**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

TP. HCM

2023

BÙI TÁ TÂN NGỌC MSSV: N19DCCN122 CHUYÊN NGÀNH: CÔNG NGHỆ PHẦN MỀM 2019-2024 Lớp: D19CQCNPM01-N 2019-2024

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**Đề tài: “Xây dựng web application**

**thi trắc nghiệm”**

**Người hướng dẫn : TH.S NGUYỄN TRUNG HIẾU**

**Sinh viên thực hiện : BÙI TÁ TÂN NGỌC**

**Mã số sinh viên : N19DCCN122**

**Lớp : D19CQCNPM01-N**

**Khóa : 2019 – 2024**

**Hệ : Đại học chính quy**

**TP.HỒ CHÍ MINH, NĂM 2023**

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**ĐỒ ÁN**

**TỐT NGHIỆP ĐẠI HỌC**

**Đề tài: “Xây dựng web application**

**thi trắc nghiệm”**

**Người hướng dẫn : TH.S NGUYỄN TRUNG HIẾU**

**Sinh viên thực hiện : BÙI TÁ TÂN NGỌC**

**Mã số sinh viên : N19DCCN122**

**Lớp : D19CQCNPM01-N**

**Khóa : 2019 – 2024**

**Hệ : Đại học chính quy**

**TP.HỒ CHÍ MINH, NĂM 2023**

# LỜI CẢM ƠN

Trong suốt khoảng thời gian hơn bốn năm tại Học viện Công nghệ Bưu chính Viễn thông Cơ sở tại TP. Hồ Chí Minh. Em đã nhận được nhiều sự quan tâm, giúp đỡ của quý Thầy Cô và bạn bè. Với lòng biết ơn sâu sắc và chân thành nhất, Em xin được gửi lời cảm ơn chân thành nhất đến các thầy cô trong Khoa Công nghệ Thông tin 2 của Học viện Công nghệ Bưu chính Viễn thông Cơ sở tại Thành phố Hồ Chí Minh đã cùng với tri thức và tâm huyết của mình để truyền đạt vốn kiến thức quý báu cho em trong suốt thời gian học tập tại trường.

Em xin được gửi lời cảm ơn chân thành nhất đến các thầy cô trong Khoa Công nghệ Thông tin 2 của Học viện Công nghệ Bưu chính Viễn thông Cơ sở tại Thành phố Hồ Chí Minh đã tạo mọi điều kiện thuận lợi để em có thể hoàn thành đồ án tốt nghiệp một cách tốt nhất. Em xin gửi lời cảm ơn chân thành nhất đến thầy phụ trách hướng dẫn trực tiếp là Thạc sĩ Nguyễn Trung Hiếu. Người đã luôn theo dõi, tích cực hỗ trợ, đưa ra những lời khuyên giá trị trong suốt quá trình thực hiện đề em có thể hoàn thành đồ án thực tập tốt nghiệp này thành công.

Trong kì đồ án tốt nghiệp này, em được phân công thực hiện đề tài: “Xây dựng web application thi trắc nghiệm”. Em xin gửi lời cảm ơn chân thành nhất đến thầy phụ trách hướng dẫn trực tiếp là Thạc sĩ Nguyễn Trung Hiếu. Người đã luôn theo dõi, tích cực hỗ trợ, đưa ra những lời khuyên giá trị trong suốt quá trình thực hiện đề tài đề em có thể hoàn thành đồ án thực tập tốt nghiệp này. Sự hướng dẫn tận tình và những lời khuyên hữu ích của thầy sẽ là những kiến thức quý giá để em có thể bước đầu bước đến môi trường làm việc chuyên nghiệp sau này. Với điều kiện thời gian cũng như kiến thức và kinh nghiệm cá nhân còn hạn chế, đồ án này không thể tránh được những thiếu sót. Em rất mong quý thầy cô có thể đưa ra nhận xét, phản hồi cũng như những hướng dẫn cụ thể về mặt chuyên môn để em có thể cải thiện sản phẩm này hoàn chỉnh hơn cũng như học hỏi được nhiều kinh nghiệm cho những dự án em sex tham gia trong tương lai.

Cuối cùng, em xin kính chúc quý thầy cô dồi dào sức khỏe và công tác thật tốt. Em xin chân thành cảm ơn!

Thành phố Hồ Chí Minh, tháng 12/2023

Bùi Tá Tân Ngọc

# MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc147091969)

[MỤC LỤC 4](#_Toc147091970)

[DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH 6](#_Toc147091971)

[KÝ HIỆU CÁC CỤM TỪ VIẾT TẮT 8](#_Toc147091972)

[MỞ ĐẦU 9](#_Toc147091973)

[CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI 10](#_Toc147091974)

[1.1. Tổng quan. 10](#_Toc147091975)

[**1.1.1. Tổng quan về ứng dụng. 10**](#_Toc147091976)

[**1.1.2. Khảo sát yêu cầu nghiệp vụ. 10**](#_Toc147091977)

[1.2. Tìm hiểu đề tài. 10](#_Toc147091978)

[**1.2.1. Lý thuyết. 10**](#_Toc147091979)

[**1.2.2. Thực hành. 10**](#_Toc147091980)

[1.3. Công cụ sử dụng. 11](#_Toc147091981)

[1.4. Mục tiêu đề tài. 11](#_Toc147091982)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ 12](#_Toc147091983)

[2.1. Giới thiệu về Node.js. 12](#_Toc147091984)

[2.2. Giới thiệu về Express.js và Sequelize. 12](#_Toc147091985)

[**2.2.1. Express.js. 12**](#_Toc147091986)

[**2.2.2. Sequeqlize. 12**](#_Toc147091987)

[2.3. Giới thiệu về React. 13](#_Toc147091988)

[2.3. React native. 13](#_Toc147091989)

[2.4. MySQL Server. 13](#_Toc147091990)

[2.5. RESTful API. 13](#_Toc147091991)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 15](#_Toc147091992)

[3.1. Mô hình hóa yêu cầu. 15](#_Toc147091993)

[**3.1.1. Tìm hiểu quy trình nghiệp vụ. 15**](#_Toc147091994)

[**3.1.2. Xác định yêu cầu chức năng. 15**](#_Toc147091995)

[**3.1.3. Các tác nhân của hệ thống. 15**](#_Toc147091996)

[3.2. Sơ đồ usecase. 15](#_Toc147091997)

[**3.2.1. Danh sách các usecase: 15**](#_Toc147091998)

[**3.2.2. Sơ đồ usecase tổng quát. 15**](#_Toc147091999)

[**3.2.3. Đặc tả usecase. 15**](#_Toc147092000)

[3.3. Mô hình thực thể ERD. 15](#_Toc147092001)

[**3.3.1. Xác định các thực thể. 15**](#_Toc147092002)

[**3.3.2. Mô hình thực thể ERD (Entity RelationShip Diagram). 15**](#_Toc147092003)

[3.4. Thiết kế cơ sở dữ liệu. 15](#_Toc147092004)

[**3.4.1. Mô hình diagram của cơ sở dữ liệu. 15**](#_Toc147092005)

[**3.4.2. Từ điển dữ liệu. 15**](#_Toc147092006)

[CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH 16](#_Toc147092007)

[4.1. Lập trình xây dựng ứng dụng. 16](#_Toc147092008)

[4.2. Xây dựng chương trình phía back-end. 16](#_Toc147092009)

[4.3. Xây dựng chương trình phía front-end. 16](#_Toc147092010)

[4.4. Xây dựng ứng dụng di động. 16](#_Toc147092011)

[CHƯƠNG 5: MINH HỌA SẢN PHẨM 17](#_Toc147092012)

[CHƯƠNG 6: KẾT LUẬN 18](#_Toc147092013)

[6.1. Kết luận 18](#_Toc147092014)

[**6.1.1. Những kết quả đạt được 18**](#_Toc147092015)

[**6.1.2. Những hạn chế 18**](#_Toc147092016)

[6.2. Hướng phát triển của sản phẩm 18](#_Toc147092017)

[PHỤ LỤC 19](#_Toc147092018)

[DANH MỤC TÀI LIỆU THAM KHẢO 20](#_Toc147092019)

# DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH

[Bảng 3.1. Yêu cầu chức năng nghiệp vụ đối với admin/gv 14](#_Toc149073420)

[Bảng 3.2. Yêu cầu chức năng nghiệp vụ của sinh viên 16](#_Toc149073421)

[Bảng 3.3. Bảng danh sách các tác nhân của hệ thống 16](#_Toc149073422)

[Bảng 3.4. Bảng mô tả danh sách usecase của hệ thống. 16](#_Toc149073423)

[Bảng 3.5. Bảng mô tả usecase đăng nhập. 18](#_Toc149073424)

[Bảng 3.6. Bảng mô tả usecase quên mật khẩu. 19](#_Toc149073425)

[Bảng 3.7. Bảng mô tả usecase quản lý tài khoản. 20](#_Toc149073426)

[Bảng 3.8. Bảng mô tả usecase quản lý thông tin cá nhân. 21](#_Toc149073427)

[Bảng 3.9. Bảng mô tả usecase quản lý môn học. 22](#_Toc149073428)

[Bảng 3.10. Bảng mô tả usecase quản lý câu hỏi. 23](#_Toc149073429)

[Bảng 3.11. Bảng mô tả usecase quản lý lịch kiểm tra. 24](#_Toc149073430)

[Bảng 3.12. Bảng mô tả usecase quản lý lịch đề thi. 25](#_Toc149073431)

[Bảng 3.13. Bảng mô tả usecase làm bài thi. 26](#_Toc149073432)

[Bảng 3.14. Bảng mô tả usecase xem kết quả. 27](#_Toc149073433)

[Bảng 3.15. Bảng mô tả usecase quản lý lớp tín chỉ. 28](#_Toc149073434)

[Bảng 3.16. Mô tả các thuộc tính trong bảng User 30](#_Toc149073435)

[Bảng 3.17. Mô tả các thuộc tính trong bảng Role 30](#_Toc149073436)

[Bảng 3.18. Mô tả các thuộc tính trong bảng RoleUser 30](#_Toc149073437)

[Bảng 3.19. Mô tả các thuộc tính trong bảng permission 31](#_Toc149073438)

[Bảng 3.20. Mô tả các thuộc tính trong bảng Role\_Permissions 31](#_Toc149073439)

[Bảng 3.21. Mô tả các thuộc tính trong bảng subjects 31](#_Toc149073440)

[Bảng 3.22. Mô tả các thuộc tính trong bảng chapters 32](#_Toc149073441)

[Bảng 3.23. Mô tả các thuộc tính trong bảng questions 32](#_Toc149073442)

[Bảng 3.24. Mô tả các thuộc tính trong bảng tests 32](#_Toc149073443)

[Bảng 3.25. Mô tả các thuộc tính trong bảng test\_schedules 33](#_Toc149073444)

[Bảng 3.26. Mô tả các thuộc tính trong bảng results 34](#_Toc149073445)

[Bảng 3.27. Mô tả các thuộc tính trong bảng groups 34](#_Toc149073446)

[Bảng 3.28. Mô tả các thuộc tính trong bảng test\_groups 34](#_Toc149073447)

[Bảng 3.29. Mô tả các thuộc tính trong bảng group\_details 35](#_Toc149073448)

[Bảng 3.30. Mô tả các thuộc tính trong bảng credit\_classes 35](#_Toc149073449)

[Bảng 3.31. Mô tả các thuộc tính trong bảng test\_details 36](#_Toc149073450)

[Bảng 3.32. Mô tả các thuộc tính trong bảng result\_details 36](#_Toc149073451)

[Bảng 3.33. Mô tả các thuộc tính trong bảng semesters 36](#_Toc149073452)

[Bảng 3.34. Mô tả các thuộc tính trong bảng assigns 36](#_Toc149073453)

[Hình 3.1. Mô hình thực thể ERD 30](#_Toc150795485)

[Hình 3.2. Mô hình diagram của cơ sở dữ liệu 30](#_Toc150795486)

[Hình 5.1. Màn hình đăng nhập. 40](#_Toc150795487)

[Hình 5.2. Màn hình trang tổng quan. 40](#_Toc150795488)

[Hình 5.3. Màn hình trang quản lý người dùng. 41](#_Toc150795489)

[Hình 5.4. Màn hình trang quản lý môn học. 41](#_Toc150795490)

[Hình 5.5. Màn hình trang quản lý câu hỏi. 42](#_Toc150795491)

[Hình 5.6. Màn hình trang tạo mới câu hỏi. 42](#_Toc150795492)

[Hình 5.7. Màn hình trang quản lý lớp tín chỉ. 43](#_Toc150795493)

[Hình 5.8. Màn hình trang phân công giảng viên cho lớp tín chỉ. 43](#_Toc150795494)

[Hình 5.9. Màn hình trang thêm sinh viên cho lớp tín chỉ. 44](#_Toc150795495)

[Hình 5.10. Màn hình trang tạo ca thi. 44](#_Toc150795496)

[Hình 5.11. Màn hình trang tạo đề thi. 45](#_Toc150795497)

[Hình 5.12. Màn hình trang phân đề thi cho lớp tín chỉ. 45](#_Toc150795498)

[Hình 5.13. Màn hình trang chi tiết đề thi. 46](#_Toc150795499)

[Hình 5.14. Màn hình trang kết quả, thống kê. 46](#_Toc150795500)

[Hình 5.15. Màn hình trang đăng nhập và trang chủ trên điện thoại. 47](#_Toc150795501)

[Hình 5.16. Màn hình trang đề thi và kết quả thi. 48](#_Toc150795502)

[Hình 5.17. Màn hình trang làm bài thi. 49](#_Toc150795503)

# KÝ HIỆU CÁC CỤM TỪ VIẾT TẮT

|  |  |
| --- | --- |
| API: Application programming interface | Giao diện lập trình ứng dụng |
| Website | Trang web |
| MSSQL Server | Microsoft SQL Server |
| Interface | Giao diện |
| Class | Lớp |
| SPA | Single page application |
| gv | Giảng viên |
| sv | Sinh viên |
| admin | Quản lý |

# MỞ ĐẦU

Hiện nay, với sự phát triển một cách bùng nổ của công nghệ, các hoạt động thường ngày của chúng ta cũng đã dần chuyển dịch sang hình thức trực tuyến tích hợp với các công nghệ tiên tiến. Ngành giáo dục cũng ngoại lệ.

Trong giáo dục, các bài thi chính là thước đo chính xác đánh giá quá trình học tập của những người học. Các hình thức thi cũng rất đa dạng như tự luận, trắc nghiệm, vấn đáp, .... Trong đấy, hình thức thi trắc nghiệm đang trở thành một xu hướng tất yếu cho rất nhiều kì thi, đặt biệt là trong những năm gần đây Bộ Giáo Dục nước ta đã đưa hình thức thi trắc nghiệm vào trong kỳ thi trung học phổ thông quốc gia với hầu hết các môn. Cùng với đó khi ngành công nghệ thông tin ngày càng phát triển, mạng lưới internet ngày càng lớn mạnh đa số học sinh, sinh viên đều có cơ hội sử dụng, tìm hiểu về các website.

Để góp phần phát triển thêm cho hình thức thi trắc nghiệm em đã nghiên cứu xây dựng ứng dụng thi trắc nghiệm nhằm tạo ra một ứng dụng có thể quản lý điều hành toàn bộ quá trình của một bài thi trắc nghiệm một cách hoàn chỉnh, thân thiện là hiệu quả.

# CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

## 1.1. Tổng quan.

### 1.1.1. Tổng quan về ứng dụng.

Ứng dụng thi trắc nghiệm là một ứng dụng cho phép sinh viên có thể tham gia làm các bài thi trắc nghiệm trực tuyến trên web application hoặc mobile application. Ứng dụng cũng cho admin có thể tạo ngân hàng câu hỏi và giảng viên có thể tạo và quản lý đề thi cho các lớp học mà mình được phân công.

### 1.1.2. Khảo sát yêu cầu nghiệp vụ.

* Qua quá trình tìm hiểu và khảo sát thông tin về nghiệp vụ của một ứng dụng thi trắc nghiệm trực tuyến, tổng kết lại một số yêu cầu nghiệp vụ chính của web app như sau:
  + Nghiệp vụ quản lý:
    - Quản lý ngân hàng câu hỏi.
    - Quản lý giảng viên và sinh viên.
  + Nghiệp vụ quản lý đề thi:
    - Quản lý đề thi, ca thi.
    - Quản lý kết quả bài thi.
  + Nghiệp vụ tham gia thi:
    - Quá trình tham gia làm bài thi.

## 1.2. Tìm hiểu đề tài.

### 1.2.1. Lý thuyết.

* Tìm hiểu nghiệp vụ liên quan đến web application thi trắc nghiệm trực tuyến.
* Tìm hiểu mô hình client server.
* Front-end: React.
* Mobile: React Native
* Back-end: Express.js.
* Tìm hiểu xác thực, phân quyền với JWT.

### 1.2.2. Thực hành.

Tính năng chung:

* + Đăng nhập, quên mật khẩu, quản lý thông tin cá nhân.

Module quản trị:

* + Quản lý sinh viên, giảng viên.
  + Phân công giảng dạy.
  + Dùng để quản lý tình hình chung của cửa hàng và các nhân viên của cửa hàng.

Module quản lý đề thi:

* + Tạo đề thi theo tỉ lệ câu hỏi và độ khó tự chọn.
  + Đặt lịch cho đề thi.
  + Xem kết quả làm bài của các đề thi mình quản lý.
  + Cấm thi.
* Module làm bài thi:
  + Tham gia làm bài thi mà mình được mời.
  + Xem lịch sử kết quả làm bài thi của mình.
  + Cảnh báo hoặc tự động nộp bài khi quy phạm quy chế thi (chuyển trang khác, ...).

## 1.3. Công cụ sử dụng.

* Visual studio.
* Android studio.
* Postman.

## 1.4. Mục tiêu đề tài.

Xây dựng được một ứng dụng quản lý và thi trắc nghiệm trực tuyến hiệu quả và thân thiện với người sử dụng, giúp nâng cao chất lượng giảng dạy của trường. Cụ thể:

* Giúp người quản lý có thể dễ dàng quản lý ngân hàng câu hỏi, giảng viên, sinh viên và phân công giảng dạy.
* Giúp giảng viên tạo và quản lý đề thi trắc nghiệm hiệu quả, đồng thời có thể dễ dàng quản lý kết quả các bài thi thuộc đề thi do giảng viên đấy ra đề.
* Giúp sinh viên có thể tham gia làm bài thi dễ dàng, nhanh chóng.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT VÀ CÔNG NGHỆ

## 2.1. Giới thiệu về Node.js.

Node.js là một một trường thực thi javascript đa nền tảng mã nguồn mở được xây dựng trên V8 Javascript engine, Node.js giúp xậy dựng các ứng dụng web một cách nhanh chóng và hiệu quả. Các ứng dụng được viết ở đồ án này sử dụng các công nghệ như ExpressJs, React, React native đều chạy trên nền Node.js.

Các ứng dụng Node.js chạy đơn luồng, tức là không tạo thêm luồng cho các request mới. Thay vào đó Node.js có cơ chế xử lý bất đồng bộ giúp có thể xử lý nhiều request đồng thời một cách hiệu quả. Khi thực hiện một thao tác, thay vì chặn luồng thực hiện và lãng phí CPU thì Node.js sẽ tiếp tục các hoạt động khác trong lúc chờ phản hồi. Điều này giúp Node.js có thể xử lý hàng nghìn kết nối đồng thời mà không gây ra gánh nặng quản lý luồng đồng thời, điều mà có thể tiềm ẩn nhiều rủi ro đáng kể.

Vì có cả các framework, thư viện front-end và back-end đều chạy trên Node.js mà các nhà phát triển front-end có thể dễ dàng học thêm các kiến thức mới về back-end và ngược lại mà không gặp trở ngại vì sự thay đổi ngôn ngữ lập trình.

## 2.2. Giới thiệu về Express.js và Sequelize.

### 2.2.1. Express.js.

Express.js là một framework phía máy chủ (server-side) được viết bằng javascript và được xây dựng trên cơ sở Node.js. Nó được sử dụng để xây dựng các ứng dụng web hoặc tạo các ứng dụng web Restful API.

Với cú pháp đơn giản và dễ tiếp cận, Express.js cho phép các lập trình viên xây dựng nhanh chóng các ứng dụng web phức tạp, từ các trang web tĩnh đơn giản đến các ứng dụng đa người dùng phức tạp.

Với Express.js, lập trình viên có thể xử lý các yêu cầu HTTP, quản lý định tuyến (routing), xử lý middleware, xử lý các lỗi (error handling), kết nối và làm việc với cơ sở dữ liệu, .... Express.js các phương thức để xử lý các yêu cầu GET, POST, PUT, DELETE và các phương thức HTTP khác.

Với sự linh hoạt và khả năng mở rộng tốt, Express.js đã trở thành một trong những lựa chọn phổ biến cho việc phát triển website phía máy chủ hiện nay.

### 2.2.2. Sequeqlize.

Sequelize là một thư viện Javascript cung cấp giải pháp ORM (Object-relational mapping) cho các cơ sở dữ liệu quan hệ như MySQL, PostgreSQL, SQL Server, giúp cho ứng dụng có thể tương tác với các cơ sở dữ liệu thông qua việc tương tác với các đối tượng và phương thức javascript. Giúp quá trình làm việc với cơ sở dữ liệu trở nên dễ kiểm soát và ứng dụng có thể thích ứng dễ dàng nếu phải chuyển sang dùng một loại cơ sở dữ liệu quan hệ khác.

Sequelize cũng có nhiều ưu điểm khác như hỗ trợ eager và lazy loading, giúp tối ưu hóa hiệu suất của ứng dụng cũng như hỗ trợ việc thực thi các transaction, giúp đảm bảo tính toàn vẹn của dữ liệu trong quá trình thực hiện các thao tác với cơ sở dữ liệu. Đồ án sẽ sử dụng sequelize trong ứng dụng Express.js để làm việc với cơ sở dữ liệu MySQL.

## 2.3. Giới thiệu về React.

React là một thư viện javascript mạnh mẽ được Facebook phát triển. React cho phép các developer có thể sử dụng để tạo ra những ứng dụng phía client (máy khách) một cách nhanh chóng và hiệu quả.

React được xây dựng các ứng dụng web applciation theo hướng SPA (single page application), SPA là một kiểu thiết kế website mà trang tại đấy các thông tin của ứng dụng web chỉ tải một lần duy nhất lúc ban đầu. Sau đó ứng dụng sẽ cập nhật lại những phần thay đổi nếu có yêu cầu mới từ người dùng mà không tải lại toàn bộ ứng dụng như các cách thiết kế trước đây. Điều này giúp ứng dụng xây dựng theo kiểu SPA trở nên thân thiện với người sử dụng hơn.

Một ứng dụng React được hợp thành từ những đơn vị thành phần (component). Chúng sử dụng jsx (Javascript XML), một cú pháp mở rộng cho phép nhúng code jsvascript vào file HTML một cách dễ dàng. Các component sẽ kết nối với nhau tạo nên một ứng dụng web. Khi có những yêu cầu thay đổi, ứng dụng sẽ chỉ thay đổi những thứ cần thay đổi, từ đấy tăng tốc đáng kể hiệu năng của ứng dụng.

## 2.3. React native.

React Native là một framework mã nguồn mở được phát triển bởi Facebook, nó cho phép các lập trình viên tạo các ứng dụng di động chạy được trên cả hệ điều hành Android và iOS với trải nghiệm và hiệu năng như các ứng dụng được viết bằng các công nghệ dành riêng cho từng hệ điều hành (tính native).

React Native tương tự như React, vì vậy cho phép lập trình viên tái sử dụng hầu như 80 – 90% đoạn code thay vì phải viết lại khi áp dụng vào một nền tảng khác. Nên các lập trình viên có thể viết code nhanh hơn và phát hành một bộ code cho cả nền tảng iOS và Android mà chỉ khác biệt nhỏ liên quan đến thiết bị. Điều này sẽ tiết kiệm được thời gian, chi phí, duy trì ít code hơn và ít bugs hơn.

## 2.4. MySQL Server.

MySQL là hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở miễn phí phổ biến hiện nay bên cạnh PostgreSQL và Microsoft SQL Server. MySQL ổn định và dễ sử dụng, có thể hoạt động trên nhiều hệ điều hành và cung cấp một hệ thống lớn các hàm tiện ích mạnh mẽ giúp nó trở thành một trong các hệ quản trị cơ sở dữ liệu được sử dụng nhiều nhất.

## 2.5. RESTful API.

REST là viết tắt của Representational State Transfer. RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource. RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile…) khác nhau giao tiếp với nhau.

Những tiêu chuẩn phổ biến nhất của REST như là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE…) và cách định dạng các URL cho ứng dụng web để quản các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API.

# CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## 3.1. Mô hình hóa yêu cầu.

### 3.1.1. Tìm hiểu quy trình nghiệp vụ.

Web application thi trắc nghiệm là một phần mềm sử dụng để quản lý và thực hiện các kỳ thi trắc nghiệm quy mô học viện. Đối tượng sử dụng ứng dụng gồm người quản lý(admin), giảng viên và sinh viên của học viện.

* Đối với người quản lý (admin): Người quản lý sử dụng ứng dụng để quản lý chung cho ứng dụng, tạo môn học, lớp tín chỉ, bộ câu hỏi ban đầu và quản lý người sử dụng của ứng dụng.
* Đối với giảng viên (gv):
  + Tạo và quản lý đề thi cho các lớp mình được phân công.
  + Quản lý các câu hỏi của mình, kết quả các bài thi mình tạo.
* Đối với sinh viên(sv):
  + Tham gia các kỳ thi mà mình được giao.
  + Xem kết quả của bản thân.
  + Thi thử

Quy trình tạo và làm một bài thi trắc nghiệm:

* **Bước 1:**  Giảng viên truy cập website và chọn tạo đề thi cho lớp môn học mà mình được phân công.
* **Bước 2:** Sinh viên truy cập vào ứng dụng để bắt đầu thực hiện bài thi:

+ Chỉ được thi khi đúng giờ làm bài.

+ Bị cảnh báo hoặc bắt buộc nộp bài thi nếu vi phạm quy chế thi (giảng viên tùy chọn mục này) như nhấn ra ngoài nhiều lần.

+ Nếu khách hàng muốn đặt hàng và thanh toán cho sản phẩm: Chọn địa chỉ nhận hàng và phương thức thanh toán sau đấy bấm nút “Đặt hàng”.

Để có thể thêm sản phẩm vào giỏ hàng cũng như đặt hàng, khách hàng phải thực hiện đăng nhập trước, nếu chưa có tài khoản thì phải đăng ký.

* **Bước 3:** Khi làm xong bài thi nhấn nút nộp bài, nếu hết thời gian làm bài, hệ thống sẽ tự động nộp bài thi.
* **Bước 4:** Sau khi nộp bài thi, trở về trang chủ hoặc có thể xem kết quả làm bài thi nếu như giảng viên tùy chọn cho phép sinh viên xem kết quả lúc tạo đề thi.

### 3.1.2. Xác định yêu cầu chức năng.

Bảng 3.. Yêu cầu chức năng nghiệp vụ đối với admin/gv

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Công việc** | **Loại công việc** | **Quy định/Công thức liên quan** | **Biểu mẫu liên quan** |
| **Nhóm yêu cầu chức năng cho cả giảng viên và admin** | | | | |
| ***Quản lý câu hỏi*** | | | | |
| 1 | Thêm câu hỏi | Lưu trữ | Ràng buộc sửa đủ và đúng kiểu dữ liệu ràng buộc. |  |
| 2 | Sửa câu hỏi | Lưu trữ | Ràng buộc sửa đủ và đúng kiểu dữ liệu ràng buộc. |  |
| 3 | Xóa câu hỏi | Lưu trữ | Chỉ có thể xóa câu hỏi chưa được sử dụng, giảng viên chỉ có thể xóa câu hỏi mình tạo |  |
| ***Quản lý lịch kiểm tra*** | | | | |
| 4 | Thêm lịch kiểm tra | Lưu trữ | Ràng buộc nhập đủ và đúng kiểu dữ liệu. |  |
| 5 | Sửa lịch kiểm tra | Lưu trữ | Ràng buộc sửa đủ và đúng kiểu dữ liệu. |  |
| 6 | Xóa lịch kiểm tra | Lưu trữ | Chỉ được xóa khi ca thi không có bài thi nào. |  |
| ***Quản lý đề kiểm tra*** | | | | |
| 7 | Xem bài kiểm tra | Tra cứu | Xem trước bài kiểm tra |  |
| 8 | Tạo bài kiểm tra | Lưu trữ | Ràng buộc nhập đủ và đúng kiểu dữ liệu. |  |
| 9 | Sửa bài kiểm tra | Lưu trữ | Chỉ được sửa trước giờ thi. |  |
| 10 | Xóa bài kiểm tra | Lưu trữ | Chỉ được xóa bài thi chưa được đăng ký. |  |
| ***Quản lý kết quả*** | | | | |
| 11 | Xem kết quả | Tra cứu | Giảng viên chỉ được xem kết quả bài thi mình ra đề. |  |
| 12 | Xuất kết quả ra file | Lưu trữ | Giảng viên chỉ được xuất kết quả bài thi mình ra đề. |  |
| ***Nhóm chức năng đăng nhập*** | | | | |
| 13 | Đăng nhập | Lưu trữ | Tài khoản được cấp bởi admin |  |
| 14 | Đặt lại mật khẩu | Lưu trữ | Địa chỉ email để đặt lại mật khẩu phải là địa chỉ email dùng để đăng ký tài khoản. |  |
| 15 | Quản lý thông tin tài khoản | Lưu trữ | Cho phép thay đổi thông tin cá nhân. |  |
| ***Yêu cầu chức năng nghiệp vụ của admin*** | | | | |
| ***Quản lý tài khoản quản trị*** | | | | |
| 16 | Thêm mới tài khoản | Lưu trữ | Ràng buộc nhập đủ và đúng kiểu dữ liệu ràng buộc về tên đăng nhập, email, hình ảnh, số điện thoại, địa chỉ, mật khẩu. |  |
| 17 | Sửa tài khoản | Lưu trữ | Ràng buộc sửa đủ và đúng kiểu dữ liệu ràng buộc về tên đăng nhập, email, hình ảnh, số điện thoại, địa chỉ, mật khẩu. |  |
| 18 | Xóa tài khoản | Lưu trữ | Không thể xóa tài khoản đang đăng nhập |  |
| 19 | Tìm kiếm tài khoản | Tra cứu | Tìm kiếm tài khoản theo tên đăng nhập, email, số điện thoại, địa chỉ. |  |
| 20 | Cấp quyền cho tài khoản | Lưu trữ | Cấp quyền cho tài khoản |  |

Bảng 3.. Yêu cầu chức năng nghiệp vụ của sinh viên

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TT** | **Công việc** | **Loại công việc** | **Quy định/ Công thức liên quan** | **Biểu mẫu liên quan** |
| **Yêu cầu chức năng nghiệp vụ cho sinh viên** | | | | |
| 1 | Xem bài kiểm tra và lịch kiểm tra của mình | Tra cứu |  |  |
| 2 | Tham gia làm bài kiểm tra | Lưu trữ |  |  |
| 3 | Xem kết quả | Tra cứu |  |  |

### 3.1.3. Các tác nhân của hệ thống.

Bảng 3.. Bảng danh sách các tác nhân của hệ thống

|  |  |  |  |
| --- | --- | --- | --- |
| **TT** | **Tác nhân** | **Ý nghĩa** | **Mô tả** |
| 1 | Quản lý  (admin) | Người chịu trách nhiệm quản lý điều hành hệ thống. | - Quản lý tài khoản.  - Quản lý phân quyền.  - Quản lý câu hỏi  - Quản lý môn học.  - Quản lý lớp học. |
| 2 | Giảng viên  (gv) | Người chịu trách nhiệm quản lý các bài thi của môn học mà mình được phân công. | - Tạo đề thi.  - Tạo câu hỏi  - Xem kết quả thi. |
| 4 | Sinh viên  (sv) | Người sử dụng ứng dụng để làm bài thi của mình. | - Xem lịch thi.  - Làm bài thi.  - Xem kết quả thi |

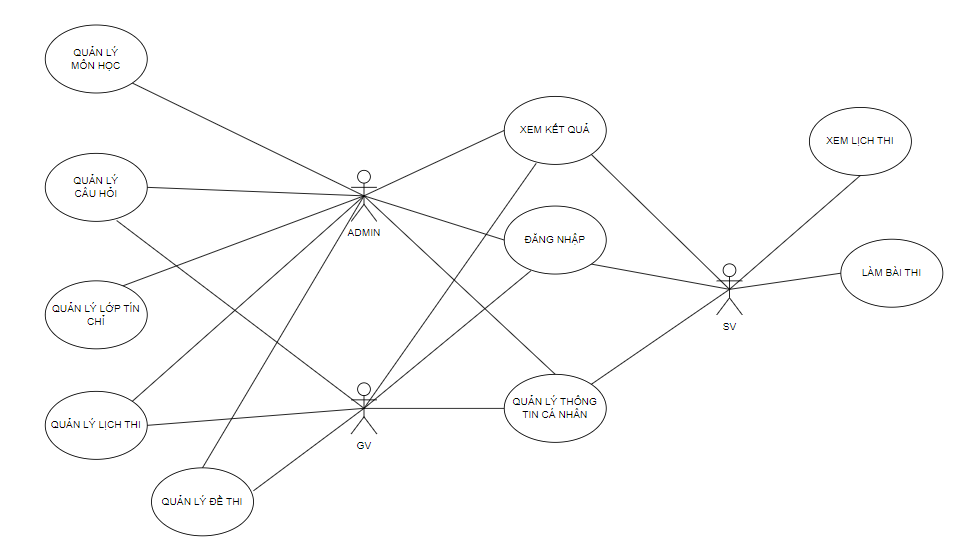
## 3.2. Sơ đồ usecase.

Bảng 3.. Bảng mô tả danh sách usecase của hệ thống.

|  |  |  |
| --- | --- | --- |
| **TT** | **Tên Use case** | **Mô tả** |
| UC01 | Đăng nhập | Use case này mô tả các bước đăng nhập của các actor vào hệ thống. |
| UC02 | Quên mật khẩu | Use case này sử dụng khi actor quên mật khẩu, actor sẽ đổi lại mật khẩu mới bằng việc gửi email. |
| UC03 | Quản lý tài khoản. | Use case này cho phép admin xem/xóa tài khoản giảng viên, sinh viên. |
| UC04 | Chỉnh sửa thông tin cá nhân | Use case này sử dụng khi actor muốn thay đổi các thông tin cá nhân của mình. |
| UC05 | Quản lý môn học | Use case này cho phép admin tạo/xem/xóa môn học, chương. |
| UC06 | Quản lý câu hỏi | Use case này cho phép admin và gv tạo/xem/xóa câu hỏi |
| UC07 | Quản lý lịch kiểm tra | Use case này cho phép admin và gv tạo/xem/xóa lịch thi. |
| UC08 | Quản lý đề thi | Use case này cho phép admin và gv tạo/xem/xóa đề thi. |
| UC10 | Làm bài thi | Use case này cho phép sv tham gia làm bài thi. |
| UC11 | Xem kết quả | Use case này cho phép admin, gv, sv xem kết quả bài thi. |
| UC12 | Quản lý lớp tín chỉ | Use case này cho phép admin tạo/xem/xóa lớp tín chỉ. |

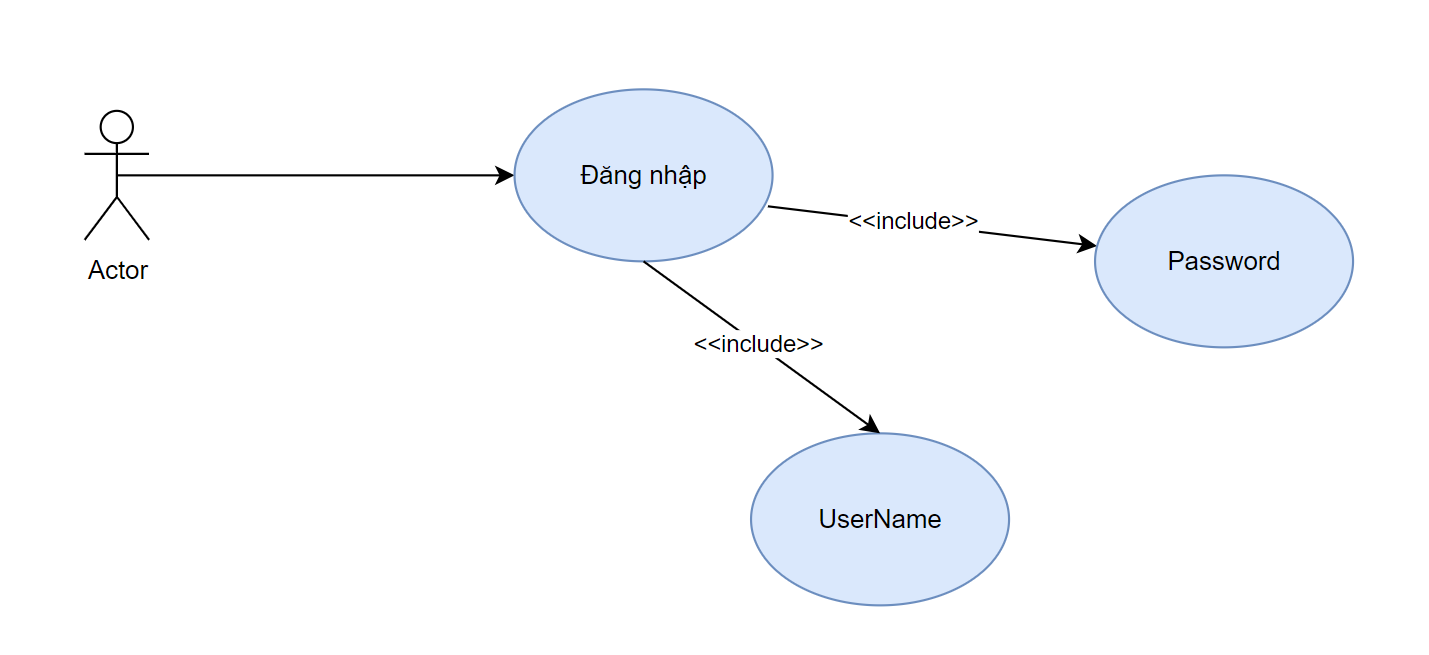
### 3.2.1. Danh sách các usecase:

### 3.2.2. Sơ đồ usecase tổng quát.



### 3.2.3. Đặc tả usecase.

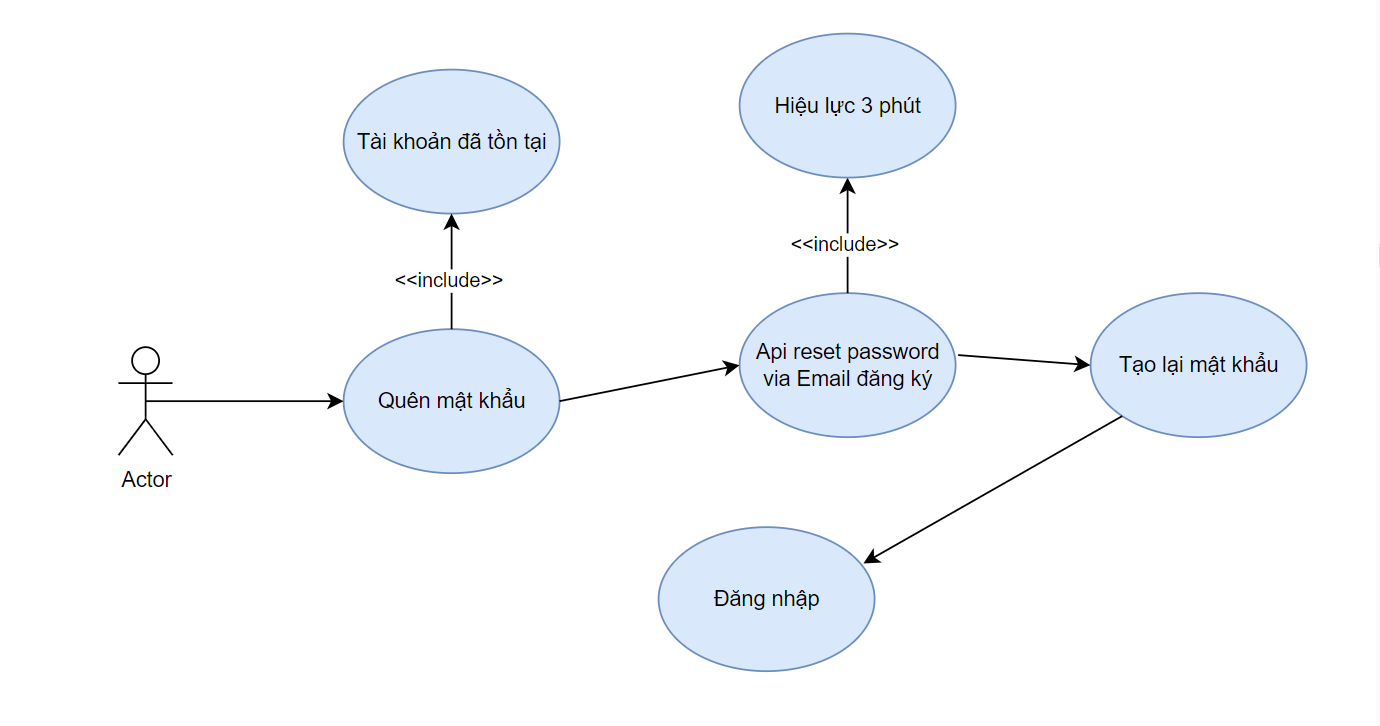
#### 3.2.3.1. Usecase “Đăng nhập”



Bảng 3.. Bảng mô tả usecase đăng nhập.

|  |  |
| --- | --- |
| **Tên use-case** | Đăng nhập |
| **Tác nhân** | User (admin, gv, sv) |
| **Tiền điều kiện** | Đã có tài khoản để truy cập hệ thống |
| **Hậu điều kiện** | Đăng nhập thành công. |
| **Mô tả** | Đăng nhập vào hệ thống. |
| **Kịch bản:**  1. Click button Đăng nhập ở trang đăng nhập.  2. Nhập các thông tin vào form đăng nhập (email, password).  3. Click Đăng nhập  4. Hệ thống kiểm tra thông tin hợp lệ, đầy đủ hay chưa.  5.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**:  4.1. Các trường chưa được điền đầy đủ thông tin, không đúng định dạng  5.1. Địa chỉ email đã được sử dụng.  5.2. Địa chỉ email không tồn tại. | |

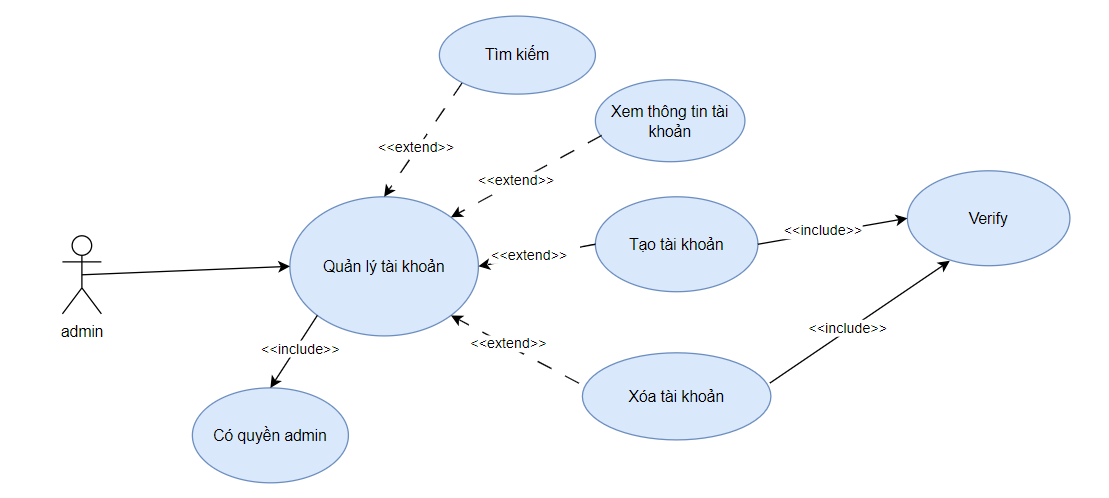
#### 3.2.3.2. Usecase “Quên mật khẩu”



Bảng 3.. Bảng mô tả usecase quên mật khẩu.

|  |  |
| --- | --- |
| **Tên use-case** | Quên mật khẩu |
| **Tác nhân** | User (gv, sv) |
| **Tiền điều kiện** | Đã có tài khoản để truy cập hệ thống |
| **Hậu điều kiện** | Reset mật khẩu thành công. |
| **Mô tả** | Tạo lại mật khẩu khi quên mật khẩu |
| **Kịch bản:**  1. Click button quên mật khẩu.  2. Nhập email đăng ký.  3. Hệ thống kiểm tra và gửi mail đặt lại mật khẩu mới.  4. User truy cập và đặt lại mật khẩu.  5.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**:  2.1 Email không tồn tại.  3.1. Mật khẩu mới không đúng định dạng. | |

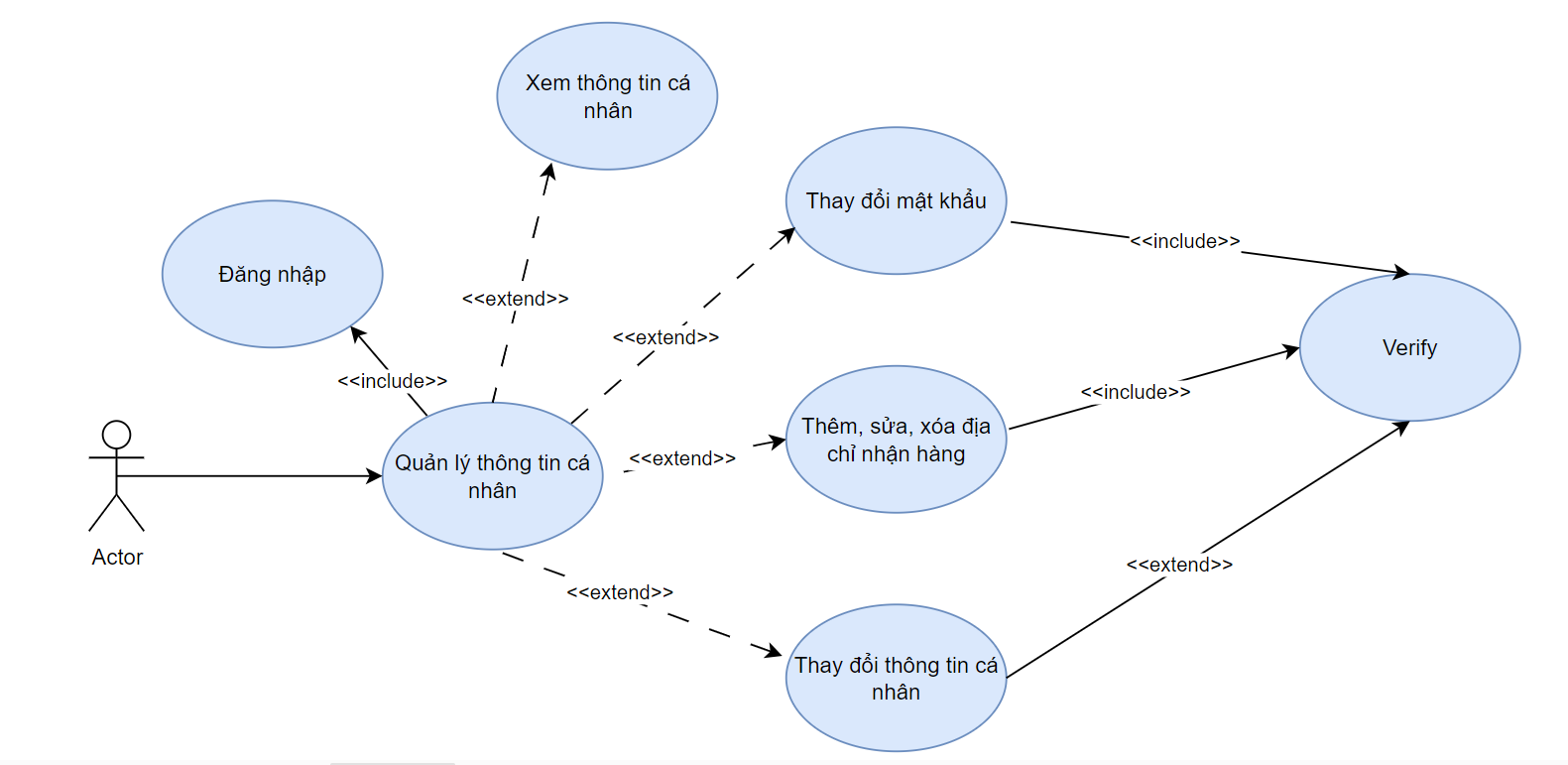
#### 3.2.3.3. Usecase “Quản lý tài khoản”



Bảng 3.. Bảng mô tả usecase quản lý tài khoản.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý tài khoản |
| **Tác nhân** | Admin |
| **Tiền điều kiện** | Đã đăng nhập thành công |
| **Hậu điều kiện** | Quản lý được danh sách giảng viên, sinh viên |
| **Mô tả** | Quản lý tài khoản |
| **Kịch bản:**  1. Click vào quản lý tài khoản.  2.Click vào một tài khoản cụ thể.  3. Xem hoặc sửa, xóa thông tin của tài khoản.  4. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**: | |

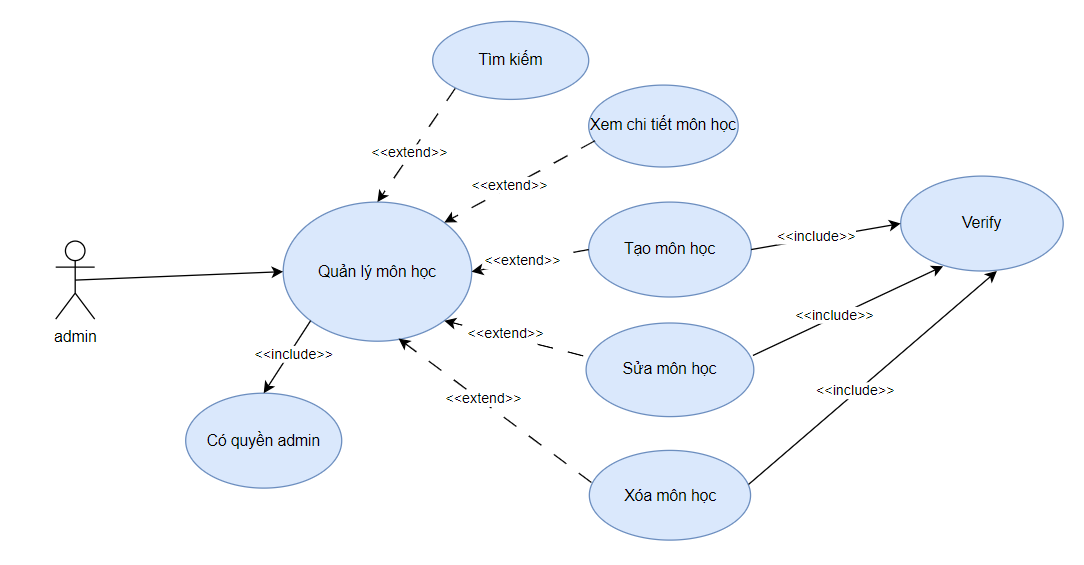
#### 3.2.3.4. Usecase “Quản lý thông tin cá nhân”



Bảng 3.. Bảng mô tả usecase quản lý thông tin cá nhân.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý thông tin cá nhân |
| **Tác nhân** | User (admin, gv, sv) |
| **Tiền điều kiện** | Đăng nhập thành công |
| **Hậu điều kiện** | Xem là quản lý được thông tin các nhân |
| **Mô tả** | Quản lý thông tin cá nhân |
| **Kịch bản:**  1. Click button thông tin user  2. Chọn thêm, sửa thông tin cá nhân  3. Nhập thông tin (nếu được yêu cầu).  4.Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**:  3.1. Thông tin nhập không đúng định dạng. | |

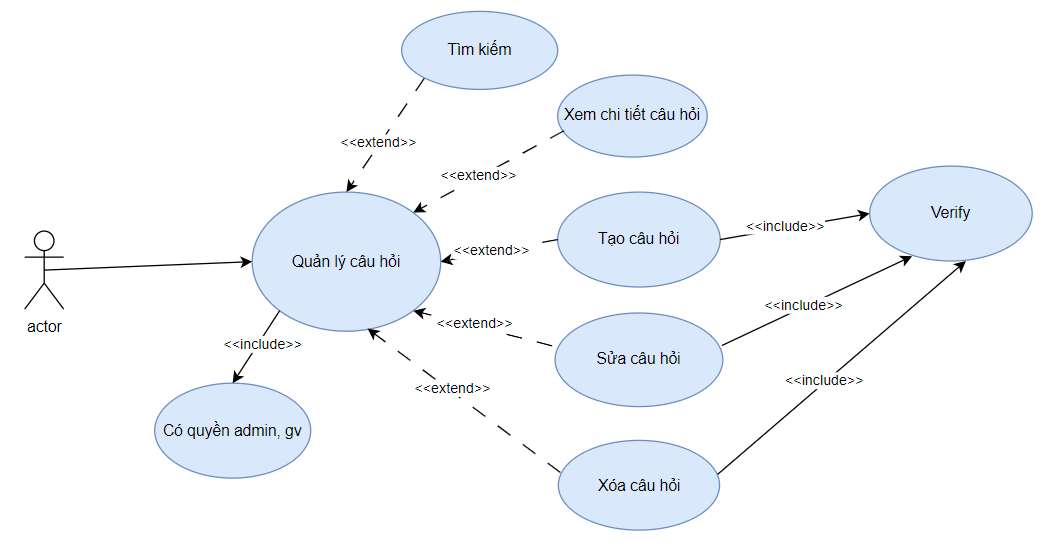
#### 3.2.3.5. Usecase “Quản lý môn học”



Bảng 3.. Bảng mô tả usecase quản lý môn học.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý môn học |
| **Tác nhân** | Admin |
| **Tiền điều kiện** |  |
| **Hậu điều kiện** | Quản lý thông tin môn học |
| **Mô tả** | Xem, thêm, sửa, xóa môn học, chương. |
| **Kịch bản:**  1. Click môn học.  2. Chọn thêm, sửa, xóa, xem, ...  3.Điền thông tin nếu yêu cầu.  4. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  3.1. Thông tin không đúng định dạng | |

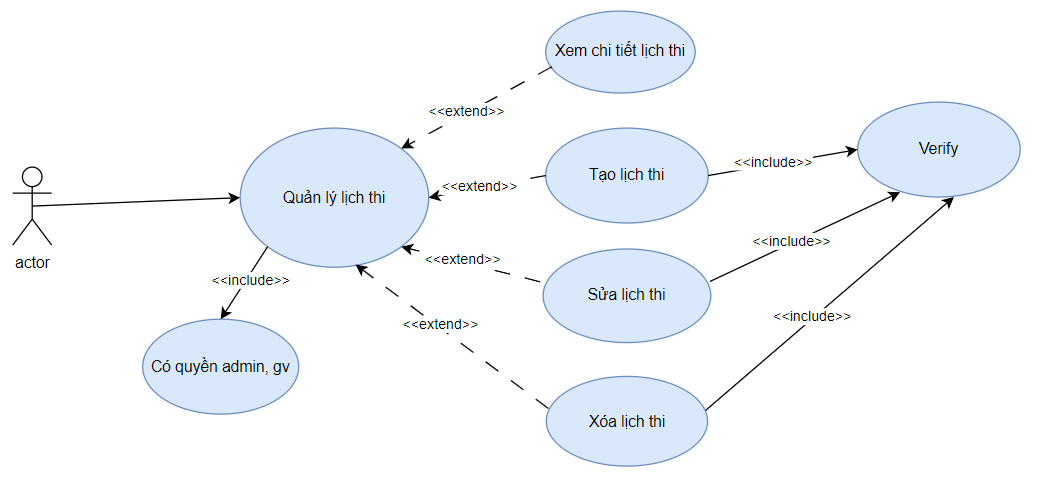
#### 3.2.3.6. Usecase “Quản lý câu hỏi”



Bảng 3.. Bảng mô tả usecase quản lý câu hỏi.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý câu hỏi |
| **Tác nhân** | Admin, gv. |
| **Tiền điều kiện** |  |
| **Hậu điều kiện** | Xem là quản lý câu hỏi. |
| **Mô tả** | Quản lý câu hỏi. |
| **Kịch bản:**  1. Click câu hỏi.  2. Chọn thêm, sửa, xóa, xem, ...  3.Điền thông tin nếu yêu cầu.  4. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  3.1. Thông tin nhập không đúng định dạng.  4.1. Thông tin không đúng ràng buộc. | |

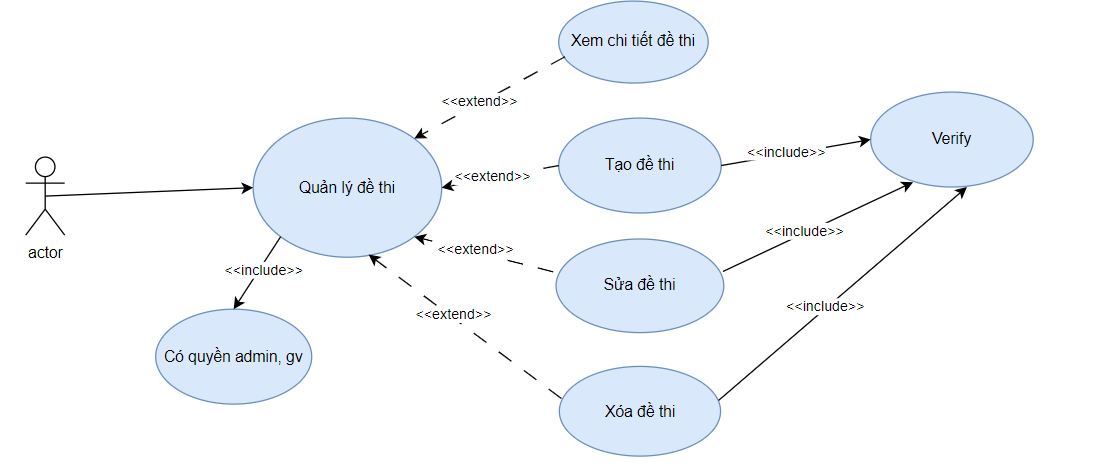
#### 3.2.3.7. Usecase “Quản lý lịch kiểm tra”



Bảng 3.. Bảng mô tả usecase quản lý lịch kiểm tra.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý lịch kiểm tra |
| **Tác nhân** | Admin, gv |
| **Tiền điều kiện** |  |
| **Hậu điều kiện** | Quản lý lịch kiểm tra |
| **Mô tả** | Mua hàng |
| **Kịch bản:**  1. Click đề kiểm tra.  2. Chọn thêm, sửa, xóa, xem, ...  3.Điền thông tin nếu yêu cầu.  4. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  3.1. Thông tin nhập không đúng định dạng.  4.1. Thông tin không đúng ràng buộc. | |

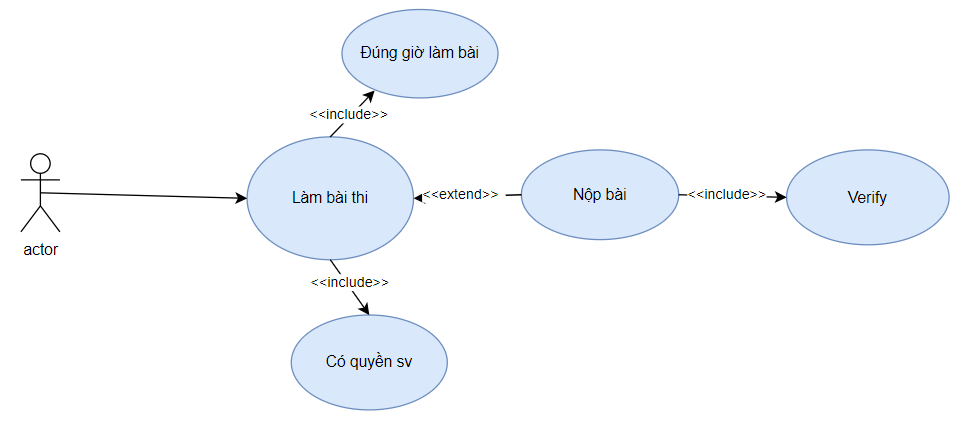
#### 3.2.3.8. Usecase “Quản lý đề thi”



Bảng 3.. Bảng mô tả usecase quản lý lịch đề thi.

|  |  |
| --- | --- |
| **Tên use-case** | Quản lý lịch đề thi |
| **Tác nhân** | Admin, gv |
| **Tiền điều kiện** | Đăng nhập thành công. |
| **Hậu điều kiện** |  |
| **Mô tả** | Quản lý lịch đề thi |
| **Kịch bản:**  1. Click đề kiểm tra.  2. Chọn thêm, sửa, xóa, xem, ...  3.Điền thông tin nếu yêu cầu.  4. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  3.1. Thông tin nhập không đúng định dạng.  4.1. Thông tin không đúng ràng buộc. | |

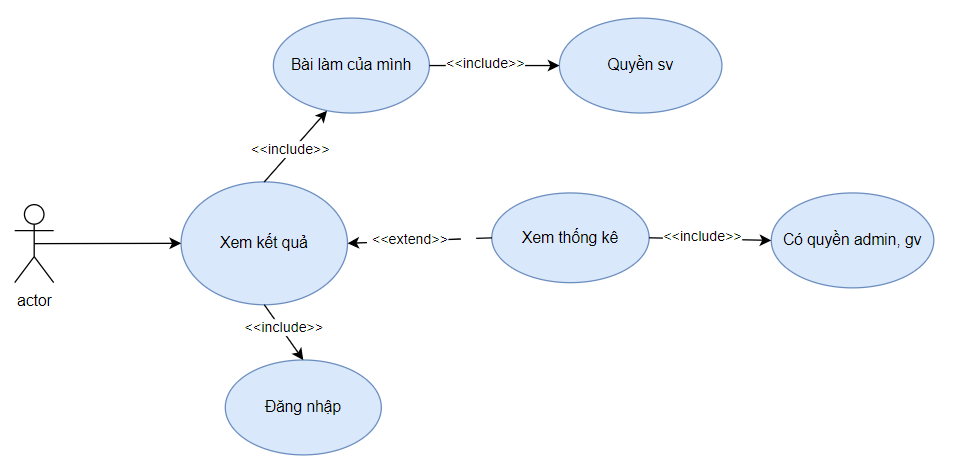
#### 3.2.3.9. Usecase “Làm bài thi”



Bảng 3.. Bảng mô tả usecase làm bài thi.

|  |  |
| --- | --- |
| **Tên use-case** | Làm bài thi |
| **Tác nhân** | Sv |
| **Tiền điều kiện** | Đăng nhập thành công |
| **Hậu điều kiện** | Làm bài thi thành công |
| **Mô tả** | Làm bài thi |
| **Kịch bản:**  1. Click vào bài thi.  2.Click vào làm bài thi  3. Làm bài thi và click vào nộp bài.  3. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  3.1. Vi phạm quy chế thi. | |

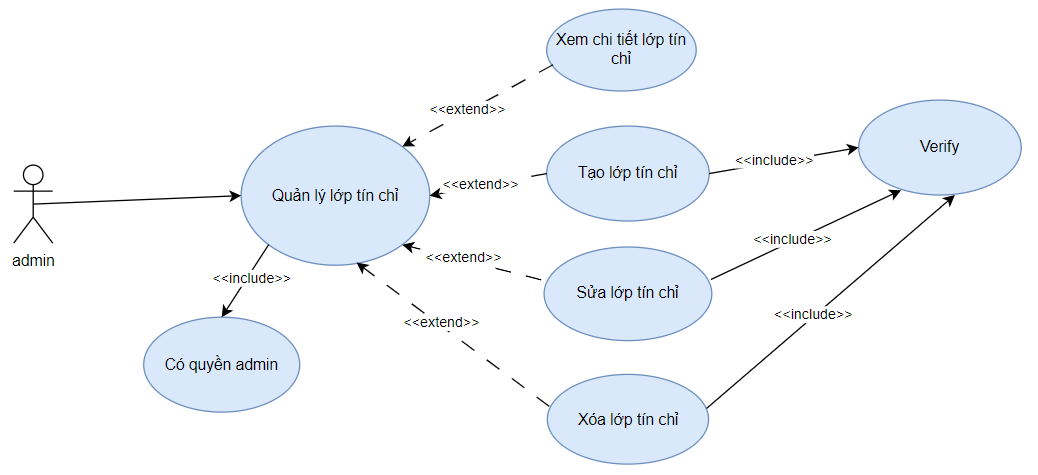
#### 3.2.3.10. Usecase “Xem kết quả”



Bảng 3.. Bảng mô tả usecase xem kết quả.

|  |  |
| --- | --- |
| **Tên use-case** | Xem kết quả |
| **Tác nhân** | Admin, gv, sv |
| **Tiền điều kiện** | Đã đăng nhập thành công |
| **Hậu điều kiện** | Xem kết quả bài thi |
| **Mô tả** | Xem kết quả |
| **Kịch bản:**  1. Click vào kết quả.  2.Click vào kết quả cụ thể.  3. Hệ thống xử lý và trả về kết quả | |
| **Ngoại lệ**: | |

#### 3.2.3.11. Usecase “Quản lý lớp tín chỉ”



Bảng 3.. Bảng mô tả usecase quản lý lớp tín chỉ.

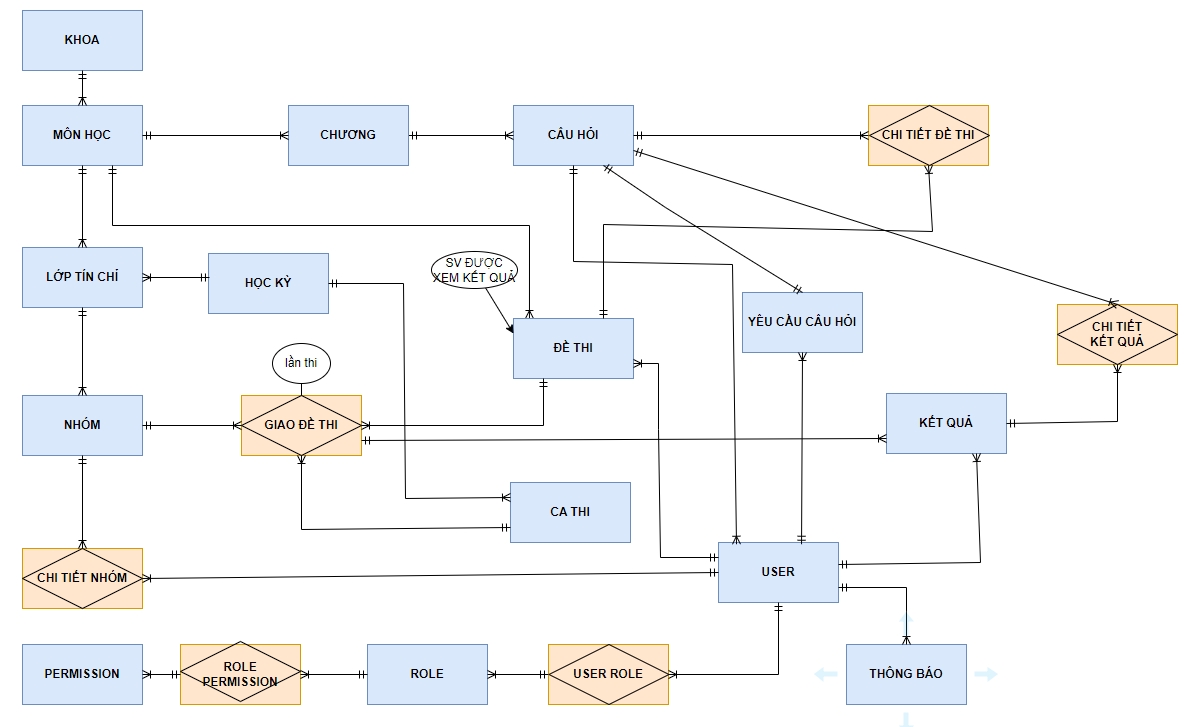
|  |  |
| --- | --- |
| **Tên use-case** | Quản lý lớp tín chỉ |
| **Tác nhân** | Admin |
| **Tiền điều kiện** | Đã đăng nhập |
| **Hậu điều kiện** | Quản lý được thông tin lớp tín chỉ. |
| **Mô tả** | Quản lý lớp tín chỉ |
| **Kịch bản:**  1. Click lớp tín chỉ.  2. Chọn thêm, sửa, xóa, xem, ...  3.Điền thông tin nếu yêu cầu.  4. Hệ thống xử lý và trả về kết quả. | |
| **Ngoại lệ**:  3.1. Thông tin nhập không đúng định dạng.  4.1. Thông tin không đúng ràng buộc. | |

## 3.3. Mô hình thực thể ERD.

### 3.3.1. Xác định các thực thể.

* User (Id, FirstName, LastName, PasswordHash, RefreshToken).
* Role (Id, Name, Normalize).
* Permission (Id, Name, Normalize).
* Khoa (Id, Name).
* Subject (Id, Name).
* Chapter (Id, Name, Index);
* CreditClass (Id, Name, Quantity).
* Group (Id, Name).
* TestSchedule (Id, Name, Date).
* Test (Id, Name, Time, Ban).
* Question (Id, Name, AnswerA, AnswerB, AnswerC, AnswerD, CorrectAnswer).
* Result (Id, Name, Mark, SubmitTime).
* Semester (Id, Year, Semester).

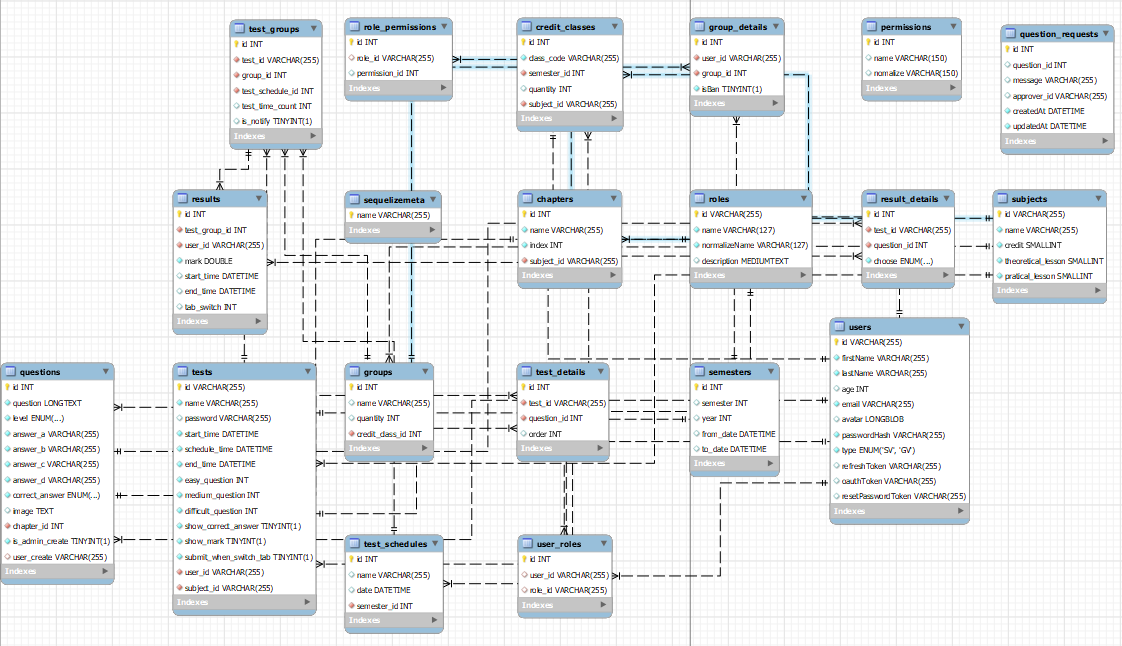
### 3.3.2. Mô hình thực thể ERD (Entity RelationShip Diagram).



Hình 3.. Mô hình thực thể ERD

## 3.4. Thiết kế cơ sở dữ liệu.

### 3.4.1. Mô hình diagram của cơ sở dữ liệu.



Hình 3.. Mô hình diagram của cơ sở dữ liệu

### 3.4.2. Từ điển dữ liệu.

#### 3.4.2.1. Bảng User

Bảng 3.. Mô tả các thuộc tính trong bảng User

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar | 255 | Mã Id | Not null  PrimaryKey |
| 2 | FirstName | nvarchar | 255 | Họ | Not null |
| 3 | LastName | nvarchar |  | Tên | Not null |
| 4 | Email | nvarchar | 45 | Địa chỉ email | Not null |
| 5 | Avatar | LONGBLOB |  | Ảnh đại diện |  |
| 6 | PasswordHash | nvarchar | 45 | Mật khẩu được mã hóa | Not null |
| 7 | OauthToken | Nvarchar | 255 |  |  |
| 8 | RefeshToken | nvarchar | 255 |  |  |
| 9 | Type | ENUM | GV, SV | Tên đăng nhập | Not null |
| 10 | IsBlock | Bit | 1 | Khóa tài khoản | Not null |
| 11 | Resetpassword Token | nvarchar | 255 | Token đặt lại mật khẩu |  |
| 12 | Age | int |  | Tuổi |  |

#### 3.4.2.2. Bảng Role

Bảng 3.. Mô tả các thuộc tính trong bảng Role

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar |  | Id | Not null  PrimaryKey |
| 2 | Name | varchar | 255 | Mã vai trò | Not null |
| 3 | NormalizedName | varchar | 255 | Mã vai trò in hoa | Not null |
| 4 | Description | text |  | Mô tả |  |

#### 3.4.2.3. Bảng UserRole

Bảng 3.. Mô tả các thuộc tính trong bảng RoleUser

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | UserId | nvarchar | 255 | Mã người dùng | Primary Key, Foreign Key,  Not null |
| 2 | RoleId | nvarchar | 255 | Mã vai trò |

#### 3.4.2.4. Bảng permissions

Bảng 3.. Mô tả các thuộc tính trong bảng permission

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | int |  | Id | Not null  PrimaryKey |
| 2 | Name | varchar | 255 | Tên | Not null |
| 3 | NormalizedName | varchar | 255 | In hoa |  |

#### 3.4.2.5. Bảng Role\_Permissions

Bảng 3.. Mô tả các thuộc tính trong bảng Role\_Permissions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | int |  | Khóa | Primary key |
| 2 | Role\_id | nvarchar | 255 | Mã vai trò | Foreign Key  Not null |
| 3 | Permission\_id | Int |  | Mã quyền | Foreign Key  Not null |

#### 3.4.2.6. Bảng subjects

Bảng 3.. Mô tả các thuộc tính trong bảng subjects

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | id | nvarchar |  | Id | Not null  PrimaryKey |
| 2 | name | Nvarchar | 255 | Tên môn học | Not null |
| 3 | credit | int |  | Số tín chỉ | Not null |
| 4 | thecretical\_lesson | int |  | Tiết lý thuyết | Not null |
| 5 | practical\_lesson | int |  | Tiết thực hành | Not null |

#### 3.4.2.7. Bảng chapters

Bảng 3.. Mô tả các thuộc tính trong bảng chapters

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | id | int |  | Id | Not null  PrimaryKey |
| 2 | name | nvarchar | 255 | Tên chương | Not null |
| 3 | index | int |  | Thứ tự | Not null |
| 4 | subject\_id | Varchar | 255 | Mã môn học | Foreign Key Not null |

#### 3.4.2.8. Bảng questions

Bảng 3.. Mô tả các thuộc tính trong bảng questions

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar | 255 | Id | Not null  PrimaryKey |
| 2 | Question | nvarchar | 255 | Tên câu hỏi | Not null |
| 3 | Level | Enum |  | Độ khó | Not null |
| 4 | answer\_a | nvarchar | 255 | Câu A | Not null |
| 5 | answer\_b | nvarchar | 255 | Câu B | Not null |
| 6 | answer\_c | nvarchar | 255 | Câu C | Not null |
| 7 | answer\_d | nvarchar | 255 | Câu D | Not null |
| 8 | correct\_answer | Enum |  | Đáp án | Not null |
| 9 | Image | Text |  | Ảnh |  |
| 10 | Chapter\_id | Int |  | Mã chương | Foreign Key Not null |
| 11 | Is\_admin\_create | Bool |  | Tạo bởi admin |  |
| 12 | User\_create | Varchar | 255 | Người tạo | Foreign Key Not null |

#### 3.4.2.9. Bảng tests

Bảng 3.. Mô tả các thuộc tính trong bảng tests

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | nvarchar | 255 | Id | Not null  PrimaryKey |
| 2 | Name | nvarchar | 255 | Tên câu hỏi | Not null |
| 3 | Password | Enum |  | Độ khó | Not null |
| 4 | Start\_time | date |  | Bắt đầu |  |
| 5 | Schedule\_time | Date |  |  |  |
| 6 | End\_time | Date |  | Kết thúc |  |
| 7 | Easy\_question | Int |  | Câu dễ |  |
| 8 | Medium\_question | Int |  | Câu trung bình |  |
| 9 | Difficult\_question | int |  | Câu khó |  |
| 10 | Show\_correct\_answer | Bool |  |  |  |
| 11 | Show\_mark | Bool |  |  |  |
| 12 | Submit\_when\_switch\_tab | bool |  |  |  |
| 13 | User\_id | varchar | 255 | Người tạo | Foreignkey  Not null |
| 14 | Subject\_id | varchar | 255 | Môn học | Foreignkey  Not null |
| 15 | Test\_schedule\_id | int |  | Lịch thi | Foreignkey  Not null |

#### 3.4.2.10. Bảng test\_schedules

Bảng 3.. Mô tả các thuộc tính trong bảng test\_schedules

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifỉer |  | Mã Id | Not null  PrimaryKey |
| 2 | Name | nvarchar |  | Tên | Not null |
| 3 | Date | date |  | Ngày | Not null |
| 4 | Semester\_id | int | 255 | Mã học kỳ | Foreignkey  Not null |

#### 3.4.2.11. Bảng results

Bảng 3.. Mô tả các thuộc tính trong bảng results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | int |  | Id | Not null  PrimaryKey |
| 2 | Mark | double | 255 | Ngày tạo | Not null |
| 3 | Start\_time | date |  | Bắt đầu | Not null |
| 4 | End\_time | Date |  | Id | Not null |
| 5 | Tab\_switch | Int |  | Số lần chuyển tab | Not null |
| 6 | Test\_group\_id | int |  | Mã nhóm | Foreign Key Not null |
| 7 | User\_id | varchar | 255 | Người dùng | Foreign Key Not null |

#### 3.4.2.12. Bảng groups

Bảng 3.. Mô tả các thuộc tính trong bảng groups

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | Int |  |  | Not null  PrimaryKey |
| 2 | Name | Varchar |  |  | Foreign Key, Not null |
| 3 | Quantity | int |  |  | Not null |
| 4 | Subject\_id | int |  |  | Foreign Key, Not null |

#### 3.4.2.13. Bảng test\_groups

Bảng 3.. Mô tả các thuộc tính trong bảng test\_groups

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | int |  | Mã Id | Not null  PrimaryKey |
| 2 | Test\_id | Int |  | Mã đơn đặt hàng | Foreign Key  Not null |
| 3 | Group\_id | Int |  | Mã trạng thái | Foreign Key  Not null |
| 4 | Test\_schedule\_id | Int |  |  | Foreign Key Not null |
| 5 | Test\_time\_count | Int |  |  |  |
| 6 | Is\_notify | Bool |  |  |  |

#### 3.4.2.14. Bảng group\_details

Bảng 3.. Mô tả các thuộc tính trong bảng group\_details

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** |  | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | Int |  |  | Id | PrimaryKey  Not null |
| 2 | User\_id | Varchar | 255 |  |  | Foreignkey  Not null |
| 3 | Group\_id | Int |  |  |  | Foreignkey  Not null |
| 4 | Is\_ban | bool |  |  |  |  |

#### 3.4.2.15. Bảng credit\_classes

Bảng 3.. Mô tả các thuộc tính trong bảng credit\_classes

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | Int |  | Id | PrimaryKey  Not null |
| 2 | Class\_code | Nvarchar | 255 | Mã lớp | Not null |
| 3 | Subject\_id | Int |  | Mã môn học | Foreignkey  Not null |
| 4 | Quantity | Int |  | Số lượng | Not null |
| 5 | Semester | Int |  | Học kỳ | Foreignkey  Not null l |

#### 3.4.2.16. Bảng test\_details

Bảng 3.. Mô tả các thuộc tính trong bảng test\_details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | Int |  | Id | PrimaryKey  Not null |
| 2 | Test\_id | Int |  | Mã đề thi | Foreignkey  Not null |
| 3 | Question\_id | Int |  | Mã câu hỏi | Foreignkey  Not null |
| 4 | Order | Int |  | Thứ tự | Not null |

#### 3.4.2.17. Bảng result\_details

Bảng 3.. Mô tả các thuộc tính trong bảng result\_details

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | uniqueidentifier |  | Id | PrimaryKey  Not null |
| 2 | Test\_id | Int |  |  | Foreignkey  Not null |
| 3 | Question\_id | Int |  |  | Foreignkey  Not null |
| 4 | Choose | Enum |  |  |  |

#### 3.4.2.18. Bảng semesters

Bảng 3.. Mô tả các thuộc tính trong bảng semesters

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | int |  | Id | PrimaryKey  Not null |
| 2 | Semester | Int |  |  | Not null |
| 3 | year | Int |  |  | Not null |
| 4 | From\_date | Date |  |  |  |
| 5 | To\_date | Date |  |  |  |

#### 3.4.2.19. Bảng assigns

Bảng 3.. Mô tả các thuộc tính trong bảng assigns

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TT** | **THUỘC TÍNH** | **KIỂU** | **MIỀN GIÁ TRỊ** | **Ý NGHĨA** | **GHI CHÚ** |
| 1 | Id | Int |  | Id | PrimaryKey  Not null |
| 2 | Credit\_class\_id | Int |  |  | Foreignkey  Not null |
| 3 | User\_id | Varchar | 255 |  | Foreignkey  Not null |

# CHƯƠNG 4: XÂY DỰNG CHƯƠNG TRÌNH

## 4.1. Lập trình xây dựng ứng dụng.

Công nghệ được sử dụng để xây dựng chương trình là Express.js để viết API cho phần back-end của webbsite, React.js cho phẩn front-end của website và React Native cho ứng dụng di động. Sử dụng hệ quản trị cơ sở dữ liệu MySQL để lưu trữ dữ liệu.

## 4.2. Xây dựng chương trình phía back-end.

Phẩn back-end của website sử dụng Express.js để viết các API cung cấp cho phía front-end sử dụng để tạo nên một sản phẩm hoàn chỉnh.

## 4.3. Xây dựng chương trình phía front-end.

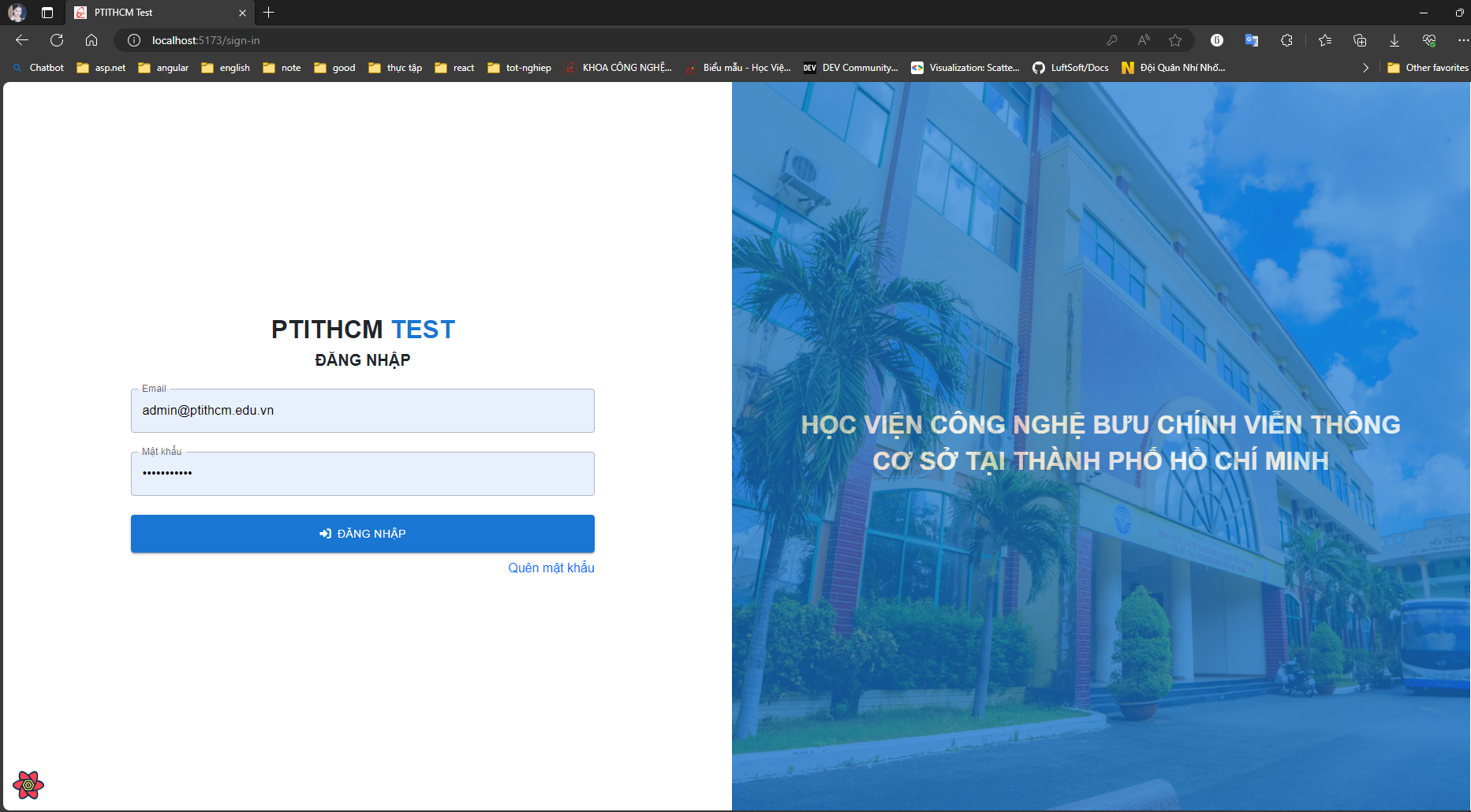
Ứng dụng phía client sử dụng framework React và các thư viện được cung cấp liên quan để xây dựng.

Ứng dụng sử dụng thư viện axios để gọi các API đến back-end để nhận, gửi các dữ liệu liên quan.

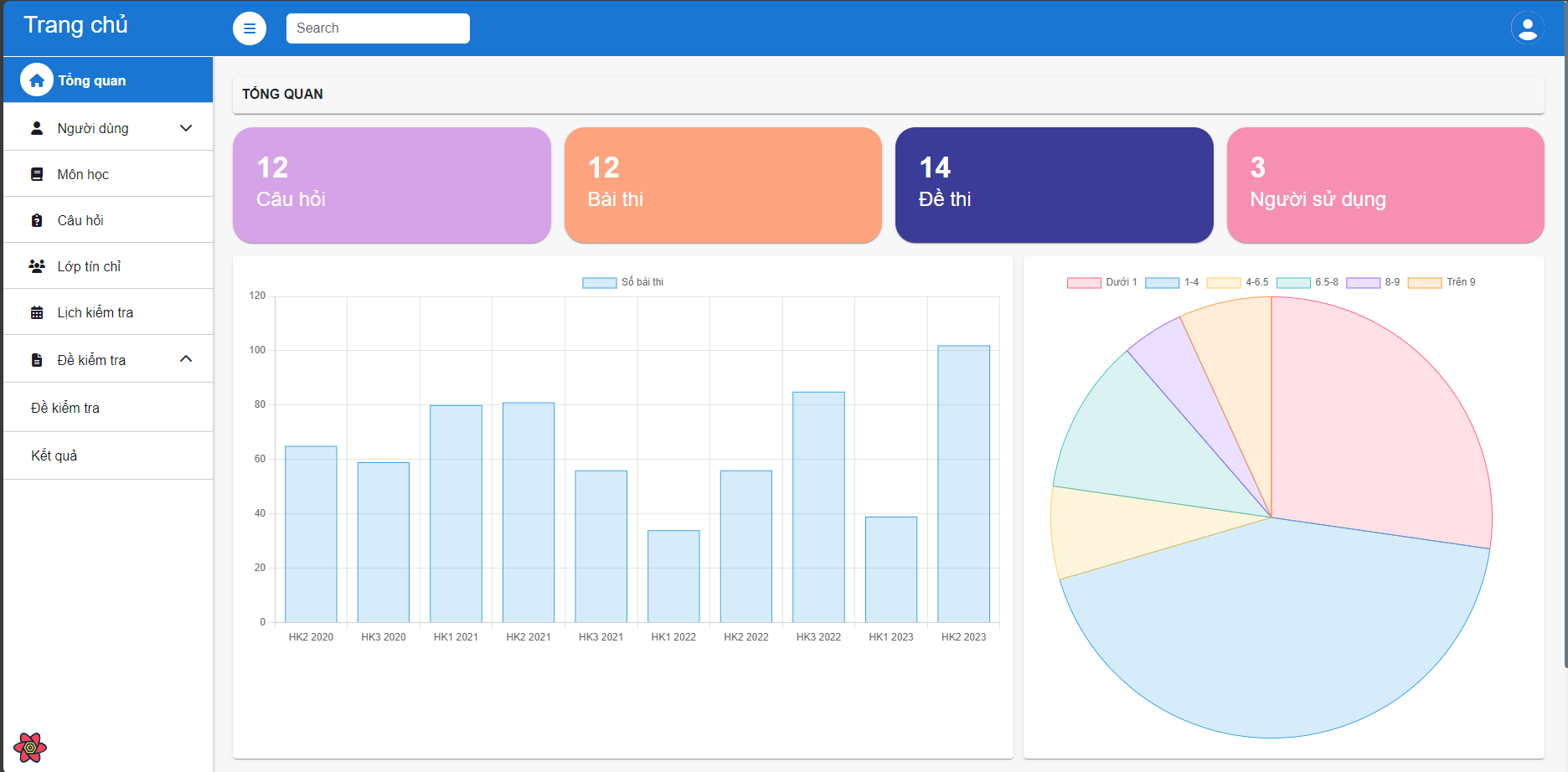
## 4.4. Xây dựng ứng dụng di động.

Phần mobile của ứng dụng được phát triển bằng React Native.

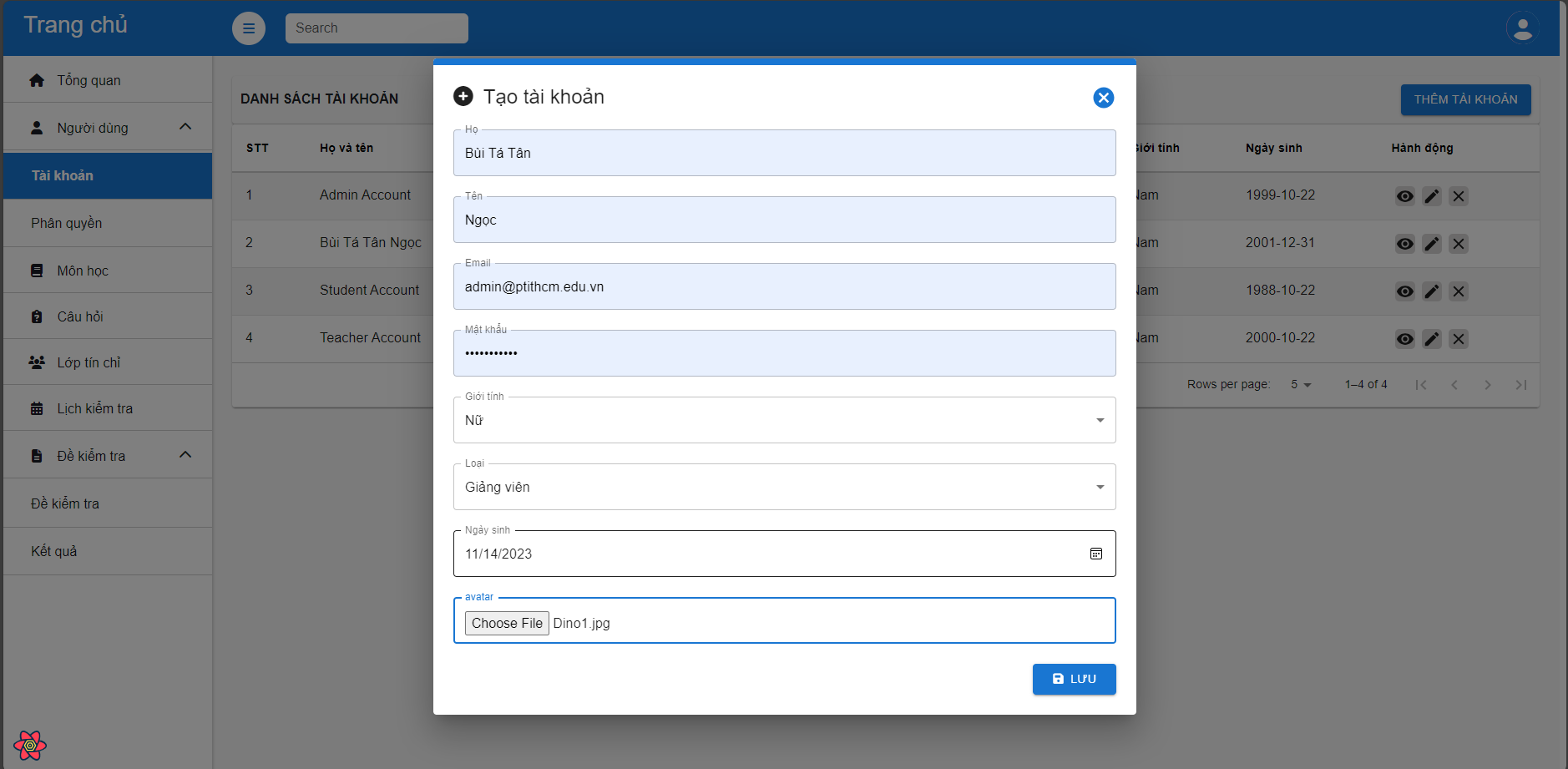
# CHƯƠNG 5: MINH HỌA SẢN PHẨM



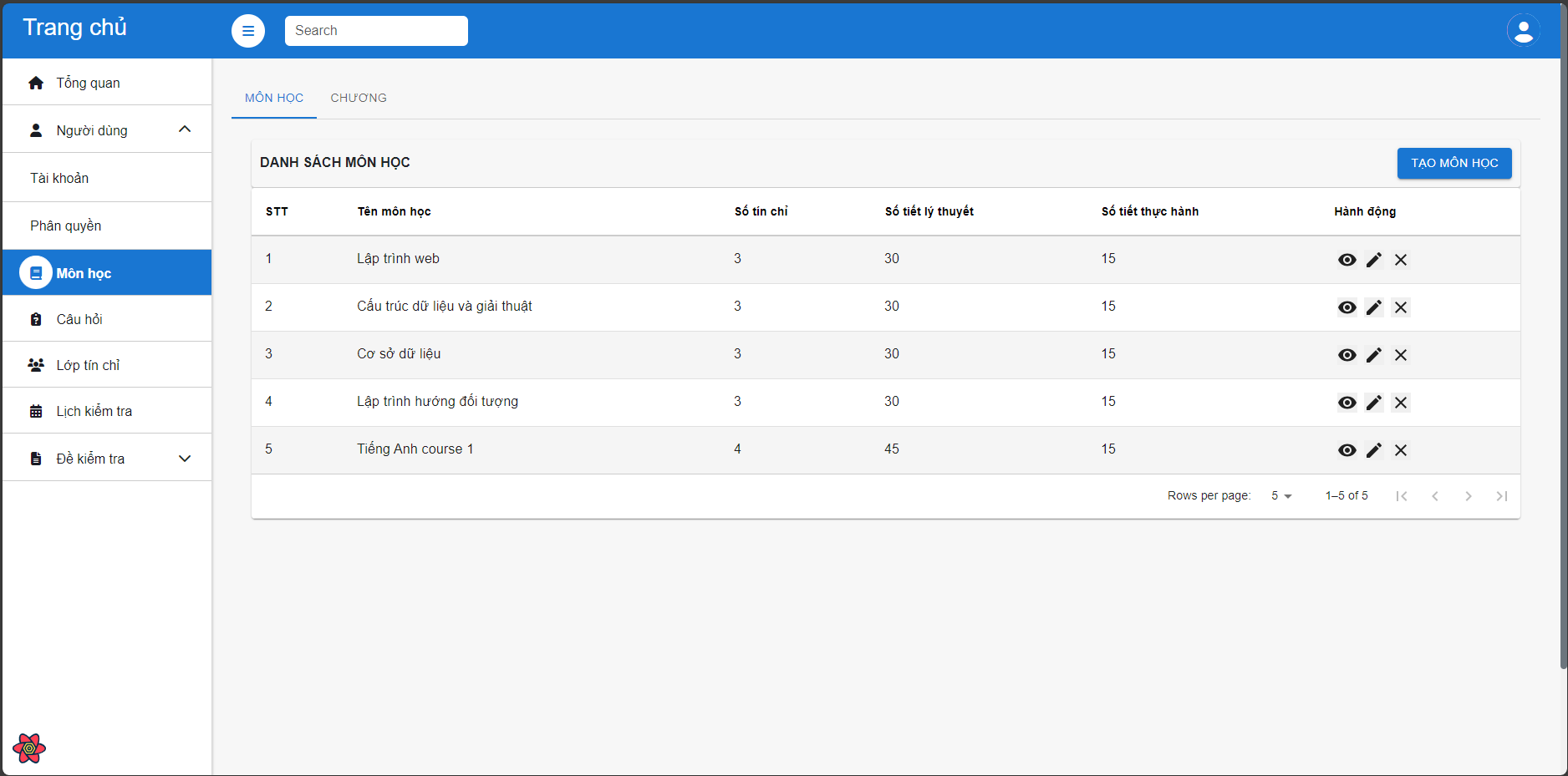
Hình 5.1. Màn hình đăng nhập.

**

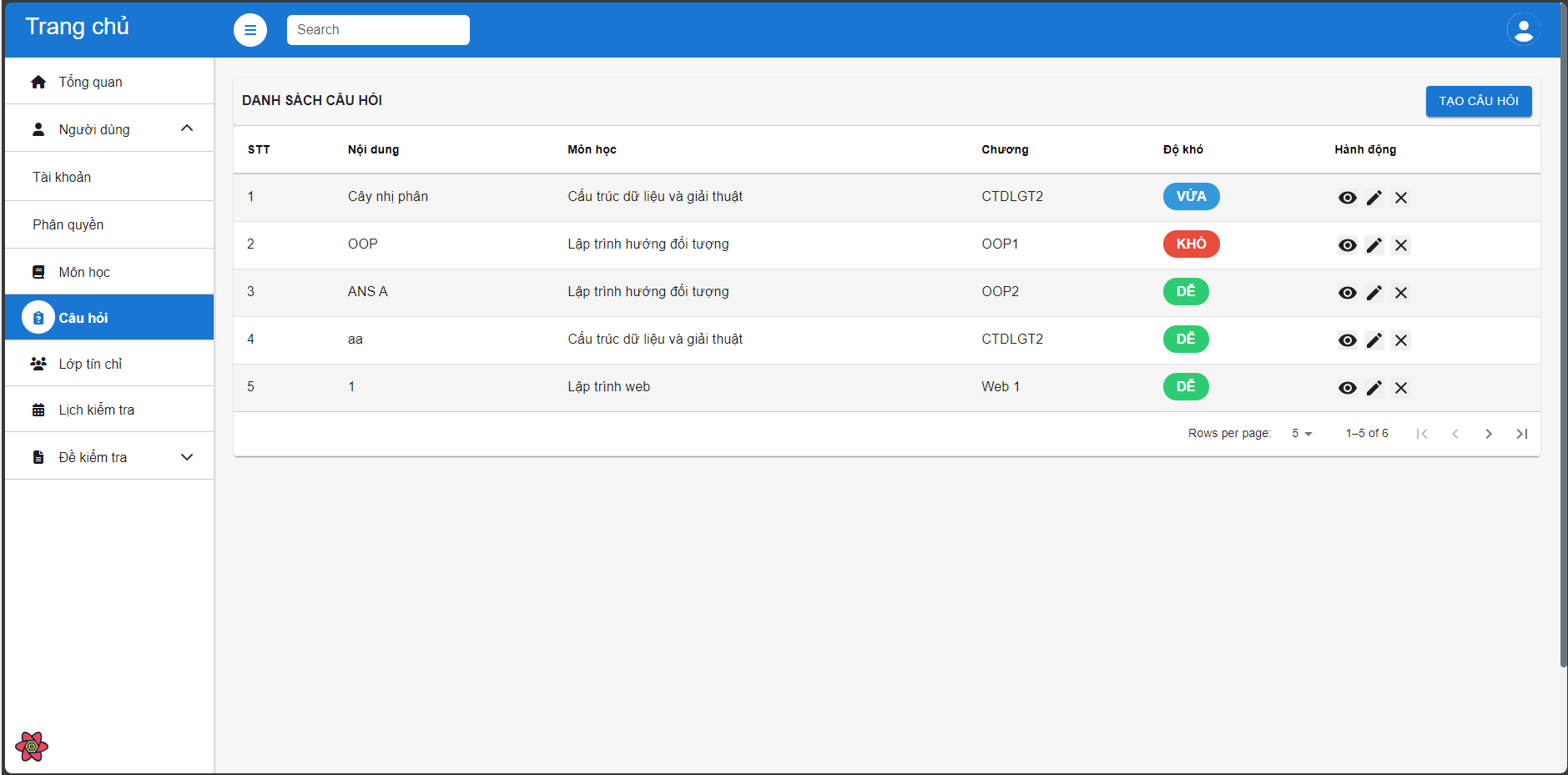
Hình 5.2. Màn hình trang tổng quan.

**

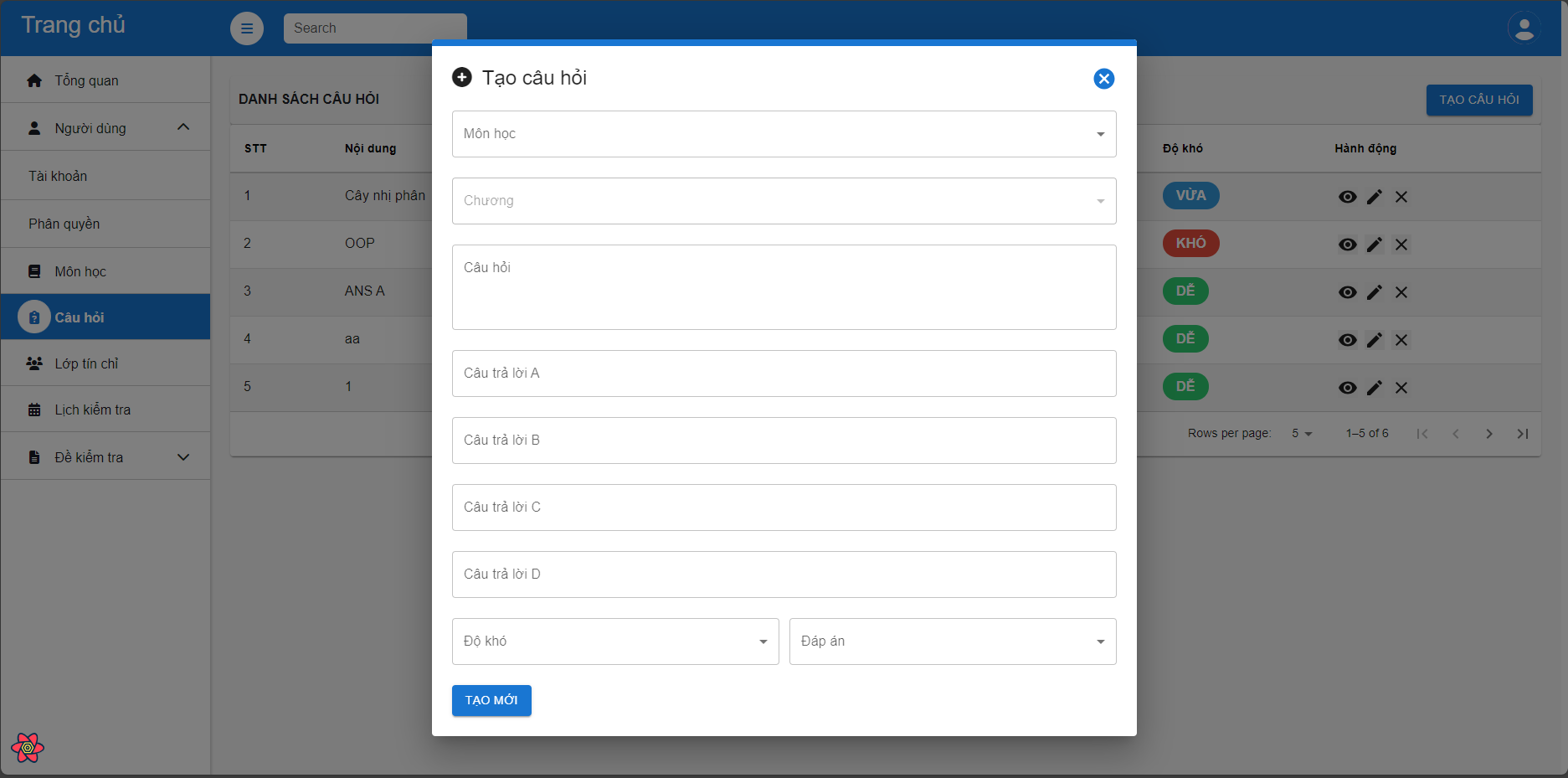
Hình 5.3. Màn hình trang quản lý người dùng.

**

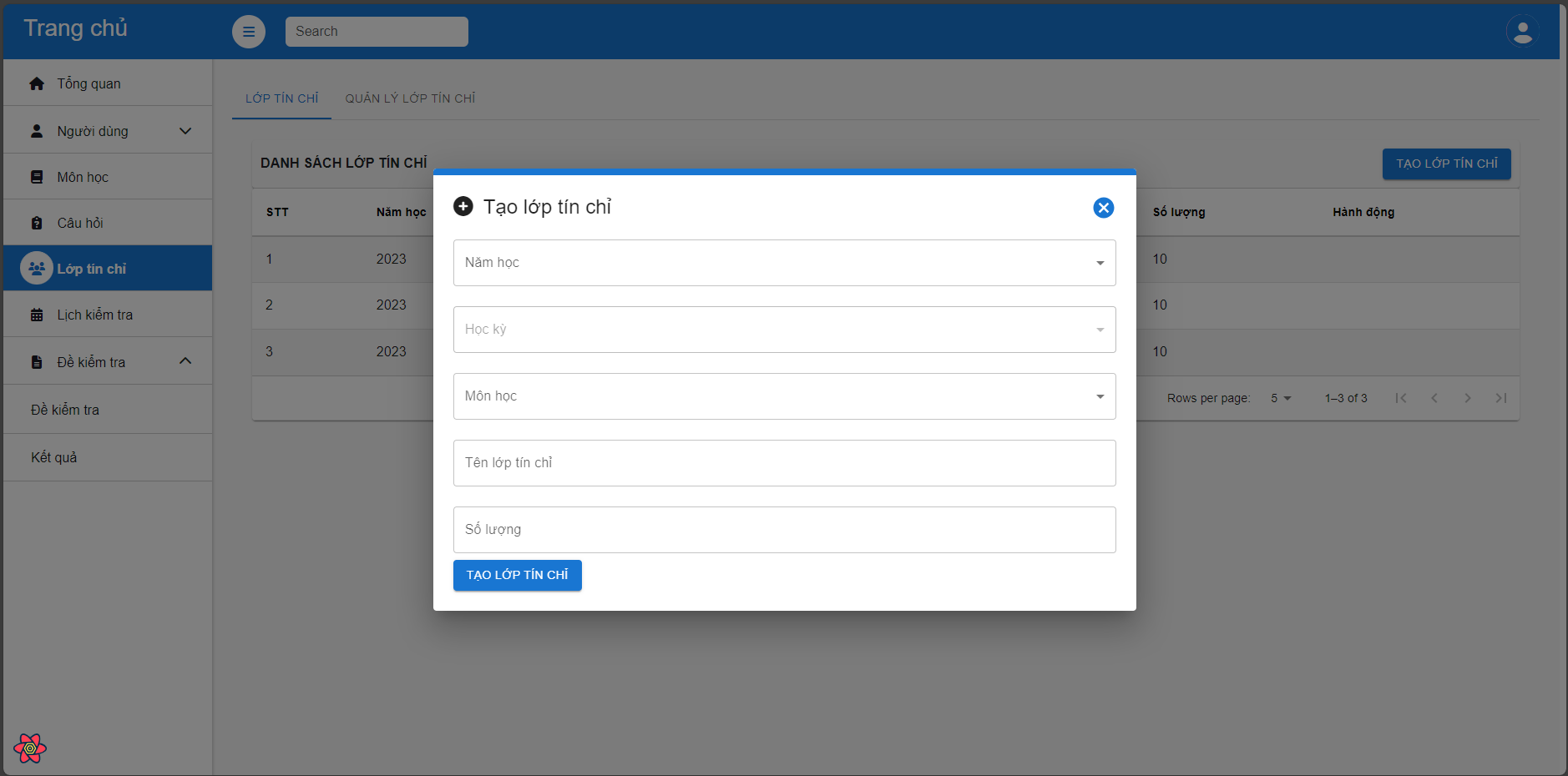
Hình 5.4. Màn hình trang quản lý môn học.

**

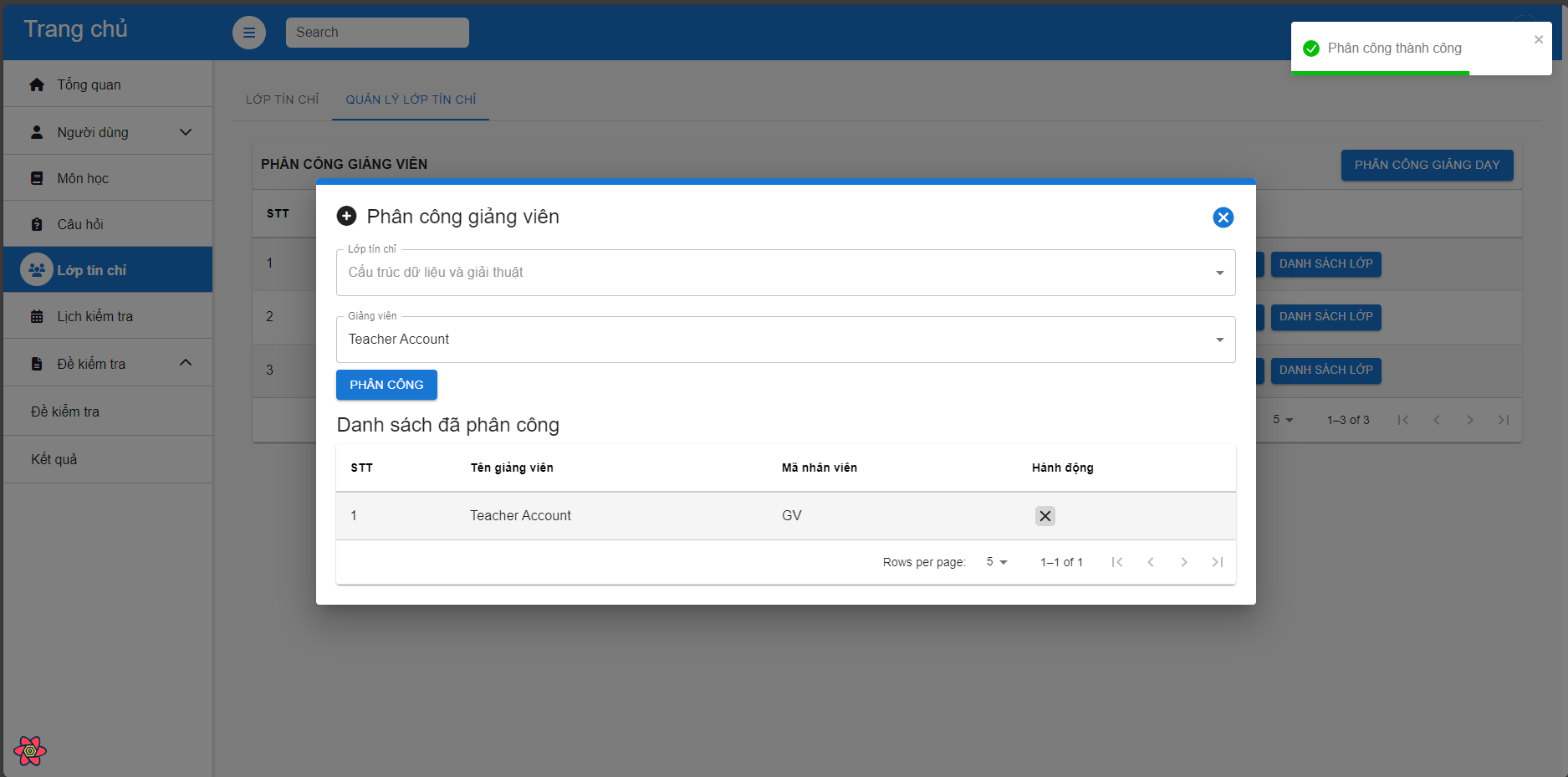
Hình 5.5. Màn hình trang quản lý câu hỏi.

**

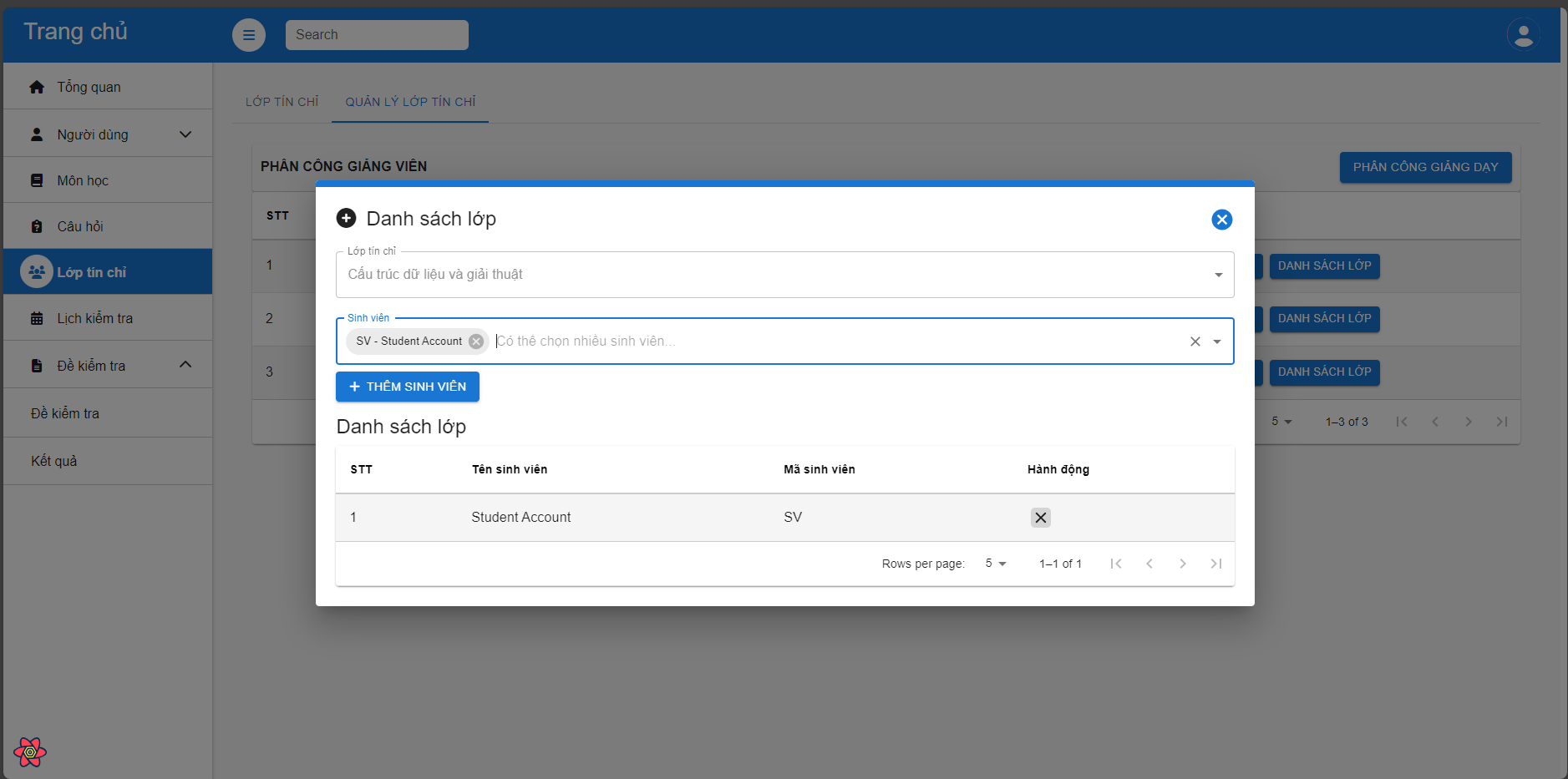
Hình 5.6. Màn hình trang tạo mới câu hỏi.

**

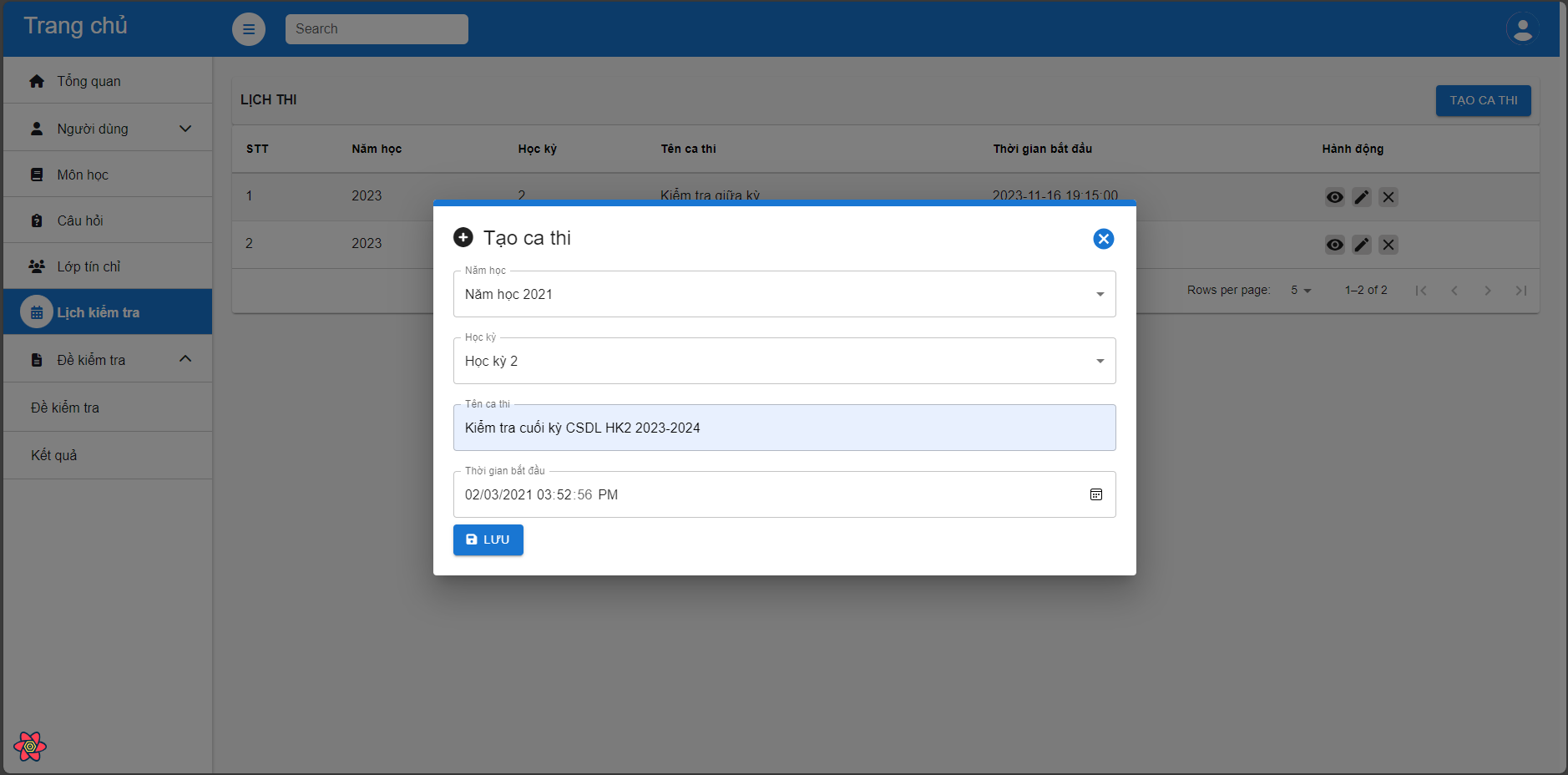
Hình 5.7. Màn hình trang quản lý lớp tín chỉ.

**

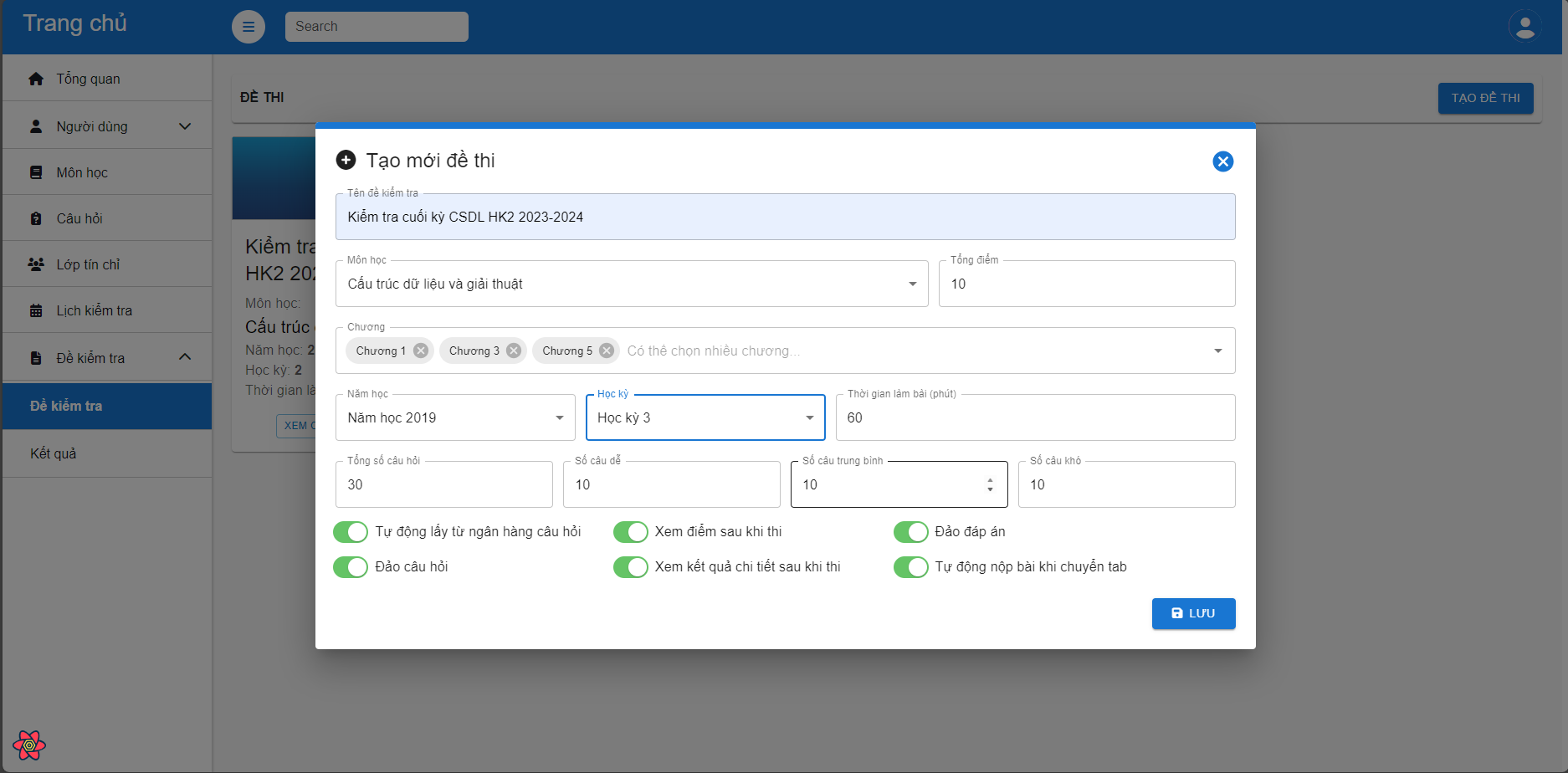
Hình 5.8. Màn hình trang phân công giảng viên cho lớp tín chỉ.

**

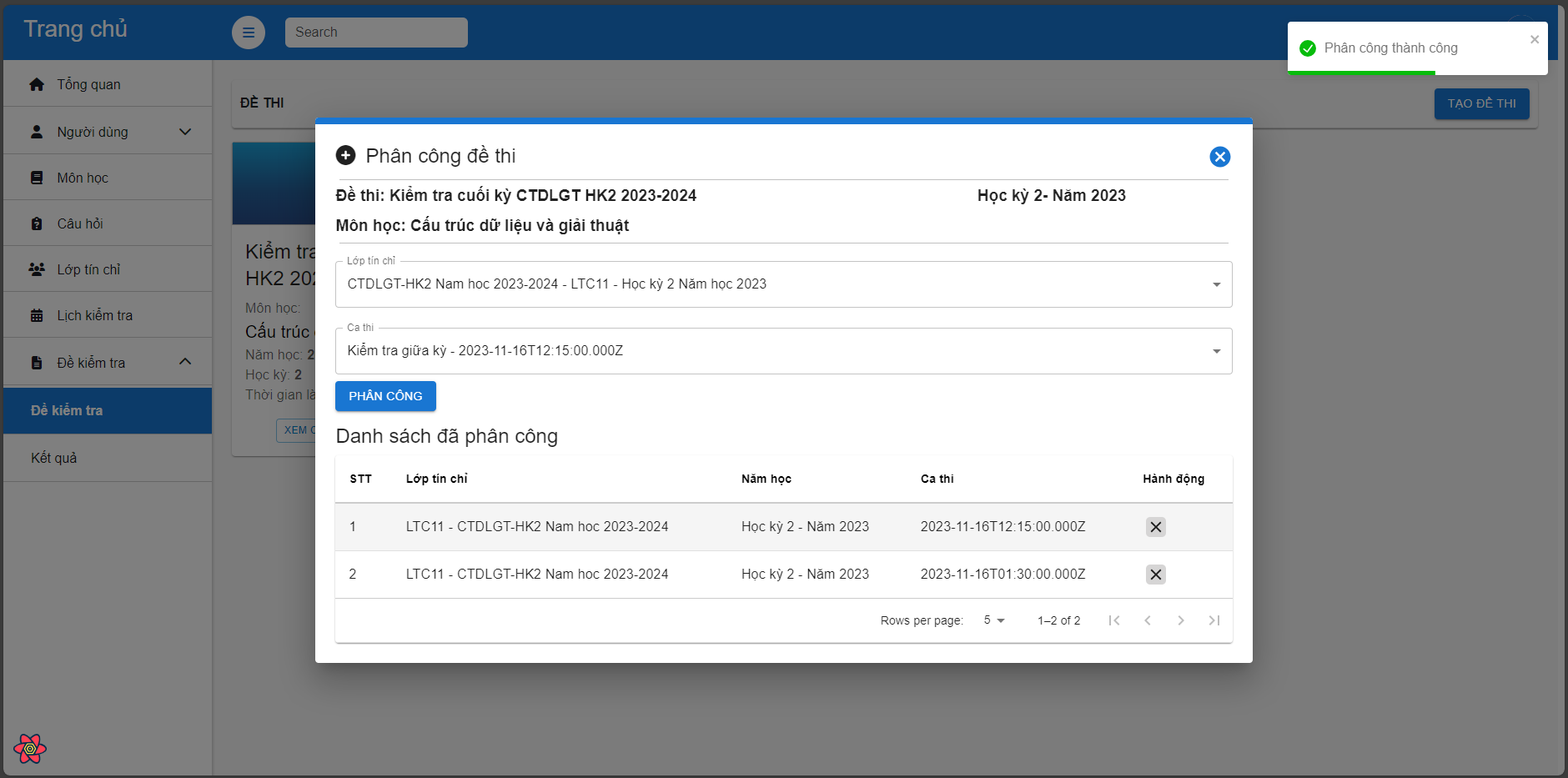
Hình 5.9. Màn hình trang thêm sinh viên cho lớp tín chỉ.

**

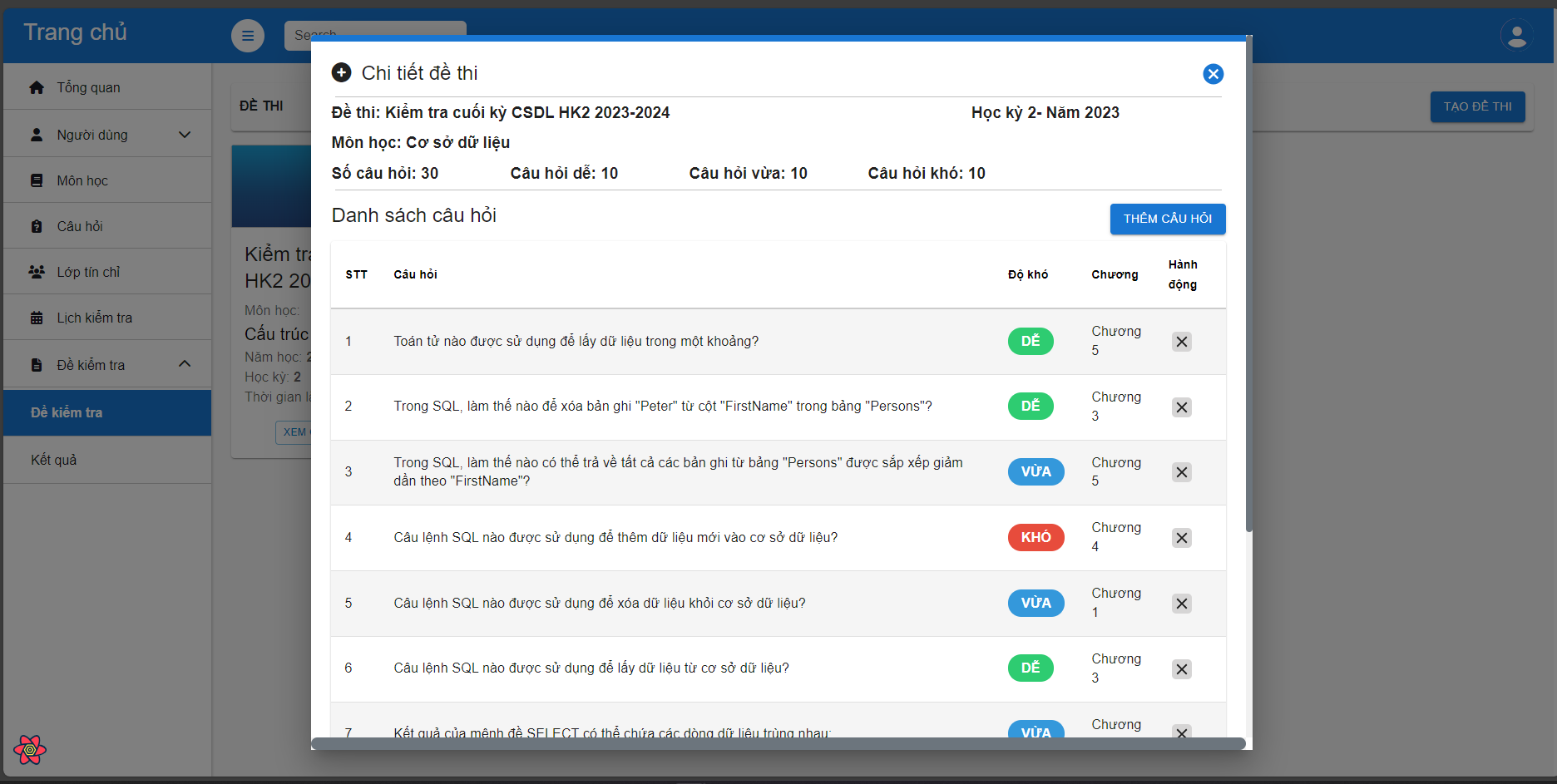
Hình 5.10. Màn hình trang tạo ca thi.

**

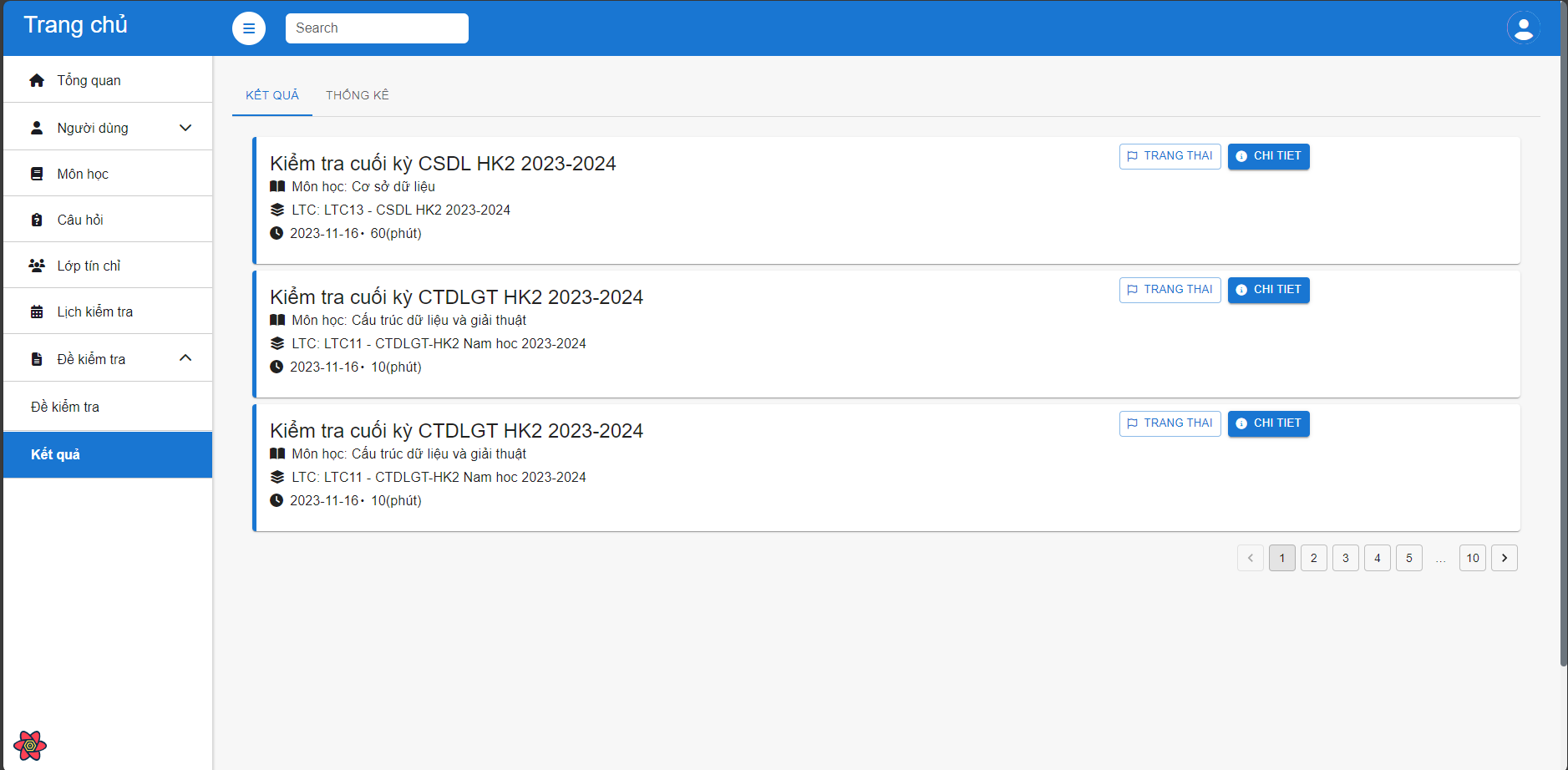
*Hình 5.11. Màn hình trang tạo đề thi.*

**

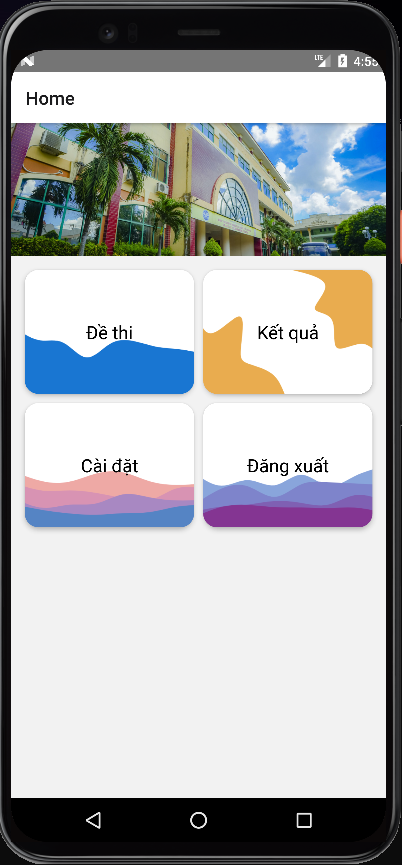
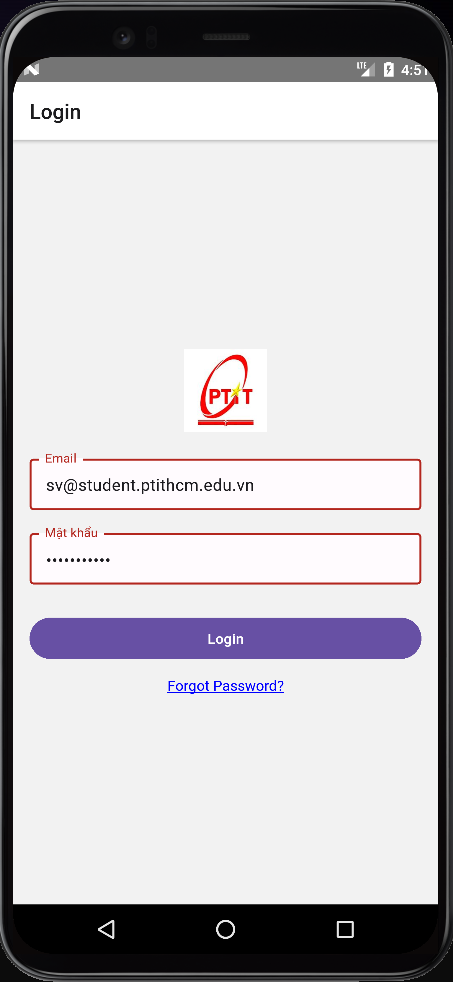
*Hình 5.12. Màn hình trang phân đề thi cho lớp tín chỉ.*

**

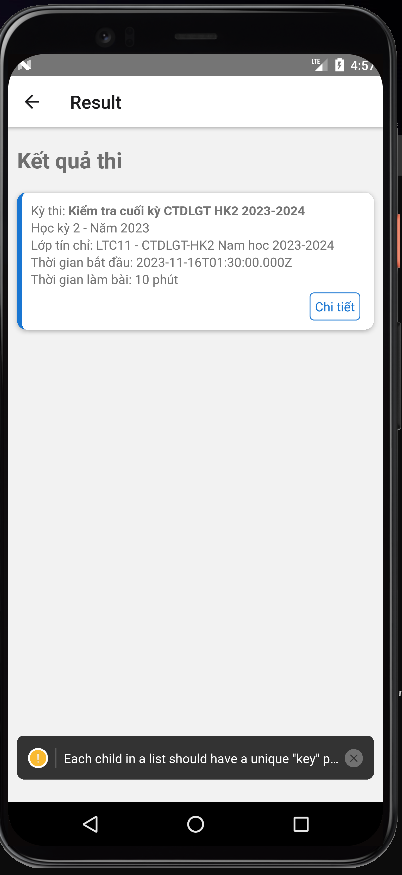
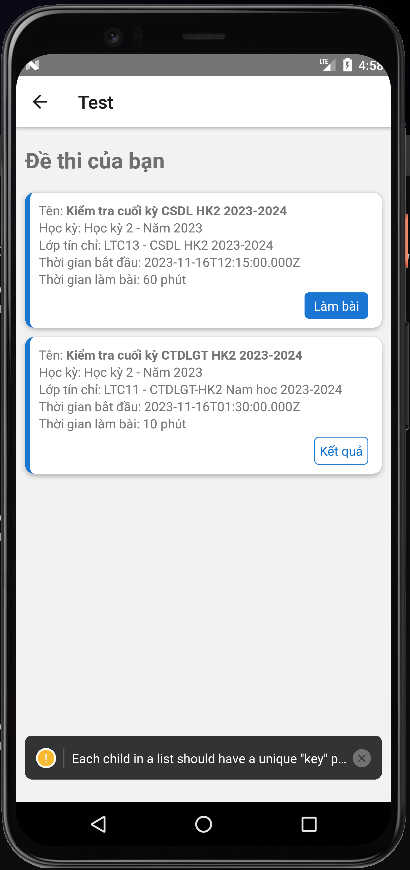
*Hình 5.13. Màn hình trang chi tiết đề thi.*

**

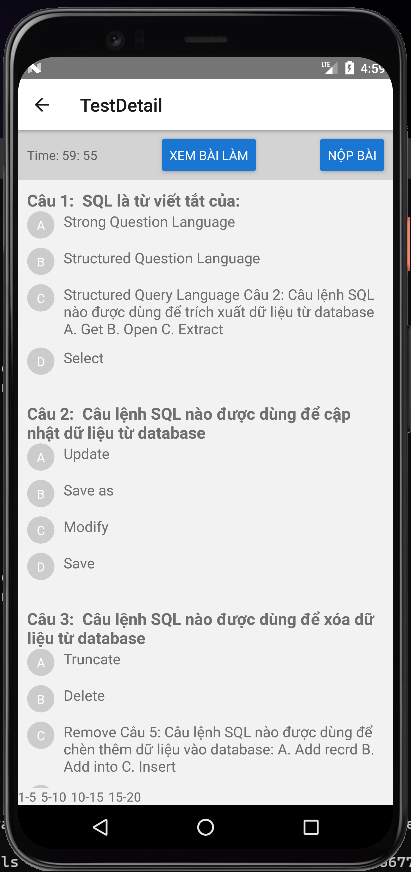
*Hình 5.14. Màn hình trang kết quả, thống kê.*

**

*Hình 5.15. Màn hình trang đăng nhập và trang chủ trên điện thoại.*

**

*Hình 5.16. Màn hình trang đề thi và kết quả thi.*

**

*Hình 5.17. Màn hình trang làm bài thi.*

# CHƯƠNG 6: KẾT LUẬN

## 6.1. Kết luận

Xây dựng một web application thi trắc nghiệm và một đề tài không quá mới, tuy nhiên với nhu cầu hiện tại với sự chuyển giao về hình thức thi từ giấy sang trắc nghiệm, từ thi offline sang online thì đề tài này có tính ứng dụng cao cũng như cần thiết trong thực tế. Quá trình nghiên cứu và phát triển đề tài cũng đã cho em nhiều kiến thức cũng như trải nghiêm mới mẻ và quý giá khi tự mình thực hiện vào tất cả các quá trình để làm một phần mềm hoàn chỉnh từ xây dựng ý tưởng đến phân tích thiết kế, phát triển, kiểm thử, ....

Do thời gian thực hiện có giới hạn, phải nghiên cứu nhiều công nghệ mới mà em lần đầu sử dụng như React, React native, đồng thời kinh nghiệm cũng như nghiệp vụ thực tế về đề tài chưa cao nên sản phẩm hiện chỉ mới đáp ứng được những yêu cầu cơ bản của người sử dụng, việc đưa sản phẩm vào thực tế chắc chắn cần phải nâng cấp thêm nhiều tính năng, hiệu năng, vấn đề bảo mật, .... Do đó, những thiếu sót là điều khó tránh khỏi. Em rất mong nhận được những góp ý của quý thầy cô để có thể cải thiện kỹ năng kinh nghiệm của mình về dự án này cũng như nhiều dự án về sau trên con đường sự nghiệp phía trước.

### 6.1.1. Những kết quả đạt được

Khảo sát, tìm hiểu được nghiệp vụ để xây dựng một web application thi trắc nghiệm trực tuyến.

Tìm hiểu Nodejs, framework ExpressJs, React, React native và các thư viện liên quan kết hợp với hệ quản trị cơ sở dữ liệu MySQL Server để áp dụng xây ứng dụng.

Phân tích và thiết kế cơ sở dữ liệu, chức năng hệ thống.

Xây dựng API cho hệ thống thông qua framework Expressjs.

Xây dựng giao diện cho hệ thống bằng ReactJs.

Xây dựng ứng dụng di động sử dụng React native.

Xây dựng cơ sở dữ liệu cho hệ thống bằng MySQLServer.

Các chức năng quản trị website:

- Quản lý người dùng.

- Quản lý kho câu hỏi, đề thi, lớp tín chỉ.

- Phân công câu hỏi, đề thi, lớp tín chỉ

- Quản lý kết quả, thống kê.

Các chức năng dành cho sinh viên:

- Đăng nhập vào hệ thống bằng email sinh viên và tài khoản được cấp.

- Tham gia thi các bài thi của các môn học mình học thông qua ứng dụng di động.

- Xem kết quả thi của mình.

- Quản lý thông tin các nhân.

### 6.1.2. Những hạn chế.

- Cần hoàn thiện một số tính năng của người dùng theo phân quyền.

- Cải thiện và hạn chế các lỗi tính năng tiềm ẩn.

## 6.2. Hướng phát triển của sản phẩm

- Tìm hiểu thêm công nghệ đang sử dụng để tối ưu hóa tốc độ xử lý, nâng cao trải nghiệm của người dùng.

- Bổ sung tính năng thi trên website, thông báo, ...

# PHỤ LỤC

(Hướng dẫn cài đặt)

* Ứng dụng phía back-end chạy bằng visual studio code.
* Chạy migration database bằng lệnh “npm run reload:testdb” và khởi tạo các dữ liệu mặc định bằng lệnh “npx sequelize-cli db:seed:all –env test”.
* Chạy ứng dụng phía back-end bằng lệnh “npm run dev”.
* Để chạy ứng dụng phía front-end, sau khi mở source code trong visual studio code, mở command line bằng tổ hợp phím “Crtl+` (~)” sau đó chạy lệnh “npm i” để tải về các thư viện cần thiết. Sau đấy chạy lệnh “npm run dev” để chạy chương trình.

# DANH MỤC TÀI LIỆU THAM KHẢO

**Tiếng Việt:**

**Tiếng Anh:**

**Danh mục các Website tham khảo:**

1. React documentation: <https://angular.io/docs>
2. Angular university: <https://angular-university.io/>
3. Angular Material: <https://material.angular.io/>
4. ASP.NET core documentation: <https://learn.microsoft.com/en-us/aspnet/core>
5. C sharp corner: <https://www.c-sharpcorner.com/>
6. Stackoverflow: <https://stackoverflow.com/>
7. SQL Server documentation: <https://learn.microsoft.com/en-us/sql/sql-server/?view=sql-server-ver16>