p>2a. Tài khoản không tồn tại:</p> <p>2a1. Hệ thống thông báo lỗi và dừng quy trình khôi phục.</p>
-----------------------	--

3.1.1 Mô hình UC của admin



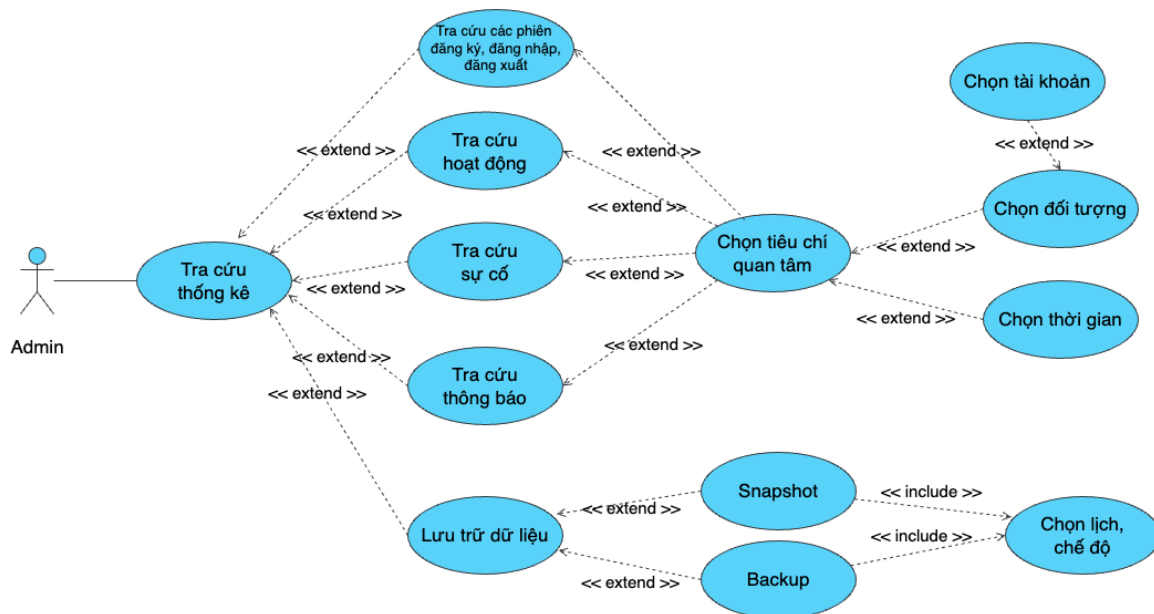
Hình 3.2: Mô hình UC admin quản trị hệ thống



Hình 3.3: Mô hình UC admin quản lý tài khoản

Use Case Name	Cập nhật tài khoản (Admin) (Hình 3.3)
Description	Admin thực hiện cập nhật thông tin hoặc trạng thái tài khoản của người dùng trên hệ thống.
Actor(s)	Admin
Trigger	Admin muốn quản lý thông tin hoặc trạng thái tài khoản người dùng.
Pre-Condition(s)	<ul style="list-style-type: none"> Admin đã đăng nhập vào hệ thống. Hệ thống có sẵn danh sách tài khoản.
Post-Condition(s)	<ul style="list-style-type: none"> Thông tin tài khoản được cập nhật. Tài khoản có thể bị khoá hoặc mở lại tùy theo thao tác.

Basic Flow	<ol style="list-style-type: none"> 1. Admin truy cập chức năng “Cập nhật tài khoản”. 2. Hệ thống hiển thị danh sách tài khoản người dùng. 3. Admin chọn một tài khoản để cập nhật. 4. Hệ thống hiển thị các lựa chọn: cập nhật thông tin cá nhân, đổi mật khẩu, khoá/mở tài khoản. 5. Admin thực hiện cập nhật theo yêu cầu. 6. Hệ thống lưu lại thay đổi và thông báo thành công.
Alternative Flow	<p>4a. Admin chọn cập nhật mật khẩu:</p> <p>4a1. Hệ thống yêu cầu chọn tiêu chí quan tâm (tài khoản, đối tượng, thời gian).</p> <p>4a2. Sau khi chọn, hệ thống cho phép nhập mật khẩu mới.</p>
Exception Flow	<p>3a. Tài khoản không tồn tại hoặc không thể chỉnh sửa:</p> <p>3a1. Hệ thống hiển thị thông báo lỗi và quay lại danh sách tài khoản.</p> <p><i>Use Case dừng lại.</i></p>

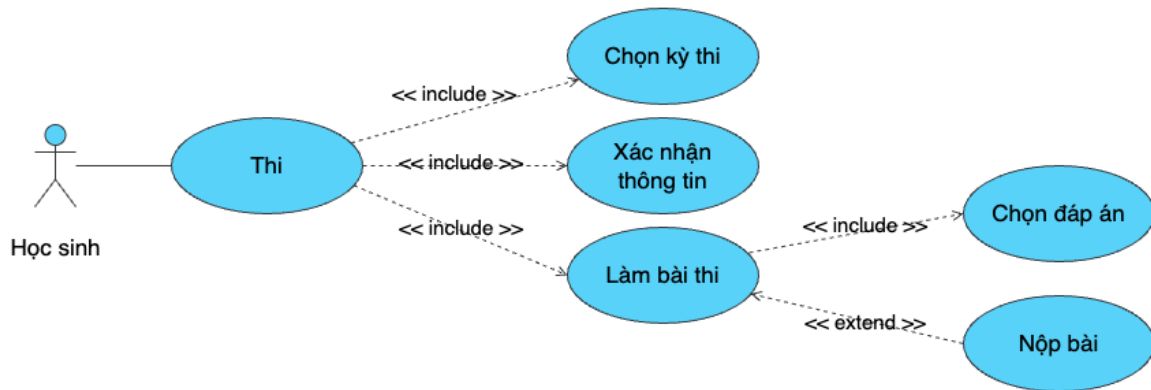


Hình 3.4: Mô hình UC admin báo cáo thống kê

Use Case Name	Tra cứu thống kê (Hình 3.4)
Description	Admin muốn truy xuất các thông tin thống kê hệ thống như đăng nhập, hoạt động, sự cố, thông báo, và lịch sử lưu trữ dữ liệu (Snapshot/Backup) nhằm giám sát và phân tích dữ liệu.
Actor(s)	Admin
Trigger	Admin truy cập vào chức năng thống kê
Pre-Condition(s)	Admin đã đăng nhập thành công vào hệ thống.
Post-Condition(s)	Dữ liệu thống kê được hiển thị hoặc xuất ra báo cáo nếu cần.

Basic Flow	<ol style="list-style-type: none"> 1. Admin truy cập chức năng "Tra cứu thống kê". 2. Hệ thống yêu cầu chọn tiêu chí quan tâm (thời gian, tài khoản, đối tượng). 3. Admin chọn các tiêu chí phù hợp. 4. Hệ thống hiển thị các loại thống kê theo tiêu chí: <ul style="list-style-type: none"> • Tra cứu các phiên đăng ký/đăng nhập/đăng xuất. • Tra cứu hoạt động hệ thống. • Tra cứu sự cố. • Tra cứu thông báo. • Tra cứu lịch sử lưu trữ dữ liệu (Snapshot, Backup). 5. Admin xem thông tin hoặc tải về nếu cần.
Alternative Flow	<p>2a. Admin không chọn đầy đủ tiêu chí.</p> <p>2a1. Hệ thống thông báo lỗi và yêu cầu chọn đủ thông tin.</p>
Exception Flow	<p>4a. Lỗi truy xuất cơ sở dữ liệu.</p> <p>4a1. Hệ thống hiển thị thông báo lỗi.</p> <p>4a2. Admin chọn thao tác khác hoặc thử lại sau.</p>

3.1.2 Mô hình UC của học sinh

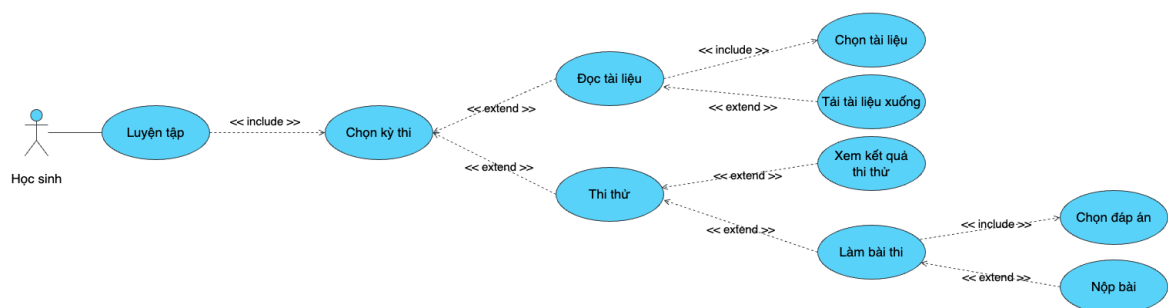


Hình 3.5: Mô hình UC học sinh vào thi

Use Case Name	Thi (Hình 3.5)
Description	Học sinh muốn tham gia kỳ thi để thực hiện bài kiểm tra theo đúng quy trình trên hệ thống.
Actor(s)	Học sinh
Trigger	Học sinh muốn thực hiện một bài thi
Pre-Condition(s)	<ul style="list-style-type: none"> • Học sinh đã đăng nhập vào hệ thống. • Thiết bị kết nối internet ổn định. • Kỳ thi đã được mở trong hệ thống.
Post-Condition(s)	<ul style="list-style-type: none"> • Bài thi được thực hiện và lưu lại trên hệ thống. • Kết quả thi sẽ được xử lý sau khi nộp bài.

Basic Flow	<ol style="list-style-type: none"> 1. Học sinh truy cập vào tính năng “Thi”. 2. Hệ thống yêu cầu học sinh chọn kỳ thi muốn tham gia. 3. Học sinh xác nhận lại thông tin cá nhân và thông tin kỳ thi. 4. Hệ thống hiển thị giao diện làm bài thi. 5. Học sinh thực hiện chọn đáp án cho từng câu hỏi. 6. Học sinh chọn lệnh “Nộp bài” khi hoàn thành bài thi. 7. Hệ thống xác nhận bài thi đã được nộp thành công và lưu kết quả.
Alternative Flow	<p>5a. Học sinh rời khỏi giao diện làm bài nhưng chưa nộp bài.</p> <p>5a1. Hệ thống tự động lưu tạm tiến trình làm bài.</p> <p>5a2. Khi học sinh quay lại, hệ thống khôi phục lại trạng thái bài làm.</p>

Exception Flow	<p>3a. Học sinh phát hiện thông tin không chính xác khi xác nhận.</p> <p>3a1. Học sinh hủy thao tác và quay lại trang chọn kỳ thi.</p> <p><i>Use Case dừng lại.</i></p> <p>6a. Kết nối internet bị gián đoạn khi đang nộp bài.</p> <p>6a1. Hệ thống lưu bài tạm thời và hiển thị thông báo lỗi.</p> <p>6a2. Khi kết nối lại, hệ thống yêu cầu nộp lại bài.</p> <p><i>Use Case tiếp tục từ bước 6.</i></p>
-----------------------	---

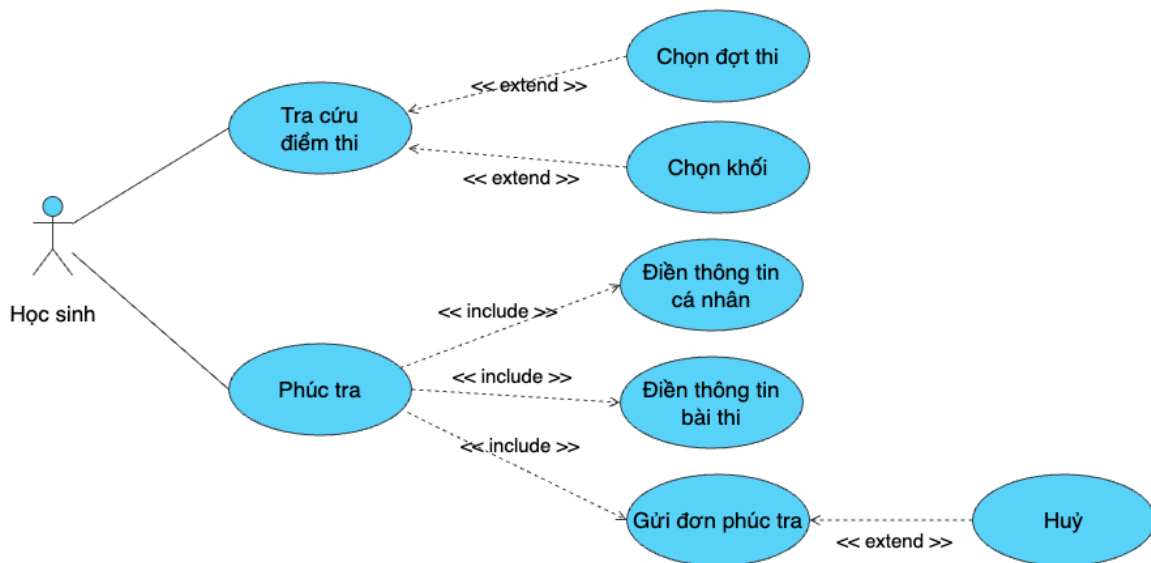


Hình 3.6: Mô hình UC học sinh luyện tập

Use Case Name	Luyện tập (Hình 3.6)
Description	Học sinh muốn tham gia luyện tập để ôn tập kiến thức và làm quen với các dạng câu hỏi qua thi thử hoặc đọc tài liệu.
Actor(s)	Học sinh
Trigger	Học sinh truy cập vào chức năng luyện tập từ ứng dụng

Pre-Condition(s)	<ul style="list-style-type: none"> • Học sinh đã đăng nhập vào hệ thống. • Các kỳ thi luyện tập hoặc tài liệu đã được cập nhật.
Post-Condition(s)	<ul style="list-style-type: none"> • Học sinh hoàn thành việc luyện tập. • Hệ thống ghi nhận kết quả (nếu có).
Basic Flow	<ol style="list-style-type: none"> 1. Học sinh chọn chức năng “Luyện tập”. 2. Hệ thống hiển thị danh sách kỳ thi luyện tập. 3. Học sinh chọn kỳ thi muốn luyện tập. 4. Học sinh lựa chọn phương thức luyện tập: đọc tài liệu hoặc thi thử. 5. Nếu chọn “Đọc tài liệu”: <ol style="list-style-type: none"> a. Học sinh chọn tài liệu. b. Học sinh có thể tải tài liệu xuống thiết bị để xem ngoại tuyến. 6. Nếu chọn “Thi thử”: <ol style="list-style-type: none"> a. Học sinh làm bài thi (chọn đáp án cho từng câu hỏi). b. Sau khi làm bài, học sinh có thể chọn “Nộp bài”. c. Hệ thống hiển thị kết quả thi thử.

Alternative Flow	<p>5a. Học sinh chỉ đọc tài liệu mà không tải xuống.</p> <p>5a1. Tài liệu hiển thị trực tiếp trên giao diện hệ thống.</p>
Exception Flow	<p>3a. Học sinh không chọn được kỳ thi do không có dữ liệu.</p> <p>3a1. Hệ thống hiển thị thông báo lỗi và đề nghị quay lại sau.</p> <p>6a. Mất kết nối khi đang làm bài thi thử.</p> <p>6a1. Hệ thống lưu tạm thời trạng thái bài làm.</p> <p>6a2. Khi kết nối lại, hệ thống cho phép tiếp tục từ câu đã làm.</p>



Hình 3.7: Mô hình UC học sinh tra cứu kết quả thi và phức tra

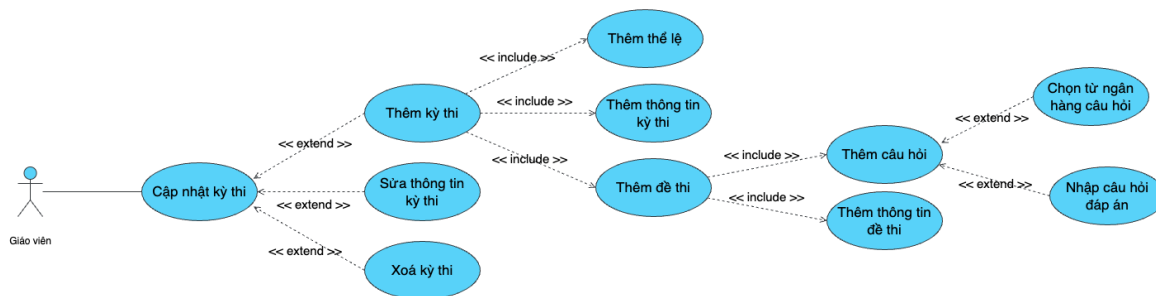
Use Case Name	Tra cứu điểm thi (Hình 3.7)
Description	Học sinh muốn tra cứu điểm thi của mình để biết kết quả thi theo đợt và khối đã chọn.

Actor(s)	Học sinh
Trigger	Học sinh muốn xem điểm thi sau kỳ thi
Pre-Condition(s)	<ul style="list-style-type: none"> • Học sinh đã đăng nhập hệ thống. • Dữ liệu điểm thi đã được cập nhật.
Post-Condition(s)	<ul style="list-style-type: none"> • Học sinh xem được điểm thi theo thông tin đã chọn.
Basic Flow	<ol style="list-style-type: none"> 1. Học sinh truy cập chức năng “Tra cứu điểm thi”. 2. Học sinh chọn hiển thị theo đợt thi hoặc khối. 3. Hệ thống hiển thị kết quả điểm thi tương ứng.

Use Case Name	Phúc tra (Hình 3.7)
Description	Học sinh muốn gửi đơn phúc tra bài thi nếu có thắc mắc về kết quả, nhưng chỉ được gửi một lần duy nhất.
Actor(s)	Học sinh
Trigger	Học sinh không đồng ý với kết quả điểm thi và muốn yêu cầu phúc tra.
Pre-Condition(s)	<ul style="list-style-type: none"> • Học sinh đã tra cứu điểm thi. • Hệ thống đang mở chức năng phúc tra. • Học sinh chưa gửi đơn phúc tra nào cho bài thi đó.

Post-Condition(s)	<ul style="list-style-type: none"> • Đơn phúc tra được gửi thành công và không thể gửi lại. • Hệ thống lưu đơn phúc tra và chuyển đến bộ phận xử lý.
Basic Flow	<ol style="list-style-type: none"> 1. Học sinh chọn chức năng “Phúc tra”. 2. Hệ thống kiểm tra xem học sinh đã gửi đơn phúc tra chưa. 3. Nếu chưa, hệ thống yêu cầu điền thông tin bài thi. 4. Học sinh gửi đơn phúc tra. 5. Hệ thống xác nhận đã tiếp nhận đơn phúc tra và khóa chức năng phúc tra cho bài thi đó.
Alternative Flow	<p>2a. Học sinh đã gửi đơn phúc tra trước đó.</p> <p>2a1. Hệ thống thông báo: “Bạn đã gửi đơn phúc tra. Không thể gửi lại.”</p> <p>2a2. Use Case dừng tại đây.</p>
Exception Flow	<p>3a. Học sinh điền thiếu thông tin bài thi.</p> <p>3a1. Hệ thống hiển thị cảnh báo và yêu cầu bổ sung.</p>

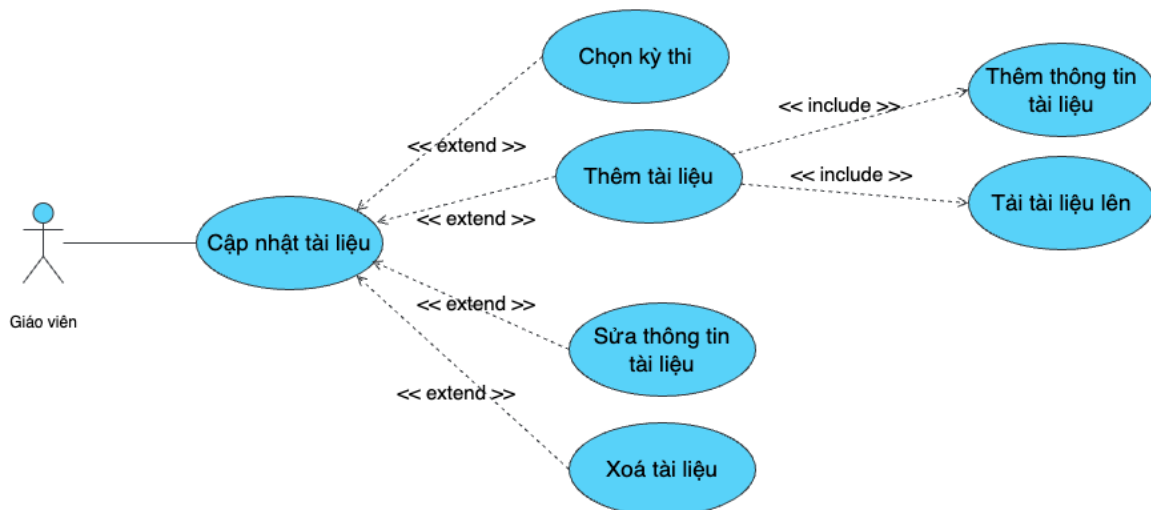
3.1.3 Mô hình UC của giáo viên



Hình 3.8: Mô hình UC giáo viên cập nhật kỳ thi

Use Case Name	Cập nhật kỳ thi (Hình 3.8)
Description	Giáo viên muốn thêm, sửa hoặc xóa kỳ thi, đồng thời cập nhật thông tin, đề thi và câu hỏi liên quan.
Actor(s)	Giáo viên
Trigger	Giáo viên truy cập chức năng quản lý kỳ thi từ hệ thống.
Pre-Condition(s)	<ul style="list-style-type: none"> Giáo viên đã đăng nhập vào hệ thống. Giáo viên có quyền thao tác với kỳ thi.
Post-Condition(s)	<ul style="list-style-type: none"> Kỳ thi được tạo mới, chỉnh sửa hoặc xóa thành công. Thông tin đề thi và câu hỏi được cập nhật nếu có.

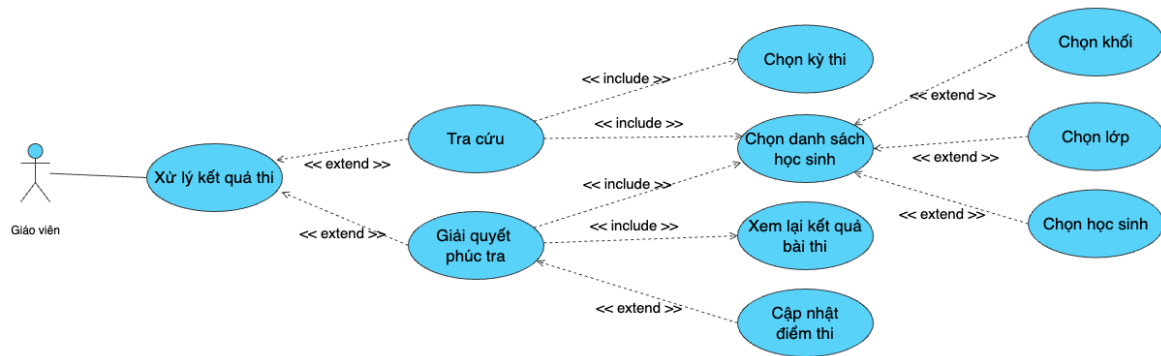
Basic Flow	<ol style="list-style-type: none"> 1. Giáo viên chọn chức năng “Cập nhật kỳ thi”. 2. Giáo viên chọn thao tác: Thêm, Sửa hoặc Xóa kỳ thi. 3. Nếu chọn “Thêm kỳ thi”: <ul style="list-style-type: none"> • Nhập thông tin kỳ thi. • Nhập thể lệ kỳ thi (nếu có). • Thêm đề thi (tùy chọn). • Với đề thi: nhập thông tin đề thi và thêm câu hỏi. • Với câu hỏi: chọn từ ngân hàng hoặc nhập mới. 4. Nếu chọn “Sửa thông tin kỳ thi”: Hệ thống hiển thị thông tin kỳ thi để chỉnh sửa. 5. Nếu chọn “Xóa kỳ thi”: Hệ thống yêu cầu xác nhận trước khi xóa.
Alternative Flow	<p>3a. Giáo viên không nhập đủ thông tin kỳ thi.</p> <p>3a1. Hệ thống yêu cầu nhập bổ sung.</p> <p>5a. Xác nhận xóa bị hủy.</p> <p>5a1. Use Case kết thúc mà không xóa kỳ thi.</p>
Exception Flow	<p>2a. Giáo viên chọn thao tác không hợp lệ.</p> <p>2a1. Hệ thống hiển thị thông báo lỗi và yêu cầu chọn lại thao tác.</p>



Hình 3.9: Mô hình UC giáo viên cập nhật tài liệu

Use Case Name	Cập nhật tài liệu (Hình 3.9)
Description	Giáo viên muốn thêm, sửa hoặc xóa tài liệu liên quan đến các kỳ thi để học sinh có thể tham khảo và ôn tập.
Actor(s)	Giáo viên
Trigger	Giáo viên truy cập chức năng cập nhật tài liệu từ hệ thống.
Pre-Condition(s)	<ul style="list-style-type: none"> Giáo viên đã đăng nhập và có quyền cập nhật tài liệu. Danh sách kỳ thi đã được tạo sẵn.
Post-Condition(s)	<ul style="list-style-type: none"> Tài liệu được thêm mới, chỉnh sửa hoặc xóa thành công. Tài liệu được liên kết với kỳ thi tương ứng.

Basic Flow	<ol style="list-style-type: none"> 1. Giáo viên chọn chức năng “Cập nhật tài liệu”. 2. Giáo viên chọn thao tác: Thêm, Sửa hoặc Xoá tài liệu. 3. Nếu chọn “Thêm tài liệu”: <ul style="list-style-type: none"> • Chọn kỳ thi tương ứng. • Nhập thông tin tài liệu. • Tải tệp tài liệu lên. 4. Nếu chọn “Sửa thông tin tài liệu”: Hệ thống hiển thị danh sách tài liệu để chỉnh sửa. 5. Nếu chọn “Xoá tài liệu”: Hệ thống yêu cầu xác nhận xoá.
Alternative Flow	<p>3a. Giáo viên không chọn kỳ thi trước khi thêm tài liệu.</p> <p>3a1. Hệ thống yêu cầu chọn kỳ thi trước khi tiếp tục.</p> <p>5a. Hủy thao tác xoá tài liệu.</p> <p>5a1. Use Case kết thúc, không xoá tài liệu.</p>
Exception Flow	<p>3b. File tài liệu không hợp lệ (định dạng không đúng hoặc vượt quá dung lượng).</p> <p>3b1. Hệ thống hiển thị thông báo lỗi và yêu cầu chọn lại.</p>



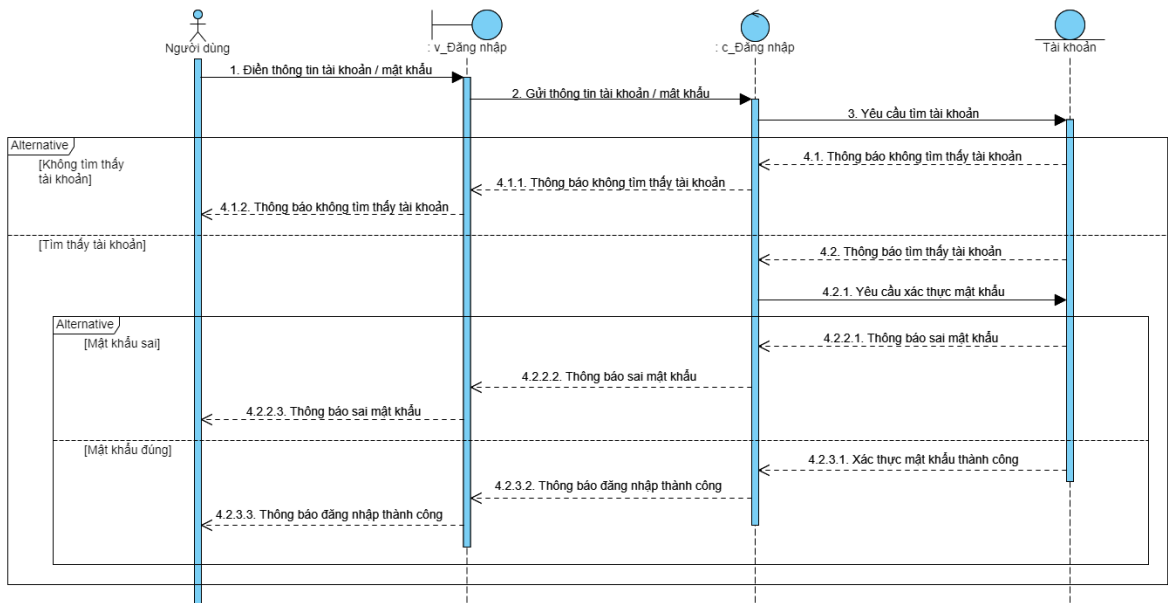
Hình 3.10: Mô hình UC giáo viên xử lý kết quả thi

Use Case Name	Xử lý kết quả thi (Hình 3.10)
Description	Là giáo viên, tôi muốn tra cứu và xử lý kết quả thi của học sinh để đảm bảo tính chính xác và công bằng trong quá trình thi cử.
Actor(s)	Giáo viên
Trigger	Giáo viên muốn kiểm tra, xác nhận hoặc điều chỉnh kết quả thi.
Pre-Condition(s)	<ul style="list-style-type: none"> Giáo viên đã đăng nhập vào hệ thống. Các kỳ thi đã được tạo và có dữ liệu học sinh dự thi.
Post-Condition(s)	<ul style="list-style-type: none"> Giáo viên xem, cập nhật hoặc giải quyết phúc tra thành công. Kết quả thi được cập nhật chính xác trên hệ thống.

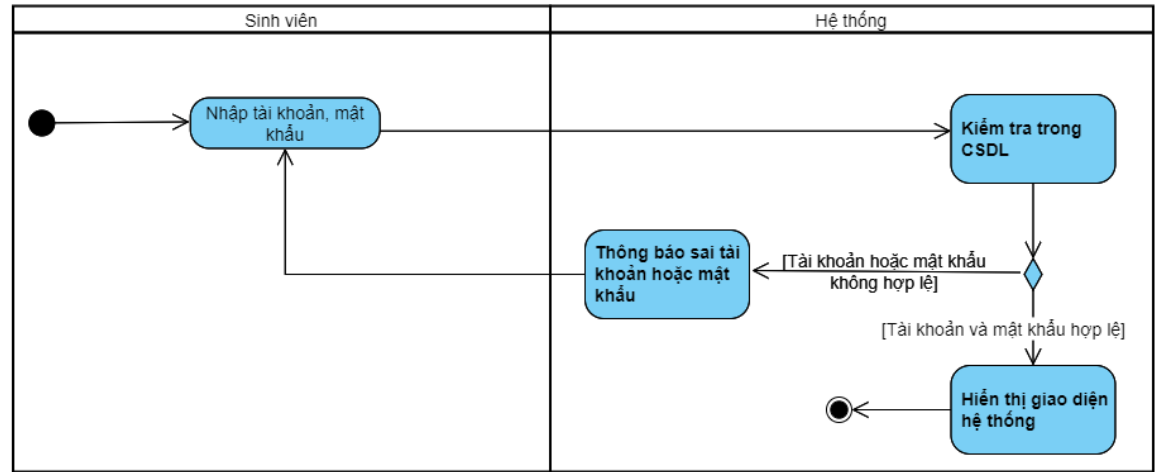
Basic Flow	<ol style="list-style-type: none"> 1. Giáo viên chọn chức năng “Xử lý kết quả thi”. 2. Hệ thống cung cấp tùy chọn “Tra cứu” hoặc “Giải quyết phúc tra”. (<i>Extend</i>) 3. Nếu chọn “Tra cứu”: <ul style="list-style-type: none"> • Chọn kỳ thi. (<i>Include</i>) • Chọn danh sách học sinh. (<i>Include</i>) <ul style="list-style-type: none"> – Chọn khối / lớp / học sinh cụ thể. (<i>Extend</i>) 4. Nếu chọn “Giải quyết phúc tra”: <ul style="list-style-type: none"> • Xem lại kết quả bài thi. (<i>Include</i>) • Cập nhật điểm thi nếu có điều chỉnh. (<i>Extend</i>)
Alternative Flow	<p>3a. Hệ thống không tìm thấy dữ liệu học sinh cho kỳ thi đã chọn.</p> <p>3a1. Hiện thị thông báo lỗi và yêu cầu kiểm tra lại kỳ thi hoặc danh sách lớp.</p> <p>4a. Giáo viên không cập nhật điểm sau khi xem lại kết quả.</p> <p>4a1. Hệ thống lưu lại trạng thái "đã xem" và không thay đổi điểm.</p>
Exception Flow	<p>2a. Giáo viên chọn sai kỳ thi hoặc chưa chọn kỳ thi.</p> <p>2a1. Hệ thống yêu cầu chọn kỳ thi hợp lệ để tiếp tục.</p>

3.2 Mô hình động

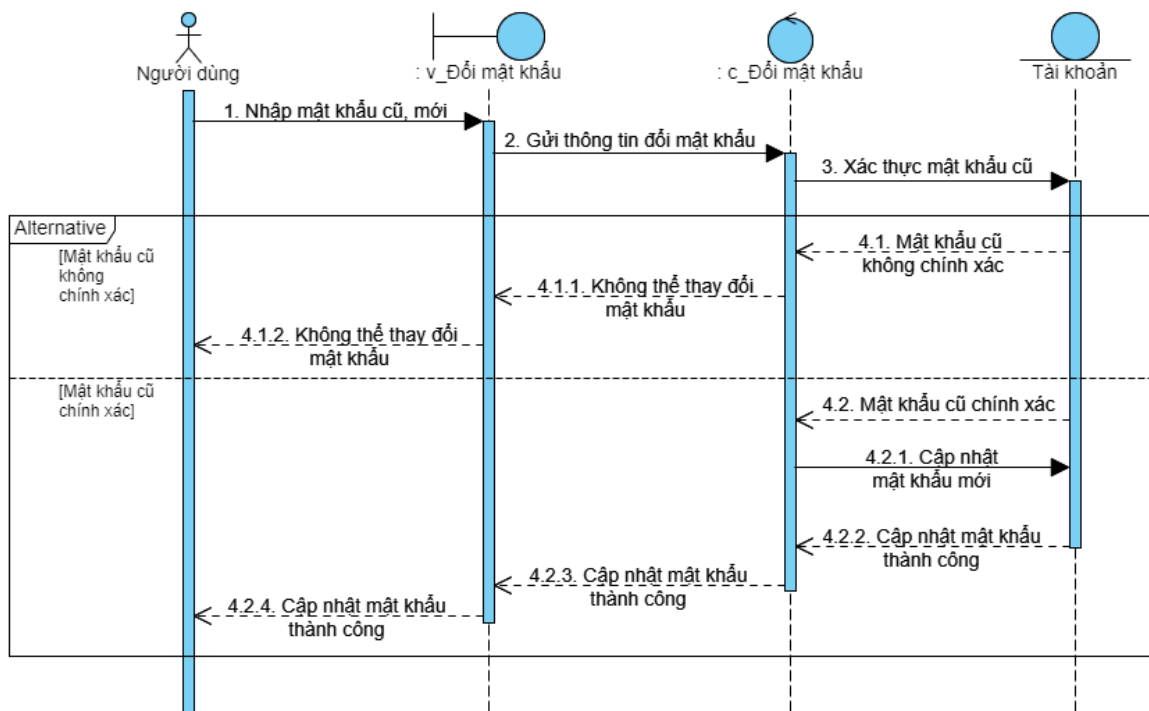
3.2.1 Hệ thống



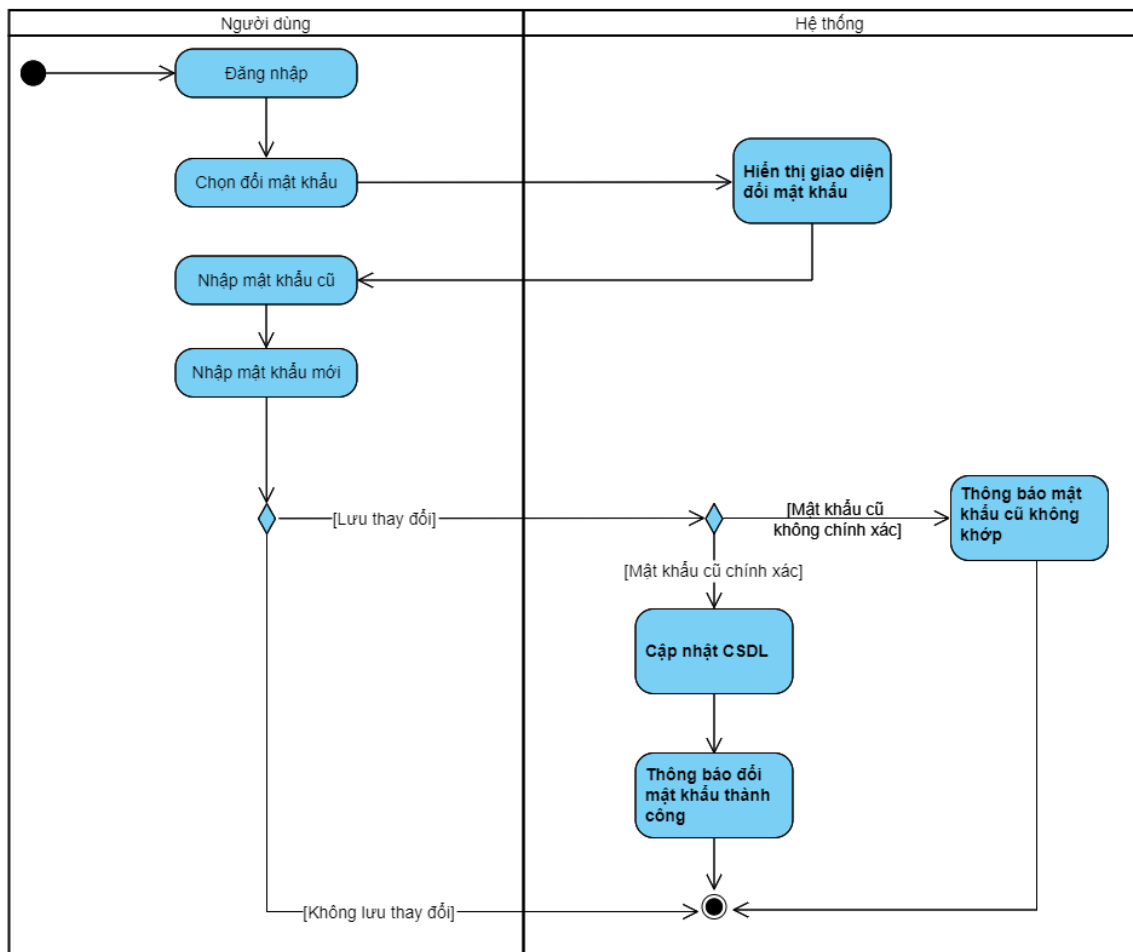
Hình 3.11: Sơ đồ tuần tự người dùng đăng nhập hệ thống



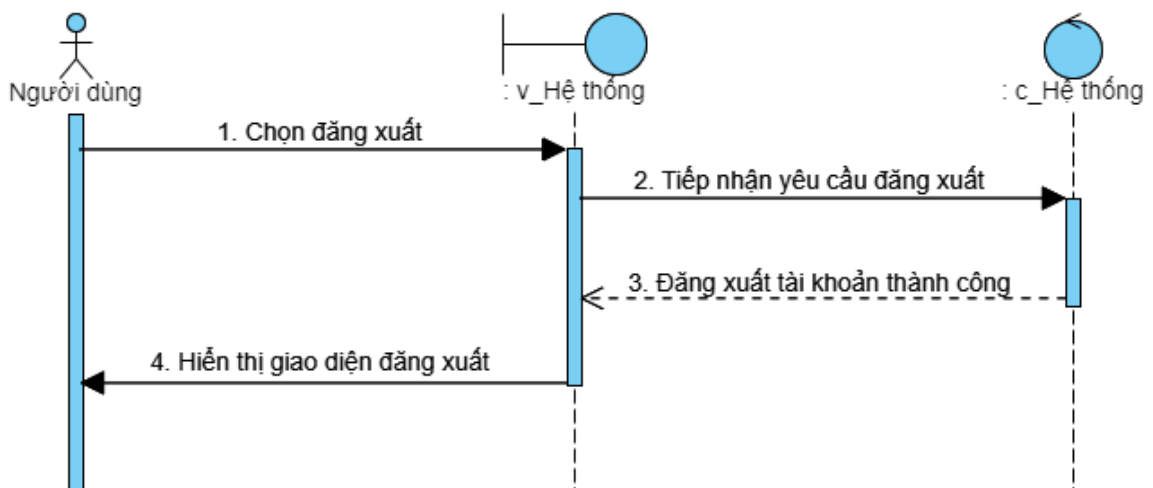
Hình 3.12: Sơ đồ hoạt động người dùng đăng nhập hệ thống



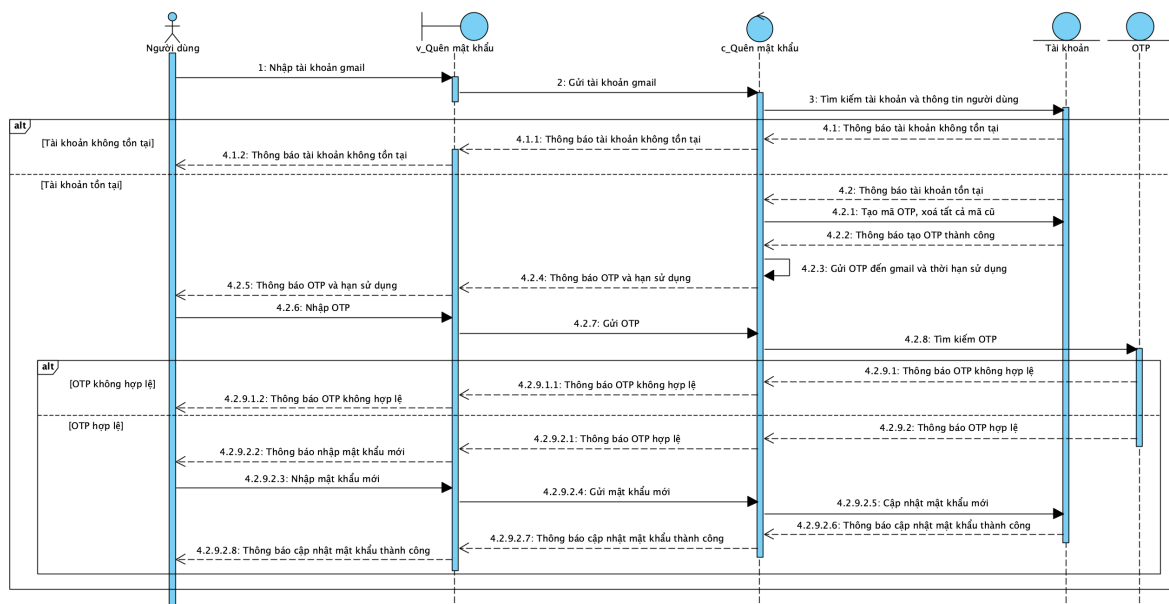
Hình 3.13: Sơ đồ tuần tự người dùng thay đổi mật khẩu



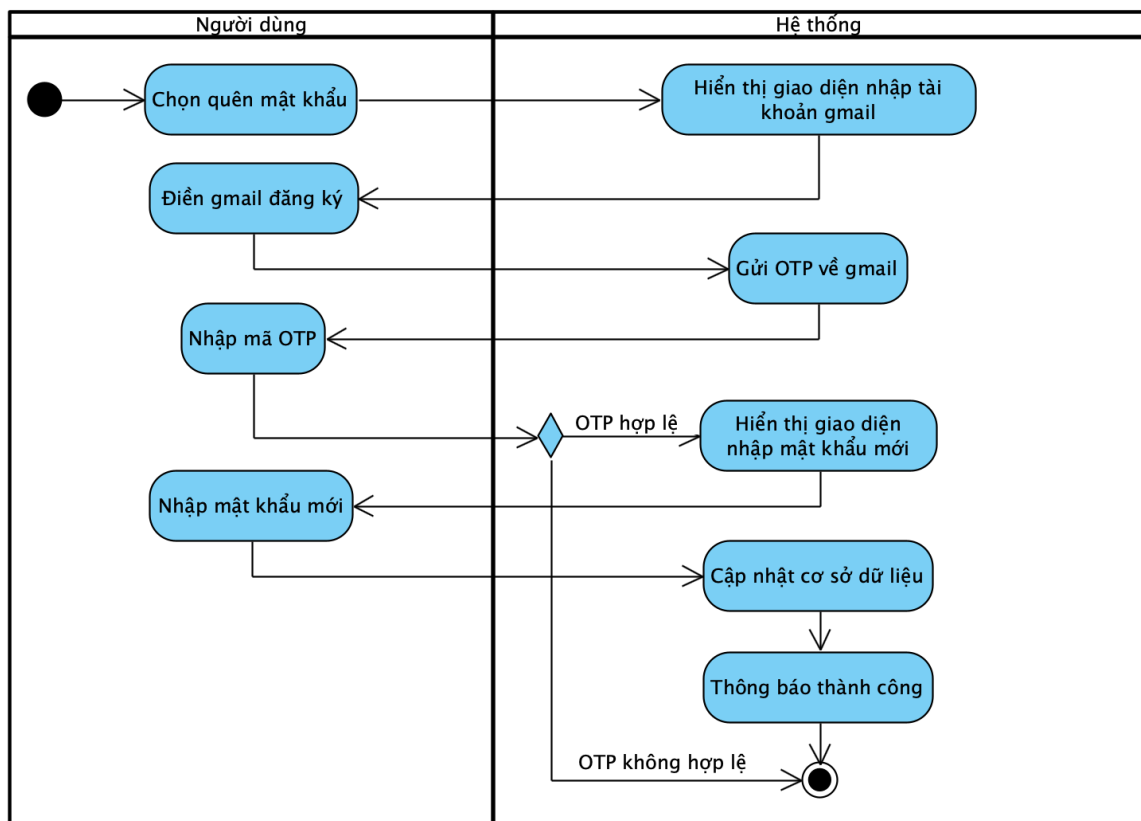
Hình 3.14: Sơ đồ hoạt động người dùng thay đổi mật khẩu



Hình 3.15: Sơ đồ tuần tự người dùng đăng xuất

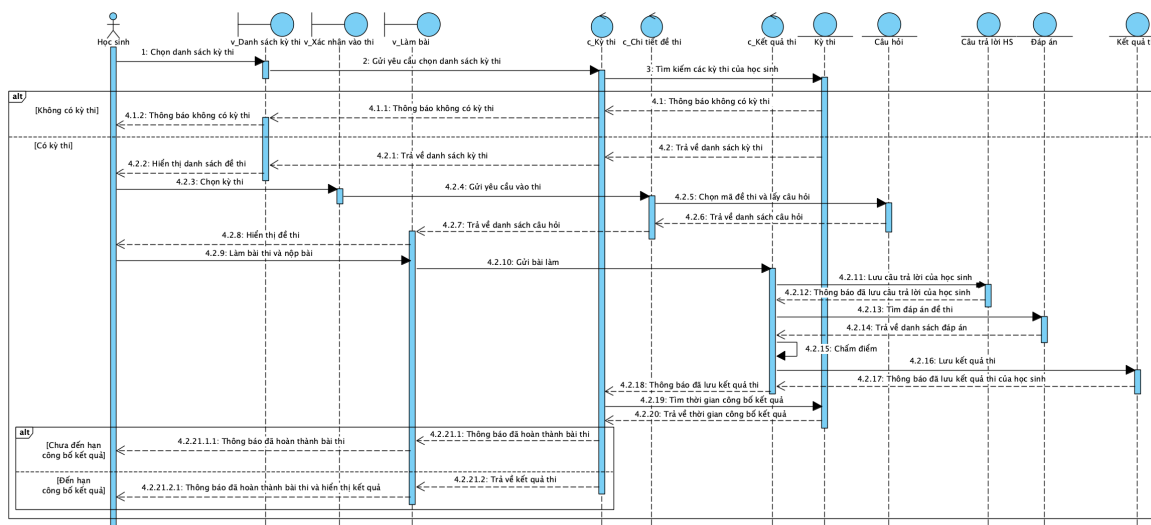


Hình 3.16: Sơ đồ tuần tự người dùng quên mật khẩu



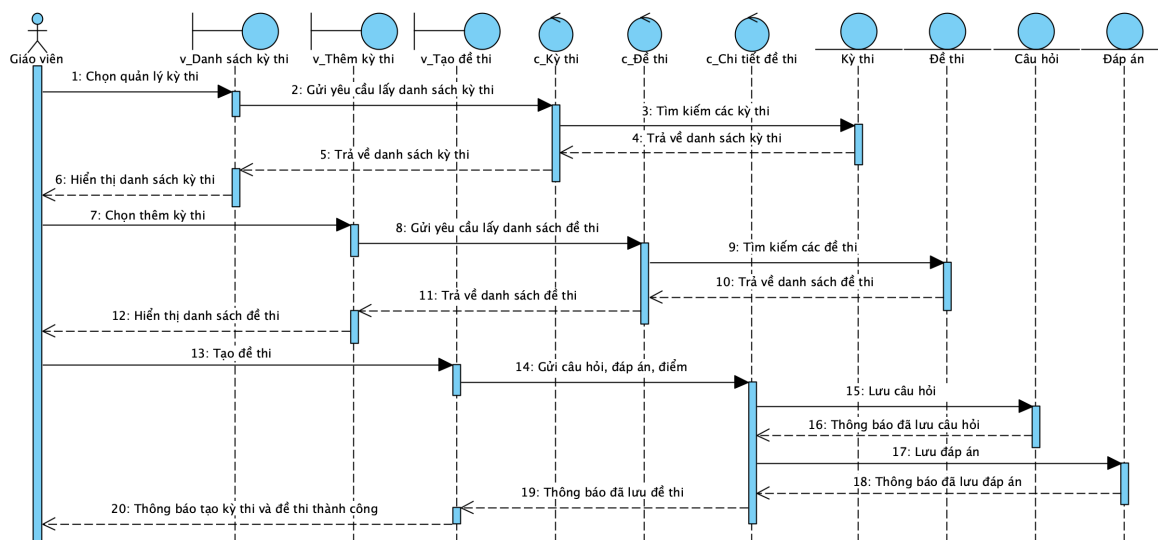
Hình 3.17: Sơ đồ hoạt động người dùng quên mật khẩu

3.2.2 Học sinh



Hình 3.18: Sơ đồ tuần tự học sinh vào thi

3.2.3 Giáo viên



Hình 3.19: Sơ đồ tuần tự giáo viên tạo kỳ thi

3.3 Sơ đồ lớp

Chương 4

Thiết kế hệ thống

4.1 Sơ đồ luồng dữ liệu

4.2 Mô hình thực thể liên kết

4.3 Danh sách các bảng

Chi tiết các bảng trong csdl

4.4 Sơ đồ dữ liệu quan hệ

Chương 5

Triển khai và thử nghiệm hệ thống

5.1 Khảo sát và thu thập dữ liệu

5.2 Giao diện hệ thống

5.2.1 Giao diện Học sinh

5.2.2 Giao diện Giáo viên

5.2.3 Giao diện Admin

Kết luận

Đồ án đã đạt được mục tiêu đề ra

Kết quả của đồ án

- 1.
- 2.

Kỹ năng đạt được

- 1.
- 2.
- 3.
- 4.
- 5.

Hướng phát triển của đồ án trong tương lai

Phụ lục A

MÃ NGUỒN CHƯƠNG TRÌNH

Nội dung mã nguồn ở đây...

Phụ lục B

HƯỚNG DẪN SỬ DỤNG SẢN PHẨM THI CÔNG

Nội dung hướng dẫn sử dụng ở đây...

Chỉ mục

use case, 10

Tài liệu tham khảo

- [1] Trung tâm Giáo dục thường xuyên Hải Phòng, “Báo cáo về chuyển đổi số trong giáo dục,” [Online]. Có sẵn tại: <http://ttgdtxhaiphong.haiphong.edu.vn/chuyen-doi-so/bao-cao-ve-chuyen-doi-so-trong-giao-duc/ct/16746/49079>.
- [2] Tạp chí Tài chính, “Chuyển đổi số trong giáo dục: Chủ động bắt nhịp, tạo bước đột phá,” 2023. [Online]. Có sẵn tại: <https://tapchitaichinh.vn/chuyen-doi-so-trong-giao-duc-chu-dong-bat-nhip-tao-buoc-dot-pha.html>.
- [3] Bộ Giáo dục và Đào tạo, “Chương trình giáo dục phổ thông mới,” 2023. [Online]. Có sẵn tại: <https://moet.gov.vn/tintuc/Pages/CT-GDPT-MOI.aspx?ItemID=8421>.
- [4] ExamSoft Worldwide Inc., “Phần mềm đánh giá bảo mật cho thi cử,” [Online]. Có sẵn tại: <https://www.examsoft.com>.
- [5] The Moodle Project, “Moodle – Nền tảng học tập mã nguồn mở,” [Online]. Có sẵn tại: <https://moodle.org>.
- [6] Tạp chí Kinh tế và Dự báo, “Chuyển đổi số trong các cơ sở giáo dục nghề nghiệp tại Việt Nam: Thực trạng và giải pháp,” 2023. [Online]. Có sẵn tại: <https://kinhtevadubao.vn/chuyen-doi-so-trong-cac-co-so-giao-duc-nghe-nghiep-tai-viet-nam-thuc-trang-va-giai-phap>.