BÌA

BÌA LÓT

# MỤC LỤC

[MỤC LỤC 1](#_Toc201399399)

[LỜI NÓI ĐẦU 3](#_Toc201399400)

[DANH MỤC HÌNH VẼ, BẢNG BIỂU 4](#_Toc201399401)

[1. Danh mục hình vẽ 4](#_Toc201399402)

[2. Danh mục bảng biểu 5](#_Toc201399403)

[CHƯƠNG I: KHẢO SÁT HỆ THỐNG 6](#_Toc201399404)

[1.1 Mô tả về môi trường hoạt động 6](#_Toc201399405)

[1.2 Khảo sát bài toán 6](#_Toc201399406)

[1.3 Ưu, nhược điểm của hệ thống cũ 10](#_Toc201399407)

[1.4 Yêu cầu của đề tài 11](#_Toc201399408)

[1.5 Công cụ lập trình 12](#_Toc201399409)

[CHƯƠNG II: PHÂN TÍCH HỆ THỐNG 14](#_Toc201399410)

[2.1 Biểu đồ Use case tổng quát 14](#_Toc201399411)

[2.2 Biểu đồ Use case Phân rã 14](#_Toc201399412)

[2.2.1 Xây dựng biểu đồ Usecase Phân rã 15](#_Toc201399413)

[2.2.2 Đặc tả yêu cầu 18](#_Toc201399414)

[2.3 Biểu đồ hoạt động 32](#_Toc201399415)

[2.4 Biểu đồ trạng thái 39](#_Toc201399416)

[2.5 Biểu đồ lớp tổng quát 40](#_Toc201399417)

[2.5.1 Các lớp dự kiến tham gia vào hệ thống 41](#_Toc201399418)

[2.6 Biểu đồ tuần tự 42](#_Toc201399419)

[2.7 Biểu đồ lớp chi tiết 48](#_Toc201399420)

[2.7.1 Lớp dự kiến 48](#_Toc201399421)

[2.7.2 Xác định thuộc tính cho lớp 51](#_Toc201399422)

[2.7.3 Xác định phương thức cho lớp 54](#_Toc201399423)

[2.7.4 Xác định mối quan hệ giữa các lớp 55](#_Toc201399424)

[CHƯƠNG III: THIẾT KẾ HỆ THỐNG 58](#_Toc201399425)

[3.1 Thiết kế cơ sở dữ liệu 58](#_Toc201399426)

[3.2 Xây dựng, thiết kế biểu đồ thành phần 64](#_Toc201399427)

[3.3 Xây dựng, thiết kế biểu đồ triển khai 66](#_Toc201399428)

[3.4 Thiết kế giao diện 69](#_Toc201399429)

[CHƯƠNG IV: CÀI ĐẶT VÀ CHẠY THỬ 73](#_Toc201399430)

[4.1 Cài đặt 73](#_Toc201399431)

[4.1.1 Các công cụ cần cài đặt 73](#_Toc201399432)

[4.1.2 Các thao tác để cài đặt chương trình 73](#_Toc201399433)

[4.2 Kiểm thử 74](#_Toc201399434)

[KẾT LUẬN 76](#_Toc201399435)

# LỜI NÓI ĐẦU

Trong thời đại công nghệ số phát triển, ứng dụng công nghệ thông tin vào giáo dục, đặc biệt trong tổ chức thi cử, ngày càng trở nên quan trọng. Hệ thống thi trắc nghiệm trực tuyến không chỉ giúp tiết kiệm thời gian, giảm tải công việc cho cán bộ coi thi, mà còn đảm bảo tính chính xác, minh bạch và khách quan trong đánh giá kết quả học tập.

Tuy nhiên, trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội và nhiều cơ sở giáo dục khác vẫn áp dụng phương pháp thi truyền thống, dẫn đến tốn thời gian, dễ sai sót trong chấm điểm và khó kiểm soát quá trình thi. Xuất phát từ thực tế đó, em đã chọn và phát triển đề tài: **“Xây dựng hệ thống thi trắc nghiệm cho Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội (HPC)”** làm đồ án tốt nghiệp, nhằm nâng cao hiệu quả tổ chức thi, bảo mật và tối ưu hóa quy trình chấm điểm.

Mục tiêu của đề tài là xây dựng một hệ thống thi trắc nghiệm trực tuyến đáp ứng nhu cầu của nhà trường, giảng viên và sinh viên. Trong quá trình thực hiện, em đã tìm hiểu công nghệ phù hợp, phân tích yêu cầu, thiết kế cơ sở dữ liệu và phát triển các chức năng chính để đảm bảo hệ thống hoạt động ổn định.

Trong quá trình làm đồ án, em đã gặp nhiều khó khăn nhưng cũng học hỏi được kiến thức bổ ích về lập trình, bảo mật và quản lý dữ liệu. Em xin chân thành cảm ơn thầy **Nguyễn Hoài Linh** – Trưởng khoa Công nghệ Thông tin – người đã hướng dẫn, nhận xét và hỗ trợ em suốt quá trình thực hiện.

Em cũng xin gửi lời cảm ơn đến Ban Giám hiệu, các thầy cô khoa CNTT, các phòng ban, gia đình, bạn bè đã tạo điều kiện và động viên em. Do thời gian có hạn và kiến thức còn hạn chế, đồ án chắc chắn còn thiếu sót. Em mong nhận được những góp ý quý báu để hoàn thiện hơn trong tương lai.

Em xin chân thành cảm ơn!

|  |  |
| --- | --- |
|  | Hà Nội, ngày 01 tháng 06 năm 2025  Sinh viên  Đàm Minh Giang |

# DANH MỤC HÌNH VẼ, BẢNG BIỂU

## 1. Danh mục hình vẽ

[Hình 2.1: Usecase tổng quát 14](#_Toc201399473)

[Hình 2.2: Phân rã use case đăng nhập 15](#_Toc201399474)

[Hình 2.3: Phân rã use case quản lý tài khoản 15](#_Toc201399475)

[Hình 2.4: Phân rã use case quản lý ngân hàng câu hỏi 16](#_Toc201399476)

[Hình 2.5: Phân rã use case Quản lý danh sách thí sinh 16](#_Toc201399477)

[Hình 2.6: Phân rã use case Quản lý kết quả thi 17](#_Toc201399478)

[Hình 2.7: Phân rã use case Báo cáo thống kê 17](#_Toc201399479)

[Hình 2.8: Phân rã use case xác thực thí sinh 18](#_Toc201399480)

[Hình 2.9: Phân rã use case làm bài 18](#_Toc201399481)

[Hình 2.10: Biểu đồ hoạt động chức năng đăng nhập 32](#_Toc201399482)

[Hình 2.11: Biểu đồ hoạt động chức năng quản lý tài khoản 33](#_Toc201399483)

[Hình 2.12: Biểu đồ hoạt động chức năng quản lý ngân hàng câu hỏi 35](#_Toc201399484)

[Hình 2.13: Biểu đồ hoạt động chức năng quản lý thí sinh 36](#_Toc201399485)

[Hình 2.14: Biểu đồ hoạt động chức năng báo cáo thống kê 37](#_Toc201399486)

[Hình 2.15: Biểu đồ hoạt động chức năng xác thực thí sinh 37](#_Toc201399487)

[Hình 2.16: Biểu đồ hoạt động chức năng làm bài 38](#_Toc201399488)

[Hình 2.17: biểu đồ trạng thái chức năng đăng nhập 39](#_Toc201399489)

[Hình xx: Biểu đồ tuần tự chức năng đăng nhập 42](#_Toc201399490)

[Hình xx: Biểu đồ tuần tự chức năng quản lý tài khoản 43](#_Toc201399491)

[Hình xx: Biểu đồ tuần tự chức năng quản lý ngân hàng câu hỏi 45](#_Toc201399492)

[Hình xx: Biểu đồ tuần tự chức năng quản lý thí sinh 46](#_Toc201399493)

[Hình xx: Biểu đồ tuần tự chức năng báo cáo thống kê 47](#_Toc201399494)

[Hình xx: Biểu đồ tuần tự chức năng xác thực thí sinh 47](#_Toc201399495)

[Hình xx: Biểu đồ tuần tự chức năng làm bài 48](#_Toc201399496)

[Hình 3.1: Biểu đồ cơ sở dữ liệu 58](#_Toc201399497)

[Hình 3.2: Biểu đồ thành phần 64](#_Toc201399498)

[Hình 3.3: Biểu đồ triển khai 67](#_Toc201399499)

[Hình 3.3: Giao diện đăng nhập hệ thống 69](#_Toc201399500)

[Hình 3.4: Giao diện trang chủ 69](#_Toc201399501)

[Hình 3.5: Giao diện quản lý ngân hàng câu hỏi 70](#_Toc201399502)

[Hình 3.6: Giao diện thêm câu hỏi 70](#_Toc201399503)

[Hình 3.7: Giao diện kỳ thi 71](#_Toc201399504)

[Hình 3.8: Giao diện tạo đề thi 71](#_Toc201399505)

[Hình 3.9: Giao diện danh sách thí sinh 72](#_Toc201399506)

[Hình 3.10: Giao diện nhập/xuất danh sách thí sinh 72](#_Toc201399507)

## 2. Danh mục bảng biểu

[Bảng 2.1: Các chức năng tương tự biểu đồ phân rã Hình 2.3 15](#_Toc201399436)

[Bảng 2.2: Đặc tả usecase đăng nhập 19](#_Toc201399437)

[Bảng 2.3: Đặc tả usecase đổi mật khẩu 20](#_Toc201399438)

[Bảng 2.4: Đặc tả usecase đăng xuất 21](#_Toc201399439)

[Bảng 2.5: Đặc tả usecase Thêm tài khoản 21](#_Toc201399440)

[Bảng 2.6: Đặc tả usecase Sửa tài khoản 22](#_Toc201399441)

[Bảng 2.7: Đặc tả usecase Xoá tài khoản 23](#_Toc201399442)

[Bảng 2.8: Các use case tương đồng với quản lý tài khoản 24](#_Toc201399443)

[Bảng 2.9: Đặc tả usecase Nhập/Xuất câu hỏi 25](#_Toc201399444)

[Bảng 2.10: Đặc tả usecase Nhập/Xuất thí sinh 26](#_Toc201399445)

[Bảng 2.11: Đặc tả usecase Xem báo cáo 27](#_Toc201399446)

[Bảng 2.12: Đặc tả usecase Xuất báo cáo 28](#_Toc201399447)

[Bảng 2.13: Đặc tả usecase Xác thực thí sinh 29](#_Toc201399448)

[Bảng 2.14: Đặc tả usecase Làm bài 29](#_Toc201399449)

[Bảng 2.15: Đặc tả usecase Nộp bài 31](#_Toc201399450)

[Bảng 2.16: Các biểu đồ hoạt động tương tự quản lý tài khoản Hình 2.16 34](#_Toc201399451)

[Bảng 2.17: Các lớp dự kiến tham gia vào hệ thống 41](#_Toc201399452)

[Bảng 2.18: Các biểu đồ tuần tự tương tự chức năng quản lý tài khoản Hình 2.16 44](#_Toc201399453)

[Bảng 2.19: Lớp dự kiến 50](#_Toc201399454)

[Bảng 2.20: Thuộc tính các lớp 54](#_Toc201399455)

[Bảng 2.21: Phương thức các lớp 55](#_Toc201399456)

[Bảng 2.22: Mối quan hệ 1-1 giữa các lớp 56](#_Toc201399457)

[Bảng 2.23: Mối quan hệ 1-n giữa các lớp 56](#_Toc201399458)

[Bảng 2.24: Mối quan hệ C-A giữa các lớp 57](#_Toc201399459)

[Bảng 3.1: Thiết kế bảng taikhoan trong CSDL 59](#_Toc201399460)

[Bảng 3.2: Thiết kế bảng sinhvien trong CSDL 59](#_Toc201399461)

[Bảng 3.3: Thiết kế bảng nganh trong CSDL 59](#_Toc201399462)

[Bảng 3.4: Thiết kế bảng khoa trong CSDL 59](#_Toc201399463)

[Bảng 3.5: Thiết kế bảng monhoc trong CSDL 59](#_Toc201399464)

[Bảng 3.6: Thiết kế bảng kythi trong CSDL 60](#_Toc201399465)

[Bảng 3.7: Thiết kế bảng sobaodanh trong CSDL 60](#_Toc201399466)

[Bảng 3.8: Thiết kế bảng cauhoi trong CSDL 60](#_Toc201399467)

[Bảng 3.9: Thiết kế bảng dapan trong CSDL 61](#_Toc201399468)

[Bảng 3.10: Thiết kế bảng theloaicauhoi trong CSDL 61](#_Toc201399469)

[Bảng 3.11: Thiết kế bảng dethi trong CSDL 62](#_Toc201399470)

[Bảng 3.12: Thiết kế bảng dethicauhoi trong CSDL 62](#_Toc201399471)

[Bảng 3.13: Thiết kế bảng baithi trong CSDL 62](#_Toc201399472)

# CHƯƠNG I: KHẢO SÁT HỆ THỐNG

## 1.1 Mô tả về môi trường hoạt động

Trong bối cảnh chuyển đổi số đang diễn ra mạnh mẽ trong lĩnh vực giáo dục, việc đưa công nghệ vào giảng dạy và kiểm tra đánh giá không còn là lựa chọn, mà đã trở nên cần thiết. Nhu cầu tổ chức các kỳ thi trực tuyến ngày càng trở nên cấp thiết, đặc biệt sau những ảnh hưởng từ đại dịch Covid-19 đã cho thấy tầm quan trọng của các hệ thống thi cử online.

Hệ thống thi trắc nghiệm được xây dựng nhằm đáp ứng nhu cầu thực tiễn của Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội (HPC) trong việc tổ chức kiểm tra, đánh giá chất lượng học tập của sinh viên một cách hiệu quả, minh bạch và tiết kiệm nguồn lực. Hệ thống này không chỉ giúp giảm tải công việc cho con người mà còn nâng cao tính khách quan và khả năng truy cập linh hoạt cho người học.

Môi trường hoạt động của hệ thống là trên nền tảng mạng Internet hoặc triển khai tương tự trong mạng nội bộ của nhà trường, đảm bảo cho giảng viên và sinh viên có thể truy cập mọi lúc, mọi nơi với thiết bị phù hợp. Hệ thống được triển khai trên các thiết bị như máy tính để bàn, laptop và cả điện thoại di động có kết nối mạng, thuận tiện cho người sử dụng trong nhiều tình huống học tập và thi cử khác nhau.

Hiện tại, nhà trường đã có sẵn hạ tầng mạng ổn định, bao gồm hệ thống máy chủ và các thiết bị mạng phù hợp, đảm bảo hỗ trợ tốt cho hoạt động vận hành của hệ thống. Bên cạnh đó, các đơn vị chức năng như Khoa Công Nghệ Thông Tin và phòng Công nghệ thông tin sẽ phối hợp chặt chẽ trong công tác quản trị, bảo trì, cập nhật hệ thống cũng như hỗ trợ kỹ thuật, nhằm đảm bảo hệ thống hoạt động ổn định, bảo mật và hiệu quả lâu dài.

## 1.2 Khảo sát bài toán

Tại Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội, các kỳ thi đóng vai trò quan trọng trong việc kiểm tra và đánh giá kết quả học tập của sinh viên. Tuy nhiên, việc tổ chức thi trắc nghiệm hiện nay vẫn chủ yếu thực hiện theo phương pháp thủ công: giảng viên tự in đề, phát đề giấy, thu bài và chấm điểm bằng tay. Cách làm này bộc lộ nhiều bất cập, đặc biệt khi quy mô lớp học ngày càng lớn và nhu cầu tổ chức thi từ xa ngày càng tăng.

Những khó khăn cụ thể có thể kể đến như:

* Tốn thời gian và công sức trong khâu chuẩn bị đề thi và xử lý bài làm.
* Việc chấm điểm thủ công dễ dẫn đến sai sót, ảnh hưởng đến tính chính xác và công bằng.
* Giám sát kỳ thi không hiệu quả, dễ phát sinh gian lận.
* Không phù hợp với tình huống học tập từ xa, đặc biệt trong các trường hợp khẩn cấp như dịch bệnh.

Trước thực trạng trên, các sinh viên, nhà trường và các giảng viên có nhu cầu xây dựng một hệ thống thi trắc nghiệm trực tuyến với các yêu cầu chính:

* Hỗ trợ tạo đề thi trắc nghiệm nhanh chóng, dễ sử dụng.
* Có thể thi trực tuyến trên nhiều thiết bị (máy tính, điện thoại…).
* Tự động chấm điểm và xuất kết quả ngay sau khi thí sinh hoàn thành bài thi.
* Cung cấp các chức năng giám sát thi để hạn chế gian lận (giám sát webcam, ghi lại nhật ký hành vi, ghi lại ).
* Dễ dàng quản lý.
* Hỗ trợ thống kê, lưu trữ và truy xuất kết quả khi cần thiết.

**Phân tích chức năng hệ thống**

Dựa trên nhu cầu thực tế tại Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội và mục tiêu ứng dụng công nghệ vào quá trình tổ chức thi, hệ thống thi trắc nghiệm trực tuyến được thiết kế với các nhóm chức năng chính như sau:

**1. Xác thực và đăng nhập người dùng**

Hệ thống hỗ trợ đăng nhập cho ba loại người dùng: **Quản trị viên**, **Giáo viên**, và **Thí sinh**, mỗi loại có quy trình xác thực riêng biệt.

* **Đăng nhập dành cho Quản trị viên và Giáo viên** (qua website):
  + Nhập tên đăng nhập và mật khẩu.
  + Xác thực thông tin tài khoản trong hệ thống quản lý người dùng.
  + Sau khi đăng nhập, chuyển đến giao diện quản trị theo vai trò.
* **Xác thực và đăng nhập dành cho Thí sinh** (qua giao diện client):
  + Xác thực bằng **Mã sinh viên** và **Số báo danh**.
  + Hệ thống kiểm tra thông tin thí sinh có nằm trong danh sách dự thi, kiểm tra hay không.
  + Nếu hợp lệ và trong thời gian thi cho phép, chuyển sang giao diện trang tổng quan.

**2. Quản lý người dùng**

* Tạo và chỉnh sửa, xoá tài khoản quản trị viên, giáo viên.
* Cấp quyền sử dụng chức năng phù hợp theo vai trò.
* Khóa/mở tài khoản, thiết lập lại mật khẩu.

**3. Quản lý danh mục đào tạo**

* Quản lý thông tin các **khoa**, **ngành**, **môn học**.
* Liên kết môn học với giáo viên phụ trách để tạo đề thi tương ứng.

**4. Quản lý ngân hàng câu hỏi**

* Tạo, chỉnh sửa, xóa và phân loại câu hỏi.
* Phân loại theo chương, chủ đề, mức độ khó.

**5. Tạo và quản lý đợt thi/kiểm tra**

* Tạo mới bài thi với các thông tin:
  + Tên bài thi/kiểm tra, môn học, số câu hỏi, thời gian làm bài.
  + Chọn câu hỏi từ ngân hàng hoặc tạo đề tự động.
  + Thiết lập thời gian thi và danh sách thí sinh tham gia (gồm mã sinh viên).

**6. Tham gia làm bài thi (Thí sinh – client)**

* Sau khi đăng nhập thành công, thí sinh truy cập trang tổng quan, xem thông tin kỳ thi sắp diễn ra.
* Khi đến thời gian thi, bấm “Bắt đầu” để chuyển sang giao diện làm bài.
* Giao diện làm bài bao gồm: câu hỏi trắc nghiệm, đồng hồ đếm ngược, và các tùy chọn trả lời.
* Sau khi hoàn thành, thí sinh bấm “Nộp bài” để gửi kết quả. Hệ thống tự động chuyển sang màn hình hiển thị điểm số.

**7. Chấm điểm và hiển thị kết quả**

* Hệ thống tự động chấm điểm và lưu lại ngay sau khi thí sinh nộp bài.
* Hiển thị kết quả: điểm số, trạng thái đạt/không đạt.
* Lưu kết quả vào hệ thống để giáo viên có thể theo dõi, thống kê và xuất báo cáo.

**Phần mềm tương tự trên thị trường**

Hiện nay trên thị trường có nhiều phần mềm và nền tảng hỗ trợ thi trắc nghiệm trực tuyến như:

* **Google Forms + Microsoft Forms**: Miễn phí, dễ sử dụng nhưng thiếu tính năng giám sát và bảo mật, quản lý.
* **Quizizz, Kahoot, Socrative**: Phù hợp cho các bài kiểm tra ngắn, tương tác cao nhưng không tối ưu cho kỳ thi quy mô lớn.
* **Viettel e-Learning, Misa EMIS**: Các giải pháp trong nước có hỗ trợ tiếng Việt, nhưng thường đi kèm phí sử dụng và hạn chế về khả năng tùy chỉnh theo yêu cầu riêng của từng trường.

Tuy có nhiều phần mềm tương tự, nhưng một hệ thống được thiết kế trọng tâm vào thi cử và riêng cho nhu cầu cụ thể của Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội sẽ giúp tối ưu quy trình tổ chức thi, dễ tích hợp với hệ thống hiện có, và thuận tiện hơn trong việc quản lý, vận hành lâu dài.

## 1.3 Ưu, nhược điểm của hệ thống cũ

**Ưu điểm:**

* **Dễ triển khai:** Phương pháp thi trắc nghiệm truyền thống không đòi hỏi nhiều về công nghệ, phù hợp với những nơi chưa có hạ tầng CNTT mạnh.
* **Đơn giản, quen thuộc:** Giảng viên và sinh viên đã quen với cách tổ chức thi truyền thống, giảm bớt thời gian làm quen với hệ thống mới.
* **Kiểm soát trực tiếp:** Cán bộ coi thi có thể dễ dàng giám sát thí sinh trong phòng thi thực tế.

**Nhược điểm:**

* **Tốn thời gian và nhân lực:** Việc phát đề, thu bài, chấm điểm thủ công mất nhiều thời gian, công sức.
* **Dễ xảy ra sai sót:** Quá trình chấm điểm thủ công có thể gây sai sót trong nhập liệu, đánh giá không chính xác.
* **Khó giám sát:** Gian lận trong phòng thi khó kiểm soát hoàn toàn, đặc biệt khi số lượng thí sinh lớn.
* **Hạn chế linh hoạt:** Khó tổ chức các kỳ thi linh hoạt, đặc biệt trong điều kiện học trực tuyến hoặc cần thi từ xa.
* **Không tối ưu lưu trữ dữ liệu:** Việc quản lý hồ sơ bài thi, điểm số và thống kê kết quả còn thủ công, không hiệu quả.

Những hạn chế trên tạo ra nhu cầu cấp thiết về việc xây dựng một hệ thống thi trắc nghiệm trực tuyến nhằm khắc phục các điểm yếu, đồng thời phát huy ưu điểm, giúp nâng cao chất lượng công tác thi cử tại trường.

## 1.4 Yêu cầu của đề tài

Đề tài “Xây dựng hệ thống thi trắc nghiệm cho Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội (HPC)” hướng đến việc giải quyết những bất cập trong quá trình tổ chức thi hiện nay, đồng thời hỗ trợ nhà trường chuyển đổi sang hình thức thi trực tuyến hiện đại. Từ thực tế đó, hệ thống cần đáp ứng các yêu cầu cụ thể sau:

**1. Yêu cầu về chức năng**

* **Quản lý người dùng phân quyền rõ ràng**: Hệ thống cần phân biệt rõ ba loại người dùng gồm **quản trị viên**, **giáo viên** và **thí sinh**, mỗi nhóm có giao diện và quyền truy cập riêng.
* **Quản lý ngân hàng câu hỏi và đề thi linh hoạt**: Cho phép giáo viên tạo, chỉnh sửa, phân loại câu hỏi; đồng thời hỗ trợ trộn đề tự động khi tổ chức bài thi.
* **Tổ chức kỳ thi đầy đủ thông tin**: Hệ thống phải hỗ trợ nhập danh sách thí sinh tham gia, cấu hình thời gian thi, tạo đề, và khởi tạo bài thi theo môn học.
* **Tham gia thi trực tuyến**: Giao diện thi thân thiện với thí sinh, hỗ trợ xác thực bằng mã sinh viên và số báo danh, cho phép làm bài trên nhiều loại thiết bị.
* **Chấm điểm và hiển thị kết quả tự động**: Sau khi nộp bài, hệ thống chấm điểm và thông báo kết quả ngay lập tức.

**2. Yêu cầu về kỹ thuật**

* **Tính chính xác và ổn định cao**: Hệ thống phải xử lý chính xác điểm số và đảm bảo hoạt động ổn định trong suốt quá trình thi.
* **Hỗ trợ đa nền tảng**: Thí sinh có thể truy cập và làm bài trên máy tính, laptop hoặc điện thoại di động có kết nối Internet.
* **Giao diện đơn giản, dễ sử dụng**: Cả giáo viên và sinh viên có thể làm quen và thao tác trên hệ thống mà không cần đào tạo chuyên sâu.
* **Khả năng mở rộng và bảo trì tốt**: Hệ thống được thiết kế theo hướng module, dễ nâng cấp hoặc tích hợp thêm tính năng sau này.

**3. Yêu cầu về bảo mật và giám sát**

* **Bảo vệ dữ liệu cá nhân và kết quả thi**: Hệ thống cần mã hóa thông tin nhạy cảm, phân quyền truy cập dữ liệu chặt chẽ.
* **Hạn chế gian lận và hỗ trợ giám sát**: Hệ thống tích hợp theo dõi webcam, ghi lại nhật ký thao tác, không cho hiển thị nội dung khác trong quá trình thi.

## 1.5 Công cụ lập trình

Để xây dựng hệ thống thi trắc nghiệm cho Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội, em dự kiến sử dụng các công cụ lập trình chính như sau:

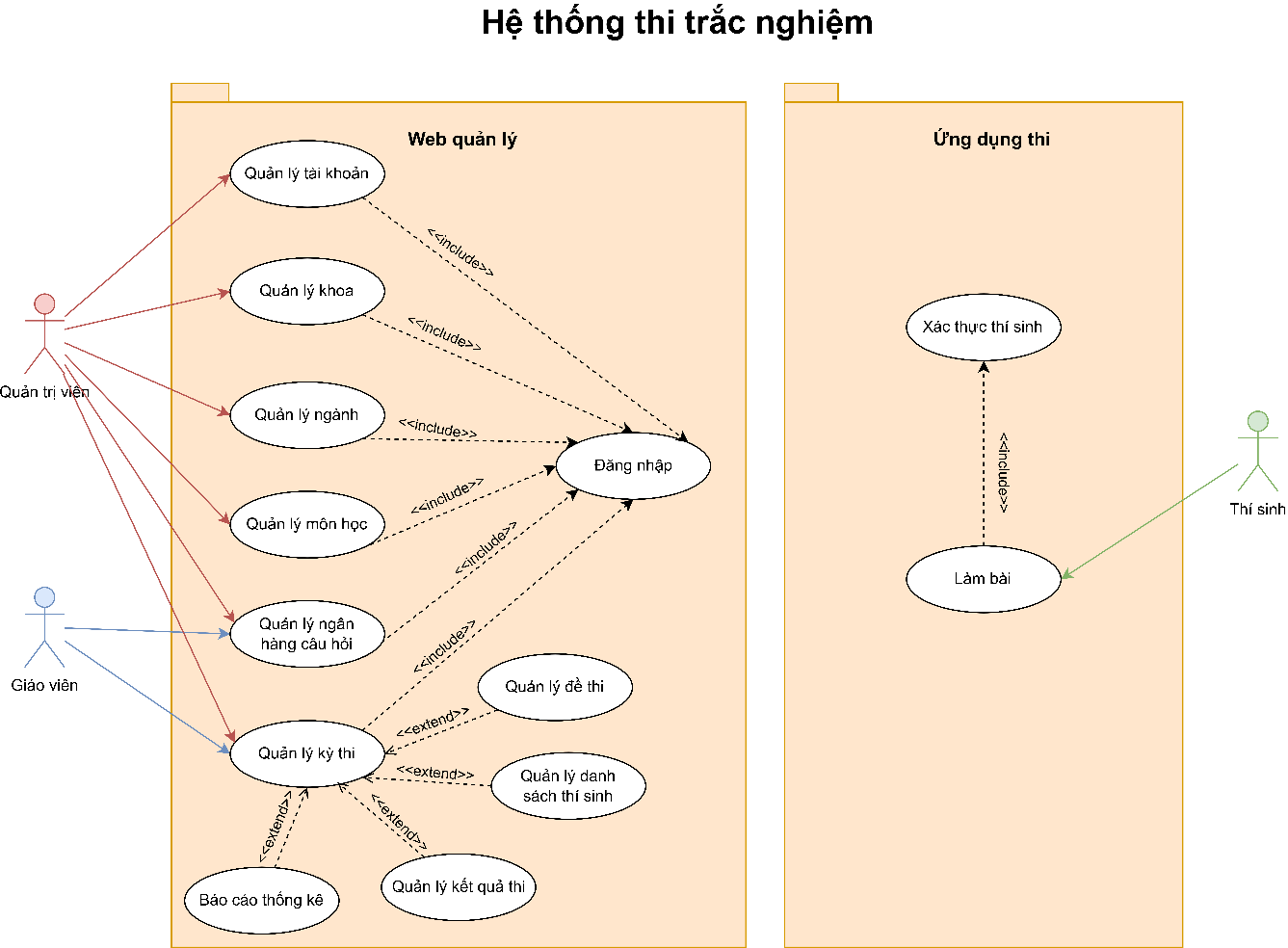
* **Server:**  
  Sử dụng ngôn ngữ lập trình **PHP** kết hợp với cơ sở dữ liệu MySQL để phát triển các chức năng xử lý phía máy chủ. PHP là công nghệ phổ biến, dễ triển khai và bảo trì, phù hợp cho việc xây dựng hệ thống web động và quản lý dữ liệu hiệu quả.
* **Client:**  
  Ứng dụng giao diện người dùng sẽ được xây dựng bằng **Flutter** — framework phát triển ứng dụng đa nền tảng của Google, cho phép xây dựng ứng dụng chạy mượt trên cả thiết bị di động (Android, iOS) và nền tảng web. Flutter giúp tạo ra giao diện đẹp, phản hồi nhanh và thân thiện với người dùng.
* **Công cụ hỗ trợ khác:**
  + Trình soạn thảo mã nguồn như Visual Studio Code, Android Studio.
  + Công cụ quản lý phiên bản Git và GitHub để kiểm soát mã nguồn.
  + Các thư viện, framework PHP và Flutter được lựa chọn phù hợp trong quá trình xây dựng để hỗ trợ phát triển nhanh và hiệu quả.

Lựa chọn PHP cho server và Flutter cho client nhằm tận dụng điểm mạnh của từng công nghệ, đồng thời đảm bảo hệ thống có hiệu suất tốt, dễ bảo trì và mở rộng trong tương lai.

# CHƯƠNG II: PHÂN TÍCH HỆ THỐNG

## 2.1 Biểu đồ Use case tổng quát

Để hình dung tổng thể cách người dùng tương tác với hệ thống thi trắc nghiệm trực tuyến, sơ đồ Use Case được xây dựng nhằm thể hiện các chức năng chính và luồng hoạt động giữa các vai trò. Hệ thống có ba tác nhân chính: **quản trị viên**, **giáo viên** và **thí sinh**, mỗi đối tượng đảm nhiệm những chức năng riêng biệt từ quản lý, tổ chức đến tham gia thi. Sơ đồ sau đây phản ánh mối quan hệ giữa các tác nhân và hệ thống trong toàn bộ quá trình triển khai một kỳ thi trực tuyến tại Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội.

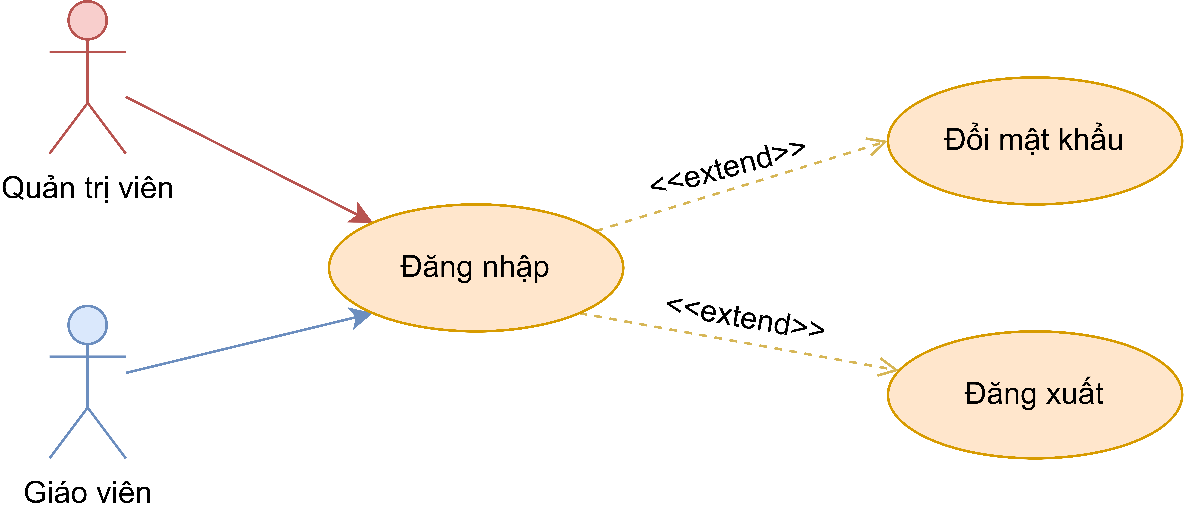


Hình 2.1: Usecase tổng quát

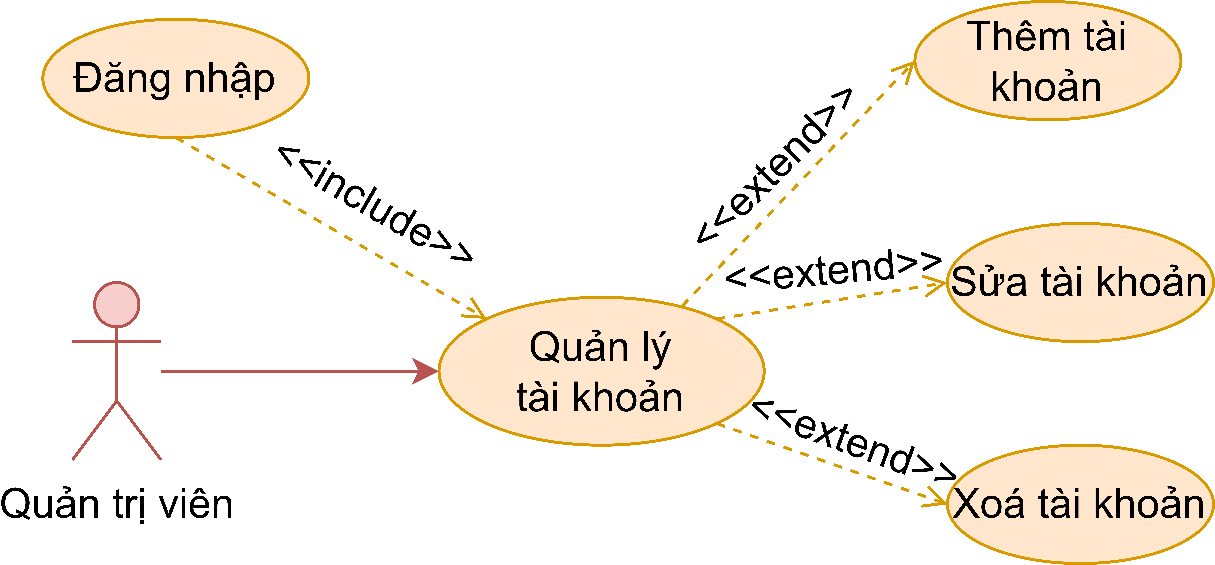
## 2.2 Biểu đồ Use case Phân rã

Từ sơ đồ Use Case tổng quát, các chức năng chính của hệ thống được phân rã chi tiết theo từng vai trò người dùng. Biểu đồ Use Case phân rã thể hiện rõ các tác vụ cụ thể mà từng tác nhân thực hiện: quản trị viên chịu trách nhiệm quản lý hệ thống và tài khoản; giáo viên tạo đề, cấu hình và giám sát kỳ thi; thí sinh thực hiện đăng nhập, làm bài và nhận kết quả. Biểu đồ này giúp làm rõ luồng tương tác trong từng phân hệ, tạo cơ sở cho việc thiết kế hệ thống một cách hợp lý và sát với yêu cầu thực tế.

### 2.2.1 Xây dựng biểu đồ Usecase Phân rã



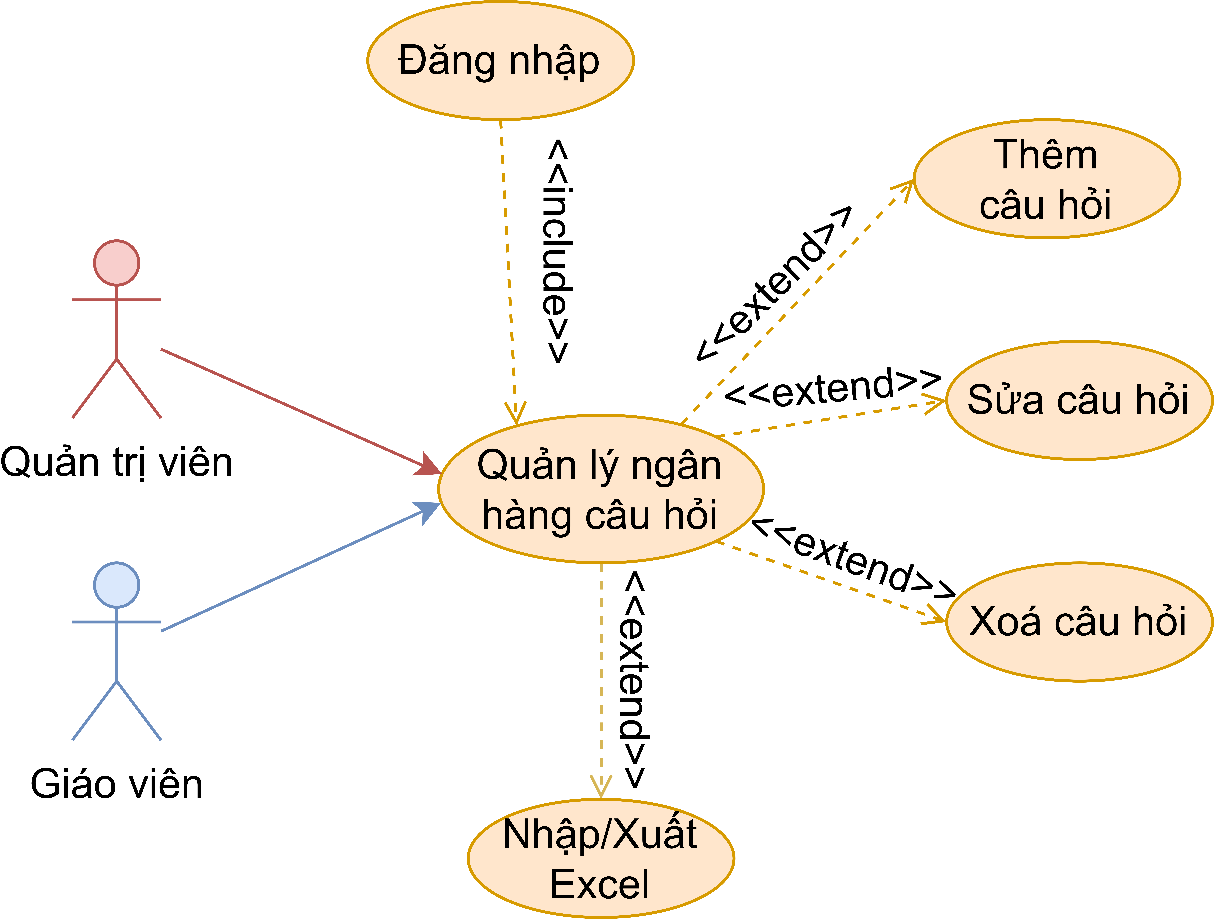
Hình 2.2: Phân rã use case đăng nhập



Hình 2.3: Phân rã use case quản lý tài khoản

|  |  |  |
| --- | --- | --- |
| **STT** | **Phân rã usecase chức năng** | **Giống với biểu đồ** |
| 1 | Quản lý khoa | Hình 2.3: Phân rã use case quản lý tài khoản |
| 2 | Quản lý ngành | Hình 2.3: Phân rã use case quản lý tài khoản |
| 3 | Quản lý môn học | Hình 2.3: Phân rã use case quản lý tài khoản |
| 4 | Quản lý kỳ thi | Hình 2.3: Phân rã use case quản lý tài khoản |
| 5 | Quản lý đề thi | Hình 2.3: Phân rã use case quản lý tài khoản |

Bảng 2.1: Các chức năng tương tự biểu đồ phân rã Hình 2.3



Hình 2.4: Phân rã use case quản lý ngân hàng câu hỏi

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.5: Phân rã use case Quản lý danh sách thí sinh

A diagram of a diagram

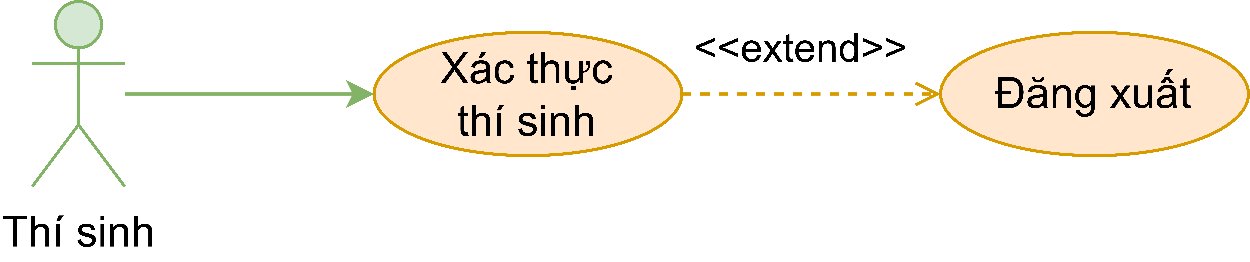
AI-generated content may be incorrect.

Hình 2.6: Phân rã use case Quản lý kết quả thi

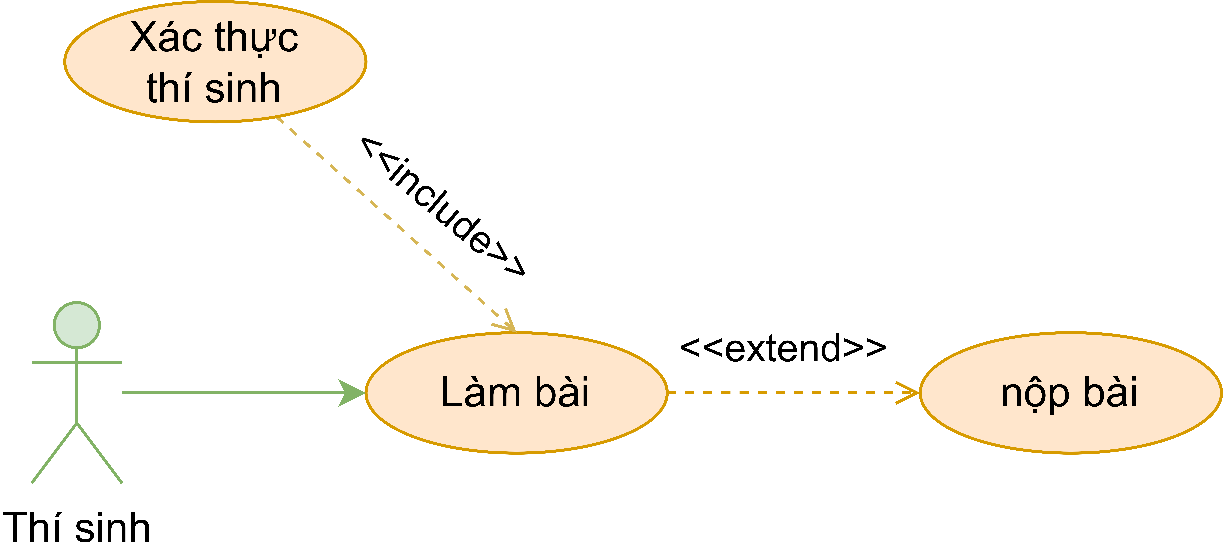
A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.7: Phân rã use case Báo cáo thống kê



Hình 2.8: Phân rã use case xác thực thí sinh



Hình 2.9: Phân rã use case làm bài

### 2.2.2 Đặc tả yêu cầu

**- Case Đăng nhập:**

|  |
| --- |
| **Usecase:** Đăng nhập. |
| **Phạm vi:** Quản trị viên, giáo viên đăng nhập vào hệ thống. |
| **Tác nhân chính:** Quản trị viên, giáo viên. |
| **Điều kiện tiên quyết:** Hệ thống hoạt động ổn định. |
| **Điều kiện thực hiện:**   * Người dùng đăng nhập vào hệ thống. * Người dùng đã có tài khoản trên hệ thống. |
| **Sự kiện kích hoạt:** Truy cập vào hệ thống khi chưa đăng nhập. |
| **Luồng sự kiên chính:**   1. Người dùng nhập thông tin đăng nhập (tên đăng nhập và mật khẩu) vào hệ thống. 2. Người dùng nhấn nút “Đăng nhập”. 3. Hệ thống kiểm tra thông tin đăng nhập và so sánh với cơ sở dữ liệu. 4. Nếu thông tin đăng nhập hợp lệ:    * Hệ thống hiển thị dữ liệu và giao diện hệ thống lên màn hình người dùng. 5. Nếu thông tin đăng nhập không hợp lệ:    * Chuyển sang Luồng phụ *A1*. |
| **Luồng phụ:**  *Luồng phụ A1:* thông tin đăng nhập không hợp lệ.   1. Hệ thống hiển thị thông báo lỗi: " Tên đăng nhập hoặc mật khẩu không đúng" trên màn hình. 2. Hệ thống yêu cầu người dùng nhập lại thông tin đăng nhập. 3. Hệ thống trả về trạng thái ban đầu, chờ người dùng nhập lại. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi không thể thực thi.   1. Hệ thống phát thông báo lỗi: "Lỗi khi thực hiện đăng nhập" (Lỗi hệ thống, mất kết nối, v.v.). 2. Hệ thống hủy thao tác, không tiếp tục xử lý. 3. Hệ thống trả về trạng thái trước đó. |

Bảng 2.2: Đặc tả usecase đăng nhập

**- Case đổi mật khẩu:**

|  |
| --- |
| **Usecase:** Đổi thông tin & mật khẩu. |
| **Phạm vi:** Quản trị viên, giáo viên cần đổi mật khẩu. |
| **Tác nhân chính:** Quản trị viên, giáo viên. |
| **Điều kiện tiên quyết:** Hệ thống hoạt động ổn định. |
| **Điều kiện thực hiện:** Người dùng đã đăng nhập vào hệ thống. |
| **Sự kiện kích hoạt:** Người dùng chọn chức năng "Đổi mật khẩu" trên giao diện hệ thống. |
| **Luồng sự kiện chính:**   1. Người dùng click vào dropdown user. 2. Người dùng chọn mục "Đổi mật khẩu mật khẩu" trên giao diện hệ thống. 3. Người dùng nhập thông tin mật khẩu cũ và mới sau đó nhấn nút "Lưu thay đổi". 4. Hệ thống kiểm tra tính hợp lệ của thông tin mới:    * Đảm bảo mật khẩu mới đáp ứng yêu cầu (độ dài, ký tự đặc biệt, v.v.).    * So sánh mật khẩu hiện tại với cơ sở dữ liệu để xác thực. 5. Nếu thông tin hợp lệ:    * Hệ thống cập nhật mật khẩu mới vào cơ sở dữ liệu.    * Hệ thống hiển thị thông báo thành công: "Đổi mật khẩu thành công!". 6. Nếu thông tin không hợp lệ:    * Chuyển sang Luồng phụ *A1*. |
| **Luồng phụ:**  *Luồng phụ A1:* **Thông tin không hợp lệ hoặc xác thực thất bại.**   1. Hệ thống hiển thị thông báo lỗi: "Mật khẩu hiện tại không chính xác". 2. Hệ thống yêu cầu người dùng kiểm tra lại và nhập lại thông tin. 3. Hệ thống quay lại trạng thái trước đó, chờ người dùng chỉnh sửa. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi không thể thực thi.   1. Hệ thống phát thông báo lỗi: "Lỗi khi thực hiện đăng nhập" (Lỗi hệ thống, mất kết nối, v.v.). 2. Hệ thống hủy thao tác, không tiếp tục xử lý. 3. Hệ thống trả về trạng thái trước đó. |

Bảng 2.3: Đặc tả usecase đổi mật khẩu

**- Case Đăng xuất:**

|  |
| --- |
| **Usecase:** Đăng xuất. |
| **Phạm vi:** Quản trị viên, Giáo viên đăng xuất khỏi hệ thống. |
| **Tác nhân chính:** Quản trị viên, giáo viên. |
| **Điều kiện tiên quyết:** Người dùng đã đăng nhập. |
| **Điều kiện thực hiện:** Người dùng đã đăng nhập vào hệ thống Edudex Quiz. |
| **Sự kiện kích hoạt:** Người dùng nhấn vào chức năng "Đăng xuất" trên giao diện hệ thống. |
| **Luồng sự kiên chính:**   1. Người dùng click vào dropdown user. 2. Người dùng chọn mục "Đăng xuất". 3. Hệ thống kiểm tra yêu cầu đăng xuất. 4. Hệ thống xóa thông tin phiên đăng nhập của người dùng. 5. Hệ thống đưa người dùng trở về giao diện màn hình đăng nhập. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi không thể thực thi đăng xuất.   1. Hệ thống phát thông báo lỗi: "Lỗi khi thực hiện đăng xuất" (lỗi hệ thống, mất kết nối, v.v.). 2. Hệ thống không thực hiện đăng xuất và vẫn giữ nguyên trạng thái phiên làm việc. 3. Hệ thống trả về trạng thái trước đó, Người dùng được yêu cầu thử lại. |

Bảng 2.4: Đặc tả usecase đăng xuất

**- Case Thêm tài khoản:**

|  |
| --- |
| **Usecase:** Thêm tài khoản. |
| **Phạm vi:** Quản trị viên thêm tài khoản người dùng vào hệ thống. |
| **Tác nhân chính:** Quản trị viên. |
| **Điều kiện tiên quyết:**   * Quản trị viên đã đăng nhập vào hệ thống. * Hệ thống đang hoạt động ổn định. |
| **Điều kiện thực hiện:**   * Thông tin tài khoản hợp lệ và tên đăng nhập chưa tồn tại trong hệ thống. |
| **Sự kiện kích hoạt:** Quản trị viên chọn chức năng "Thêm tài khoản mới" trên giao diện. |
| **Luồng sự kiên chính:**   1. Quản trị viên chọn chức năng "Thêm tài khoản". 2. Hệ thống hiển thị form nhập thông tin tài khoản. 3. Quản trị viên nhập đầy đủ thông tin. 4. Nhấn nút “Thêm”. 5. Hệ thống kiểm tra tính hợp lệ và tính duy nhất của tên đăng nhập. 6. Nếu hợp lệ:    * Hệ thống lưu thông tin vào cơ sở dữ liệu.    * Hiển thị thông báo “Thêm tài khoản thành công”.    * Trở về danh sách tài khoản. |
| **Luồng phụ:** *Luồng phụ A1:*Thêm tài khoản với tên đăng nhập đã tồn tại.   1. Hệ thống hiển thị thông báo lỗi: “Tên đăng nhập đã tồn tại”. 2. Yêu cầu người dùng nhập lại tên đăng nhập khác. 3. Quay lại bước 3 của luồng chính.   *Luồng phụ A2:* Thiếu thông tin bắt buộc.   1. Hệ thống hiển thị thông báo lỗi: “Vui lòng nhập đầy đủ thông tin”. 2. Quay lại form nhập thông tin. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi hệ thống khi thêm tài khoản.   1. Hệ thống hiển thị thông báo lỗi: “Lỗi hệ thống, không thể thêm tài khoản”. 2. Hủy thao tác thêm mới. 3. Trở về trạng thái trước đó. |

Bảng 2.5: Đặc tả usecase Thêm tài khoản

**- Case Sửa tài khoản:**

|  |
| --- |
| **Usecase:** Sửa tài khoản. |
| **Phạm vi:** Quản trị viên cập nhật thông tin tài khoản. |
| **Tác nhân chính:** Quản trị viên. |
| **Điều kiện tiên quyết:**   * Quản trị viên đã đăng nhập vào hệ thống. * Hệ thống hoạt động ổn định. |
| **Điều kiện thực hiện:**   * Tài khoản cần sửa tồn tại trong hệ thống. * Quản trị viên có quyền sửa thông tin tài khoản. |
| **Sự kiện kích hoạt:** Quản trị viên chọn chức năng “Sửa” đối với một tài khoản từ danh sách tài khoản. |
| **Luồng sự kiên chính:**   1. Quản trị viên truy cập danh sách tài khoản. 2. Quản trị viên chọn tài khoản cần chỉnh sửa. 3. Hệ thống hiển thị form thông tin tài khoản hiện tại. 4. Quản trị viên cập nhật các thông tin. 5. Nhấn nút “Lưu thay đổi” để xác nhận thay đổi. 6. Hệ thống kiểm tra tính hợp lệ của dữ liệu. 7. Nếu hợp lệ:    * Hệ thống cập nhật thông tin trong cơ sở dữ liệu.    * Hiển thị thông báo “Cập nhật tài khoản thành công”.    * Quay lại danh sách tài khoản. |
| **Luồng phụ:** *Luồng phụ A1:*Dữ liệu nhập vào không hợp lệ.   1. Hệ thống hiển thị thông báo lỗi: “Thông tin không hợp lệ”. 2. Quay lại bước 4 để người dùng chỉnh sửa lại.   *Luồng phụ A2:* Thiếu thông tin bắt buộc.   1. Hệ thống hiển thị thông báo lỗi: “Vui lòng nhập đầy đủ thông tin”. 2. Quay lại form nhập thông tin. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi hệ thống khi cập nhật tài khoản.   1. Hệ thống hiển thị thông báo lỗi: “Lỗi hệ thống, không thể cập nhật tài khoản”. 2. Hủy thao tác cập nhật. 3. Quay về trạng thái trước đó. |

Bảng 2.6: Đặc tả usecase Sửa tài khoản

**- Case Xoá tài khoản:**

|  |
| --- |
| **Usecase:** Xoá tài khoản. |
| **Phạm vi:** Quản trị viên xóa tài khoản người dùng khỏi hệ thống. |
| **Tác nhân chính:** Quản trị viên. |
| **Điều kiện tiên quyết:**   * Quản trị viên đã đăng nhập vào hệ thống. * Hệ thống hoạt động ổn định. |
| **Điều kiện thực hiện:**   * Tài khoản cần xóa phải tồn tại. * Không được xóa chính tài khoản đang đăng nhập hoặc tài khoản đang liên quan đến kỳ thi đang và sắp diễn ra. |
| **Sự kiện kích hoạt:** Quản trị viên chọn thao tác “Xóa” từ danh sách tài khoản. |
| **Luồng sự kiên chính:**   1. Quản trị viên truy cập danh sách tài khoản. 2. Chọn tài khoản cần xóa. 3. Hệ thống hiển thị hộp thoại xác nhận xóa. 4. Quản trị viên xác nhận thao tác xóa. 5. Hệ thống kiểm tra điều kiện xóa. 6. Nếu hợp lệ:    * Xóa tài khoản khỏi hệ thống.    * Hiển thị thông báo “Xóa tài khoản thành công”.    * Quay lại danh sách tài khoản. |
| **Luồng phụ:** *Luồng phụ A1:*Người dùng hủy thao tác xóa.   1. Quản trị viên chọn “Hủy” tại bước xác nhận. 2. Hệ thống hủy thao tác xóa và quay lại danh sách tài khoản. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi hệ thống khi xóa tài khoản.   1. Hệ thống hiển thị thông báo lỗi: “Lỗi hệ thống, không thể xóa tài khoản”. 2. Hủy thao tác. 3. Trả về trạng thái trước đó. |

Bảng 2.7: Đặc tả usecase Xoá tài khoản

|  |  |  |
| --- | --- | --- |
| **STT** | **Đặc tả usecase chức năng** | **Giống với bảng đặc tả** |
| 1 | Thêm khoa | Bảng 2.4: Đặc tả usecase Thêm tài khoản |
| 2 | Sửa khoa | Bảng 2.5: Đặc tả usecase Sửa tài khoản |
| 3 | Xoá khoa | Bảng 2.6: Đặc tả usecase Xoá tài khoản |
| 4 | Thêm ngành | Bảng 2.4: Đặc tả usecase Thêm tài khoản |
| 5 | Sửa ngành | Bảng 2.5: Đặc tả usecase Sửa tài khoản |
| 6 | Xoá ngành | Bảng 2.6: Đặc tả usecase Xoá tài khoản |
| 7 | Thêm môn học | Bảng 2.4: Đặc tả usecase Thêm tài khoản |
| 8 | Sửa môn học | Bảng 2.5: Đặc tả usecase Sửa tài khoản |
| 9 | Xoá môn học | Bảng 2.6: Đặc tả usecase Xoá tài khoản |
| 10 | Thêm câu hỏi | Bảng 2.4: Đặc tả usecase Thêm tài khoản |
| 11 | Sửa câu hỏi | Bảng 2.5: Đặc tả usecase Sửa tài khoản |
| 12 | Xoá câu hỏi | Bảng 2.6: Đặc tả usecase Xoá tài khoản |
| 13 | Thêm kỳ thi | Bảng 2.4: Đặc tả usecase Thêm tài khoản |
| 14 | Sửa kỳ thi | Bảng 2.5: Đặc tả usecase Sửa tài khoản |
| 15 | Xoá kỳ thi | Bảng 2.6: Đặc tả usecase Xoá tài khoản |
| 16 | Thêm đề thi | Bảng 2.4: Đặc tả usecase Thêm tài khoản |
| 17 | Sửa đề thi | Bảng 2.5: Đặc tả usecase Sửa tài khoản |
| 18 | Xoá đề thi | Bảng 2.6: Đặc tả usecase Xoá tài khoản |
| 19 | Thêm thí sinh | Bảng 2.4: Đặc tả usecase Thêm tài khoản |
| 20 | Sửa thí sinh | Bảng 2.5: Đặc tả usecase Sửa tài khoản |
| 21 | Xoá thí sinh | Bảng 2.6: Đặc tả usecase Xoá tài khoản |

Bảng 2.8: Các use case tương đồng với quản lý tài khoản

**- Case Nhập/Xuất câu hỏi:**

|  |
| --- |
| **Usecase:** Nhập/Xuất câu hỏi |
| **Phạm vi:** Quản trị viên thao tác nhập hoặc xuất danh sách thí sinh từ/ra file. |
| **Điều kiện tiên quyết:**   * Người dùng đã đăng nhập hệ thống. * Hệ thống hoạt động ổn định. * Môn học đã được tạo. |
| **Điều kiện thực hiện:**   * Tệp tin nhập đúng định dạng và cấu trúc yêu cầu. |
| **Sự kiện kích hoạt:** Người dùng chọn chức năng “Nhập/Xuất câu hỏi” từ giao diện quản lý ngân hàng câu hỏi. |
| **Luồng sự kiên chính:**   1. Người dùng truy cập giao diện quản lý ngân hàng câu hỏi. 2. Chọn chức năng “Nhập/Xuất câu hỏi”. 3. Hệ thống hiển thị giao diện cho phép chọn hành động: Nhập hoặc Xuất. 4. Nếu chọn “Nhập”: 5. Người dùng chọn và tải lên tệp Excel/CSV. 6. Hệ thống kiểm tra định dạng, nội dung và tính hợp lệ của dữ liệu. 7. Nếu hợp lệ:    * Hệ thống lưu câu hỏi vào ngân hàng theo đúng môn học và thể loại.    * Thông báo: “Nhập câu hỏi thành công”. 8. Nếu chọn “Xuất”:    * Hệ thống tạo tệp Excel chứa danh sách câu hỏi của môn học/ngân hàng.    * Cung cấp liên kết để tải tệp về.    * Thông báo: “Xuất câu hỏi thành công”. |
| **Luồng phụ:**  *Luồng phụ A1:* File nhập sai định dạng hoặc thiếu dữ liệu   1. Hệ thống thông báo lỗi: “Tệp không đúng định dạng hoặc thiếu thông tin”. 2. Yêu cầu chọn lại tệp hợp lệ. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* **Lỗi hệ thống khi nhập/xuất**   1. Có lỗi khi xử lý file hoặc ghi dữ liệu vào hệ thống. 2. Hệ thống hiển thị: “Có lỗi xảy ra trong quá trình xử lý. Vui lòng thử lại sau”. 3. Hủy thao tác và trở về trạng thái ban đầu. |

Bảng 2.9: Đặc tả usecase Nhập/Xuất câu hỏi

**- Case Nhập/Xuất thí sinh:**

|  |
| --- |
| **Usecase:** Nhập/Xuất thí sinh |
| **Phạm vi:** Quản trị viên thao tác nhập hoặc xuất danh sách thí sinh từ/ra file. |
| **Tác nhân chính:** Quản trị viên, Giáo viên |
| **Điều kiện tiên quyết:**   * Người dùng đã đăng nhập hệ thống. * Hệ thống hoạt động ổn định. * Kỳ thi đã được tạo và đang trong giai đoạn chuẩn bị. |
| **Điều kiện thực hiện:**   * Tệp tin nhập đúng định dạng và cấu trúc yêu cầu. * Thí sinh chưa bị trùng mã sinh viên trong cùng kỳ thi. |
| **Sự kiện kích hoạt:** Quản trị viên chọn chức năng “Nhập/Xuất thí sinh” từ giao diện quản lý kỳ thi. |
| **Luồng sự kiên chính:**   1. Quản trị viên truy cập giao diện quản lý kỳ thi. 2. Quản trị viên chọn chức năng “Nhập/Xuất danh sách thí sinh”. 3. Hệ thống hiển thị giao diện cho phép chọn hành động: nhập hoặc xuất. 4. Nếu chọn “Nhập”:    * Quản trị viên chọn và tải lên file Excel/CSV từ máy tính.    * Hệ thống kiểm tra định dạng và dữ liệu của file.    * Nếu hợp lệ:      + Hệ thống lưu thông tin thí sinh vào kỳ thi.      + Tự động sinh số báo danh không trùng.      + Thông báo “Nhập danh sách thành công”. 5. Nếu chọn “Xuất”:    * Hệ thống tạo file Excel chứa danh sách thí sinh của kỳ thi.    * Cung cấp liên kết để tải file về. 6. Thông báo “Xuất danh sách thành công”. |
| **Luồng phụ:**  *Luồng phụ A1:* File nhập sai định dạng hoặc thiếu dữ liệu   1. Hệ thống hiển thị thông báo lỗi: “Tệp không đúng định dạng hoặc thiếu thông tin”. 2. Yêu cầu chọn lại file hợp lệ.   *Luồng phụ A2:* Mã sinh viên bị trùng trong file nhập hoặc trong kỳ thi   1. Hệ thống cảnh báo: “Một số mã sinh viên đã tồn tại”. 2. Hiển thị danh sách dòng lỗi để người dùng xử lý thủ công. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi hệ thống khi nhập xuất   1. Mã sinh viên bị trùng trong file nhập hoặc trong kỳ thi 2. Hệ thống cảnh báo: “Một số mã sinh viên đã tồn tại”. 3. Hiển thị danh sách dòng lỗi để người dùng xử lý thủ công. |

Bảng 2.10: Đặc tả usecase Nhập/Xuất thí sinh

**- Case Xem báo cáo:**

|  |
| --- |
| **Usecase:** Xem báo cáo |
| **Phạm vi:** Quản trị viên, giáo viên xem các báo cáo thống kê của kỳ thi |
| **Tác nhân chính:** Quản trị viên, Giáo viên |
| **Điều kiện tiên quyết:**   * Người dùng đã đăng nhập hệ thống * Có dữ liệu để báo cáo (bài thi đã nộp, thí sinh đã tham gia, v.v.) |
| **Điều kiện thực hiện:**   * Kỳ thi đã tồn tại * Đã có ít nhất một thí sinh nộp bài thi |
| **Sự kiện kích hoạt:** Người dùng truy cập vào một kỳ thi và chọn chức năng “Xem báo cáo” |
| **Luồng sự kiên chính:**   1. Người dùng chọn mục “Xem báo cáo” trên giao diện. 2. Hệ thống truy vấn dữ liệu và tổng hợp báo cáo. 3. Hệ thống hiển thị báo cáo trực quan bằng bảng và/hoặc biểu đồ |
| **Luồng phụ:**  *Luồng phụ A1:* Không có dữ liệu phù hợp:   1. Hệ thống không tìm thấy dữ liệu khớp với tiêu chí lọc 2. Hiển thị thông báo: “Không tìm thấy dữ liệu báo cáo phù hợp” 3. Cho phép người dùng chọn lại điều kiện lọc. |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi truy xuất dữ liệu:   1. Hệ thống gặp lỗi khi truy vấn dữ liệu hoặc tính toán thống kê 2. Hiển thị thông báo: “Không thể hiển thị báo cáo. Vui lòng thử lại sau.” 3. Hủy thao tác và quay lại giao diện chọn điều kiện |

Bảng 2.11: Đặc tả usecase Xem báo cáo

**- Case Xuất báo cáo:**

|  |
| --- |
| **Usecase:** Xuất báo cáo |
| **Phạm vi:** Quản trị viên, giáo viên in hoặc xuất báo cáo thống kê ra file |
| **Tác nhân chính:** Quản trị viên, Giáo viên |
| **Điều kiện tiên quyết:**   * Người dùng đã đăng nhập hệ thống * Đã có dữ liệu báo cáo được hiển thị trên giao diện |
| **Điều kiện thực hiện:**   * Kỳ thi đã tồn tại * Đã có ít nhất một thí sinh nộp bài thi |
| **Sự kiện kích hoạt:** Người dùng nhấn nút “Xuất báo cáo” tại màn hình xem báo cáo |
| **Luồng sự kiên chính:**   1. Người dùng nhấn nút “Xuất báo cáo”. 2. Hệ thống hiển thị lựa chọn in hoặc xuất file 3. Người dùng chọn định dạng mong muốn. 4. Hệ thống tiến hành xử lý in hoặc tạo file. 5. Nếu thành công:    * Hệ thống cung cấp liên kết tải về file    * Hiển thị thông báo: “Xuất báo cáo thành công” 6. Người dùng tải file về máy |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi khi tạo file báo cáo:   1. Hệ thống gặp lỗi khi xử lý dữ liệu hoặc tạo file 2. Hiển thị thông báo: “Xuất báo cáo thất bại. Vui lòng thử lại sau.” 3. Hủy thao tác và quay về giao diện báo cáo |

Bảng 2.12: Đặc tả usecase Xuất báo cáo

**- Case Xác thực thí sinh:**

|  |
| --- |
| **Usecase:** Xác thực thí sinh |
| **Phạm vi:** Thí sinh đăng nhập vào hệ thống tại máy client để chuẩn bị làm bài thi |
| **Tác nhân chính:** Thí sinh |
| **Điều kiện tiên quyết:**   * Kỳ thi đã được tạo và đang mở trạng thái sẵn sàng * Thí sinh đã được đăng ký và có mã sinh viên + số báo danh * Máy client có kết nối đến máy chủ hệ thống |
| **Điều kiện thực hiện:**   * Mã sinh viên và số báo danh chính xác, khớp với thông tin kỳ thi * Số báo danh chưa được sử dụng cho bài thi đã nộp |
| **Sự kiện kích hoạt:** Thí sinh truy cập ứng dụng client |
| **Luồng sự kiên chính:**   1. Thí sinh truy cập vào giao diện thi tại máy client. 2. Nhập mã sinh viên và số báo danh. 3. Nhấn nút “Xác thực”. 4. Hệ thống gửi thông tin xác thực lên máy chủ. 5. Máy chủ kiểm tra thông tin:    * Mã sinh viên và số báo danh tồn tại    * Thuộc đúng kỳ thi đang diễn ra    * Chưa thi hoặc chưa nộp bài 6. Nếu hợp lệ:    * Hệ thống hiển thị giao diện bài thi tương ứng    * Cho phép thí sinh bắt đầu làm bài 7. Nếu không hợp lệ    * Chuyển sang luồng phụ *A1* |
| **Luồng phụ:**  *Luồng phụ A1:* Thông tin không hợp lệ:   1. Mã sinh viên hoặc số báo danh sai, không tồn tại, hoặc không khớp 2. Hệ thống hiển thị thông báo: “Sai mã sinh viên hoặc số báo danh. Vui lòng kiểm tra lại.” 3. Cho phép nhập lại.   *Luồng phụ A2:* Thí sinh đã thi hoặc đã nộp bài:   1. Hệ thống phát hiện số báo danh đã thi xong hoặc đã nộp 2. Hiển thị thông báo: “Thí sinh đã hoàn thành bài thi. Không thể làm lại.” 3. Ngăn chặn truy cập |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Lỗi hệ thống hoặc mất kết nối:   1. Mất kết nối giữa client và máy chủ, hoặc hệ thống xảy ra lỗi 2. Hiển thị thông báo: “Không thể xác thực. Vui lòng thử lại sau.” 3. Cho phép thí sinh quay lại hoặc thử lại sau |

Bảng 2.13: Đặc tả usecase Xác thực thí sinh

**- Case Làm bài:**

|  |
| --- |
| **Usecase:** Làm bài |
| **Phạm vi:** Thí sinh thực hiện bài thi tại máy client |
| **Tác nhân chính:** Thí sinh |
| **Điều kiện tiên quyết:**   * Thí sinh đã được xác thực thành công * Kỳ thi đã sẵn sàng * Hệ thống và máy client hoạt động ổn định |
| **Điều kiện thực hiện:**   * Thí sinh đăng nhập đúng mã sinh viên và số báo danh * Kỳ thi đang trong thời gian cho phép làm bài |
| **Sự kiện kích hoạt:** Thí sinh bấm “bắt đầu làm bài” sau khi thí sinh được xác thực |
| **Luồng sự kiên chính:**   1. Hệ thống hiển thị giao diện làm bài: danh sách câu hỏi và các phương án lựa chọn. 2. Thí sinh lần lượt chọn đáp án cho từng câu hỏi. 3. Trong quá trình làm bài, thí sinh có thể:    * Chuyển qua lại giữa các câu hỏi    * Theo dõi thời gian làm bài 4. Hệ thống tự động lưu tạm thời đáp án định kỳ hoặc theo từng thao tác. 5. Thí sinh có thể nhấn nút “Nộp bài” khi hoàn thành sớm. 6. Khi hết thời gian:    * Hệ thống tự động khóa bài thi và chuyển sang giai đoạn nộp bài |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Thoát giao diện làm bài trước khi nộp:   1. Thí sinh buộc thoát ứng dụng hoặc gặp lỗi thiết bị. 2. Hệ thống lưu tạm thời bài làm. 3. Cho phép đăng nhập lại nếu vẫn trong thời gian thi. |

Bảng 2.14: Đặc tả usecase Làm bài

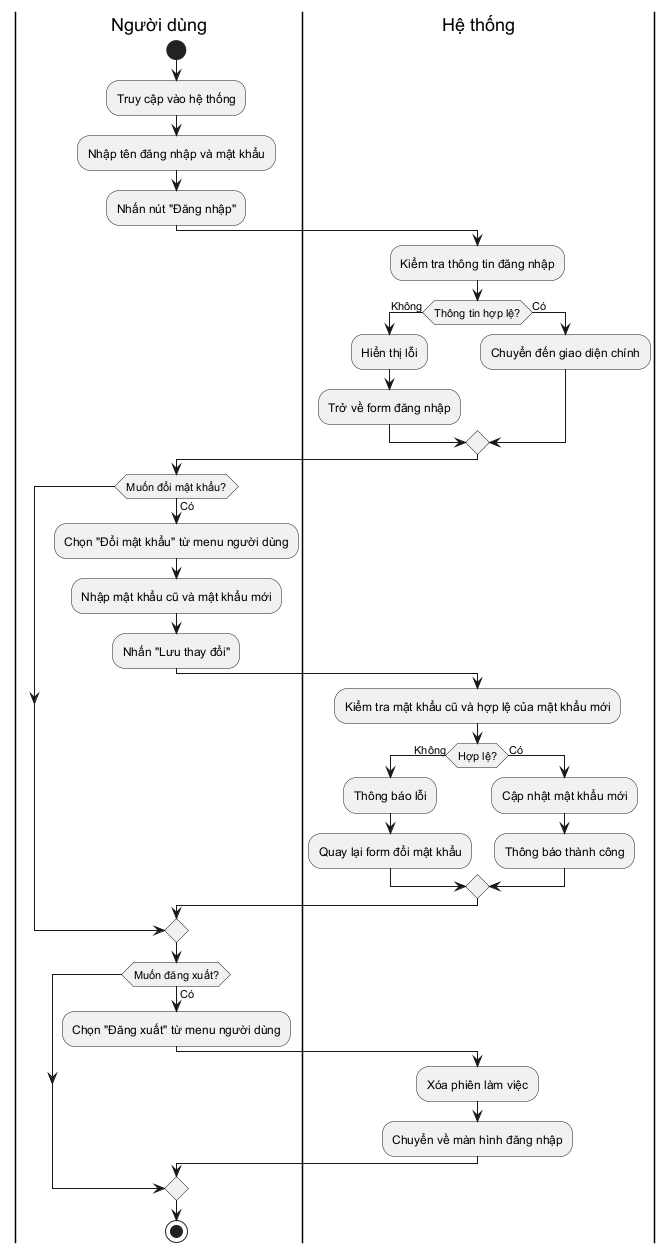
**- Case Nộp bài:**

|  |
| --- |
| **Usecase:** Nộp bài |
| **Phạm vi:** Thí sinh hoàn thành bài thi và nộp bài lên hệ thống |
| **Tác nhân chính:** Thí sinh |
| **Điều kiện tiên quyết:**   * Thí sinh đang làm bài thi và bài thi chưa được nộp * Hệ thống đang hoạt động bình thường * Thí sinh đã được xác thực và làm bài theo đúng quy trình |
| **Điều kiện thực hiện:**   * Thí sinh nhấn “Nộp bài” hoặc hết thời gian làm bài * Dữ liệu bài làm được hệ thống ghi nhận đầy đủ |
| **Sự kiện kích hoạt:** Thí sinh nhấn nút “Nộp bài” hoặc hệ thống tự động nộp bài khi hết giờ |
| **Luồng sự kiên chính:**   1. Thí sinh nhấn nút “Nộp bài” trên giao diện thi hoặc hệ thống tự động kích hoạt khi hết giờ. 2. Hệ thống hiển thị hộp thoại xác nhận: “Bạn có chắc chắn muốn nộp bài không?”. 3. Thí sinh xác nhận. 4. Hệ thống gửi toàn bộ dữ liệu bài làm lên server. 5. Server xử lý:    * So sánh đáp án thí sinh với đáp án đúng    * Tính điểm (số câu đúng, tổng điểm) 6. Server trả kết quả về client gồm:    * Số điểm đạt được    * Số câu đúng / Tổng số câu 7. Hệ thống hiển thị kết quả cho thí sinh trên giao diện 8. Hệ thống ghi nhận trạng thái bài thi là “Đã nộp” và không cho phép sửa lại |
| **Luồng phụ:**  *Luồng phụ A1:* Xác nhận nộp bài bị hủy:   1. Thí sinh chọn “Hủy” khi hệ thống hỏi xác nhận nộp bài 2. Hệ thống quay lại giao diện làm bài |
| **Luồng ngoại lệ:**  *Luồng ngoại lệ E1:* Mất kết nối khi gửi bài:   1. Mạng mất kết nối trong lúc gửi bài lên server 2. Hệ thống hiển thị thông báo: “Không thể gửi bài. Đang chờ kết nối lại…” 3. Hệ thống tự động gửi lại khi có mạng trở lại, hoặc hiển thị thông báo lỗi nếu không gửi được   *Luồng ngoại lệ E2:* Luồng ngoại lệ E2 – Lỗi tính điểm trên server:   1. Server không thể tính điểm (lỗi logic, dữ liệu thiếu, lỗi hệ thống) 2. Hiển thị thông báo: “Lỗi xử lý bài thi. Vui lòng liên hệ giám thị.” 3. Hệ thống ghi nhận bài đã nộp nhưng **không hiển thị điểm ngay** |

Bảng 2.15: Đặc tả usecase Nộp bài

## 2.3 Biểu đồ hoạt động

Biểu đồ hoạt động được sử dụng để biểu diễn luồng xử lý nghiệp vụ hoặc quá trình thực hiện một chức năng trong hệ thống. Biểu đồ mô tả các hành động, quyết định và luồng điều khiển từ lúc bắt đầu đến khi kết thúc quy trình. Qua sơ đồ này, ta có thể hình dung rõ cách các tác vụ được thực hiện tuần tự hoặc song song, cũng như các điều kiện rẽ nhánh ảnh hưởng đến quá trình hoạt động của hệ thống.



Hình 2.10: Biểu đồ hoạt động chức năng đăng nhập

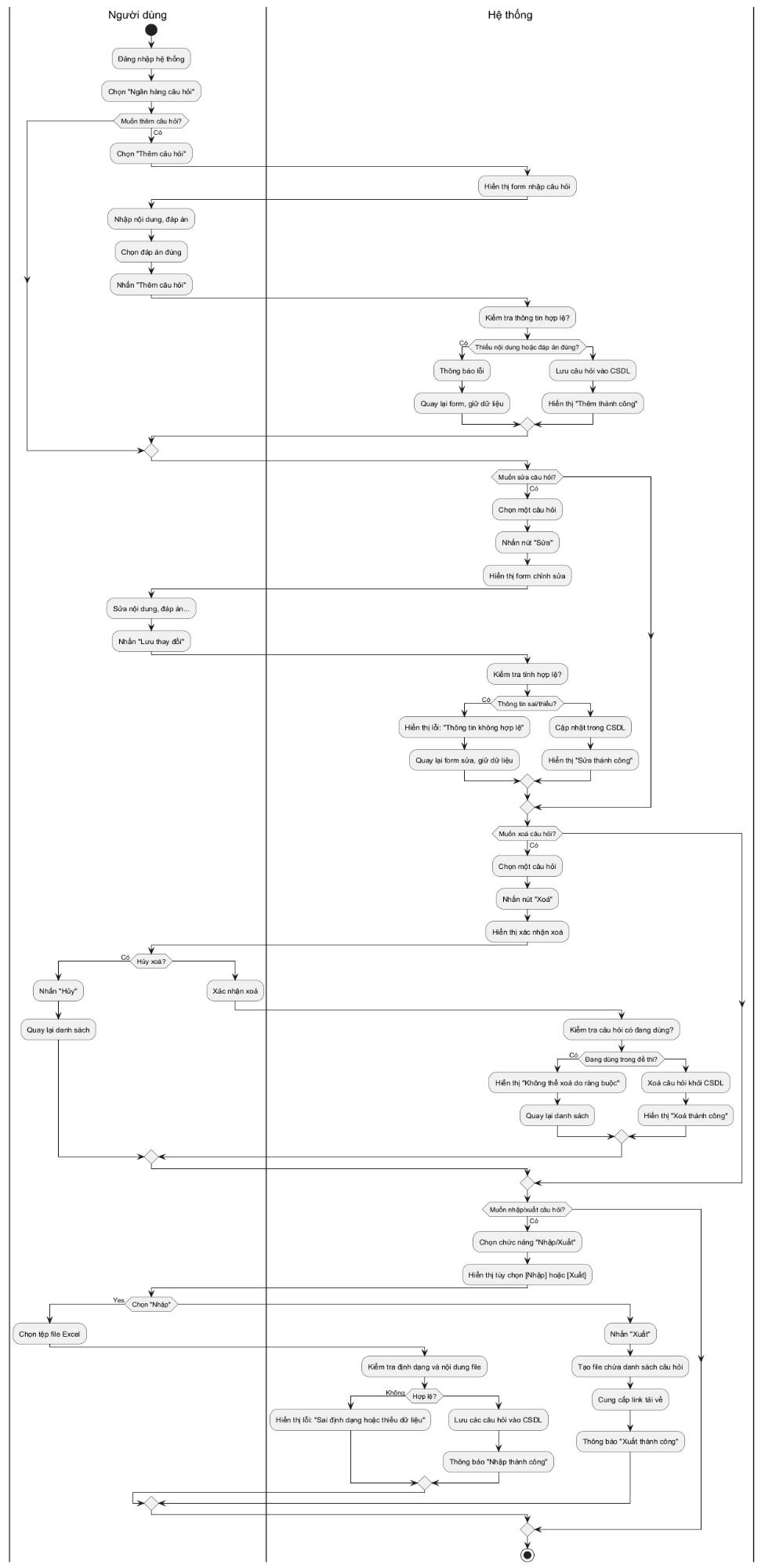
A diagram of a flowchart

AI-generated content may be incorrect.

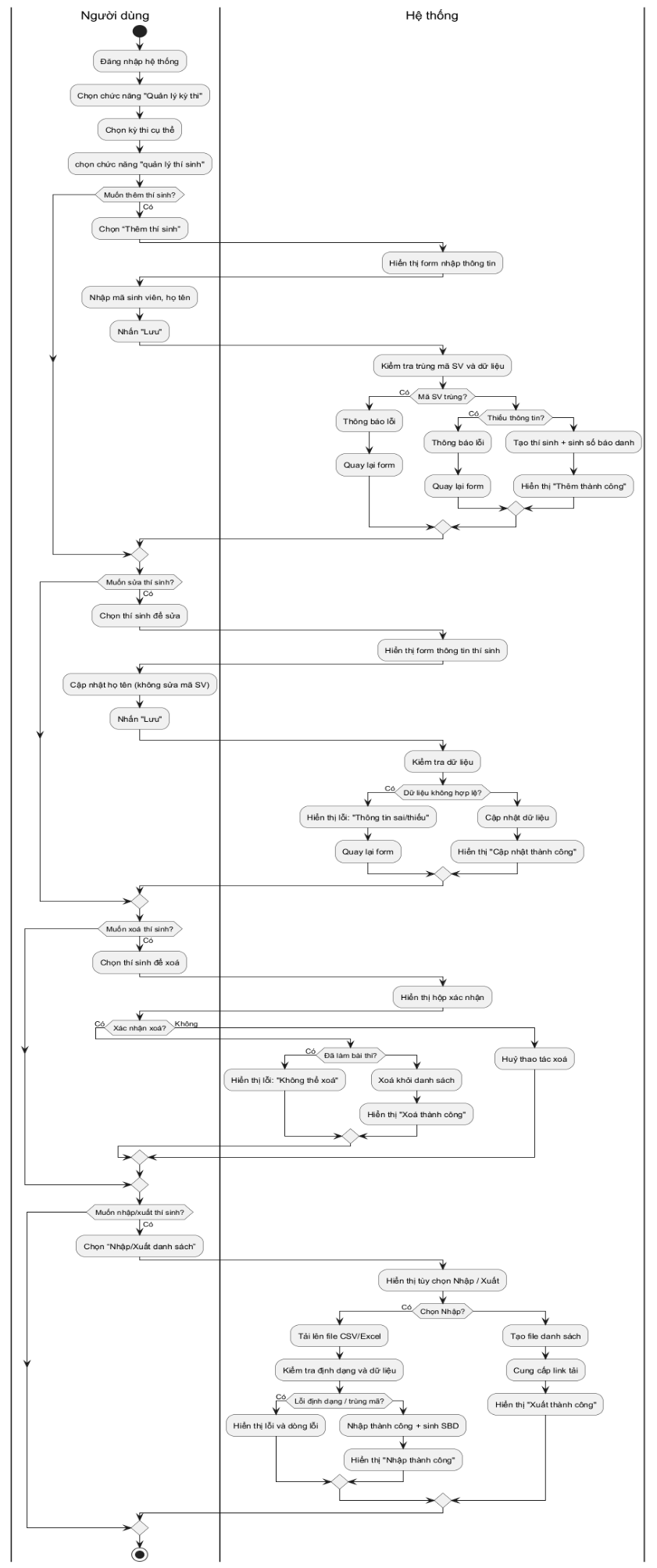
Hình 2.11: Biểu đồ hoạt động chức năng quản lý tài khoản

|  |  |  |
| --- | --- | --- |
| **STT** | **Biểu đồ hoạt động chức năng** | **Giống với biểu đồ hoạt động** |
| 1 | Quản lý khoa | Hình 2.16: Biểu đồ hoạt động chức năng quản lý tài khoản |
| 2 | Quản lý ngành | Hình 2.16: Biểu đồ hoạt động chức năng quản lý tài khoản |
| 3 | Quản lý môn học | Hình 2.16: Biểu đồ hoạt động chức năng quản lý tài khoản |
| 4 | Quản lý kỳ thi | Hình 2.16: Biểu đồ hoạt động chức năng quản lý tài khoản |
| 5 | Quản lý đề thi | Hình 2.16: Biểu đồ hoạt động chức năng quản lý tài khoản |

Bảng 2.16: Các biểu đồ hoạt động tương tự quản lý tài khoản Hình 2.16



Hình 2.12: Biểu đồ hoạt động chức năng quản lý ngân hàng câu hỏi



Hình 2.13: Biểu đồ hoạt động chức năng quản lý thí sinh

A diagram of a diagram

AI-generated content may be incorrect.

Hình 2.14: Biểu đồ hoạt động chức năng báo cáo thống kê

A diagram of a company

AI-generated content may be incorrect.

Hình 2.15: Biểu đồ hoạt động chức năng xác thực thí sinh

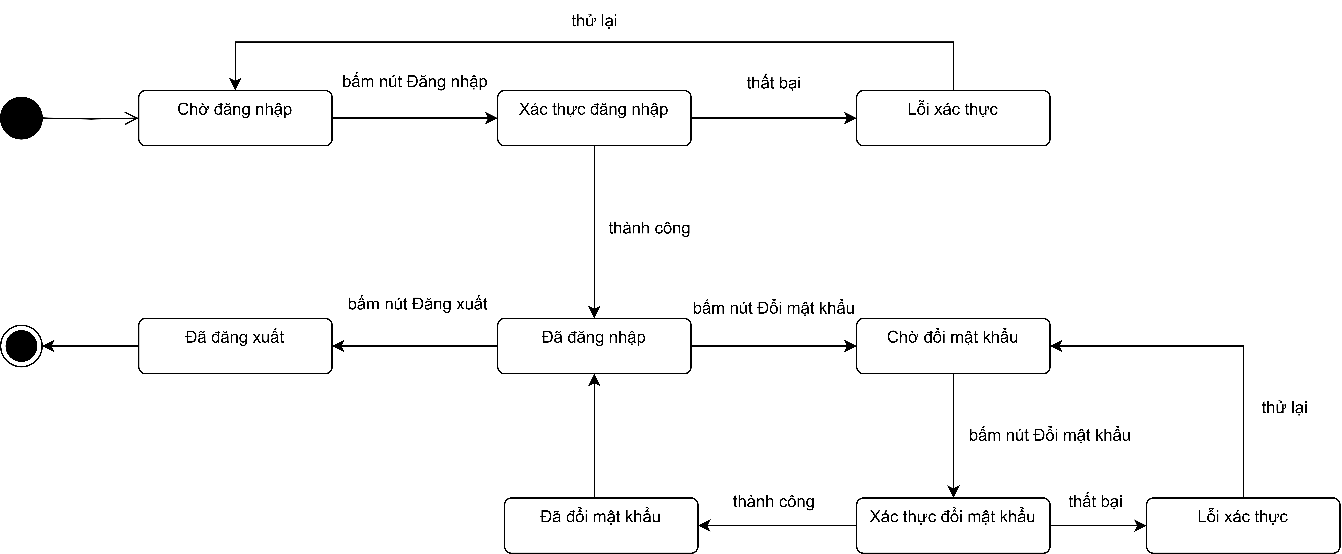
A diagram of a chat

AI-generated content may be incorrect.

Hình 2.16: Biểu đồ hoạt động chức năng làm bài

## 2.4 Biểu đồ trạng thái

Biểu đồ trạng thái được sử dụng để mô tả các trạng thái khác nhau của một đối tượng trong hệ thống và cách nó chuyển đổi giữa các trạng thái đó dựa trên các sự kiện hoặc điều kiện nhất định. Mỗi trạng thái thể hiện một giai đoạn cụ thể trong vòng đời của đối tượng. Biểu đồ giúp xác định rõ các phản ứng của hệ thống khi có thay đổi xảy ra, từ đó hỗ trợ kiểm soát luồng xử lý và tính nhất quán của dữ liệu.

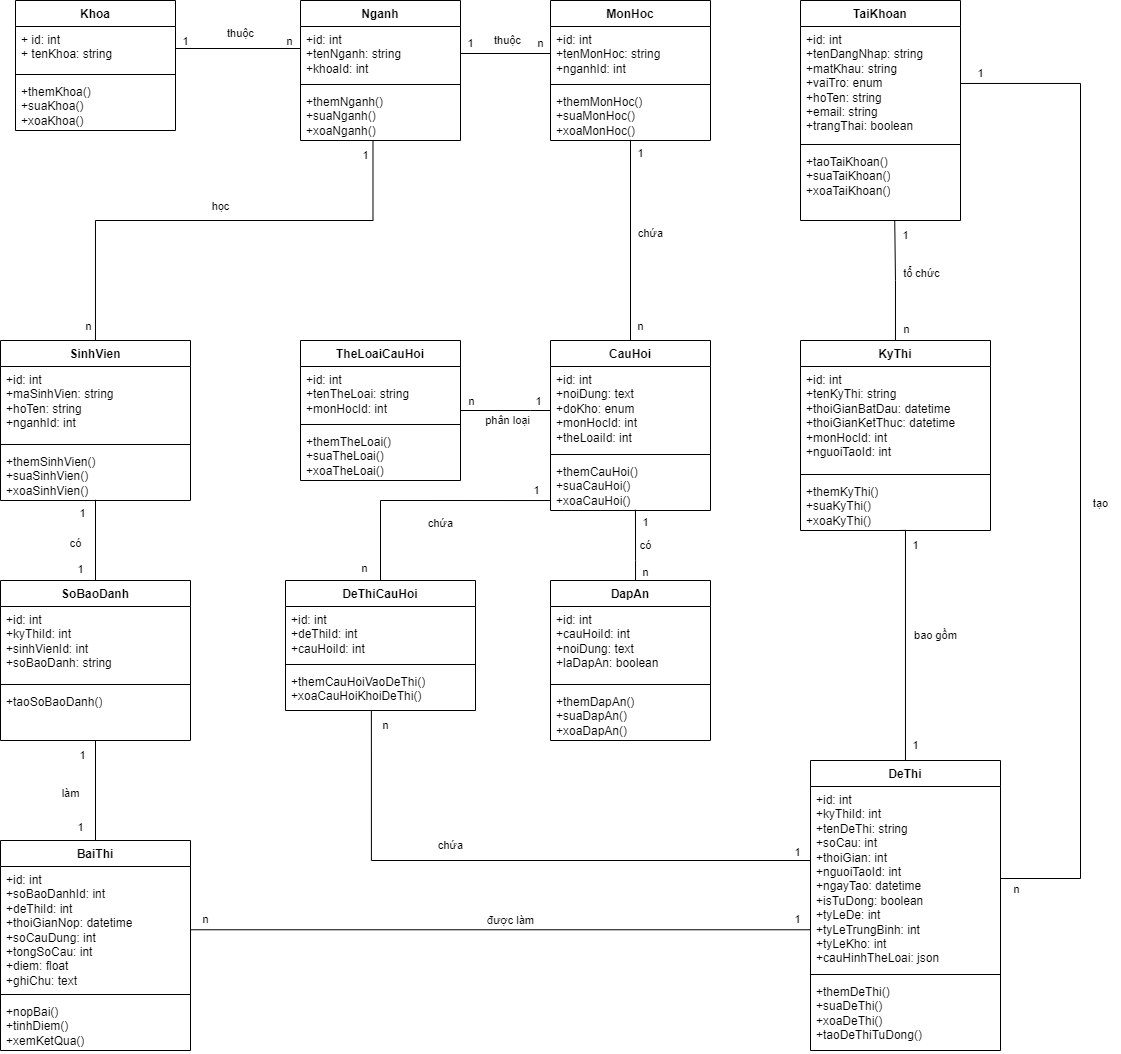


Hình 2.17: biểu đồ trạng thái chức năng đăng nhập

…

## 2.5 Biểu đồ lớp tổng quát

Biểu đồ lớp tổng quát thể hiện các thành phần cốt lõi của hệ thống thi trắc nghiệm và cách chúng liên kết với nhau. Trong hệ thống này, người dùng có thể là quản trị viên hoặc giáo viên, được quản lý thông qua lớp tài khoản với quyền hạn phù hợp. Cấu trúc đào tạo được tổ chức theo mô hình từ khoa, đến ngành, rồi đến từng môn học cụ thể. Ngân hàng câu hỏi được xây dựng dựa trên các câu hỏi, đáp án và thể loại câu hỏi, phục vụ cho việc tạo đề thi một cách linh hoạt. Việc tổ chức kỳ thi được thực hiện thông qua các lớp quản lý kỳ thi, đề thi và các câu hỏi trong từng đề. Thí sinh tham gia thi sẽ được quản lý qua danh sách sinh viên và số báo danh tương ứng. Tất cả các thành phần trong hệ thống được liên kết thông qua các mối quan hệ rõ ràng để đảm bảo tính toàn vẹn dữ liệu, đồng thời hỗ trợ đầy đủ các chức năng như tạo đề, phân công thí sinh, làm bài và chấm điểm tự động.

Hình xx: biểu đồ lớp tổng quát

### 2.5.1 Các lớp dự kiến tham gia vào hệ thống

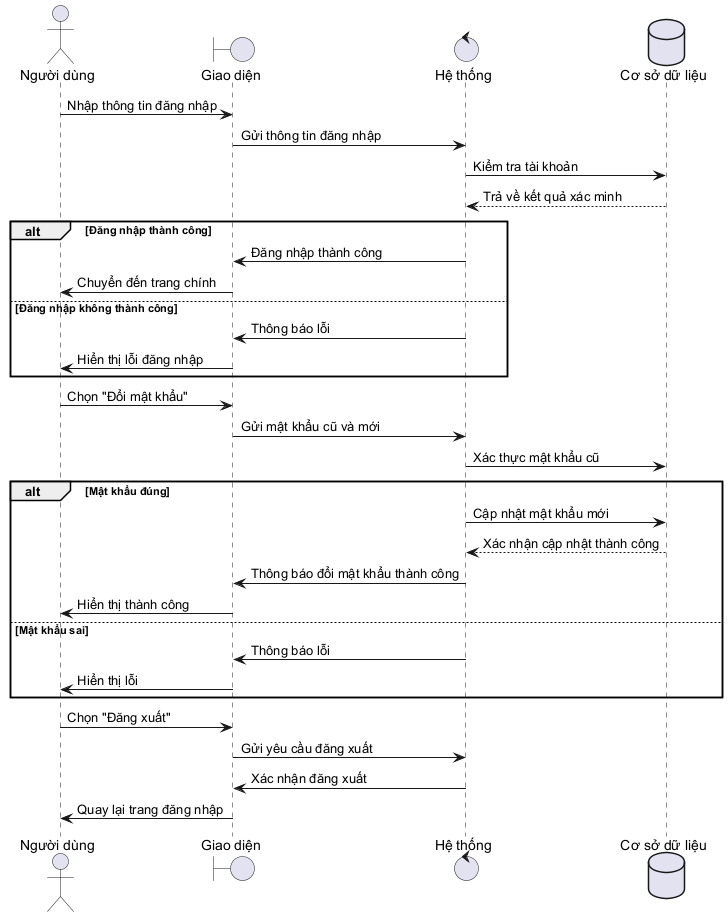
Hệ thống gồm nhiều lớp đại diện cho các thực thể và chức năng chính, phục vụ cho việc quản lý dữ liệu và xử lý nghiệp vụ thi trắc nghiệm. Bảng sau trình bày tên lớp và mô tả ngắn gọn vai trò của từng lớp:

|  |  |
| --- | --- |
| **Tên lớp** | **Mô tả** |
| **Khoa** | Đại diện cho khoa trong trường. Cung cấp các thao tác thêm, sửa, xóa thông tin khoa. |
| **Ngành** | Quản lý các ngành thuộc khoa. Cho phép thêm, sửa, xóa ngành và liên kết ngành với khoa tương ứng. |
| **Môn học** | Quản lý môn học theo từng ngành. Cho phép thêm, sửa, xóa môn học và gắn với ngành cụ thể. |
| **Tài khoản** | Quản lý tài khoản đăng nhập cho giáo viên và quản trị viên. Bao gồm thông tin đăng nhập, vai trò, trạng thái và chức năng tạo/sửa/xóa tài khoản. |
| **Sinh viên** | Quản lý thông tin thí sinh, gồm mã sinh viên, họ tên và ngành học. Cung cấp các phương thức thêm, sửa, xóa sinh viên. |
| **Thể loại câu hỏi** | Đại diện cho loại câu hỏi trong mỗi môn học (ví dụ: lý thuyết, thực hành). Hỗ trợ thêm, sửa, xóa thể loại. |
| **Câu hỏi** | Quản lý nội dung câu hỏi, độ khó, môn học và thể loại. Hỗ trợ các thao tác thêm mới, chỉnh sửa và xóa câu hỏi. |
| **Đáp án** | Quản lý các phương án trả lời của một câu hỏi, bao gồm nội dung và xác định đáp án đúng. Cho phép thêm, sửa, xóa đáp án. |
| **Kỳ thi** | Đại diện cho một đợt thi của một môn học cụ thể. Lưu thông tin thời gian thi, người tạo và hỗ trợ thao tác quản lý kỳ thi. |
| **Đề thi** | Quản lý đề thi của kỳ thi, bao gồm số câu hỏi, thời gian làm bài, cách tạo (thủ công/tự động), tỷ lệ độ khó và loại câu hỏi. |
| **Đề thi câu hỏi** | Lớp liên kết giữa đề thi và các câu hỏi cụ thể. Cho phép thêm hoặc xóa câu hỏi khỏi đề thi. |
| **Số báo danh** | Quản lý mã số báo danh của từng sinh viên trong kỳ thi. Hỗ trợ tự động sinh và gán số báo danh không trùng lặp. |
| **Bài thi** | Đại diện cho kết quả làm bài của thí sinh. Ghi nhận thời gian nộp, số câu đúng/sai, điểm và ghi chú. Cho phép nộp bài, tính điểm và xem kết quả. |

Bảng 2.17: Các lớp dự kiến tham gia vào hệ thống

## 2.6 Biểu đồ tuần tự

Biểu đồ tuần tự được sử dụng để mô tả trình tự các tương tác giữa các đối tượng trong hệ thống theo dòng thời gian. Biểu đồ thể hiện rõ cách các đối tượng gửi và nhận thông điệp để thực hiện một chức năng cụ thể. Thông qua biểu đồ này, ta có thể quan sát được thứ tự thực thi các phương thức, luồng xử lý logic và sự phối hợp giữa các thành phần trong hệ thống một cách trực quan và logic. Dưới đây là biểu đồ tuần tự cho các chức năng của hệ thống:



Hình xx: Biểu đồ tuần tự chức năng đăng nhập

A white sheet of paper with black text

AI-generated content may be incorrect.

Hình xx: Biểu đồ tuần tự chức năng quản lý tài khoản

|  |  |  |
| --- | --- | --- |
| **STT** | **Biểu đồ tuần tự chức năng** | **Giống với biểu đồ tuần tự** |
| 1 | Quản lý khoa | Hình xx: Biểu đồ tuần tự chức năng quản lý tài khoản |
| 2 | Quản lý ngành | Hình xx: Biểu đồ tuần tự chức năng quản lý tài khoản |
| 3 | Quản lý môn học | Hình xx: Biểu đồ tuần tự chức năng quản lý tài khoản |
| 4 | Quản lý kỳ thi | Hình xx: Biểu đồ tuần tự chức năng quản lý tài khoản |
| 5 | Quản lý đề thi | Hình xx: Biểu đồ tuần tự chức năng quản lý tài khoản |

Bảng 2.18: Các biểu đồ tuần tự tương tự chức năng quản lý tài khoản Hình 2.16

A white sheet of paper with black text

AI-generated content may be incorrect.

Hình xx: Biểu đồ tuần tự chức năng quản lý ngân hàng câu hỏi

A white sheet of paper with black text

AI-generated content may be incorrect.

Hình xx: Biểu đồ tuần tự chức năng quản lý thí sinh

A diagram of a diagram

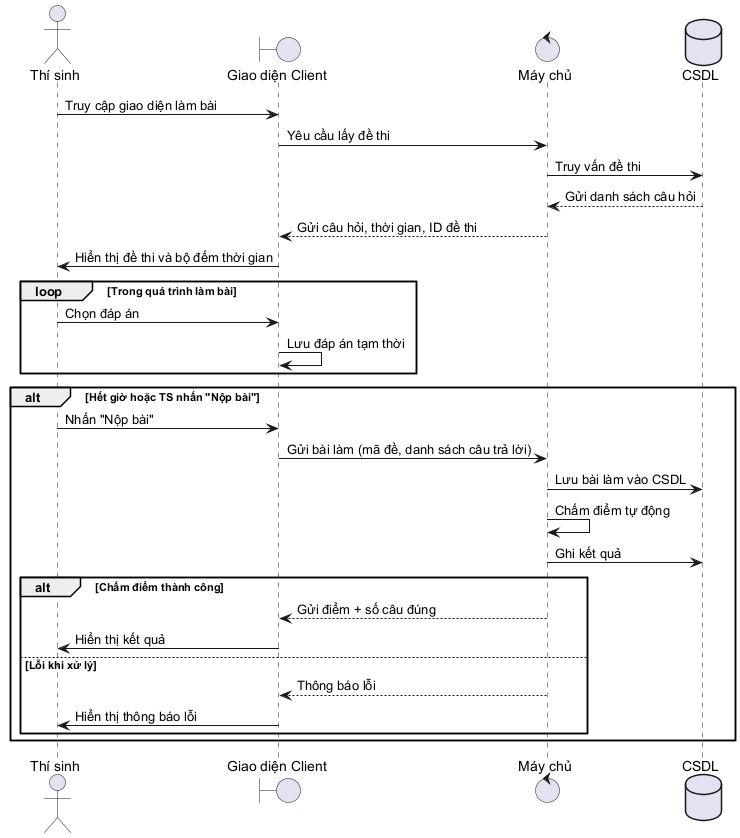
AI-generated content may be incorrect.

Hình xx: Biểu đồ tuần tự chức năng báo cáo thống kê

A diagram of a diagram

AI-generated content may be incorrect.

Hình xx: Biểu đồ tuần tự chức năng xác thực thí sinh



Hình xx: Biểu đồ tuần tự chức năng làm bài

## 2.7 Biểu đồ lớp chi tiết

### 2.7.1 Lớp dự kiến

|  |  |  |
| --- | --- | --- |
| **Lớp** | **Thuộc tính** | **Phương thức** |
| Khoa | id: int,  tenKhoa: string | +themKhoa(),  +suaKhoa(),  +xoaKhoa() |
| Nganh | id: int,  tenNganh: string,  khoaId: int | +themNganh(),  +suaNganh(),  +xoaNganh() |
| MonHoc | id: int,  tenMonHoc: string,  nganhId: int | +themMonHoc(),  +suaMonHoc(),  +xoaMonHoc() |
| TaiKhoan | id: int,  tenDangNhap: string,  matKhau: string,  vaiTro: enum,  hoTen: string,  trangThai: boolean | +taoTaiKhoan(),  +suaTaikhoan(),  +xoaTaiKhoan() |
| SinhVien | id: int,  maSinhVien: string,  hoTen: string,  nganhId: int | +themSinhVien(),  +suaSinhVien(),  +xoaSinhVien() |
| TheLoaiCauHoi | id: int,  tenTheLoai: string,  monHocId: int | +themTheLoai(),  +suaTheLoai(),  +xoaTheLoai() |
| CauHoi | id: int,  noiDung: text,  doKho: enum,  monHocId: int,  theLoaiId: int | +themCauHoi(),  +suaCauHoi(),  +xoaCauHoi() |
| DapAn | id: int,  cauHoiId: int,  noiDung: text,  laDapAn: boolean | +themDapAn(),  +suaDapAn(),  +xoaDapAn() |
| KyThi | id: int,  tenKyThi: string,  thoiGianBatDau: datetime,  thoiGianKetThuc: datetime,  monHocId: int,  nguoiTaoId: int | +themKyThi(),  +suaKyThi(),  +xoaKyThi() |
| DeThi | id: int,  kyThiId: int,  tenDeThi: string,  soCau: int,  thoiGian: int,  nguoiTaoId: int,  ngayTao: datetime,  isTuDong: boolean,  tyLeDe: int,  tyLeKho: int, cauHoiTheoTheLoai: json | +themDeThi(),  +suaDeThi(),  +xoaDeThi(),  +taoDeThiTuDong() |
| DeThiCauHoi | id: int,  deThiId: int,  cauHoiId: int | +themCauHoiVaoDeThi(),  +xoaCauHoiDeThi() |
| SoBaoDanh | id: int,  kyThiId: int,  sinhVienId: int,  soBaoDanh: string | +taoSoBaoDanh() |
| BaiThi | id: int,  soBaoDanhId: int,  deThiId: int,  thoiGianNop: datetime,  soCauDung: int,  soCauSai: int,  diem: float,  ghiChu: text | +nopBai(),  +tinhDiem(),  +xemKetQua() |

Bảng 2.19: Lớp dự kiến

### 2.7.2 Xác định thuộc tính cho lớp

|  |  |  |  |
| --- | --- | --- | --- |
| **Lớp** | **Thuộc tính** | **Kiểu Dữ Liệu** | **Mô tả** |
| Khoa | id | int | Mã định danh khoa |
| tenKhoa | string | Tên khoa |
| Nganh | id | int | Mã định danh ngành |
| tenNganh | string | Tên ngành |
| khoaId | int | Khóa ngoại đến Khoa |
| MonHoc | id | int | Mã định danh môn học |
| tenMonHoc | string | Tên môn học |
| nganhId | int | Khóa ngoại đến Ngành |
| TaiKhoan | id | int | Mã định danh tài khoản |
| tenDangNhap | string | Tên đăng nhập |
| matKhau | string | Mật khẩu đã mã hóa |
| vaiTro | enum | admin hoặc giaovien |
| hoTen | string | Họ tên người dùng |
| trangThai | boolean | Tình trạng hoạt động |
| SinhVien | id | int | Mã định danh sinh viên |
| maSinhVien | string | Mã sinh viên duy nhất |
| hoTen | string | Họ tên sinh viên |
| nganhId | int | Khóa ngoại đến Ngành |
| TheLoaiCauHoi | id | int | Mã định danh thể loại |
| tenTheLoai | string | Tên thể loại câu hỏi |
| monHocId | int | Khóa ngoại đến Môn học |
| CauHoi | id | int | Mã định danh câu hỏi |
| noiDung | text | Nội dung câu hỏi |
| doKho | enum | Mức độ: de, trungbinh, kho |
| monHocId | int | Khóa ngoại đến Môn học |
| theLoaiId | int | Khóa ngoại đến Thể loại (có thể null) |
| DapAn | id | int | Mã định danh đáp án |
| cauHoiId | int | Khóa ngoại đến Câu hỏi |
| noiDung | text | Nội dung đáp án |
| laDapAn | boolean | Xác định đúng/sai |
| KyThi | id | int | Mã định danh kỳ thi |
| tenKyThi | string | Tên kỳ thi |
| thoiGianBatDau | datetime | Bắt đầu thi |
| thoiGianKetThuc | datetime | Kết thúc thi |
| monHocId | int | Môn học thi |
| nguoiTaoId | int | Tài khoản người tạo |
| DeThi | id | int | Mã định danh đề thi |
| kyThiId | int | Thuộc kỳ thi nào |
| tenDeThi | string | Tên đề thi |
| soCau | int | Tổng số câu hỏi |
| thoiGian | int | Thời gian làm bài (phút) |
| nguoiTaoId | int | Người tạo đề |
| ngayTao | datetime | Ngày tạo đề |
| isTuDong | boolean | Có tạo tự động không |
| tyLeDe | int | Tỷ lệ câu dễ |
| tyLeKho | int | Tỷ lệ câu khó |
| cauHoiTheoTheLoai | json | Cấu hình câu hỏi theo thể loại |
| DeThiCauHoi | id | int | Mã định danh dòng |
| deThiId | int | Khóa ngoại đến Đề thi |
| cauHoiId | int | Khóa ngoại đến Câu hỏi |
| SoBaoDanh | id | int | Mã định danh |
| kyThiId | int | Kỳ thi tương ứng |
| sinhVienId | int | Sinh viên dự thi |
| soBaoDanh | string | Số báo danh không trùng |
| BaiThi | id | int | Mã định danh bài thi |
| soBaoDanhId | int | Liên kết số báo danh |
| deThiId | int | Đề thi đã làm |
| thoiGianNop | datetime | Thời gian nộp bài |
| soCauDung | int | Số câu trả lời đúng |
| soCauSai | int | Số câu sai |
| diem | float | Điểm số |
| ghiChu | text | Ghi chú thêm nếu có |

Bảng 2.20: Thuộc tính các lớp

### 2.7.3 Xác định phương thức cho lớp

|  |  |  |
| --- | --- | --- |
| **Lớp** | **Phương thức** | **Mô tả** |
| Khoa | +themKhoa() | Thêm mới một khoa |
| +suaKhoa() | Cập nhật thông tin khoa |
| +xoaKhoa() | Xóa khoa khỏi hệ thống |
| Nganh | +themNganh() | Thêm mới ngành học |
| +suaNganh() | Cập nhật ngành học |
| +xoaNganh() | Xóa ngành học |
| MonHoc | +themMonHoc() | Thêm mới môn học |
| +suaMonHoc() | Sửa thông tin môn học |
| +xoaMonHoc() | Xóa môn học khỏi hệ thống |
| TaiKhoan | +taoTaiKhoan() | Tạo mới tài khoản admin hoặc giáo viên |
| +suaTaiKhoan() | Cập nhật tài khoản |
| +xoaTaiKhoan() | Xóa tài khoản |
| SinhVien | +themSinhVien() | Thêm mới sinh viên |
| +suaSinhVien() | Sửa thông tin sinh viên |
| +xoaSinhVien() | Xóa sinh viên |
| TheLoaiCauHoi | +themTheLoai() | Thêm thể loại câu hỏi mới |
| +suaTheLoai() | Cập nhật thể loại câu hỏi |
| +xoaTheLoai() | Xóa thể loại |
| CauHoi | +themCauHoi() | Thêm câu hỏi mới vào ngân hàng |
| +suaCauHoi() | Chỉnh sửa nội dung câu hỏi |
| +xoaCauHoi() | Xóa câu hỏi |
| DapAn | +themDapAn() | Thêm đáp án cho câu hỏi |
| +suaDapAn() | Sửa đáp án |
| +xoaDapAn() | Xóa đáp án |
| KyThi | +themKyThi() | Tạo kỳ thi mới |
| +suaKyThi() | Chỉnh sửa thông tin kỳ thi |
| +xoaKyThi() | Xóa kỳ thi |
| DeThi | +themDeThi() | Tạo đề thi mới thủ công hoặc tự động |
| +suaDeThi() | Cập nhật đề thi |
| +xoaDeThi() | Xóa đề thi |
| +taoDeThiTuDong() | Tạo đề thi tự động dựa trên cấu hình |
| DeThiCauHoi | +themCauHoiVaoDeThi() | Thêm câu hỏi vào đề thi |
| +xoaCauHoiDeThi() | Gỡ câu hỏi khỏi đề thi |
| SoBaoDanh | +taoSoBaoDanh() | Sinh số báo danh duy nhất cho mỗi sinh viên trong kỳ thi |
| BaiThi | +nopBai() | Gửi bài thi của thí sinh |
| +tinhDiem() | Chấm điểm bài thi tự động |
| +xemKetQua() | Hiển thị kết quả và phân tích |

Bảng 2.21: Phương thức các lớp

### 2.7.4 Xác định mối quan hệ giữa các lớp

**a. Mối quan hệ 1-1 (One-to-One)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Lớp A** | **Lớp B** | **Mối quan hệ** | **Mô tả** |
| SoBaoDanh | SinhVien | Mỗi sinh viên có duy nhất 1 số báo danh trong một kỳ thi | Xác định thí sinh tham gia kỳ thi duy nhất |
| BaiThi | SoBaoDanh | Mỗi bài thi gắn với một số báo danh duy nhất | Bài thi đại diện cho một lượt làm của thí sinh |

Bảng 2.22: Mối quan hệ 1-1 giữa các lớp

**b. Mối quan hệ 1-n (One-to-Many)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Lớp Cha (1)** | **Lớp Con (n)** | **Mối quan hệ** | **Mô tả** |
| Khoa | Nganh | Một khoa có nhiều ngành | Tổ chức phân ngành theo khoa |
| Nganh | MonHoc | Một ngành có nhiều môn học | Mỗi môn học thuộc một ngành |
| MonHoc | CauHoi | Một môn học có nhiều câu hỏi | Câu hỏi được phân theo môn |
| CauHoi | DapAn | Một câu hỏi có nhiều đáp án | Mỗi đáp án thuộc về một câu |
| KyThi | DeThi | Một kỳ thi có thể có nhiều đề thi | Tạo nhiều đề khác nhau cho cùng 1 kỳ |
| DeThi | DeThiCauHoi | Một đề thi chứa nhiều câu hỏi | Mỗi đề gồm nhiều câu hỏi cụ thể |
| TaiKhoan | KyThi, DeThi | Một giáo viên (tài khoản) có thể tạo nhiều kỳ thi, đề thi | Liên kết giữa người tạo và nội dung thi |

Bảng 2.23: Mối quan hệ 1-n giữa các lớp

**c. Mối quan hệ hợp thành - kết hợp (Composition - Aggregation)**

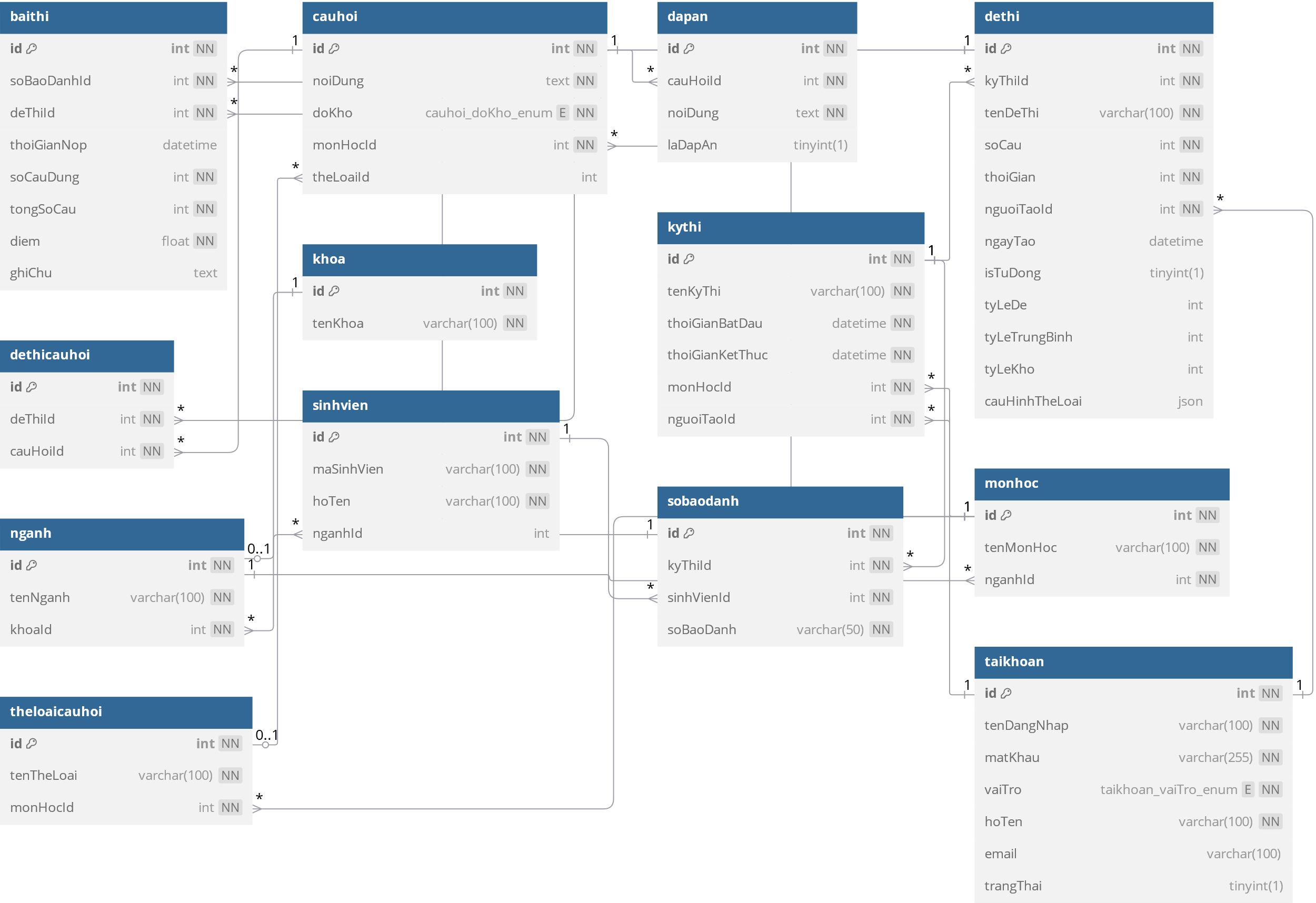
|  |  |  |  |
| --- | --- | --- | --- |
| **Lớp Tổng thể** | **Lớp Thành phần** | **Loại quan hệ** | **Mô tả** |
| DeThi | DeThiCauHoi | Hợp thành (Composition) | Khi đề thi bị xóa, các câu hỏi trong đề cũng bị xóa theo |
| KyThi | SoBaoDanh | Hợp thành (Composition) | Số báo danh tồn tại cùng kỳ thi – khi kỳ thi bị xóa thì SBD cũng mất |
| DeThi | CauHoi | Kết hợp (Aggregation) | Câu hỏi được lấy từ ngân hàng – xóa đề không xóa câu hỏi gốc |
| MonHoc | TheLoaiCauHoi | Kết hợp (Aggregation) | Thể loại thuộc về môn học, có thể tái sử dụng |

Bảng 2.24: Mối quan hệ C-A giữa các lớp

# CHƯƠNG III: THIẾT KẾ HỆ THỐNG

## 3.1 Thiết kế cơ sở dữ liệu

Thiết kế cơ sở dữ liệu dựa vào biểu đồ lớp, xác định các bảng, khóa chính, khóa ngoại, và mối quan hệ giữa các bảng đảm bảo tính toàn vẹn và hiệu quả trong việc lưu trữ và truy xuất dữ liệu



Hình 3.1: Biểu đồ cơ sở dữ liệu

**Các bảng:**

**1. Bảng taikhoan (Tài khoản)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID tài khoản |
| 2 | tenDangNhap | varchar(100) | False | Tên đăng nhập |
| 3 | matKhau | varchar(255) | False | Mật khẩu |
| 4 | vaiTro | enum('admin','giaovien') | False | Vai trò tài khoản |
| 5 | hoTen | varchar(100) | False | Họ tên |
| 6 | email | varchar(100) | True | Email |
| 7 | trangThai | tinyint(1) | True (default) | Trạng thái hoạt động (1: hoạt động) |

Bảng 3.1: Thiết kế bảng taikhoan trong CSDL

**2. Bảng sinhvien (Sinh viên)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **rỗng** | **Mô tả** |
| 1 | id | int | False | ID sinh viên |
| 2 | maSinhVien | varchar(100) | False | Mã sinh viên |
| 3 | hoTen | varchar(100) | False | Họ tên sinh viên |
| 4 | nganhId | int | True | Khóa ngoại đến ngành |

Bảng 3.2: Thiết kế bảng sinhvien trong CSDL

**3. Bảng nganh (Ngành học)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID ngành học |
| 2 | tenNganh | varchar(100) | False | Tên ngành học |
| 3 | khoaId | int | False | Khóa ngoại đến bảng khoa |

Bảng 3.3: Thiết kế bảng nganh trong CSDL

**4. Bảng khoa (Khoa)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID khoa |
| 2 | tenKhoa | varchar(100) | False | Tên khoa |

Bảng 3.4: Thiết kế bảng khoa trong CSDL

**5. Bảng monhoc (Môn học)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID môn học |
| 2 | tenMonHoc | varchar(100) | False | Tên môn học |
| 3 | nganhId | int | False | Khóa ngoại đến bảng nganh |

Bảng 3.5: Thiết kế bảng monhoc trong CSDL

**6. Bảng kythi (Kỳ thi)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID kỳ thi |
| 2 | tenKyThi | varchar(100) | False | Tên kỳ thi |
| 3 | thoiGianBatDau | datetime | False | Thời gian bắt đầu |
| 4 | thoiGianKetThuc | datetime | False | Thời gian kết thúc |
| 5 | monHocId | int | False | Môn thi (FK đến monhoc) |
| 6 | nguoiTaoId | int | False | Người tạo (FK đến taikhoan) |

Bảng 3.6: Thiết kế bảng kythi trong CSDL

**7. Bảng sobaodanh (Số báo danh)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID số báo danh |
| 2 | kyThiId | int | False | FK đến kỳ thi |
| 3 | sinhVienId | int | False | FK đến sinh viên |
| 4 | soBaoDanh | varchar(50) | False | Mã số báo danh (duy nhất) |

Bảng 3.7: Thiết kế bảng sobaodanh trong CSDL

**8. Bảng cauhoi (Câu hỏi)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID câu hỏi |
| 2 | noiDung | text | False | Nội dung câu hỏi |
| 3 | doKho | enum('de','trungbinh','kho') | False | Độ khó câu hỏi |
| 4 | monHocId | int | False | FK đến môn học |
| 5 | theLoaiId | int | True | FK đến thể loại câu hỏi |

Bảng 3.8: Thiết kế bảng cauhoi trong CSDL

**9. Bảng dapan (Đáp án)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID đáp án |
| 2 | cauHoiId | int | False | FK đến câu hỏi |
| 3 | noiDung | text | False | Nội dung đáp án |
| 4 | laDapAn | tinyint(1) | True | Có phải đáp án đúng (1/0) |

Bảng 3.9: Thiết kế bảng dapan trong CSDL

**10. Bảng theloaicauhoi (Thể loại câu hỏi)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID thể loại |
| 2 | tenTheLoai | varchar(100) | False | Tên thể loại |
| 3 | monHocId | int | False | FK đến môn học |

Bảng 3.10: Thiết kế bảng theloaicauhoi trong CSDL

**11. Bảng dethi (Đề thi)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID đề thi |
| 2 | kyThiId | int | False | FK đến kỳ thi |
| 3 | tenDeThi | varchar(100) | False | Tên đề thi |
| 4 | soCau | int | False | Tổng số câu |
| 5 | thoiGian | int | False | Thời gian làm bài (phút) |
| 6 | nguoiTaoId | int | False | FK đến người tạo (tài khoản) |
| 7 | ngayTao | datetime | True (default) | Ngày tạo đề |
| 8 | isTuDong | tinyint(1) | True (default) | Có phải tạo tự động |
| 9 | tyLeDe | int | True (default) | Tỷ lệ câu dễ |
| 10 | tyLeTrungBinh | int | True (default) | Tỷ lệ câu trung bình |
| 11 | tyLeKho | int | True (default) | Tỷ lệ câu khó |
| 12 | cauHinhTheLoai | json | True | Cấu hình thể loại câu hỏi |

Bảng 3.11: Thiết kế bảng dethi trong CSDL

**12. Bảng dethicauhoi (Câu hỏi trong đề thi)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID |
| 2 | deThiId | int | False | FK đến đề thi |
| 3 | cauHoiId | int | False | FK đến câu hỏi |

Bảng 3.12: Thiết kế bảng dethicauhoi trong CSDL

**13. Bảng baithi (Bài thi)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Tên** | **Kiểu dữ liệu** | **Rỗng** | **Mô tả** |
| 1 | id | int | False | ID bài thi |
| 2 | soBaoDanhId | int | False | FK đến số báo danh |
| 3 | deThiId | int | False | FK đến đề thi |
| 4 | thoiGianNop | datetime | True | Thời gian nộp bài |
| 5 | soCauDung | int | False | Số câu làm đúng |
| 6 | tongSoCau | int | False | Tổng số câu trong đề |
| 7 | diem | float | False | Điểm đạt được |
| 8 | ghiChu | text | True | Ghi chú thêm |

Bảng 3.13: Thiết kế bảng baithi trong CSDL

**Quan hệ giữa các bảng:**

1.Bảng **taikhoan** và bảng **dethi**, **kythi**

* Mối quan hệ: 1-n (một tài khoản có thể tạo nhiều kỳ thi và đề thi).
* Khóa ngoại:
  + **dethi.nguoiTaoId** tham chiếu đến **taikhoan.id**.
  + **kythi.nguoiTaoId** tham chiếu đến **taikhoan.id**.

2.Bảng **sinhvien** vàbảng **sobaodanh**

* Mối quan hệ: 1-n (mỗi sinh viên có thể được cấp nhiều số báo danh cho các kỳ thi khác nhau).
* Khóa ngoại:
  + **sobaodanh.sinhVienId** tham chiếu đến **sinhvien.id**.

3.Bảng **sobaodanh** và bảng **baithi**

* Mối quan hệ: 1-n (mỗi số báo danh chỉ được thi một đề, nhưng có thể có nhiều bài thi nếu mở rộng).
* Khóa ngoại:
  + **baithi.soBaoDanhId** tham chiếu đến **sobaodanh.id**.

4. Bảng **cauhoi** và bảng **dapan**

* Mối quan hệ: 1-n (mỗi câu hỏi có nhiều đáp án).
* Khóa ngoại:
  + **dapan.cauHoiId** tham chiếu đến **cauhoi.id**.

5. Bảng **dethi** và bảng **dethicauhoi**

* Mối quan hệ: 1-n (mỗi đề thi chứa nhiều câu hỏi).
* Khóa ngoại:
  + **dethicauhoi.deThiId** tham chiếu đến **dethi.id**.

6. Bảng **cauhoi** và bảng **dethicauhoi**

* Mối quan hệ: 1-n (mỗi câu hỏi có thể xuất hiện trong nhiều đề).
* Khóa ngoại:
  + **dethicauhoi.cauHoiId** tham chiếu đến **cauhoi.id**.

7. Bảng **monhoc** và bảng **cauhoi, theloaicauhoi, kythi**

* Mối quan hệ: 1-n
* Khóa ngoại:
  + **cauhoi.monHocId, theloaicauhoi.monHocId, kythi.monHocId** tham chiếu đến **monhoc.id**.

8. Bảng **nganh** và bảng **monhoc, sinhvien**

* Mối quan hệ: 1-n
* Khóa ngoại:
  + **monhoc.nganhId, sinhvien.nganhId** tham chiếu đến **nganh.id**.

9. Bảng **khoa** và bảng **nganh**

* Mối quan hệ: 1-n
* Khóa ngoại:
  + **nganh.khoaId** tham chiếu đến **khoa.id**.

## 3.2 Xây dựng, thiết kế biểu đồ thành phần

Biểu đồ thành phần (Component Diagram) mô tả cấu trúc phần mềm bằng cách hiển thị các thành phần (components), mô-đun, thư viện và cách chúng tương tác với nhau. Nó giúp hiểu cách các phần của hệ thống được tổ chức và giao tiếp, hỗ trợ thiết kế và bảo trì phần mềm.

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.2: Biểu đồ thành phần

**1. Thành phần trong hệ thống**

**1.1. Ứng dụng thi (Student Client)**

Ứng dụng dành cho thí sinh làm bài thi, phát triển bằng Flutter, có thể chạy trên các hệ điều hành phổ biến như: Android, iOS, Windows, Linux, MacOS.

**Thành phần chính:**

* **Module xác thực thí sinh**
  + Thực hiện xác minh danh tính dựa trên mã sinh viên và số báo danh.
  + Kết nối đến API backend để kiểm tra thông tin hợp lệ trước khi làm bài.
* **Module làm bài thi**
  + Hiển thị nội dung đề thi nhận được từ hệ thống.
  + Cho phép chọn đáp án, lưu tạm thời và nộp bài khi hoàn thành.

**1.2. Máy chủ Web (Web Server)**

Hệ thống phía máy chủ phát triển bằng PHP (Laravel), chịu trách nhiệm quản lý toàn bộ dữ liệu, xác thực người dùng và xử lý nghiệp vụ thi.

**Thành phần chính:**

* **Dịch vụ API**
  + Nhận yêu cầu HTTP từ ứng dụng Flutter.
  + Điều phối các dịch vụ nội bộ để xử lý dữ liệu và trả kết quả về cho client.
* **Module xác thực đăng nhập**
  + Quản lý xác thực tài khoản giáo viên, admin qua tên đăng nhập và mật khẩu.
  + Cấp quyền truy cập phù hợp theo vai trò người dùng.
* **Module Ngân hàng câu hỏi**
  + Cho phép thêm, sửa, xóa câu hỏi trắc nghiệm.
  + Câu hỏi được phân loại theo môn học, độ khó và thể loại.
* **Module Kỳ thi**
  + Tạo mới kỳ thi, thiết lập thời gian, danh sách thí sinh tham gia.
  + Kết nối với ngân hàng câu hỏi để tạo đề thi.
* **Ứng dụng PHP (Laravel)**
  + Điều phối toàn bộ các module xử lý nghiệp vụ.
  + Triển khai giao diện quản trị (web admin).
* **Cơ sở dữ liệu MySQL**
  + Lưu trữ toàn bộ dữ liệu của hệ thống bao gồm người dùng, kỳ thi, câu hỏi, bài làm, kết quả...
  + Tất cả các module đều truy xuất và ghi dữ liệu thông qua các dịch vụ được bảo vệ.

**Các module nghiệp vụ liên quan:**

* **Module Tài khoản:** Quản lý tài khoản admin, giáo viên.
* **Module Khoa:** Quản lý danh sách khoa trong hệ thống.
* **Module Ngành:** Gắn với khoa, dùng để phân loại môn học.
* **Module Môn học:** Dùng để phân loại ngân hàng câu hỏi và đề thi.

**2. Luồng hoạt động của hệ thống**

**2.1. Luồng hoạt động của Ứng dụng thí sinh (Client Flutter)**

1. Thí sinh mở ứng dụng được phát triển bằng Flutter trên thiết bị di động.
2. Ứng dụng gửi yêu cầu xác thực thí sinh đến máy chủ thông qua giao tiếp HTTP.
3. Sau khi xác thực thành công, ứng dụng nhận đề thi và hiển thị cho thí sinh làm bài.
4. Khi hoàn thành, bài thi được gửi lên máy chủ để lưu trữ và xử lý.
5. Thí sinh có thể xem lại bài thi và kết quả nếu được phép.

**2.2. Luồng hoạt động của Máy chủ (Web Server)**

1. Máy chủ nhận yêu cầu từ ứng dụng Flutter qua các API.
2. Dịch vụ API sẽ định tuyến yêu cầu đến các module chuyên trách:
   * **Xác thực đăng nhập**: kiểm tra thông tin đăng nhập của thí sinh hoặc giáo viên.
   * **Ngân hàng câu hỏi**: truy xuất các câu hỏi theo đề thi.
   * **Kỳ thi**: quản lý thông tin kỳ thi, phát đề thi, lưu bài thi nộp.
3. Dữ liệu từ các module được xử lý thông qua ứng dụng PHP, sau đó truy xuất đến cơ sở dữ liệu MySQL.
4. Cơ sở dữ liệu trung tâm lưu trữ toàn bộ thông tin như: tài khoản, câu hỏi, kỳ thi, bài thi, v.v.

3.3 Xây dựng, thiết kế biểu đồ triển khai  
Biểu đồ triển khai (Deployment Diagram) mô tả cách phần mềm được triển khai trên phần cứng, bao gồm các máy chủ, thiết bị khách, kết nối mạng, và môi trường thực thi. Nó giúp hiểu rõ hệ thống hoạt động như thế nào trong thực tế, xác định các thành phần phần cứng và cách chúng giao tiếp với nhau.

A diagram of a circuit

AI-generated content may be incorrect.

Hình 3.3: Biểu đồ triển khai

**1. Thành phần trong hệ thống**

**1.1. Máy thí sinh (Client)**

Thiết bị mà thí sinh sử dụng để làm bài thi trực tuyến, được cài đặt ứng dụng Flutter.

**Thành phần chính:**

* **Ứng dụng thi Flutter**
  + Gửi yêu cầu HTTP đến ứng dụng PHP trên máy chủ để:
    - Xác thực thí sinh
    - Lấy đề thi
    - Gửi bài làm sau khi hoàn thành
  + Hiển thị đề thi và thu nhận câu trả lời từ thí sinh

**1.2. Máy chủ web (Web Server)**

Thiết bị xử lý trung tâm, đảm nhiệm vai trò xử lý logic nghiệp vụ và truy xuất dữ liệu.

**Thành phần chính:**

* **Ứng dụng PHP (Server API)**
  + Nhận và xử lý yêu cầu HTTP (RESTful API) từ ứng dụng Flutter
  + Kết nối đến cơ sở dữ liệu MySQL để đọc/ghi dữ liệu bài thi, thí sinh, đề thi, kết quả,...
* **Cơ sở dữ liệu MySQL**
  + Lưu trữ toàn bộ dữ liệu hệ thống: tài khoản, kỳ thi, đề thi, câu hỏi, bài làm, kết quả,...

**2. Luồng hoạt động của hệ thống**

**2.1. Luồng hoạt động của Ứng dụng thí sinh (Client Flutter)**

1. Thí sinh mở ứng dụng và nhập thông tin xác thực (mã sinh viên, số báo danh).
2. Ứng dụng gửi yêu cầu HTTP đến ứng dụng PHP trên máy chủ để kiểm tra và lấy đề thi tương ứng.
3. Đề thi được hiển thị cho thí sinh làm bài.
4. Sau khi hoàn thành, bài làm được gửi về ứng dụng PHP thông qua API.
5. Hệ thống xử lý, lưu kết quả và phản hồi lại trạng thái thành công hoặc lỗi.

**2.2. Luồng hoạt động của Ứng dụng PHP (Web Server)**

1. Nhận yêu cầu xác thực, lấy đề, gửi bài... từ ứng dụng Flutter qua giao thức HTTP.
2. Xử lý nghiệp vụ như:
   * Kiểm tra mã sinh viên, số báo danh
   * Trích xuất đề thi từ CSDL
   * Ghi nhận kết quả bài làm
3. Tương tác với MySQL để lưu trữ và truy vấn dữ liệu.
4. Trả phản hồi (dạng JSON) về lại ứng dụng thí sinh.

**2.3. Mạng kết nối**

* Giao tiếp giữa ứng dụng Flutter và ứng dụng PHP diễn ra qua giao thức **HTTP Request** nội bộ.

## 3.4 Thiết kế giao diện

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.3: Giao diện đăng nhập hệ thống

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.4: Giao diện trang chủ

A screenshot of a chat

AI-generated content may be incorrect.

Hình 3.5: Giao diện quản lý ngân hàng câu hỏi

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.6: Giao diện thêm câu hỏi

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.7: Giao diện kỳ thi

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.8: Giao diện tạo đề thi

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.9: Giao diện danh sách thí sinh

A screenshot of a computer

AI-generated content may be incorrect.

Hình 3.10: Giao diện nhập/xuất danh sách thí sinh

# CHƯƠNG IV: CÀI ĐẶT VÀ CHẠY THỬ

## 4.1 Cài đặt

Hệ thống bao gồm hai phần chính: phía server được lập trình bằng PHP kết hợp với cơ sở dữ liệu MySQL, và phía client là ứng dụng Flutter đã được biên dịch sẵn dưới dạng thực thi (.exe) trên hệ điều hành Windows. Quá trình cài đặt dưới đây hướng dẫn cách thiết lập môi trường máy chủ, cơ sở dữ liệu và triển khai ứng dụng cho người dùng cuối.

### 4.1.1 Các công cụ cần cài đặt

**Phía server (PHP – MySQL):**

* **XAMPP** (hoặc phần mềm tương đương) cung cấp dịch vụ Apache và MySQL.
* **phpMyAdmin** để thao tác với cơ sở dữ liệu một cách trực quan.
* **Trình duyệt web** (Chrome, Edge, Firefox...) để truy cập giao diện quản trị.

**Phía client (ứng dụng Flutter):**

* Hệ điều hành Windows 10 trở lên.

### 4.1.2 Các thao tác để cài đặt chương trình

**A. Cài đặt máy chủ:**

1. Cài XAMPP, bật Apache và MySQL.
2. Tạo cơ sở dữ liệu mới trong phpMyAdmin (ví dụ: tracnghiem\_hpc).
3. Import tệp database.sql vào cơ sở dữ liệu.
4. Giải nén mã nguồn PHP vào thư mục htdocs.
5. Cấu hình kết nối DB trong file /include/config.php (dòng 4-8).
6. Truy cập trang quản trị bằng địa chỉ <http://localhost/> hoặc domain đã được cài đặt từ trước.

**B. Triển khai ứng dụng client:**

1. Mở thư mục chứa file thực thi SetupEdudexQ.exe.
2. Sau khi cài đặt xong phần mềm tự động tạo 1 shortcut ở màn hình chính.
3. Chạy phần mềm trên máy để bắt đầu sử dụng.

## 4.2 Kiểm thử

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Cases** | **Expected Result** | **Date** | **Tester** | **Result** | **Pass/Fail** |
| Kiểm tra chức năng đăng nhập: + Đăng nhập với tài khoản hợp lệ | - Đăng nhập thành công vào hệ thống | 21/06/2025 | Đàm Minh Giang | Đăng nhập thành công | Pass |
| Kiểm tra đăng nhập với tài khoản để trống | - Cảnh báo không hợp lệ ở ô nhập tên tài khoản | 21/06/2025 | Đàm Minh Giang | Cảnh báo lỗi hiển thị đúng | Pass |
| Kiểm tra đăng nhập với mật khẩu để trống | - Cảnh báo không hợp lệ ở ô nhập mật khẩu | 21/06/2025 | Đàm Minh Giang | Cảnh báo lỗi hiển thị đúng | Pass |
| Kiểm tra đăng nhập với tài khoản không tồn tại | - Hiển thị thông báo lỗi: “Tên đăng nhập hoặc mật khẩu không đúng” | 21/06/2025 | Đàm Minh Giang | Thông báo lỗi hiển thị đúng | Pass |
| Kiểm tra chức năng thêm câu hỏi mới | - Giao diện thêm câu hỏi hiển thị  - Nhập đúng dữ liệu, hệ thống lưu thành công | 21/06/2025 | Đàm Minh Giang | Câu hỏi được thêm vào CSDL Thông báo " Thêm câu hỏi thành công!" | Pass |
| Kiểm tra thêm câu hỏi khi thiếu nội dung | - Hiển thị thông báo lỗi: “Vui lòng điền vào trường này” | 21/06/2025 | Đàm Minh Giang | Cảnh báo lỗi đúng yêu cầu | Pass |
| Kiểm tra tạo kỳ thi mới | - Giao diện tạo kỳ thi hiển thị  - Lưu kỳ thi thành công và có trong danh sách kỳ thi | 21/06/2025 | Đàm Minh Giang | Kỳ thi mới được tạo và lưu CSDL | Pass |
| Kiểm tra nhập danh sách thí sinh từ file | - Tệp đúng định dạng được hệ thống chấp nhận - Sinh số báo danh tự động | 21/06/2025 | Đàm Minh Giang | Danh sách được nhập, số báo danh được sinh | Pass |
| Kiểm tra nhập file sai định dạng | - Hiển thị lỗi: “Chỉ hỗ trợ file Excel (.xlsx, .xls)!” | 21/06/2025 | Đàm Minh Giang | Lỗi hiển thị đúng | Pass |
| Kiểm tra làm bài thi | - Hiển thị câu hỏi theo đề  - Cho phép chọn đáp án và nộp bài | 21/06/2025 | Đàm Minh Giang | Làm bài thành công, kết quả được ghi nhận | Pass |

# KẾT LUẬN

Sau quá trình nghiên cứu, tìm hiểu và triển khai, đồ án “Xây dựng hệ thống thi trắc nghiệm cho Trường Cao Đẳng Công Nghệ Bách Khoa Hà Nội (HPC)” đã được hoàn thành với đầy đủ các chức năng cơ bản theo đúng yêu cầu đề ra. Hệ thống được phát triển theo mô hình client-server, trong đó phía server đảm nhận vai trò xử lý và lưu trữ dữ liệu trung tâm, còn phía client là nơi tương tác trực tiếp với người dùng.

Phía server được xây dựng bằng PHP, sử dụng MySQL để quản lý cơ sở dữ liệu gồm tài khoản người dùng, kỳ thi, câu hỏi, kết quả... Server cũng cung cấp các API RESTful hỗ trợ xác thực, lấy đề thi, lưu bài làm và xử lý kết quả.

Phía client được xây dựng bằng Flutter, hỗ trợ đa nền tảng, giao diện thân thiện, dễ sử dụng. Ứng dụng được xây dựng hiệu quả cho phép thí sinh đăng nhập, làm bài thi và xem kết quả.

Hệ thống có thể triển khai linh hoạt trong mạng LAN hoặc Internet, đảm bảo tính ổn định, bảo mật và khả năng kiểm soát tốt trong quá trình tổ chức thi. Thiết kế cơ sở dữ liệu theo chuẩn 3NF giúp giảm thiểu dư thừa, tối ưu truy vấn và dễ bảo trì.

Tuy phần lớn các chức năng quan trọng đã được hoàn thiện, hệ thống vẫn còn tiềm năng nâng cấp như: cải thiện giao diện người dùng, tăng cường bảo mật, tích hợp phân tích kết quả và nâng cao tính năng chống gian lận.

Thông qua quá trình thực hiện đồ án, em đã vận dụng hiệu quả các kiến thức chuyên ngành về lập trình, thiết kế hệ thống và cơ sở dữ liệu, đồng thời rèn luyện khả năng giải quyết vấn đề và tư duy triển khai phần mềm thực tế.

Em xin chân thành cảm ơn quý thầy/cô khoa Công Nghệ Thông Tin, đặc biệt là Thầy Nguyễn Hoài Linh – người đã tận tình hướng dẫn, đóng góp nhiều ý kiến quý báu cho em trong suốt quá trình thực hiện đồ án.