**TRƯỜNG ĐẠI HỌC PHƯƠNG ĐÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN – TRUYỀN THÔNG**

**NGÀNH CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI**

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ**

**KÍ TÚC XÁ TRÊN NỀN WEB SỬ DỤNG CÔNG NGHỆ LARAVEL**

HÀ NỘI – 2025

**TRƯỜNG ĐẠI HỌC PHƯƠNG ĐÔNG**

**KHOA CÔNG NGHỆ THÔNG TIN – TRUYỀN THÔNG**

**NGÀNH CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN TỐT NGHIỆP**

**ĐỀ TÀI**

**XÂY DỰNG ỨNG DỤNG QUẢN LÝ**

**KÍ TÚC XÁ TRÊN NỀN WEB SỬ DỤNG CÔNG NGHỆ LARAVEL**

Sinh viên thực hiện: **Bùi Văn Chiến**

Mã số sinh viên: **521100198**

Giảng viên hướng dẫn: **ThS. Nguyễn Thị Đoan Trang**

HÀ NỘI – 2025

# LỜI CẢM ƠN

Trước tiên, em xin gửi lời tri ân sâu sắc đến Ban Giám Hiệu Trường Đại học Phương Đông cùng quý thầy cô trong khoa, những người đã tạo điều kiện thuận lợi và hỗ trợ em trong suốt quá trình học tập và thực hiện khóa luận này.

Đặc biệt, em xin bày tỏ lòng biết ơn chân thành đến giảng viên hướng dẫn, người đã tận tình chỉ bảo, định hướng và góp ý quý báu để em có thể hoàn thiện đề tài một cách tốt nhất. Những kiến thức và kinh nghiệm mà thầy/cô chia sẻ không chỉ giúp em hoàn thành đồ án tốt nghiệp mà còn là hành trang quý giá cho tương lai.

Không thể không nhắc đến gia đình, bạn bè và đồng nghiệp, những người luôn ở bên cạnh, động viên và tiếp thêm động lực để em có thể hoàn thành khóa luận này một cách tốt nhất.

Cuối cùng, dù đã nỗ lực hết mình, nhưng chắc chắn khóa luận vẫn còn những thiếu sót. Em rất mong nhận được những ý kiến đóng góp quý báu từ thầy cô và các bạn để có thể hoàn thiện và phát triển hơn trong tương lai.

Xin chân thành cảm ơn!

# MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc198476035)

[MỤC LỤC 4](#_Toc198476036)

[DANH MỤC HÌNH ẢNH 7](#_Toc198476037)

[DANH MỤC BẢNG 10](#_Toc198476038)

[BẢNG KÍ TỰ VIẾT TẮT 11](#_Toc198476039)

[LỜI MỞ ĐẦU 12](#_Toc198476040)

[CHƯƠNG 1: TỔNG QUAN NGHIÊN CỨU 13](#_Toc198476041)

[1.1. Khảo sát yêu cầu 13](#_Toc198476042)

[1.2. Công nghệ sử dụng trong dự án 15](#_Toc198476043)

[1.2.1. Giới thiệu về ngôn ngữ PHP 15](#_Toc198476044)

[1.2.2. Giới thiệu về thư viện Laravel 16](#_Toc198476045)

[1.2.3. Giới thiệu về Mysql 20](#_Toc198476046)

[1.2.4. Giới thiệu về Framework Bootstrap 21](#_Toc198476047)

[1.2.5. Giới thiệu về Git và GitHub 22](#_Toc198476048)

[CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 24](#_Toc198476049)

[2.1. Tổng quan về nghiệp vụ 24](#_Toc198476050)

[2.2. Phân tích chức năng của hệ thống 25](#_Toc198476051)

[2.2.1. Biểu đồ usecase tổng quan hệ thống 25](#_Toc198476052)

[2.2.2. Đặc tả usecase 25](#_Toc198476053)

[2.3. Phân tích yêu cầu phi chức năng 32](#_Toc198476054)

[2.3.1. Đăng nhập 32](#_Toc198476055)

[2.3.2. Đăng ký 32](#_Toc198476056)

[2.3.3. Quên mật khẩu 33](#_Toc198476057)

[2.3.4. Thêm danh mục 33](#_Toc198476058)

[2.3.5. Cập nhật danh mục 34](#_Toc198476059)

[2.3.6. Xóa danh mục 34](#_Toc198476060)

[2.3.7. Thêm phòng KTX 35](#_Toc198476061)

[2.3.8. Xóa phòng KTX 35](#_Toc198476062)

[2.3.9. Cập nhật phòng KTX 36](#_Toc198476063)

[2.3.10. Cập nhật trạng thái hóa đơn 36](#_Toc198476064)

[2.4. Phân tích biểu đồ 37](#_Toc198476065)

[2.4.1. Biểu đồ tuần tự “Đăng nhập” 37](#_Toc198476066)

[2.4.2. Biểu đồ tuần tự “Đăng ký” 37](#_Toc198476067)

[2.4.3. Biểu đồ tuần tự “Đăng ký KTX” 38](#_Toc198476068)

[2.4.4. Biểu đồ tuần tự “Thêm danh mục” 38](#_Toc198476069)

[2.4.5. Biểu đồ tuần tự “Cập nhật danh mục” 39](#_Toc198476070)

[2.4.6. Biểu đồ tuần tự “Xóa danh mục” 39](#_Toc198476071)

[2.4.7. Biểu đồ tuần tự “Thêm phòng KTX” 40](#_Toc198476072)

[2.4.8. Biểu đồ tuần tự “Cập nhật phòng KTX” 40](#_Toc198476073)

[2.4.9. Biểu đồ tuần tự “Xóa phòng KTX” 41](#_Toc198476074)

[2.4.10. Biểu đồ tuần tự “Cập nhật trạng thái hóa đơn” 41](#_Toc198476075)

[2.5. Phân rã usecase 42](#_Toc198476076)

[2.5.1. Phân rã usecase “Đăng ký ở KTX” 42](#_Toc198476077)

[2.5.2. Phân rã usecase “Quản lý thông tin” 43](#_Toc198476078)

[2.5.3. Phân rã usercase “Xem thông tin bài viêt” 44](#_Toc198476079)

[2.5.4. Phân rã usercase “Phản hồi hóa đơn” 46](#_Toc198476080)

[2.6. Thiết kế cơ sở dữ liệu 47](#_Toc198476081)

[2.6.1. Biểu đồ lớp 47](#_Toc198476082)

[2.6.2. Bảng dữ liệu User 47](#_Toc198476083)

[2.6.3. Bảng dữ liệu Categories 48](#_Toc198476084)

[2.6.4. Bảng dữ liệu Rooms 48](#_Toc198476085)

[2.6.5. Bảng dữ liệu Members 49](#_Toc198476086)

[2.6.6. Bảng dữ liệu Pays 49](#_Toc198476087)

[2.6.7. Bảng dữ liệu Bill 49](#_Toc198476088)

[2.6.8. Bảng dữ liệu BillDetail 50](#_Toc198476089)

[2.6.9. Bảng dữ liệu News 50](#_Toc198476090)

[2.6.10. Bảng dữ liệu News\_like 51](#_Toc198476091)

[2.6.11. Bảng dữ liệu Banners 51](#_Toc198476092)

[2.6.12. Bảng dữ liệu Bank 51](#_Toc198476093)

[CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH 53](#_Toc198476094)

[3.1. Giao diện người dùng 53](#_Toc198476095)

[3.1.1. Trang đăng nhập 53](#_Toc198476096)

[3.1.2. Trang đăng ký 53](#_Toc198476097)

[3.1.3. Trang quên mật khẩu 54](#_Toc198476098)

[3.1.4. Trang chủ - Người dùng 54](#_Toc198476099)

[3.1.5. Trang thông tin bài viết 55](#_Toc198476100)

[3.1.6. Trang danh mục phòng 55](#_Toc198476101)

[3.1.7. Trang đăng ký phòng KTX 56](#_Toc198476102)

[3.1.8. Trang thanh toán 56](#_Toc198476103)

[3.1.9. Trang hoá đơn 56](#_Toc198476104)

[3.1.10. Trang chi tiết hoá đơn 57](#_Toc198476105)

[3.1.11. Trang quản lý thông tin người dùng 57](#_Toc198476106)

[3.2. Giao diện quản trị 58](#_Toc198476107)

[3.2.1. Trang chủ - Quản trị 58](#_Toc198476108)

[3.2.2. Trang quản lý danh mục 58](#_Toc198476109)

[3.2.3. Trang quản lý phòng ở 59](#_Toc198476110)

[3.2.4. Trang quản lý thành viên 60](#_Toc198476111)

[3.2.5. Trang quản lý tài chính 61](#_Toc198476112)

[3.2.6. Trang quản lý hoá đơn 62](#_Toc198476113)

[3.2.7. Trang quản lý tin tức 63](#_Toc198476114)

[3.2.8. Trang quản lý tính năng 64](#_Toc198476115)

[3.2.9. Trang quản lý tài khoản 64](#_Toc198476116)

[KẾT LUẬN 66](#_Toc198476117)

[TÀI LIỆU THAM KHẢO 67](#_Toc198476118)

[LỜI CAM ĐOAN 68](#_Toc198476119)

**DANH MỤC HÌNH ẢNH**

**CHƯƠNG 1**

[Hình 1.1. Thư viện Laravel 16](#_Toc197960305)

[Hình 1.2. Thư viện Bootstrap 21](#_Toc197960306)

**CHƯƠNG 2**

[Hình 2.1. Biểu đồ usecase tổng quan hệ thống 25](#_Toc197961040)

[Hình 2.2. Biểu đồ usecase đăng nhập 25](#_Toc197961041)

[Hình 2.3. Biểu đồ usecase đăng ký 26](#_Toc197961042)

[Hình 2.4. Biểu đồ usecase đăng ký KTX 27](#_Toc197961043)

[Hình 2.5. Biểu đồ usecase quản trị danh mục 28](#_Toc197961044)

[Hình 2.6. Biểu đồ usecase quản trị phòng KTX 29](#_Toc197961045)

[Hình 2.7. Biểu đồ usecase quản trị hóa đơn 30](#_Toc197961046)

[Hình 2.8. Biểu đồ usecase quản trị tài khoản 31](#_Toc197961047)

[Hình 2.9. Sơ đồ hoạt động chức năng đăng nhập 32](#_Toc197961048)

[Hình 2.10. Sơ đồ hoạt động chức năng đăng ký 32](#_Toc197961049)

[Hình 2.11. Sơ đồ hoạt động chức năng quên mật khẩu 33](#_Toc197961050)

[Hình 2.12. Biểu diễn hoạt động chức năng thêm danh mục 33](#_Toc197961051)

[Hình 2.13. Biểu diễn hoạt động chức năng cập nhật danh mục 34](#_Toc197961052)

[Hình 2.14. Biểu diễn hoạt động chức năng xóa danh mục 34](#_Toc197961053)

[Hình 2.15. Sơ đồ hoạt động chức năng thêm phòng KTX 35](#_Toc197961054)

[Hình 2.16. Sơ đồ hoạt động chức năng xóa phòng KTX 35](#_Toc197961055)

[Hình 2.17. Sơ đồ hoạt động chức năng cập nhật phòng KTX 36](#_Toc197961056)

[Hình 2.18. Sơ đồ hoạt động chức năng cập nhật trạng thái hóa đơn 36](#_Toc197961057)

[Hình 2.19. Biểu đồ tuần tự Đăng nhập 37](#_Toc197961058)

[Hình 2.20. Biểu đồ tuần tự Đăng ký 37](#_Toc197961059)

[Hình 2.21. Biểu đồ tuần tự chức năng đăng ký KTX 38](#_Toc197961060)

[Hình 2.22. Biểu đồ tuần tự chức năng thêm danh mục 38](#_Toc197961061)

[Hình 2.23. Biểu đồ tuần tự chức năng Cập nhật danh mục 39](#_Toc197961062)

[Hình 2.24. Biểu đồ tuần tự chức năng xóa danh mục 39](#_Toc197961063)

[Hình 2.25. Biểu đồ tuần tự chức năng thêm phòng KTX 40](#_Toc197961064)

[Hình 2.26. Biểu đồ tuần tự chức năng cập nhật sản phẩm 40](#_Toc197961065)

[Hình 2.27. Biểu đồ tuần tự chức năng xóa sản phẩm 41](#_Toc197961066)

[Hình 2.28. Biểu đồ tuần tự chức năng cập nhập trạng thái hóa đơn 41](#_Toc197961067)

[Hình 2.29. Phân rã usecase “Đăng kí” 42](#_Toc197961068)

[Hình 2.30. Phân rã usecase “Quản lý thông tin” 43](#_Toc197961069)

[Hình 2.31. Phân rã usecase “Xem thông tin bài viết” 44](#_Toc197961070)

[Hình 2.32. Phân rã usecase “Phản hồi hóa đơn” 46](#_Toc197961071)

[Hình 2.33. Biểu đồ lớp 47](#_Toc197961072)

**CHƯƠNG 3**

[Hình 3.1. Trang đăng nhập 53](#_Toc197961349)

[Hình 3.2. Trang đăng ký 53](#_Toc197961350)

[Hình 3.3. Trang quên mật khẩu 54](#_Toc197961351)

[Hình 3.4. Trang chủ 54](#_Toc197961352)

[Hình 3.5. Trang thôn tin bài viết 55](#_Toc197961353)

[Hình 3.6. Trang thông tin danh mục phòng 55](#_Toc197961354)

[Hình 3.7. Trang đăng ký phòng KTX 56](#_Toc197961355)

[Hình 3.8. Trang thanh toán 56](#_Toc197961356)

[Hình 3.9. Trang hóa đơn 56](#_Toc197961357)

[Hình 3.10. Trang chi tiết hóa đơn 57](#_Toc197961358)

[Hình 3.11. Trang quản lý thông tin người dung 57](#_Toc197961359)

[Hình 3.12. Trang chủ - quản trị 58](#_Toc197961360)

[Hình 3.13. Trang quản lý danh mục 58](#_Toc197961361)

[Hình 3.14. Trang thêm danh mục 58](#_Toc197961362)

[Hình 3.15. Trang sửa danh mục 59](#_Toc197961363)

[Hình 3.16. Trang quản lý ở KTX 59](#_Toc197961364)

[Hình 3.17. Trang thêm phòng KTX 59](#_Toc197961365)

[Hình 3.18. Trang sửa phòng KTX 60](#_Toc197961366)

[Hình 3.19. Trang quản lý thành viên KTX 60](#_Toc197961367)

[Hình 3.20. Trang thêm thành viên KTX 60](#_Toc197961368)

[Hình 3.21. Trang sửa thông tin thành viên KTX 61](#_Toc197961369)

[Hình 3.22. Trang quản lý tài chính 61](#_Toc197961370)

[Hình 3.23. Trang thêm thanh toán theo phòng 61](#_Toc197961371)

[Hình 3.24. Trang thêm thanh toán theo thành viên 62](#_Toc197961372)

[Hình 3.25. Trang quản lý hóa đơn 62](#_Toc197961373)

[Hình 3.26. Trang chi tiết hóa đơn 62](#_Toc197961374)

[Hình 3.27. Trang sửa trạng thái hóa đơn 63](#_Toc197961375)

[Hình 3.28. Trang quản lý tin tức 63](#_Toc197961376)

[Hình 3.29. Trang thêm tin tức 63](#_Toc197961377)

[Hình 3.30. Trang sửa tin tức 64](#_Toc197961378)

[Hình 3.31. Trang quản lý tính năng 64](#_Toc197961379)

[Hình 3.32. Trang quản lý tài khoản 64](#_Toc197961380)

[Hình 3.33. Trang thêm tài khoản 65](#_Toc197961381)

[Hình 3.34. Trang sửa tài khoản 65](#_Toc197961382)

**DANH MỤC BẢNG**

**Chương 1**

[Bảng 1.1. Các lệnh quan trọng trong Laravel 18](#_Toc197961657)

[Bảng 1.2. Cấu trúc thư mục app 18](#_Toc197961658)

[Bảng 1.3. Cấu trúc thư mục config 19](#_Toc197961659)

[Bảng 1.4. Cấu trúc thư mục database 19](#_Toc197961660)

[Bảng 1.5. Cấu trúc thư mục resources 19](#_Toc197961661)

[Bảng 1.6. Cấu trúc thư mục routes 19](#_Toc197961662)

**Chương 2**

[Bảng 2.1. Đặc tả usecase “Đăng nhập” 26](#_Toc197961780)

[Bảng 2.2. Đặc tả usecase “Đăng ký” 26](#_Toc197961781)

[Bảng 2.3. Đặc tả usecase “Đăng ký ở KTX” 27](#_Toc197961782)

[Bảng 2.4. Đặc tả usecase “Danh mục” - Quản trị 28](#_Toc197961783)

[Bảng 2.5. Đặc tả usecase “Phòng KTX” - Quản trị 29](#_Toc197961784)

[Bảng 2.6. Đặc tả usecase “Hóa đơn” - Quản trị 30](#_Toc197961785)

[Bảng 2.7. Đặc tả usecase “Tài khoản” - Quản trị 31](#_Toc197961786)

[Bảng 2.8. Bảng dữ liệu User 47](#_Toc197961787)

[Bảng 2.9. Bảng dữ liệu Categories 48](#_Toc197961788)

[Bảng 2.10. Bảng dữ liệu Rooms 48](#_Toc197961789)

[Bảng 2.11. Bảng dữ liệu Members 49](#_Toc197961790)

[Bảng 2.12. Bảng dữ liệu Pays 49](#_Toc197961791)

[Bảng 2.13. Bảng dữ liệu Bill 49](#_Toc197961792)

[Bảng 2.14. Bảng dữ liệu BillDetail 50](#_Toc197961793)

[Bảng 2.15. Bảng dữ liệu News 50](#_Toc197961794)

[Bảng 2.16. Bảng dữ liệu News\_like 51](#_Toc197961795)

[Bảng 2.17. Bảng dữ liệu Banners 51](#_Toc197961796)

[Bảng 2.18. Bảng dữ liệu Bank 51](#_Toc197961797)

# BẢNG KÍ TỰ VIẾT TẮT

|  |  |
| --- | --- |
| **Kí tự** | **Ý Nghĩa** |
| HT | Hệ thống |
| QTV | Quản trị viên |
| KH | Khách hàng |
| KTX | Kí túc xá |
| User | Người dùng |
| Admin | Quản trị viên |

**LỜI MỞ ĐẦU**

Trong thời đại công nghệ thông tin phát triển mạnh mẽ như hiện nay, việc ứng dụng các giải pháp công nghệ vào công tác quản lý là xu hướng tất yếu nhằm nâng cao hiệu quả và giảm thiểu chi phí vận hành. Một trong những lĩnh vực cần được hiện đại hóa là công tác quản lý ký túc xá – nơi ở tập trung của sinh viên tại các trường đại học, cao đẳng.

Quá trình quản lý ký túc xá truyền thống thường gặp nhiều hạn chế như thủ công, thiếu tính đồng bộ, tốn thời gian và dễ xảy ra sai sót trong quá trình theo dõi thông tin sinh viên, thu chi, quản lý phòng ở, tình trạng thanh toán. Điều này đặt ra yêu cầu cấp thiết phải xây dựng một hệ thống quản lý ký túc xá hiệu quả, linh hoạt và dễ sử dụng.

Với mục tiêu đó, em đã lựa chọn đề tài "Xây dựng ứng dụng quản lý ký túc xá trên nền web sử dụng công nghệ Laravel" nhằm tạo ra một công cụ hỗ trợ nhà quản lý thực hiện các nghiệp vụ một cách thuận tiện, chính xác và nhanh chóng hơn. Ứng dụng được phát triển bằng Laravel – một trong những framework PHP mạnh mẽ và phổ biến hiện nay, với khả năng tổ chức mã nguồn tốt, bảo mật cao và hỗ trợ phát triển nhanh.

Đồ án không chỉ giúp em vận dụng những kiến thức đã học về lập trình web, cơ sở dữ liệu, phân tích hệ thống mà còn rèn luyện kỹ năng thiết kế và triển khai một hệ thống thực tế. Em hy vọng rằng sản phẩm của mình có thể được áp dụng và phát triển thêm để phục vụ nhu cầu thực tiễn trong tương lai.

# CHƯƠNG 1: TỔNG QUAN NGHIÊN CỨU

**1.1. Khảo sát yêu cầu**

* **Giới thiệu**

Trong thời đại số hóa ngày nay, việc xây dựng một hệ thống quản lý ký túc xá không chỉ giúp tối ưu hóa quy trình quản lý mà còn nâng cao trải nghiệm của sinh viên khi đăng ký và sử dụng dịch vụ ký túc xá. Hệ thống quản lý ký túc xá trên nền web sẽ giúp ban quản lý dễ dàng theo dõi thông tin sinh viên, tình trạng phòng, thanh toán và các thông báo quan trọng. Đồng thời, sinh viên cũng có thể đăng ký phòng, kiểm tra thông tin và thực hiện các thủ tục trực tuyến một cách thuận tiện và nhanh chóng.

* **Mục đích**

Báo cáo này nhằm khảo sát và phân tích các yêu cầu cơ bản cũng như nâng cao cho một hệ thống quản lý ký túc xá trực tuyến. Thông qua đó, hệ thống sẽ được xây dựng với các tính năng phù hợp, đáp ứng tốt nhu cầu của sinh viên và ban quản lý ký túc xá, đảm bảo quy trình hoạt động hiệu quả và thuận tiện hơn.

* **Yêu cầu cơ bản cho “Ứng dụng quản lý KTX” trên nền web**

**Trang chủ:**

* Giao diện thân thiện, thu hút và dễ sử dụng.
* Hiển thị thông tin tổng quan về ký túc xá, bao gồm số lượng phòng, dịch vụ tiện ích và quy định chung.
* Cung cấp các thông báo quan trọng từ ban quản lý ký túc xá.
* Hỗ trợ tìm kiếm nhanh thông tin phòng còn trống.

**Danh mục phòng:**

* Danh sách các loại phòng (phòng đơn, phòng đôi, phòng tập thể, phòng VIP).
* Hiển thị số lượng phòng còn trống theo từng loại.
* Bố cục trực quan, dễ dàng điều hướng giữa các danh mục phòng.

**Quản lý đăng ký phòng:**

* Hiển thị danh sách phòng đã đăng ký.
* Hỗ trợ sinh viên xem lại trạng thái đăng ký và thời gian thuê.
* Cập nhật số lượng sinh viên trong phòng và thông tin hợp đồng thuê.
* Cho phép hủy hoặc gia hạn thuê phòng trực tuyến.

**Thanh toán:**

* Tích hợp các phương thức thanh toán trực tuyến an toàn (VNPay, Momo, chuyển khoản ngân hàng).
* Cập nhật trạng thái thanh toán và gửi xác nhận khi hoàn tất.
* Hỗ trợ xem lịch sử thanh toán và in hóa đơn.

**Quản lý tài khoản:**

* Đăng ký và đăng nhập tài khoản cho sinh viên và ban quản lý ký túc xá.
* Quản lý thông tin cá nhân, cập nhật hồ sơ sinh viên.
* Hiển thị danh sách các đơn đăng ký phòng và lịch sử thanh toán.
* Hỗ trợ sinh viên gửi yêu cầu sửa chữa, phản ánh về chất lượng phòng.

**Quản lý hệ thống (Dành cho ban quản lý ký túc xá):**

* Quản lý danh sách sinh viên đang ở trong ký túc xá.
* Quản lý thông tin phòng, cập nhật trạng thái phòng trống hoặc đã cho thuê.
* Xem và xử lý các yêu cầu phản hồi, bảo trì từ sinh viên.
* Xuất báo cáo về tình trạng thuê phòng, doanh thu và số lượng sinh viên đang ở.
* **Yêu cầu nâng cao**

**Tích hợp hệ thống thanh toán:**

* Tích hợp các cổng thanh toán điện tử phổ biến như PayPal, Stripe, và các hình thức thanh toán khác.
* Tối ưu jóa cho thiết bị di động (Responsive Design)
* Đảm bảo website hoạt động tốt trên mọi loại thiết bị, bao gồm cả điện thoại di động và máy tính bảng.

**Tích hợp hệ thống quản lý nội dung (CMS):**

* Cho phép quản trị viên dễ dàng cập nhật nội dung sản phẩm và thông tin khuyến mãi trên website.

**1.2. Công nghệ sử dụng trong dự án**

1. ***Giới thiệu về ngôn ngữ PHP***

PHP (Hypertext Preprocessor) là một ngôn ngữ lập trình kịch bản (scripting language) chạy trên máy chủ, được thiết kế chủ yếu để phát triển các ứng dụng web động. PHP có thể nhúng trực tiếp vào mã HTML và hỗ trợ nhiều cơ sở dữ liệu khác nhau như MySQL, PostgreSQL, MongoDB.

**Ưu điểm của PHP**

1. **Dễ học và dễ sử dụng**: Cú pháp đơn giản, tương tự như C và JavaScript.
2. **Mã nguồn mở và miễn phí**: Không tốn phí sử dụng.
3. **Tích hợp tốt với cơ sở dữ liệu**: PHP hỗ trợ nhiều hệ quản trị cơ sở dữ liệu như MySQL, PostgreSQL, SQL Server.
4. **Hiệu suất cao**: Chạy tốt trên các server có tài nguyên hạn chế.
5. **Hỗ trợ cộng đồng lớn**: Có nhiều thư viện và framework hỗ trợ như Laravel, CodeIgniter, Symfony.
6. **Đa nền tảng**: Chạy được trên Windows, macOS, Linux.

**Nhược điểm của PHP**

1. **Hiệu suất không cao với ứng dụng lớn**: PHP có thể chậm hơn so với Java hoặc C# khi xử lý ứng dụng lớn.
2. **Không bảo mật cao nếu không được lập trình cẩn thận**: Dễ bị tấn công nếu không bảo vệ tốt.
   * 1. ***Giới thiệu về thư viện Laravel***

Laravel là một framework PHP mã nguồn mở được phát triển bởi Taylor Otwell vào năm 2011 nhằm cung cấp một giải pháp tốt hơn so với CodeIgniter, đặc biệt là về authentication và authorization.



Hình 1.1. Thư viện Laravel

* **Điểm nổi bật của Laravel**

Laravel áp dụng mô hình MVC (Model-View-Controller) giúp tổ chức mã nguồn gọn gàng, dễ bảo trì và phát triển. Model chịu trách nhiệm về dữ liệu, View hiển thị giao diện, còn Controller xử lý logic. Điều này giúp phân tách trách nhiệm rõ ràng, giúp dự án dễ dàng quản lý và mở rộng. Eloquent ORM mạnh mẽ: Laravel tích hợp Eloquent ORM, giúp thao tác với cơ sở dữ liệu một cách dễ dàng và trực quan hơn so với việc viết truy vấn SQL thủ công. ORM hỗ trợ đầy đủ các quan hệ như 1-1, 1-n, n-n, giúp làm việc với dữ liệu hiệu quả và nhanh chóng. Blade Template Engine – Giao diện linh hoạt: Blade là công cụ template mạnh mẽ trong Laravel, hỗ trợ tái sử dụng giao diện thông qua layout và component. Nó cho phép sử dụng các câu lệnh điều kiện, vòng lặp ngay trong view mà không ảnh hưởng đến hiệu suất do có tính năng cache template.

Hệ thống Routing linh hoạt: Laravel có cơ chế định tuyến đơn giản nhưng rất linh hoạt. Nó hỗ trợ group routes, middleware và có thể dễ dàng tùy chỉnh URL động, giúp ứng dụng mở rộng tốt hơn.

Hệ thống Middleware kiểm soát request: Middleware giúp kiểm soát request trước khi đến Controller. Có thể dùng để kiểm tra authentication, CSRF protection, logging, caching.

Hệ thống Authentication & Authorization tích hợp sẵn: Laravel hỗ trợ xác thực người dùng đơn giản với lệnh. Hỗ trợ role-based access control (RBAC) giúp phân quyền dễ dàng.

Artisan Command – Công cụ dòng lệnh tiện lợi: Artisan CLI giúp tự động hóa nhiều tác vụ như tạo controller, model, migration, seeder, factory.

Tích hợp Queue, Task Scheduling, Event Broadcasting: Laravel hỗ trợ Queue giúp xử lý tác vụ nền, Scheduler giúp lập lịch cron job dễ dàng. Event Broadcasting giúp xử lý real-time ứng dụng như chat, thông báo.

Hỗ trợ Test & Debugging tốt: Laravel tích hợp PHPUnit giúp viết test tự động dễ dàng. Có công cụ Laravel Telescope giúp debug hiệu quả.

* Những hạn chế Laravel

Hiệu suất thấp hơn so với các framework khác: Laravel có nhiều tính năng tích hợp sẵn, điều này làm cho nó nặng hơn so với các framework nhẹ như Lumen, CodeIgniter. Tuy nhiên, có thể tối ưu bằng caching, queue, database indexing.

Học tập ban đầu có thể khó với người mới: Dù Laravel có cú pháp dễ hiểu, nhưng với người mới bắt đầu PHP + MVC + OOP, việc làm quen với ORM, Blade, Middleware, Service Providers có thể khó khăn.

Quá phụ thuộc vào Composer và thư viện bên thứ ba: Laravel yêu cầu Composer để quản lý package, điều này có thể gây khó khăn với người chưa quen. Một số package phụ thuộc vào bên thứ ba, có thể gây vấn đề về bảo mật hoặc tương thích phiên bản.

Không phù hợp với các ứng dụng quá nhỏ hoặc đơn giản: Laravel mạnh mẽ nhưng nếu chỉ cần xây dựng website tĩnh hoặc API đơn giản, dùng Laravel có thể là thừa thãi. Trong trường hợp này, có thể sử dụng Lumen (phiên bản nhẹ của Laravel).

Cập nhật nhanh có thể gây mất tương thích: Laravel cập nhật phiên bản thường xuyên (mỗi năm một lần), có thể gây mất tương thích khi nâng cấp. Điều này buộc lập trình viên phải cập nhật mã nguồn và kiểm tra lại toàn bộ ứng dụng.

**Các lệnh quan trọng sau khi cài đặt Laravel**

Bảng 1.1. Các lệnh quan trọng trong Laravel

|  |  |
| --- | --- |
| Lệnh | Mô tả |
| php artisan serve | Khởi chạy server Laravel |
| php artisan migrate | Chạy migration để tạo bảng database |
| php artisan make:controller TenController | Tạo một controller mới |
| php artisan make:model TenModel | Tạo một model mới |
| Php artisan make:migration create\_ten\_table | Tạo một migration mới |
| composer update | Cập nhật các package Laravel |
| php artisan route:list | Xem danh sách route |

**Thư mục app (Chứa Model, Controller, Middleware,...)**

Bảng 1.2. Cấu trúc thư mục app

|  |  |
| --- | --- |
| Thư mục | Chức năng |
| Models/ | Chứa các Model của ứng dụng, giúp làm việc với cơ sở dữ liệu bằng Eloquent ORM. |
| Http/ | Chứa Controllers, Middleware, Requests phục vụ xử lý request từ người dùng. |
| Http/Controllers/ | Chứa các file Controller để điều khiển logic của ứng dụng. |
| Http/Middleware/ | Chứa các Middleware giúp kiểm soát request trước khi xử lý. |
| Providers/ | Chứa các file Service Provider, giúp đăng ký các dịch vụ và xử lý dependency injection. |

**Thư mục config (Cấu hình ứng dụng)**

Bảng 1.3. Cấu trúc thư mục config

|  |  |
| --- | --- |
| File | Chức năng |
| config/app.php | Cấu hình chung của ứng dụng như timezone, locale, key. |
| config/database.php | Cấu hình database như MySQL, PostgreSQL, SQLite. |
| config/mail.php | Cấu hình email trong Laravel. |

**Thư mục database (Migration, Seeder, Factory)**

Bảng 1.4. Cấu trúc thư mục database

|  |  |
| --- | --- |
| Thư mục | Chức năng |
| migrations/ | Chứa các file migration giúp quản lý bảng trong database. |
| seeders/ | Chứa các Seeder để thêm dữ liệu mẫu vào database. |
| factories/ | Chứa các Factory giúp tạo dữ liệu giả (fake data). |

**Thư mục resources (Views, CSS, JS).**

Bảng 1.5. Cấu trúc thư mục resources

|  |  |
| --- | --- |
| Thư mục | Chức năng |
| views/ | Chứa các file giao diện (Blade templates). |
| lang/ | Chứa file ngôn ngữ (đa ngôn ngữ cho ứng dụng). |
| css/ | Chứa file CSS riêng của Laravel. |
| js/ | Chứa file JavaScript riêng. |

**Thư mục routes (Web, API, Console Routes)**

Bảng 1.6. Cấu trúc thư mục routes

|  |  |
| --- | --- |
| File | Chức năng |
| web.php | Chứa các route cho ứng dụng web. |
| api.php | Chứa các route API. |

* **Model trong Laravel**

Khái niệm: Model trong Laravel đại diện cho một bảng trong cơ sở dữ liệu. Laravel sử dụng Eloquent ORM (Object-Relational Mapping) để tương tác với database theo kiểu hướng đối tượng.

Tạo Model trong Laravel**:** “php artisan make:model Product”

* **View trong Laravel**

Khái niệm: View là phần hiển thị dữ liệu cho người dùng. Laravel sử dụng Blade Template Engine, giúp tái sử dụng mã HTML và dễ dàng nhúng dữ liệu từ Controller.

Cấu trúc View: Các file views thường được tạo và lưu trữ dưới dạng \*.blade.php

* **Controller trong Laravel**

Khái niệm: Controller là nơi xử lý logic của ứng dụng. Controller nhận request từ người dùng, xử lý và trả về response (View hoặc JSON).

Tạo Controller: “php artisan make:controller ProductController”.

* + 1. ***Giới thiệu về Mysql***

Khái niệm: MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở, được sử dụng rộng rãi để lưu trữ dữ liệu trong các ứng dụng web.

Thông tin cơ bản

* Được phát triển bởi MySQL AB, hiện thuộc sở hữu của Oracle Corporation.
* Sử dụng SQL (Structured Query Language) để thao tác dữ liệu.
* Hỗ trợ các hệ điều hành như Windows, macOS, Linux.

**Ưu điểm của MySQL**

* Mã nguồn mở, miễn phí.
* Tốc độ xử lý nhanh, đáng tin cậy.
* Dễ sử dụng với cú pháp SQL đơn giản.
* Hỗ trợ nhiều hệ điều hành và tích hợp tốt với PHP.

**Nhược điểm của MySQL**

* Hạn chế về xử lý giao dịch phức tạp so với các hệ quản trị cơ sở dữ liệu như PostgreSQL.
* Không hỗ trợ tốt NoSQL như MongoDB.

**Cài đặt MySQL:** Cài đặt và sử dụng thông qua XAMPP. Có nhiều phiên bản phù hợp cho các hệ điều hành.

* + 1. ***Giới thiệu về Framework Bootstrap***

Bootstrap là một framework CSS phổ biến giúp phát triển giao diện web nhanh chóng và dễ dàng. Được phát triển bởi Twitter, Bootstrap cung cấp sẵn các thành phần UI, hệ thống lưới (grid), kiểu dáng CSS, và các tiện ích JavaScript, giúp lập trình viên xây dựng giao diện đáp ứng (responsive) mà không cần viết quá nhiều mã CSS.



Hình 1.2. Thư viện Bootstrap

* **Tại sao nên dùng Bootstrap**

Tiết kiệm thời gian: Bootstrap cung cấp nhiều thành phần UI có sẵn như nút bấm (buttons), biểu mẫu (forms), bảng (tables), điều hướng (navbars), giúp lập trình viên không cần thiết kế từ đầu.

Thiết kế giao diện Responsive dễ dàng: Bootstrap sử dụng hệ thống Grid System giúp giao diện hiển thị tốt trên máy tính, tablet, điện thoại mà không cần viết nhiều CSS.

Dễ sử dụng & tùy chỉnh: Bootstrap có tài liệu hướng dẫn đầy đủ, dễ học. Hỗ trợ tùy chỉnh CSS để phù hợp với thiết kế riêng của từng dự án.

Tương thích với Laravel: Laravel hỗ trợ tích hợp Bootstrap dễ dàng qua npm, CDN hoặc cài đặt thủ công. Bootstrap giúp xây dựng giao diện nhanh hơn khi kết hợp với Blade Template Engine của Laravel.

* + 1. ***Giới thiệu về Git và GitHub***

**Git là gì?**

* Git là một hệ thống kiểm soát phiên bản phân tán (DVCS) mã nguồn mở.
* Nó cho phép theo dõi các thay đổi trong tệp tin và thư mục theo thời gian, giúp bạn quản lý lịch sử dự án một cách hiệu quả.
* Git hoạt động trên nguyên tắc "snapshot", lưu trữ các thay đổi dưới dạng các bản ghi (commit).
* Nó hỗ trợ làm việc nhóm bằng cách cho phép nhiều người cùng làm việc trên cùng một dự án mà không gây xung đột.
* Git cho phép tạo các nhánh (branch) để phát triển tính năng mới hoặc sửa lỗi mà không ảnh hưởng đến mã nguồn chính.

**Các lợi ích chính của Git:**

* Kiểm soát phiên bản: Theo dõi mọi thay đổi và dễ dàng quay lại các phiên bản trước đó.
* Làm việc nhóm: Hỗ trợ cộng tác hiệu quả giữa nhiều người.
* Quản lý nhánh: Cho phép phát triển song song và thử nghiệm tính năng mới.
* Sao lưu và phục hồi: Giúp bảo vệ mã nguồn và dễ dàng khôi phục khi cần thiết.
* Tốc độ và hiệu năng: Git được thiết kế để hoạt động nhanh chóng và hiệu quả.

**GitHub là gì?**

* GitHub là một dịch vụ lưu trữ kho mã nguồn dựa trên nền tảng web, sử dụng Git làm hệ thống kiểm soát phiên bản.
* Nó cung cấp một nền tảng để lưu trữ, quản lý và chia sẻ các dự án Git.
* GitHub cung cấp các công cụ và tính năng để hỗ trợ làm việc nhóm, chẳng hạn như yêu cầu kéo (pull requests), đánh giá mã (code reviews) và theo dõi vấn đề (issue tracking).
* GitHub là một nền tảng mạng xã hội cho các nhà phát triển, nơi họ có thể chia sẻ mã nguồn, khám phá các dự án mã nguồn mở và kết nối với nhau.

**Mối quan hệ giữa Git và GitHub:**

* Git là công cụ kiểm soát phiên bản, còn GitHub là nền tảng lưu trữ và cộng tác dựa trên Git.
* Bạn sử dụng Git để quản lý các thay đổi mã nguồn cục bộ trên máy tính của mình, sau đó sử dụng GitHub để lưu trữ và chia sẻ mã nguồn đó trên đám mây.
* GitHub cung cấp giao diện web và các công cụ bổ sung để làm việc với Git một cách dễ dàng hơn.

**Các thuật ngữ Git và GitHub quan trọng:**

* Repository (Kho lưu trữ): Nơi chứa mã nguồn và lịch sử thay đổi.
* Commit (Bản ghi): Một bản ghi các thay đổi trong mã nguồn.
* Branch (Nhánh): Một phiên bản phát triển song song của mã nguồn.
* Pull request (Yêu cầu kéo): Một yêu cầu hợp nhất các thay đổi từ một nhánh vào nhánh khác.
* Clone (Sao chép): Tạo một bản sao của kho lưu trữ Git trên máy tính của bạn.
* Push (Đẩy): Tải các thay đổi từ kho lưu trữ cục bộ lên kho lưu trữ từ xa (GitHub).
* Pull (Kéo): Tải các thay đổi từ kho lưu trữ từ xa về kho lưu trữ cục bộ.

**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

* 1. **Tổng quan về nghiệp vụ**

Hệ thống quản lý ký túc xá được thiết kế nhằm tối ưu hóa quy trình đăng ký, quản lý và theo dõi tình trạng phòng ở. Hệ thống có hai tác nhân chính: sinh viên và quản trị viên.

Sinh viên là người sử dụng hệ thống để đăng ký thuê phòng và theo dõi thông tin cá nhân trong ký túc xá. Các chức năng chính của sinh viên bao gồm:

Duyệt danh sách phòng: Xem thông tin chi tiết về các phòng trống, bao gồm giá thuê, số lượng giường, tiện ích và quy định.

* Đăng ký phòng: Chọn phòng mong muốn và gửi yêu cầu đăng ký trực tuyến.
* Quản lý hợp đồng thuê: Theo dõi trạng thái đăng ký, gia hạn hoặc hủy hợp đồng khi cần thiết.
* Thanh toán trực tuyến: Hỗ trợ các phương thức thanh toán an toàn như VNPay, Momo hoặc chuyển khoản ngân hàng.
* Gửi phản hồi và yêu cầu hỗ trợ: Sinh viên có thể báo cáo sự cố phòng ở, yêu cầu bảo trì hoặc gửi góp ý về chất lượng dịch vụ.

Quản trị viên là người chịu trách nhiệm quản lý toàn bộ hệ thống ký túc xá. Các chức năng chính của quản trị viên bao gồm:

* Quản lý danh sách phòng: Cập nhật trạng thái phòng (trống/đã thuê), điều chỉnh thông tin phòng, giá thuê và tiện ích.
* Duyệt đơn đăng ký: Xác nhận hoặc từ chối yêu cầu thuê phòng của sinh viên.
* Quản lý hợp đồng thuê: Theo dõi lịch sử thuê phòng, gia hạn hoặc chấm dứt hợp đồng khi cần.
* Quản lý thanh toán: Kiểm tra trạng thái thanh toán, xuất báo cáo doanh thu và xử lý các yêu cầu hoàn tiền.
* Xử lý phản hồi: Giải quyết các yêu cầu bảo trì phòng ở, tiếp nhận ý kiến đóng góp từ sinh viên.
* Quản lý hệ thống: Theo dõi hoạt động của người dùng, phân quyền tài khoản và đảm bảo hệ thống hoạt động ổn định.

Việc phân chia chức năng rõ ràng giữa các tác nhân giúp hệ thống quản lý ký túc xá hoạt động hiệu quả, minh bạch và thuận tiện cho cả sinh viên và ban quản lý. Hệ thống không chỉ giúp tối ưu hóa quy trình quản lý phòng ở mà còn góp phần nâng cao trải nghiệm cho sinh viên khi sử dụng dịch vụ ký túc xá tại trường.

* 1. **Phân tích chức năng của hệ thống**

1. ***Biểu đồ usecase tổng quan hệ thống***



Hình 2.1. Biểu đồ usecase tổng quan hệ thống

1. ***Đặc tả usecase***

*2.2.2.1. Đặc tả usecase “Đăng nhập”*

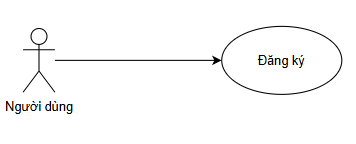


Hình 2.2. Biểu đồ usecase đăng nhập

Bảng 2.1. Đặc tả usecase “Đăng nhập”

|  |  |
| --- | --- |
| **Tên usecase** | **Đăng nhập** |
| Mô tả | Usecase cho phép khách hàng đăng nhập vào hệ thống |
| Actor | Khách hàng, quản trị viên |
| Điều kiện kích hoạt | Người dùng chọn chức năng đăng nhập từ hệ thống |
| Tiền điều kiện | Người dùng đã đăng ký tài khoản |
| Hậu điều kiện | Đăng nhập thành công |
| Luồng sự kiện chính | 1. KH, QTV: Chọn chức năng đăng nhập 2. HT: Hiển thị form 3. KH, QTV: Nhập thông tin 4. HT: Kiểm tra thông tin của KH và QTV    1. HT: Thông báo thông tin sai hoặc chưa đăng ký.    2. HT: Chuyển đến trang chủ và thông báo đăng nhập thành công |

*2.2.2.2. Đặc tả usecase “Đăng ký”*

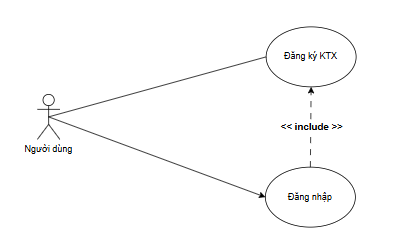


Hình 2.3. Biểu đồ usecase đăng ký

Bảng 2.2. Đặc tả usecase “Đăng ký”

|  |  |
| --- | --- |
| **Tên usecase** | **Đăng ký** |
| Mô tả | Usecase cho phép khách hàng đăng ký tài khoản tại website |
| Actor | Khách hàng |
| Điều kiện kích hoạt | Người dùng chọn chức năng đăng ký từ hệ thống |
| Tiền điều kiện | Người dùng chưa có tài khoản trên hệ thống |
| Hậu điều kiện | Đăng ký thành công |
| Luồng sự kiện chính | 1. KH: Đăng ký 2. HT: hiển thị form đăng ký 3. KH: Nhập thông tin 4. KH: Nhấn nút đăng ký 5. HT: Kiểm tra thông tin   6.1 HT: Thông báo thông tin trùng, sai,.. Đăng ký thất bại  6.2 HT: Thông báo đăng ký thành công. Chuyển đến trang đăng nhập |

*2.2.2.3. Đặc tả usecase “Đăng ký ở KTX”*

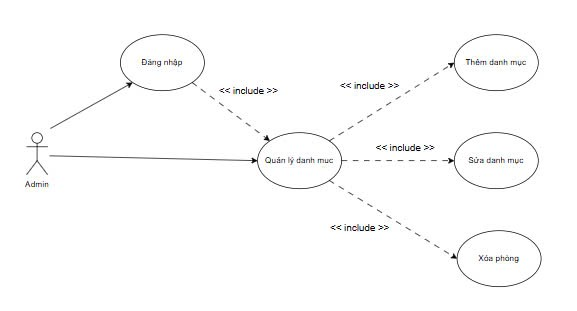


Hình 2.4. Biểu đồ usecase đăng ký KTX

Bảng 2.3. Đặc tả usecase “Đăng ký ở KTX”

|  |  |
| --- | --- |
| **Tên usecase** | **Đặt hàng** |
| Mô tả | Usecase cho phép khách hàng đăngh ký |
| Actor | Khách hàng |
| Điều kiện kích hoạt | Người dùng chọn chức năng đặt hàng |
| Tiền điều kiện | Người dùng sau khi đến trang đăng ký và nhấn nút đăng ký |
| Hậu điều kiện | Người dùng đăng ký thành công |
| Luồng sự kiện chính | 1. KH: Chọn chức năng đắng ký KTX 2. HT: Hiển thị thông tin các hàng và form nhập thông tin 3. KH: Nhập thông tin thanh toán 4. HT: Kiểm tra thông tin    1. HT: Thông tin hợp lệ. Đặt sân thành công    2. HT: Thông tin không hợp lệ. Thông báo lỗi |

*2.2.2.4. Đặc tả usecase “Danh mục” - Quản trị*

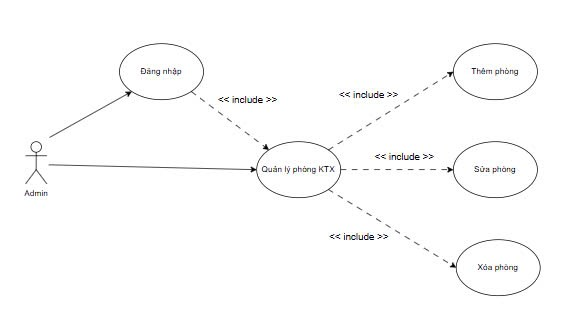


Hình 2.5. Biểu đồ usecase quản trị danh mục

Bảng 2.4. Đặc tả usecase “Danh mục” - Quản trị

|  |  |
| --- | --- |
| **Tên usecase** | **Quản lý danh mục** |
| Mô tả | Usecase cho phép QTV quản lý danh mục |
| Actor | QTV |
| Điều kiện kích hoạt | QTV chọn chức năng quản lý danh mục |
| Tiền điều kiện | QTV sau khi thêm, sửa, xóa, tìm kiếm danh mục |
| Hậu điều kiện | QTV thêm, sửa, xóa, tìm kiếm danh mục thành công |
| Luồng sự kiện chính | 1. QTV chọn chức năng quản lý danh mục 2. HT hiện thị form tạo mới hoặc cập nhật danh mục 3. QTV nhập thông tin 4. HT kiểm tra thông tin   5.1 HT: thông tin đúng thêm hoặc cập nhật thành công  5.2 HT: thông tin sai thêm hoặc cập nhất thất bại |

*2.2.2.5. Đặc tả usecase “Phòng KTX” - Quản trị*

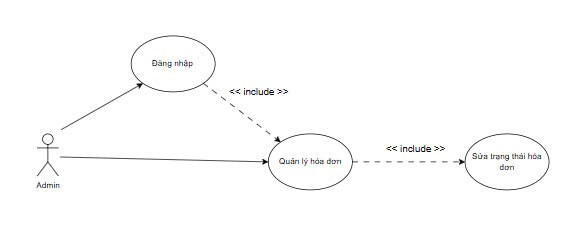


Hình 2.6. Biểu đồ usecase quản trị phòng KTX

Bảng 2.5. Đặc tả usecase “Phòng KTX” - Quản trị

|  |  |
| --- | --- |
| **Tên usecase** | **Quản lý phòng** |
| Mô tả | Usecase cho phép QTV quản lý phòng |
| Actor | QTV |
| Điều kiện kích hoạt | QTV chọn chức năng quản lý phòng |
| Tiền điều kiện | QTV sau khi thêm, sửa, xóa, tìm kiếm phòng |
| Hậu điều kiện | QTV thêm, sửa, xóa, tìm kiếm phòng thành công |
| Luồng sự kiện chính | 1. QTV chọn chức năng quản lý phòng  2. HT hiện thị form tạo mới hoặc cập nhật  3. QTV nhập thông tin  4. HT kiểm tra thông tin  5.1 HT: thông tin đúng thêm hoặc cập nhật thành công   * 1. 5.2 HT: thông tin sai thêm hoặc cập nhất thất bại |

*2.2.2.6. Đặc tả usecase “Hóa đơn” - Quản trị*

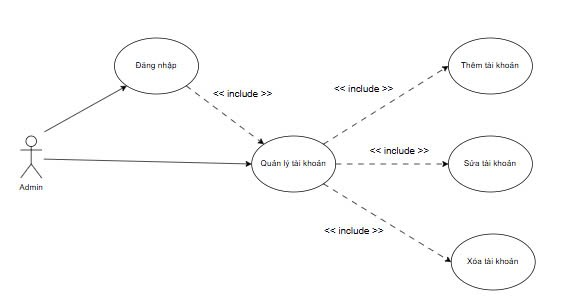


Hình 2.7. Biểu đồ usecase quản trị hóa đơn

Bảng 2.6. Đặc tả usecase “Hóa đơn” - Quản trị

|  |  |
| --- | --- |
| **Tên usecase** | **Quản lý hóa đơn** |
| Mô tả | Usecase cho phép QTV sửa trạng thái hóa đơn |
| Actor | QTV |
| Điều kiện kích hoạt | Người dùng nhấn vào mục sửa |
| Tiền điều kiện | Người dùng sau khi đăng nhập thành công vào hệ thống và ở mục “Thông tin hóa đơn” |
| Hậu điều kiện | Hệ thống cập nhật trạng thái hóa đơn |
| Luồng sự kiện chính | 1. QTV: chọn chức năng sửa 2. HT: cập nhật trạng thái hóa đơn thành công rồi chuyển về trang danh sách |

*2.2.2.7. Đặc tả usecase “Tài khoản” - Quản trị*

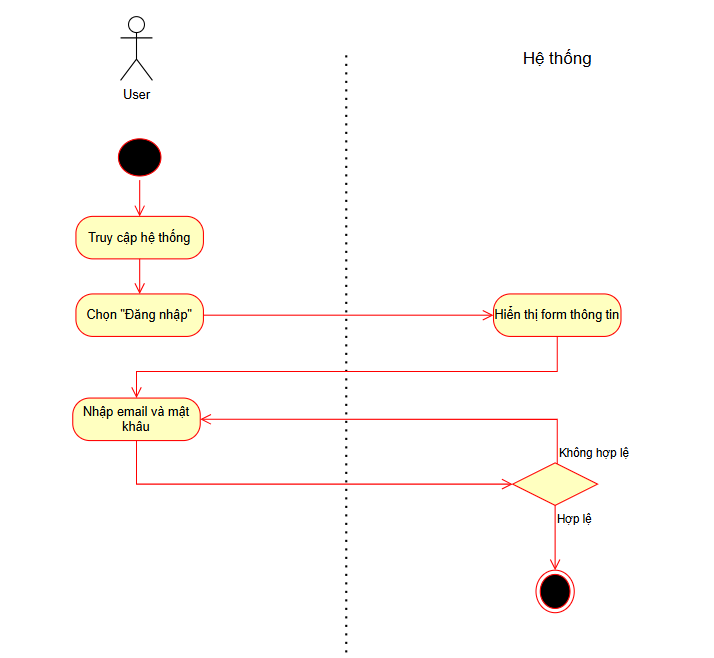


Hình 2.8. Biểu đồ usecase quản trị tài khoản

Bảng 2.7. Đặc tả usecase “Tài khoản” - Quản trị

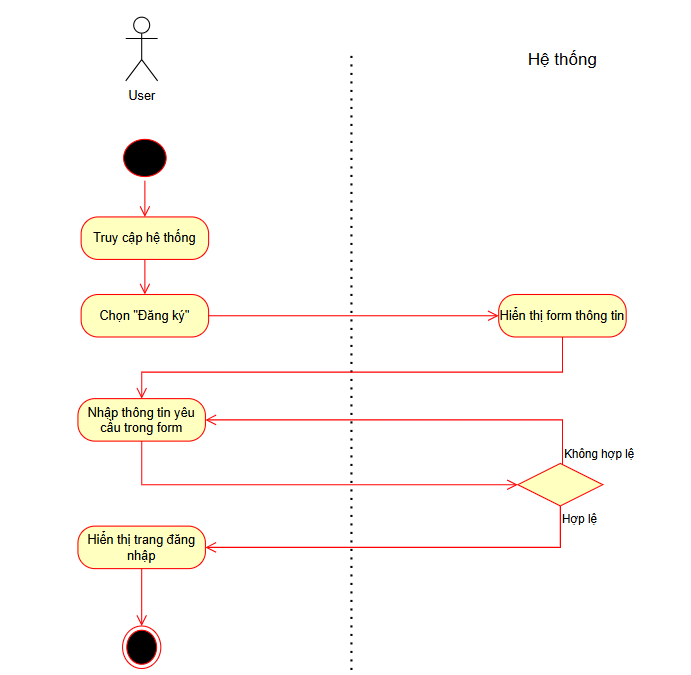
|  |  |
| --- | --- |
| **Tên usecase** | **Quản lý người dùng** |
| Mô tả | Usecase cho phép QTV thêm sửa xóa người dùng |
| Actor | QTV |
| Điều kiện kích hoạt | Người dùng nhấn vào mục thêm sửa xóa |
| Tiền điều kiện | Người dùng sau khi đăng nhập thành công vào hệ thống và ở mục “Thông tin cá nhân người dùng” |
| Hậu điều kiện | Hệ thống cập nhật danh sách người dùng |
| Luồng sự kiện chính | 1. QTV: chọn chức năng thêm, sửa, xóa 2. HT: cập nhật thông tin người dùng rồi chuyển về trang danh sách |

* 1. **Phân tích yêu cầu phi chức năng**
     1. ***Đăng nhập***



Hình 2.9. Sơ đồ hoạt động chức năng đăng nhập

* + 1. ***Đăng ký***



Hình 2.10. Sơ đồ hoạt động chức năng đăng ký

* + 1. ***Quên mật khẩu***

A diagram of a work flow

Description automatically generated

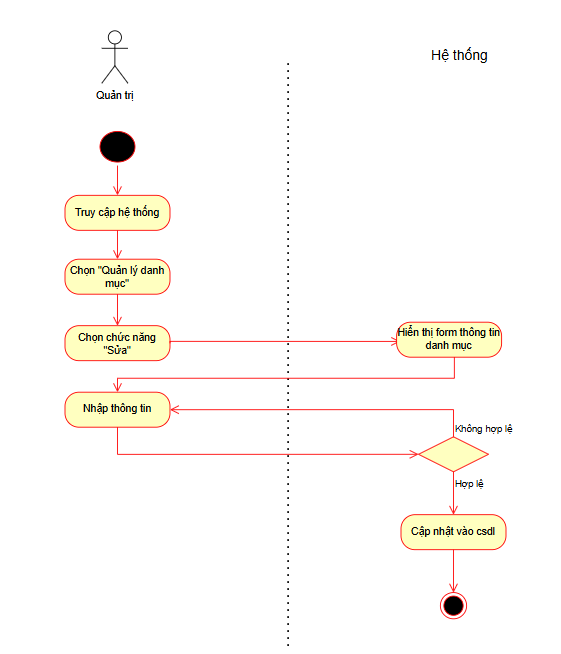
Hình 2.11. Sơ đồ hoạt động chức năng quên mật khẩu

* + 1. ***Thêm danh mục***



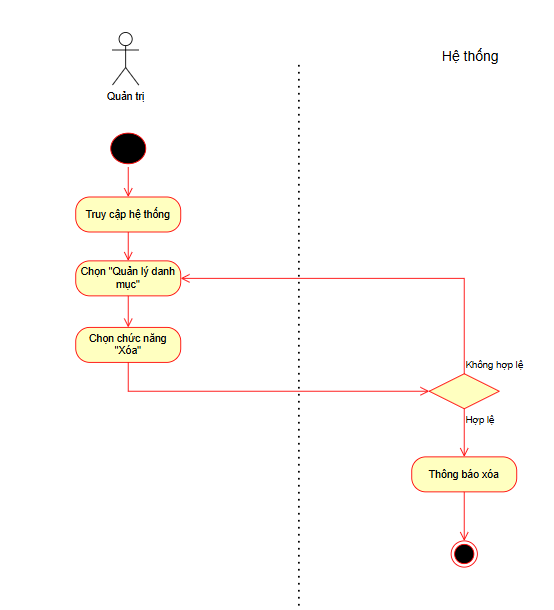
Hình 2.12. Biểu diễn hoạt động chức năng thêm danh mục

* + 1. ***Cập nhật danh mục***



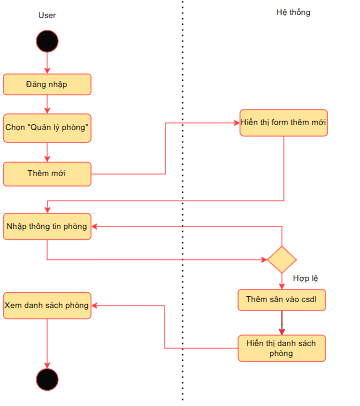
Hình 2.13. Biểu diễn hoạt động chức năng cập nhật danh mục

* + 1. ***Xóa danh mục***



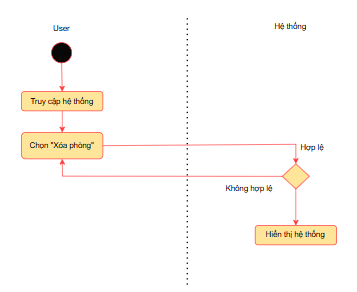
Hình 2.14. Biểu diễn hoạt động chức năng xóa danh mục

* + 1. ***Thêm phòng KTX***



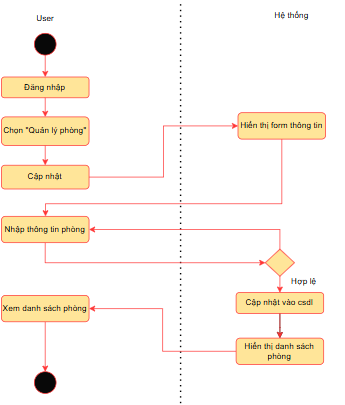
Hình 2.15. Sơ đồ hoạt động chức năng thêm phòng KTX

* + 1. ***Xóa phòng KTX***



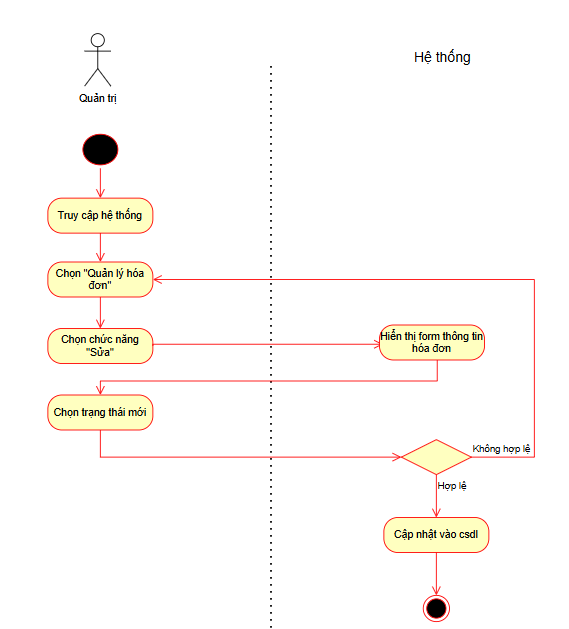
Hình 2.16. Sơ đồ hoạt động chức năng xóa phòng KTX

***2.3.9. Cập nhật phòng KTX***



Hình 2.17. Sơ đồ hoạt động chức năng cập nhật phòng KTX

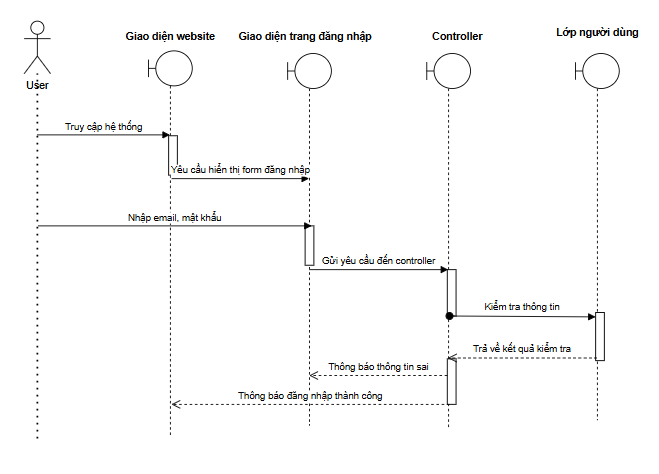
* + 1. ***Cập nhật trạng thái hóa đơn***



Hình 2.18. Sơ đồ hoạt động chức năng cập nhật trạng thái hóa đơn

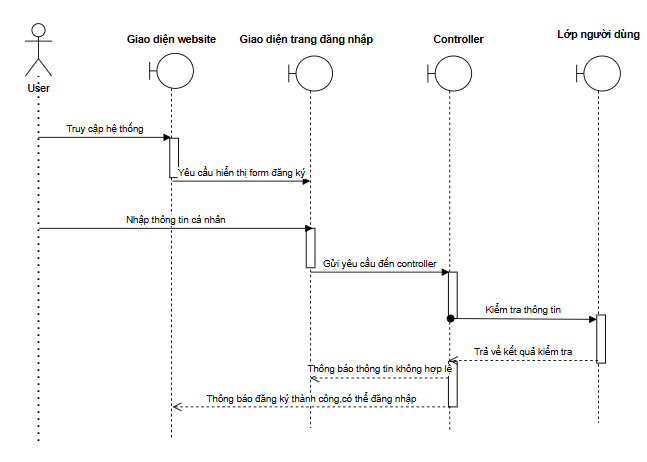
**2.4****. Phân tích biểu đồ**

1. ***Biểu đồ tuần tự “Đăng nhập”***



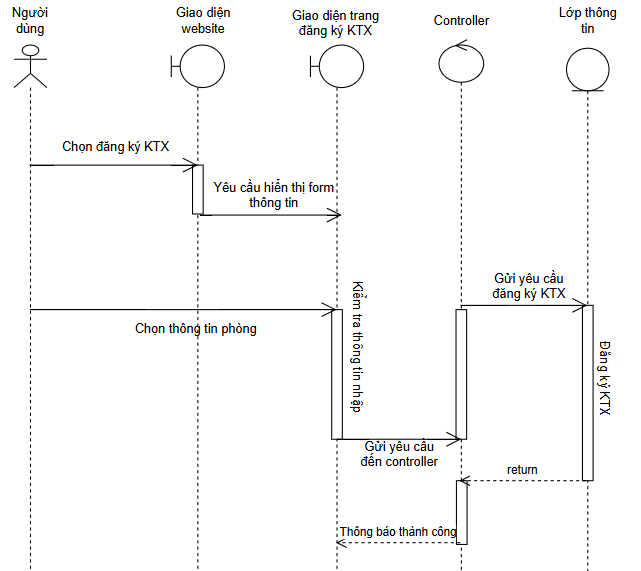
Hình 2.19. Biểu đồ tuần tự Đăng nhập

* + 1. ***Biểu đồ tuần tự “Đăng ký”***



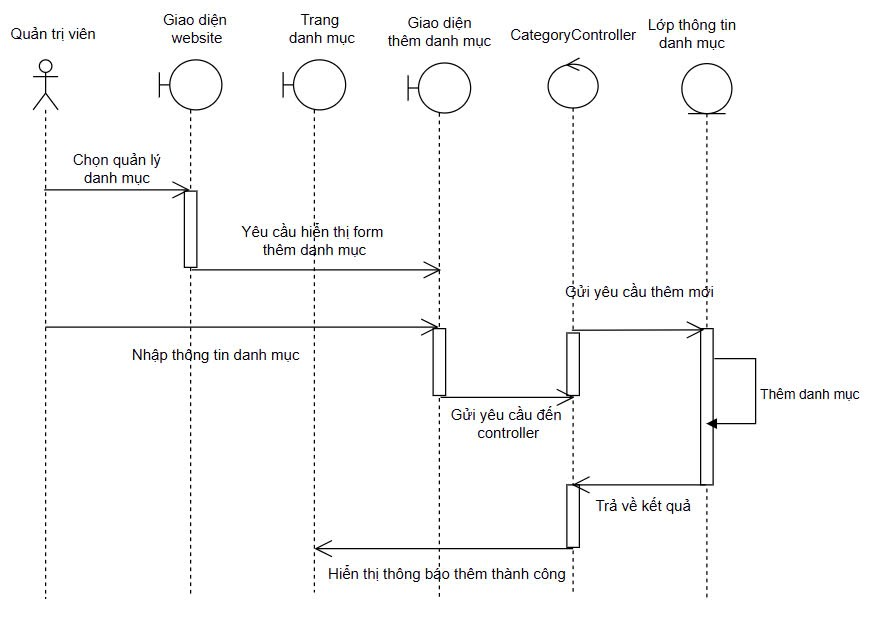
Hình 2.20. Biểu đồ tuần tự Đăng ký

* + 1. ***Biểu đồ tuần tự “Đăng ký KTX”***



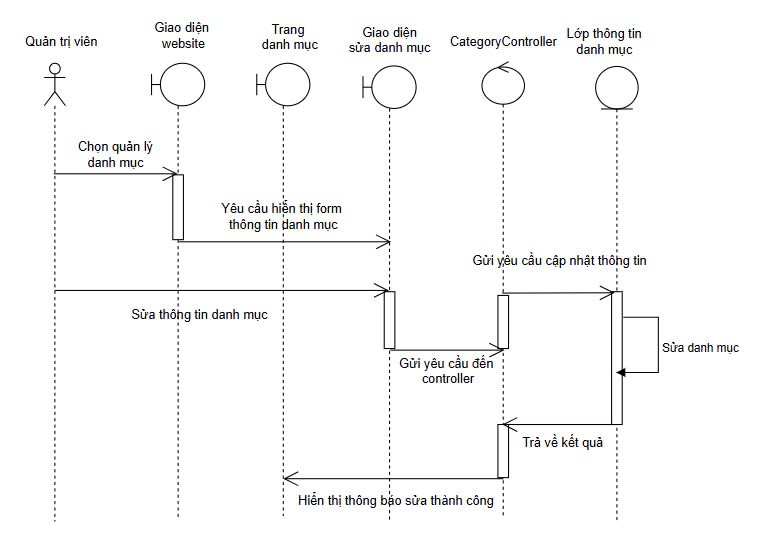
Hình 2.21. Biểu đồ tuần tự chức năng đăng ký KTX

* + 1. ***Biểu đồ tuần tự “Thêm danh mục”***



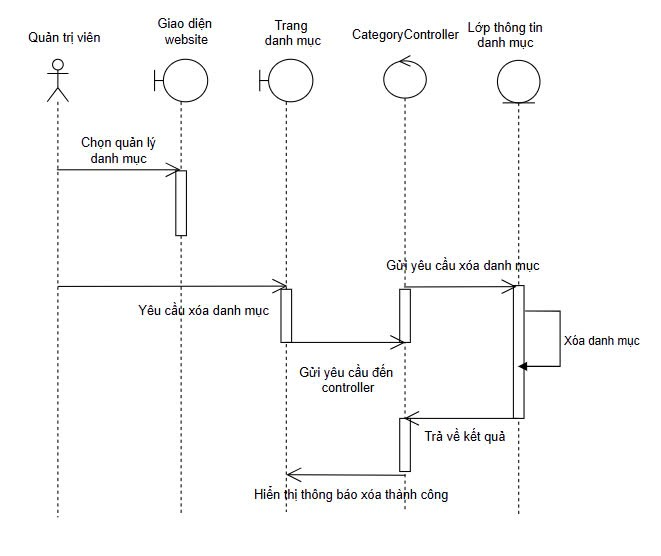
Hình 2.22. Biểu đồ tuần tự chức năng thêm danh mục

* + 1. ***Biểu đồ tuần tự “Cập nhật danh mục”***



Hình 2.23. Biểu đồ tuần tự chức năng Cập nhật danh mục

* + 1. ***Biểu đồ tuần tự “Xóa danh mục”***



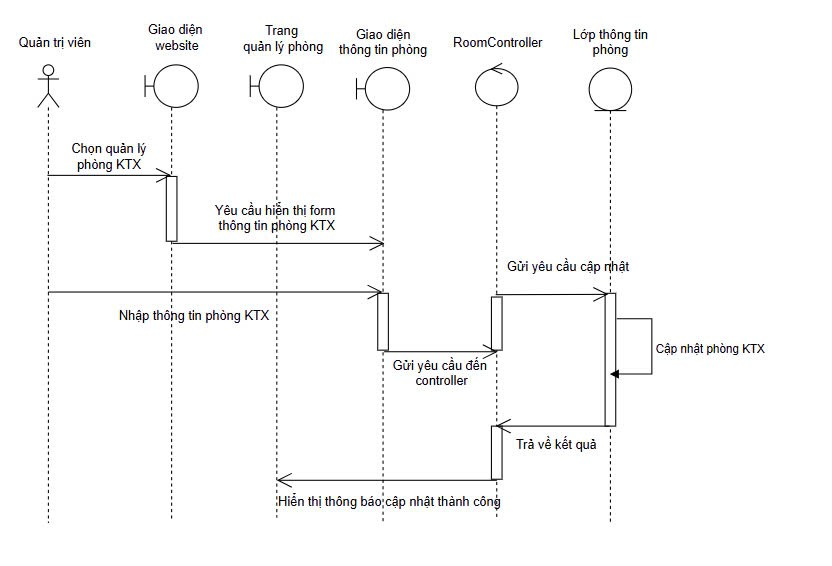
Hình 2.24. Biểu đồ tuần tự chức năng xóa danh mục

* + 1. ***Biểu đồ tuần tự “Thêm phòng KTX”***



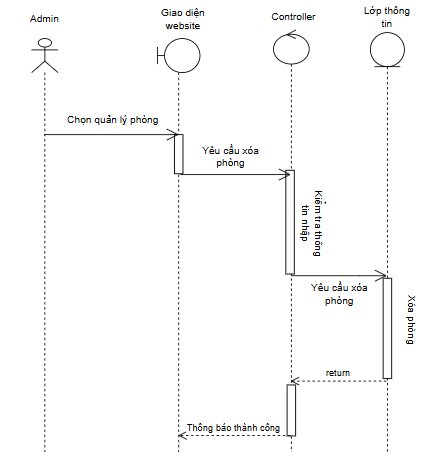
Hình 2.25. Biểu đồ tuần tự chức năng thêm phòng KTX

* + 1. ***Biểu đồ tuần tự “Cập nhật phòng KTX”***



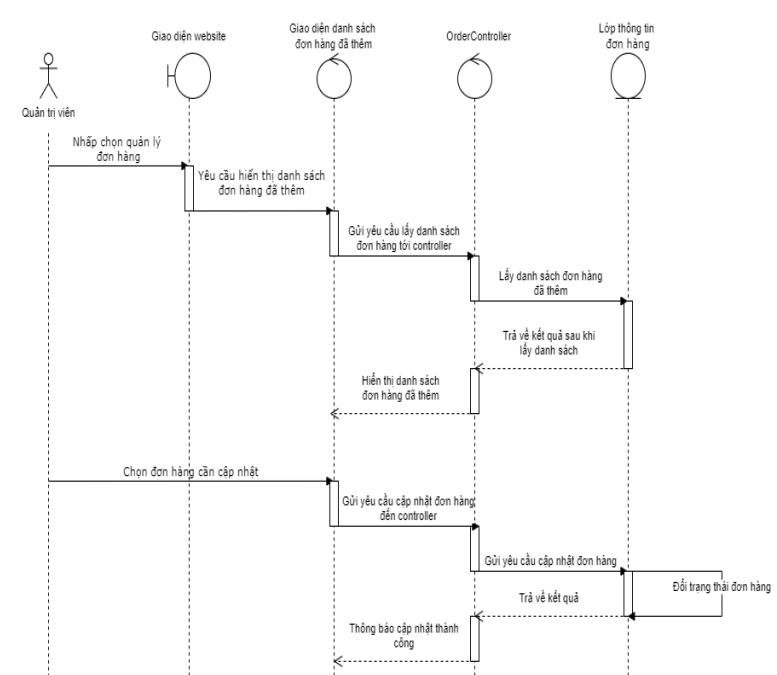
Hình 2.26. Biểu đồ tuần tự chức năng cập nhật sản phẩm

* + 1. ***Biểu đồ tuần tự “Xóa phòng KTX”***



Hình 2.27. Biểu đồ tuần tự chức năng xóa sản phẩm

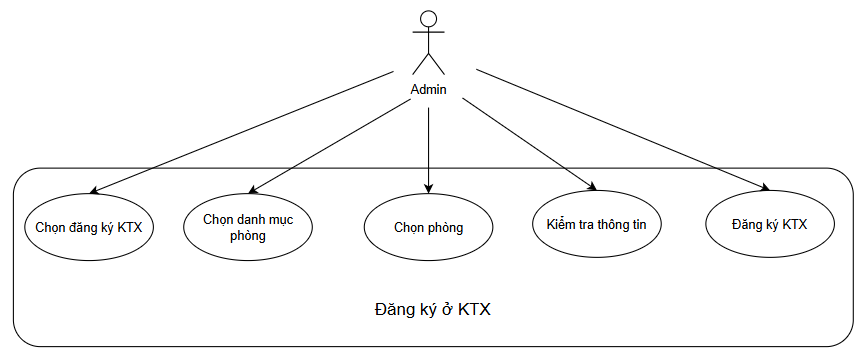
* + 1. ***Biểu đồ tuần tự “Cập nhật trạng thái hóa đơn”***



Hình 2.28. Biểu đồ tuần tự chức năng cập nhập trạng thái hóa đơn

* 1. **Phân rã usecase**

1. ***Phân rã usecase “Đăng ký ở KTX”***



Hình 2.29. Phân rã usecase “Đăng kí”

**Chọn đăng ký KTX:** Đây là bước đầu tiên trong quá trình đăng ký, nơi người dùng lựa chọn tính năng đăng ký KTX trên hệ thống. Trước khi tiếp tục, hệ thống sẽ kiểm tra điều kiện đăng ký của sinh viên. Nếu sinh viên đã từng ở KTX trước đó hoặc có vi phạm nội quy, hệ thống có thể từ chối yêu cầu đăng ký hoặc yêu cầu bổ sung thông tin. Việc kiểm tra điều kiện này giúp đảm bảo sự công bằng và quản lý tốt hơn danh sách sinh viên đăng ký.

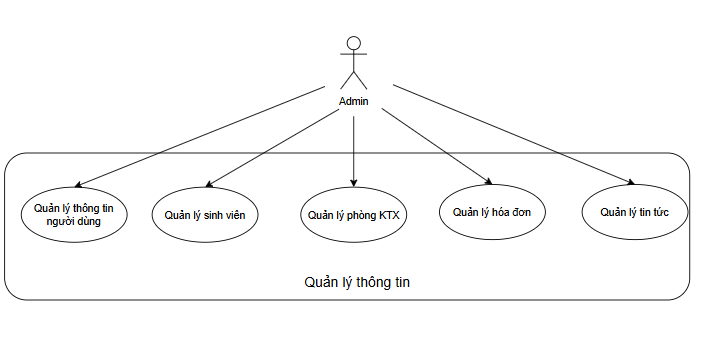
**Chọn danh mục phòng:** Sau khi vượt qua bước kiểm tra điều kiện, sinh viên sẽ tiến hành lựa chọn danh mục phòng phù hợp với nhu cầu. Hệ thống cung cấp danh sách các loại phòng như phòng đơn, phòng đôi, phòng nhóm, phòng VIP, kèm theo thông tin chi tiết về tiện ích, giá cả, số lượng giường. Để giúp sinh viên dễ dàng tìm được phòng mong muốn, hệ thống cung cấp bộ lọc tìm kiếm theo tiêu chí như giá cả, vị trí, số lượng giường hoặc các tiện ích đặc biệt.

**Chọn phòng:** Khi danh mục phòng đã được chọn, hệ thống sẽ hiển thị danh sách các phòng còn trống. Sinh viên có thể xem chi tiết từng phòng bao gồm diện tích, số lượng giường, giá thuê, các tiện ích đi kèm như điều hòa, máy giặt, bàn học… Bằng cách này, người dùng có thể đưa ra quyết định chính xác trước khi tiếp tục đăng ký. Nếu phòng đã hết chỗ hoặc không đáp ứng nhu cầu, sinh viên có thể quay lại để chọn một danh mục phòng khác.

**Kiểm tra thông tin:** Trước khi hoàn tất đăng ký, hệ thống sẽ yêu cầu sinh viên kiểm tra lại toàn bộ thông tin cá nhân cũng như thông tin về phòng đã chọn. Các thông tin cần xác nhận bao gồm họ tên, mã số sinh viên, số điện thoại liên hệ, loại phòng, thời gian thuê phòng. Nếu có sai sót, sinh viên có thể quay lại chỉnh sửa trước khi xác nhận đăng ký. Việc kiểm tra này giúp tránh các lỗi nhập liệu không mong muốn và đảm bảo thông tin chính xác.

**Đăng ký KTX:** Sau khi kiểm tra thông tin, sinh viên tiến hành xác nhận đăng ký. Hệ thống sẽ xử lý yêu cầu, cập nhật dữ liệu vào hệ thống quản lý KTX và hiển thị thông báo kết quả. Nếu đăng ký thành công, hệ thống sẽ gửi xác nhận qua email hoặc hiển thị thông tin trên giao diện. Trường hợp có lỗi, ví dụ như phòng đã hết chỗ hoặc xảy ra vấn đề trong quá trình xử lý, hệ thống sẽ yêu cầu sinh viên chọn lại phòng hoặc liên hệ với quản lý KTX để được hỗ trợ.

1. ***Phân rã usecase “Quản lý thông tin”***



Hình 2.30. Phân rã usecase “Quản lý thông tin”

**Các bước phân rã:**

Quản lý thông tin người dùng: Đây là phân hệ cho phép quản trị viên thực hiện các thao tác liên quan đến thông tin tài khoản của người dùng trong hệ thống. Chức năng chính bao gồm việc thêm mới, chỉnh sửa, xóa và tìm kiếm người dùng theo nhiều tiêu chí khác nhau.

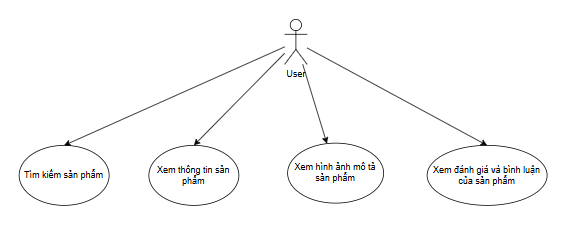
Quản lý sinh viên: Phân hệ này tập trung vào việc quản lý danh sách sinh viên trong hệ thống. Các chức năng chính bao gồm thêm mới, chỉnh sửa, xóa thông tin sinh viên, cũng như theo dõi hồ sơ cá nhân và tình trạng học tập của từng sinh viên. Quản trị viên có thể tìm kiếm sinh viên theo mã số, họ tên hoặc các thông tin khác để dễ dàng quản lý.

Quản lý phòng KTX: Đây là phân hệ dành riêng cho việc quản lý phòng ký túc xá của sinh viên. Chức năng chính bao gồm quản lý danh sách phòng, danh sách sinh viên đang ở trong từng phòng và theo dõi tình trạng đăng ký phòng. Admin có thể duyệt yêu cầu đăng ký phòng, cập nhật thông tin phòng còn trống và theo dõi lịch sử lưu trú của sinh viên.

Quản lý hóa đơn: Hệ thống hỗ trợ quản lý hóa đơn liên quan đến sinh viên, bao gồm các khoản phí ký túc xá, học phí hoặc các khoản thu khác. Quản trị viên có thể tạo mới hóa đơn, theo dõi trạng thái thanh toán, xuất báo cáo và gửi thông báo đến sinh viên khi cần thiết. Hệ thống cũng có thể hỗ trợ chức năng lọc hóa đơn theo thời gian, trạng thái thanh toán và loại phí.

Quản lý tin tức: Đây là phân hệ dành cho việc quản lý các bài viết, thông báo và tin tức nội bộ trong hệ thống. Quản trị viên có thể tạo bài viết mới, chỉnh sửa nội dung, phân loại tin tức theo danh mục và duyệt bài trước khi đăng tải. Chức năng này giúp cung cấp thông tin chính xác, kịp thời đến người dùng, đặc biệt là sinh viên và nhân viên trong hệ thống.

1. ***Phân rã usercase “Xem thông tin bài viêt”***



Hình 2.31. Phân rã usecase “Xem thông tin bài viết”

**Các bước phân rã**

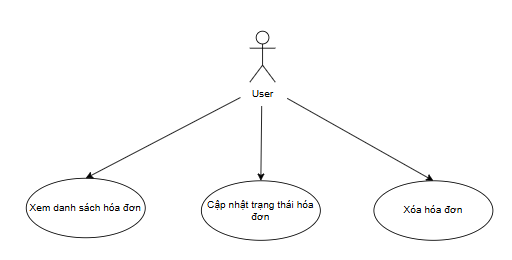
Tìm kiếm bài viết: Người dùng có thể tìm kiếm bài viết thông qua tên sản phẩm, thông tin liên quan hoặc từ khóa có trong nội dung bài viết. Hệ thống sẽ hiển thị danh sách các bài viết phù hợp với từ khóa tìm kiếm, giúp người dùng dễ dàng tiếp cận thông tin họ quan tâm. Kết quả tìm kiếm có thể được sắp xếp theo độ liên quan, thời gian đăng tải hoặc mức độ phổ biến để hỗ trợ trải nghiệm người dùng tốt hơn.

Xem thông tin bài viết: Sau khi tìm kiếm, người dùng có thể truy cập vào trang chi tiết của bài viết để xem đầy đủ thông tin liên quan. Nội dung hiển thị bao gồm tên sản phẩm, giá cả, loại sản phẩm, ngày sản xuất, ngày hết hạn (nếu có), mô tả chi tiết về sản phẩm và hướng dẫn sử dụng. Hệ thống đảm bảo cung cấp đầy đủ thông tin giúp người dùng hiểu rõ về sản phẩm trước khi đưa ra quyết định mua hàng hoặc sử dụng dịch vụ.

Xem hình ảnh mô tả bài viết: Bên cạnh thông tin văn bản, người dùng cũng có thể xem các hình ảnh mô tả sản phẩm để có cái nhìn trực quan hơn. Đối với sách, hệ thống sẽ hiển thị ảnh bìa và các hình ảnh minh họa liên quan nếu có. Ngoài ra, hệ thống có thể hỗ trợ tính năng xem trước một số trang nội dung để người dùng có thể đánh giá trước khi quyết định mua sách. Điều này giúp người dùng có trải nghiệm chân thực hơn và tăng mức độ tin tưởng đối với sản phẩm.

Xem bình luận của bài viết: Người dùng có thể xem các đánh giá, nhận xét từ những người đã mua hoặc sử dụng sản phẩm trước đó.

1. ***Phân rã usercase “Phản hồi hóa đơn”***



Hình 2.32. Phân rã usecase “Phản hồi hóa đơn”

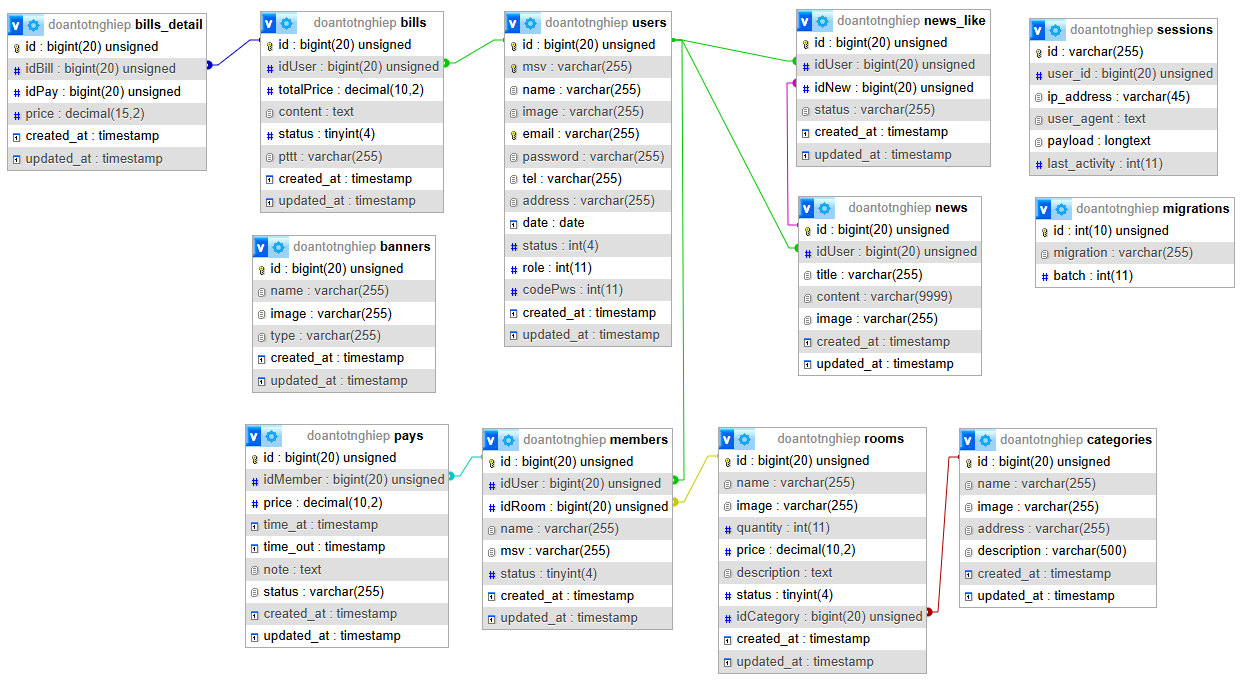
**Các bước phân rã**

**Mô tả:**

* Xem danh sách hóa đơn: Quản trị có thể xem danh sách các hóa đơn mà họ nhận được từ sinh viên.
* Cập nhật trạng thái hóa đơn: Quản trị có khả năng cập nhật trạng thái
* Gửi thông báo cho sinh viên: Quản trị có thể gửi thông báo cho sinh viên để thông báo về trạng thái hóa đơn hoặc các thông tin khác liên quan đến hóa đươn.

**Mục đích:**

* Xem danh sách hóa đơn: Cung cấp cho quản trị cái nhìn tổng quan về các hóa đơn hiện có để quản lý và xử lý chúng một cách hiệu quả.
* Cập nhật trạng thái hóa đơn: Cho phép người bán hàng thông báo về trạng thái và tiến độ của các hóa đơn.
  1. **Thiết kế cơ sở dữ liệu**
     1. ***Biểu đồ lớp***



Hình 2.33. Biểu đồ lớp

* + 1. ***Bảng dữ liệu User***

Bảng 2.8. Bảng dữ liệu User

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| Id | bigint(20) unsigned | Khóa chính |
| Msv | varchar(255) | Mã sinh viên |
| Name | varchar(255) | Tên sinh viên |
| Image | varchar(255) | Hình ảnh |
| Email | varchar(255) | Email |
| Password | varchar(255) | Mật khẩu |
| Tel | varchar(255) | Số điện thoại |
| address | varchar(255) | Địa chỉ |
| date | date | Ngày sinh |
| status | int(4) | Trạng thái |
| role | int(11) | Phân quyền |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

* + 1. ***Bảng dữ liệu Categories***

Bảng 2.9. Bảng dữ liệu Categories

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| name | varchar(255) | Tên |
| image | varchar(255) | Hình ảnh |
| address | varchar(255) | Địa chỉ |
| description | varchar(500) | Mô tả |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

* + 1. ***Bảng dữ liệu Rooms***

Bảng 2.10. Bảng dữ liệu Rooms

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| name | varchar(255) | Tên |
| image | varchar(255) | Hình ảnh |
| quantity | decimal(10,2) | Số lượng |
| price | decimal(10,2) | Giá tiền |
| description | varchar(255) | Địa chỉ |
| status | int | Trạng thái |
| idCategory | bigint(20) unsigned | FK - Category |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

* + 1. ***Bảng dữ liệu Members***

Bảng 2.11. Bảng dữ liệu Members

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| idUser | bigint(20) unsigned | FK - User |
| idRoom | bigint(20) unsigned | FK - Room |
| name | varchar(255) | Tên |
| msv | varchar(255) | Mã sinh viên |
| status | int | Trạng thái |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

* + 1. ***Bảng dữ liệu Pays***

Bảng 2.12. Bảng dữ liệu Pays

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| idMember | bigint(20) unsigned | FK - Member |
| price | decimal(10,2) | Giá tiền |
| time\_at | date | Thời gian tạo |
| time\_out | date | Thời gian hết hạn |
| note | varchar(255) | Ghi chú |
| status | varchar(255) | Trạng thái |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

* + 1. ***Bảng dữ liệu Bill***

Bảng 2.13. Bảng dữ liệu Bill

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| idUser | bigint(20) unsigned | FK - User |
| totalPrice | decimal(10,2) | Tổng tiền |
| content | text | Nội dung |
| status | int | Trạng thái |
| pttt | varchar(255) | Phương thức thanh toán |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

* + 1. ***Bảng dữ liệu BillDetail***

Bảng 2.14. Bảng dữ liệu BillDetail

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| idBill | bigint(20) unsigned | FK - Bill |
| idPay | bigint(20) unsigned | FK - Pay |
| price | decimal(10,2) | Giá tiền |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

* + 1. ***Bảng dữ liệu News***

Bảng 2.15. Bảng dữ liệu News

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| idUser | bigint(20) unsigned | FK - User |
| title | varchar(255) | Tiêu đề |
| content | varchar(9999) | Nội dung |
| image | varchar(255) | Hình ảnh |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

***2.6.10. Bảng dữ liệu News\_like***

Bảng 2.16. Bảng dữ liệu News\_like

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| idUser | bigint(20) unsigned | FK - User |
| idNew | bigint(20) unsigned | FK - New |
| status | varchar(255) | Trạng thái |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

***2.6.11. Bảng dữ liệu Banners***

Bảng 2.17. Bảng dữ liệu Banners

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| name | varchar(255) | Tên |
| image | varchar(255) | Hình ảnh |
| type | varchar(255) | Kiểu loại |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

***2.6.12. Bảng dữ liệu Bank***

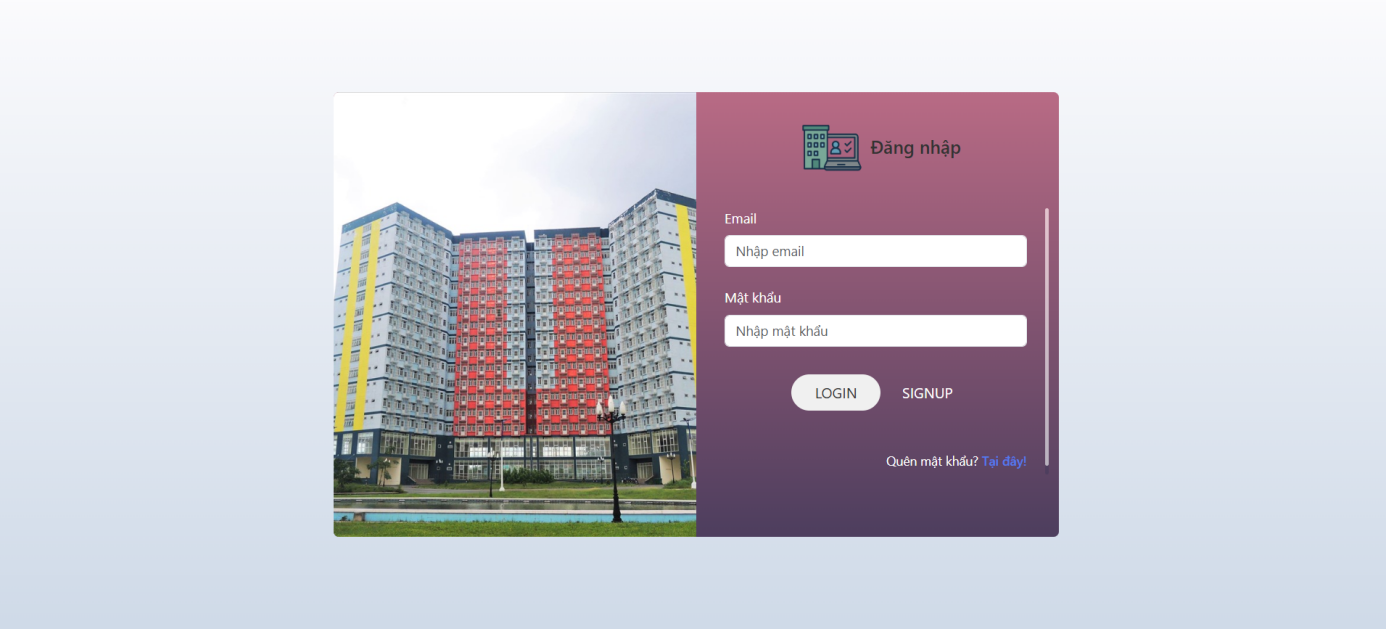
Bảng 2.18. Bảng dữ liệu Bank

|  |  |  |
| --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Mô tả |
| id | bigint(20) unsigned | Khóa chính |
| bank | varchar(255) | Tên ngân hàng |
| stk | varchar(255) | Số tài khoản |
| ctk | varchar(255) | Chủ tài khoản |
| taikhoan | varchar(255) | Tài khoản |
| matkhau | varchar(255) | Mật khẩu |
| created\_at | timestamp | Thời gian tạo |
| updated\_at | timestamp | Thời gian cập nhật |

**CHƯƠNG 3: XÂY DỰNG CHƯƠNG TRÌNH**

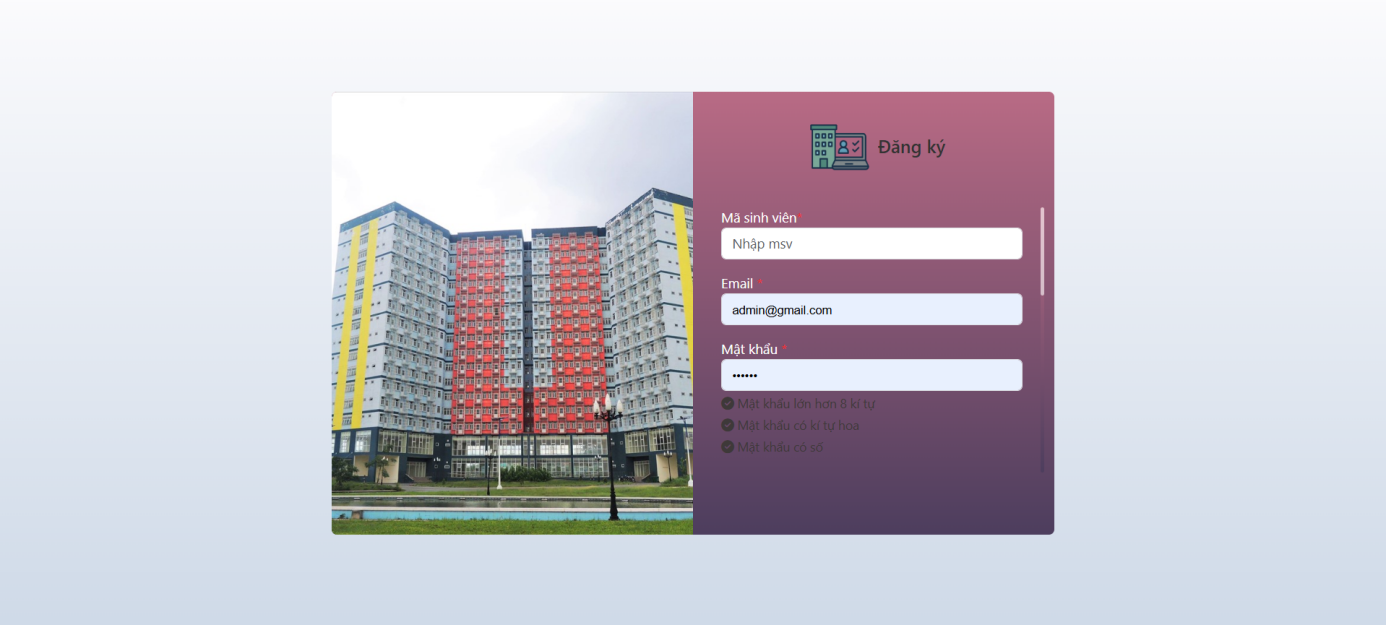
## 3.1. Giao diện người dùng

### *3.1.1. Trang đăng nhập*



Hình 3.1. Trang đăng nhập

### *3.1.2. Trang đăng ký*



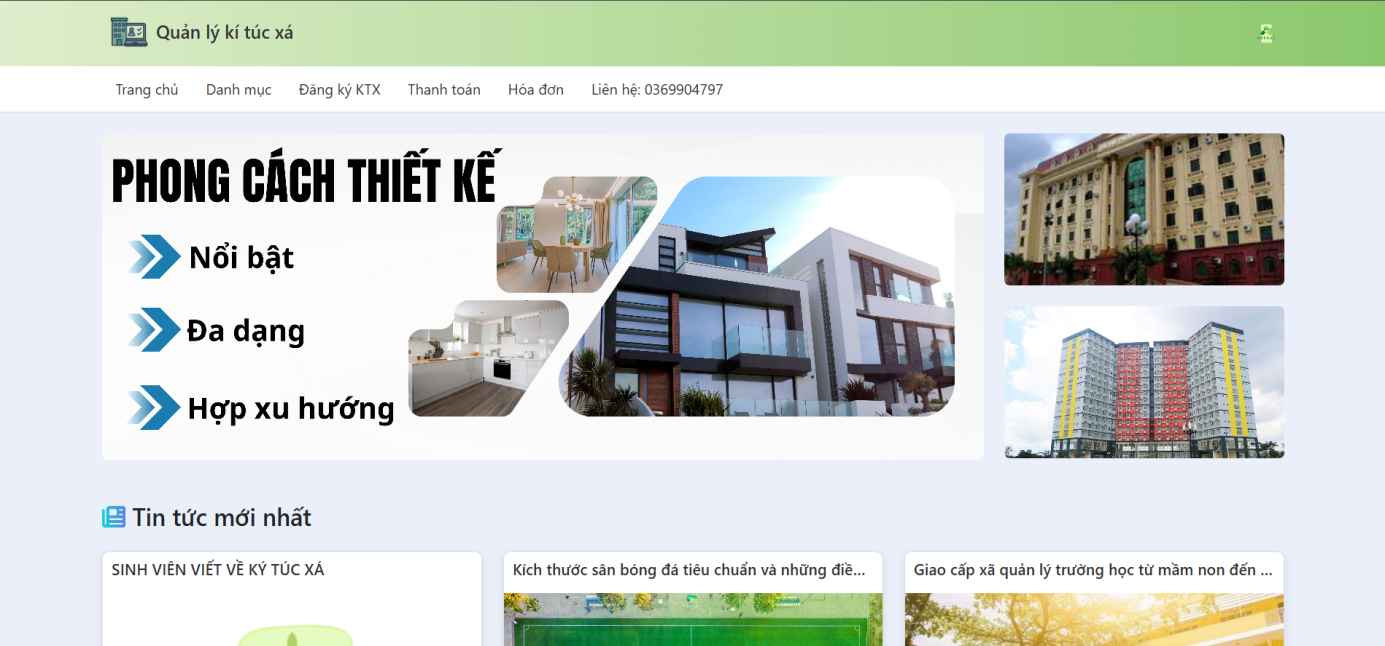
Hình 3.2. Trang đăng ký

### *3.1.3. Trang quên mật khẩu*



Hình 3.3. Trang quên mật khẩu

### *3.1.4. Trang chủ - Người dùng*



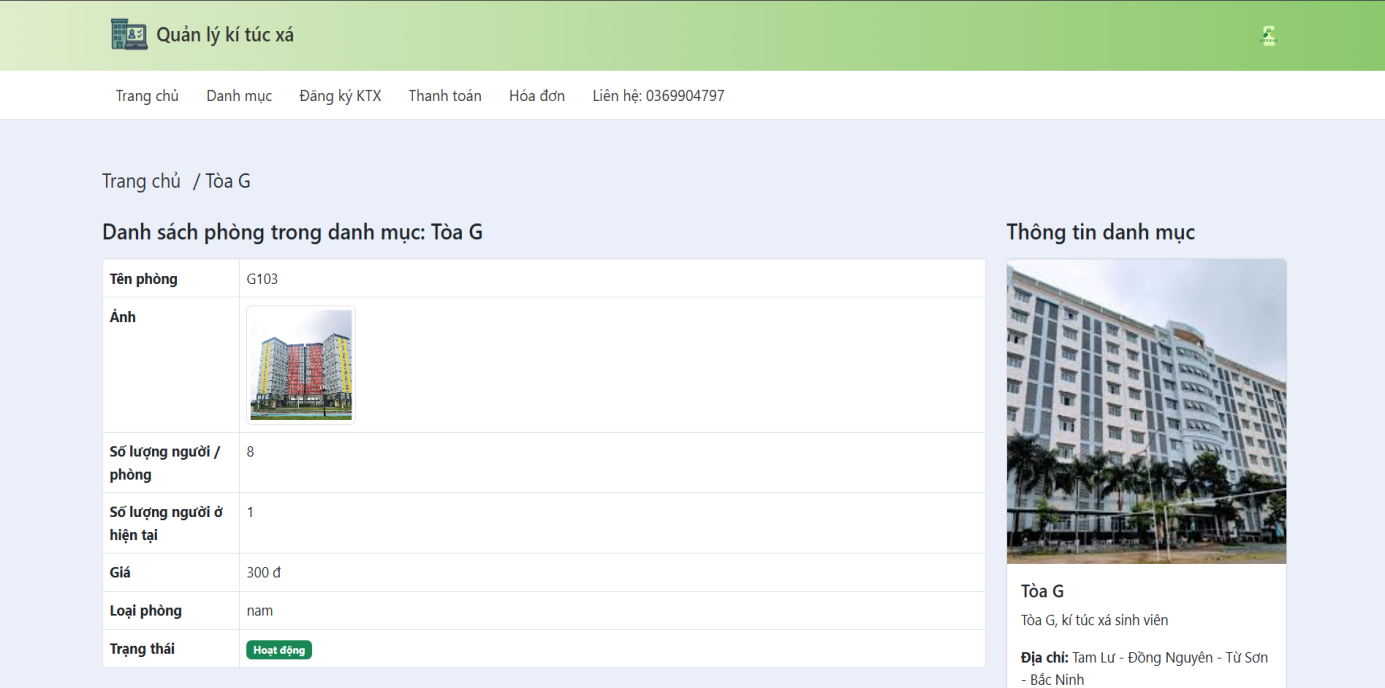
Hình 3.4. Trang chủ

### *3.1.5. Trang thông tin bài viết*



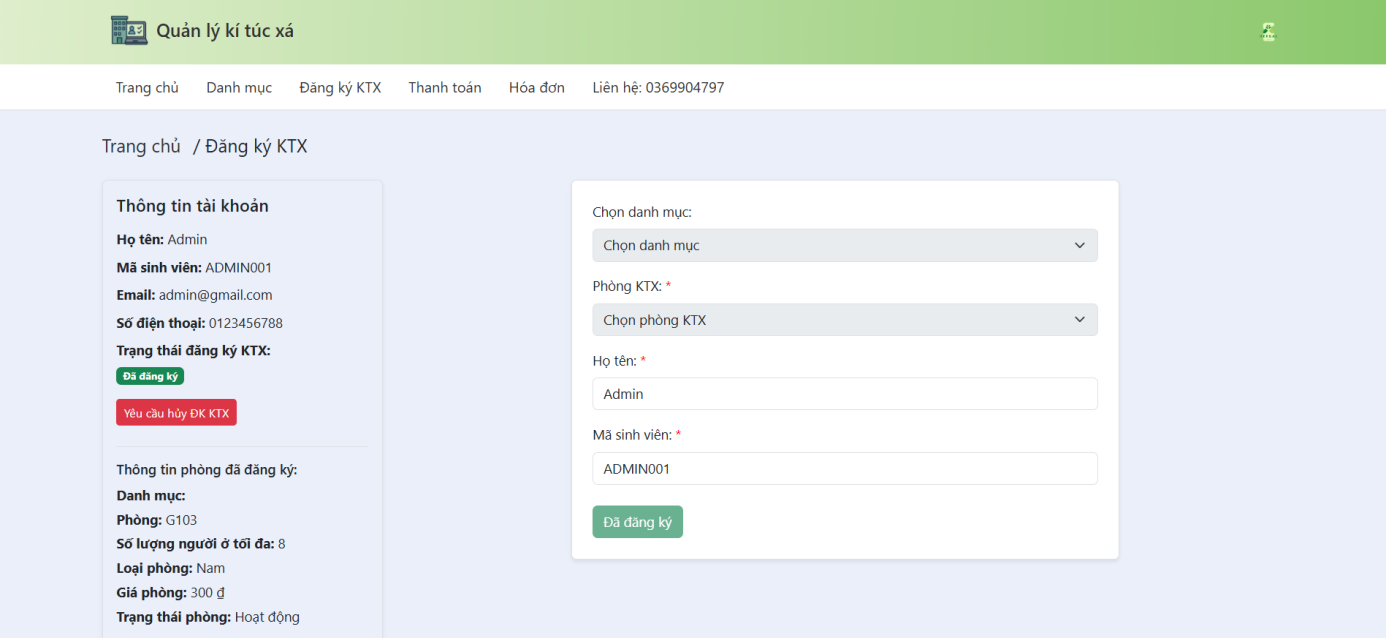
Hình 3.5. Trang thông tin bài viết

### *3.1.6. Trang danh mục phòng*



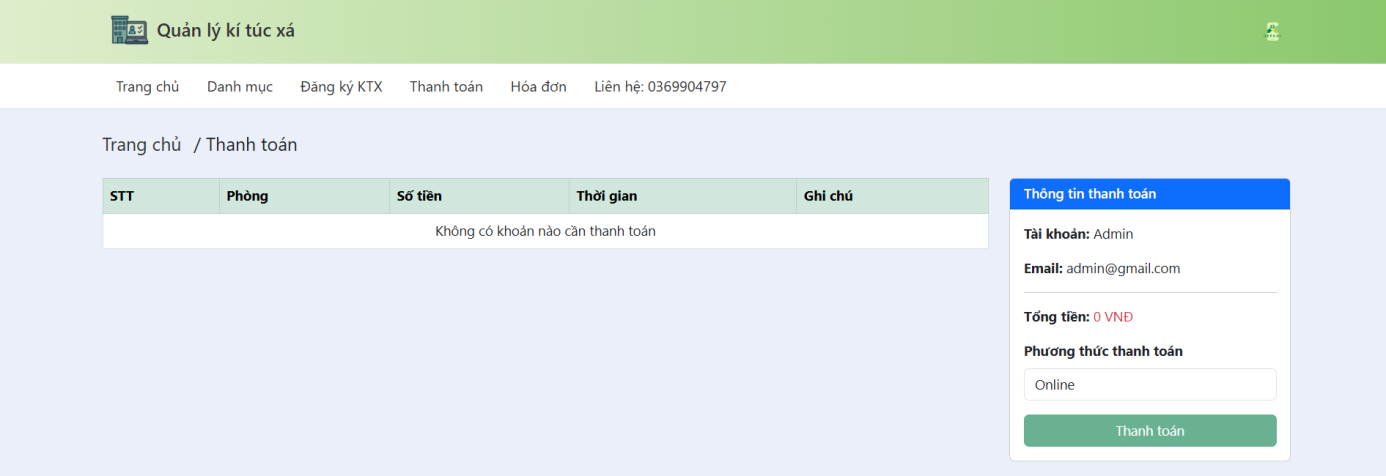
Hình 3.6. Trang thông tin danh mục phòng

### *3.1.7. Trang đăng ký phòng KTX*



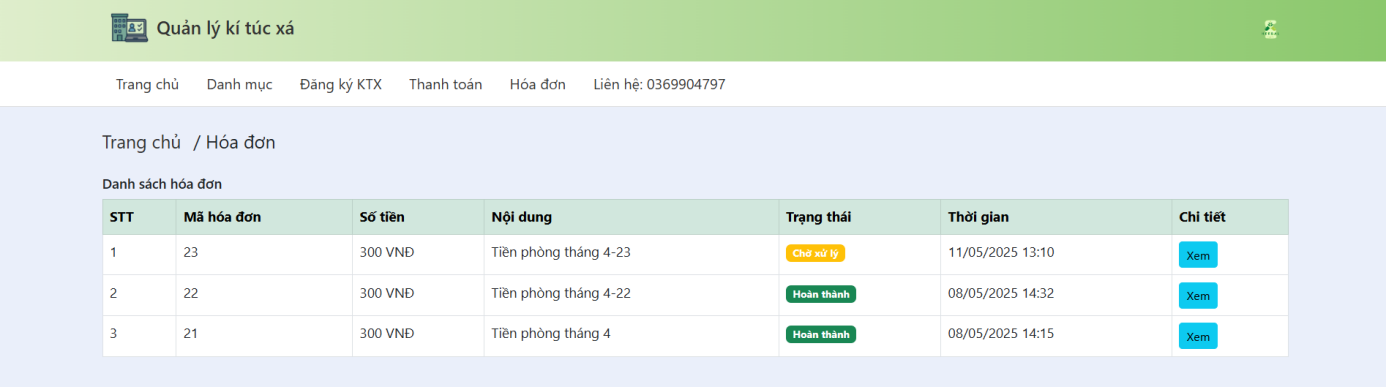
Hình 3.7. Trang đăng ký phòng KTX

### *3.1.8. Trang thanh toán*

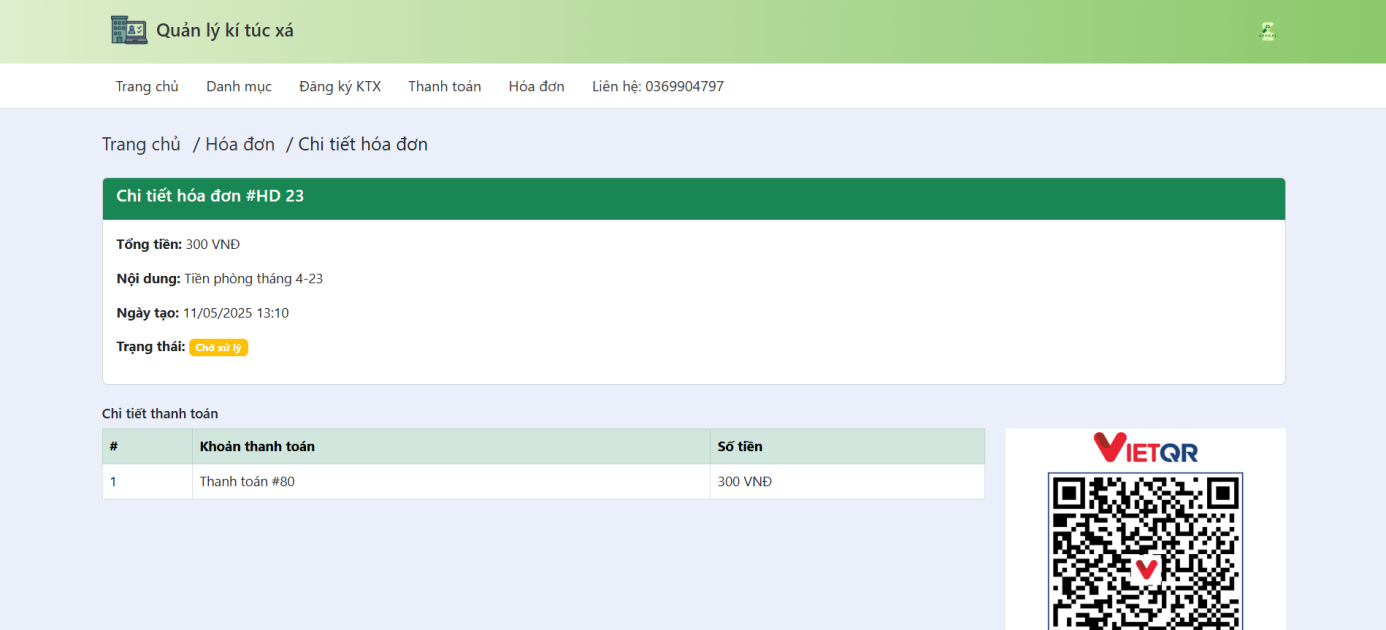


Hình 3.8. Trang thanh toán

### *3.1.9. Trang hoá đơn*

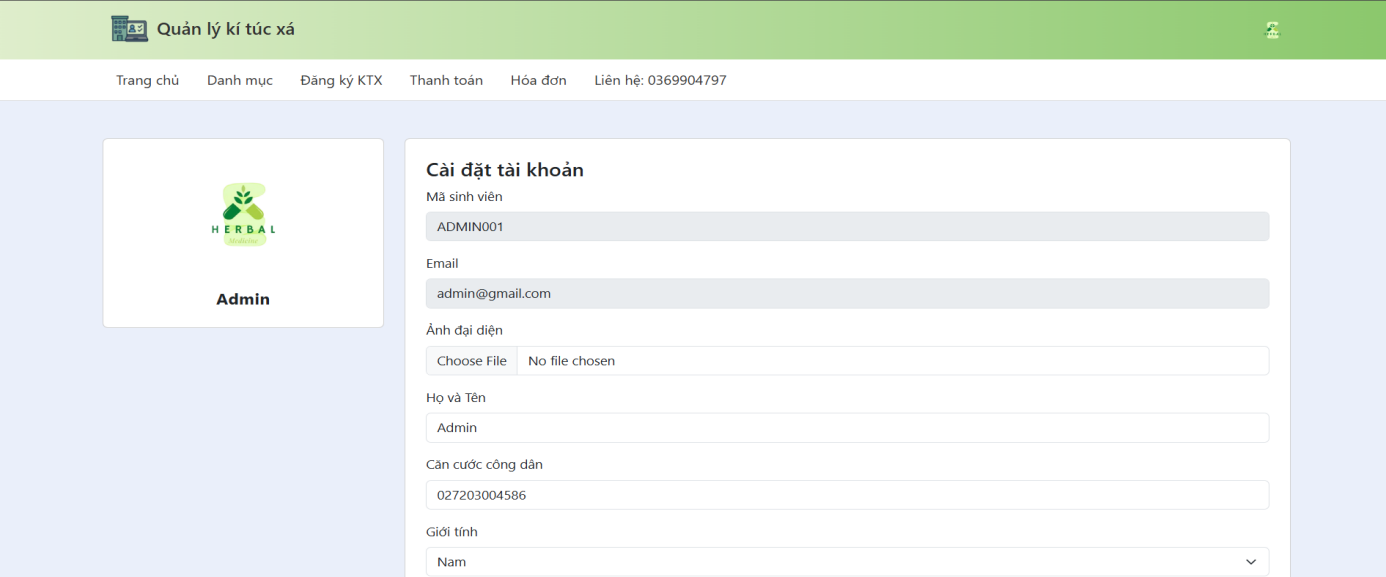
Hình 3.9. Trang hóa đơn

### *3.1.10. Trang chi tiết hoá đơn*



Hình 3.10. Trang chi tiết hóa đơn

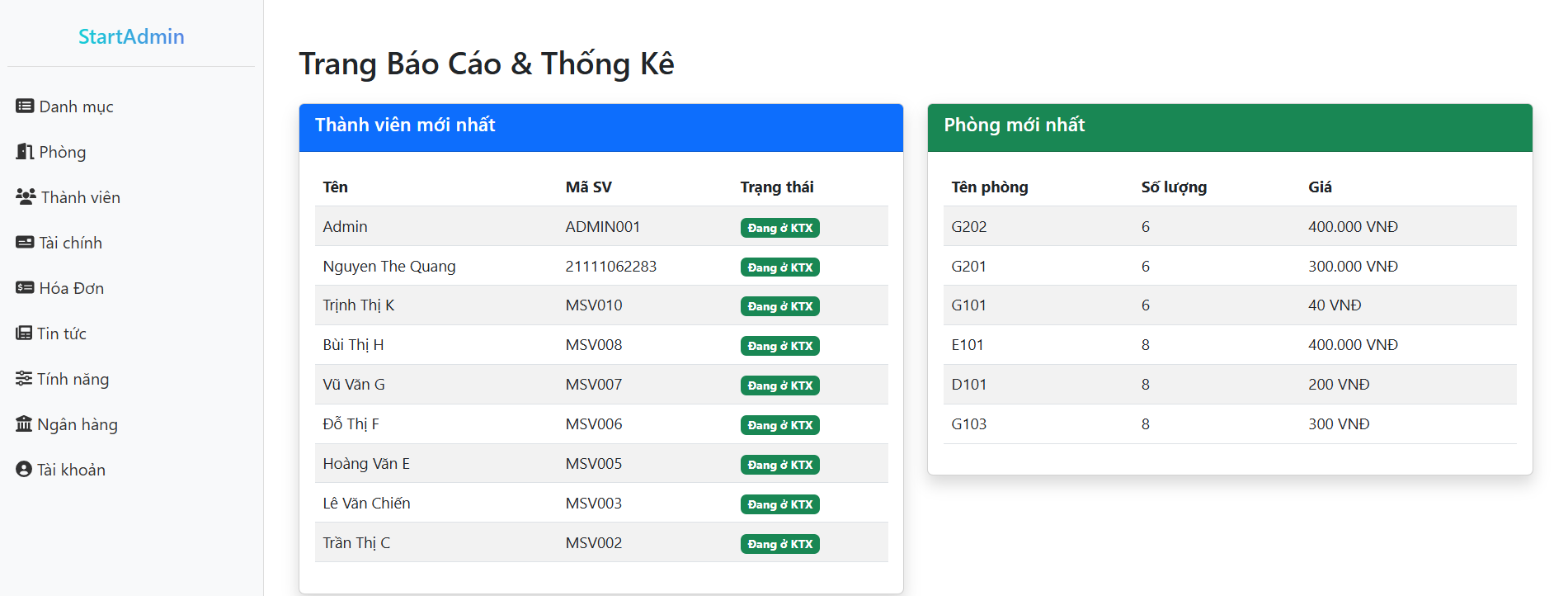
### *3.1.11. Trang quản lý thông tin người dùng*



Hình 3.11. Trang quản lý thông tin người dung

## 3.2. Giao diện quản trị

### *3.2.1. Trang chủ - Quản trị*

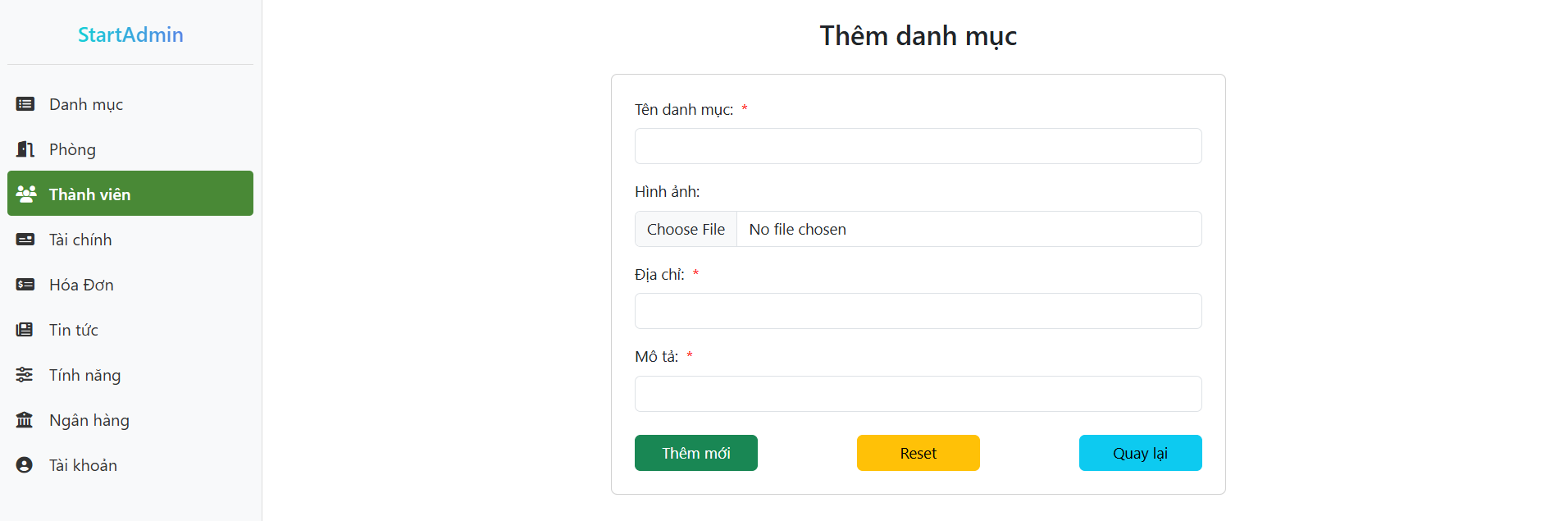


Hình 3.12. Trang chủ - quản trị

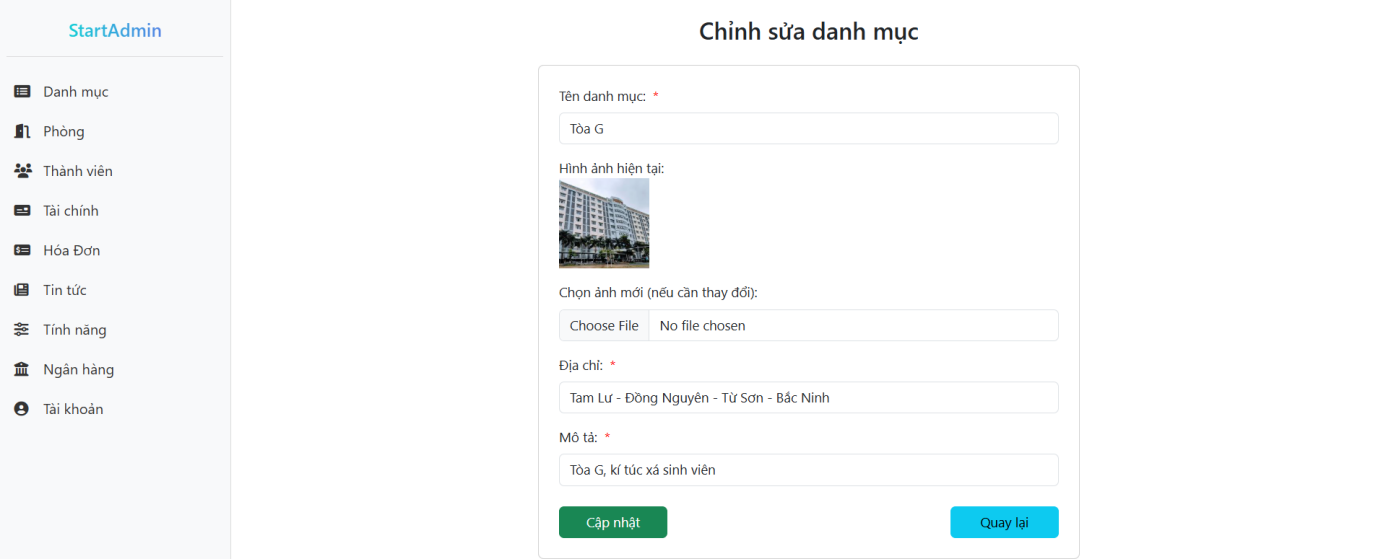
### *3.2.2. Trang quản lý danh mục*



Hình 3.13. Trang quản lý danh mục

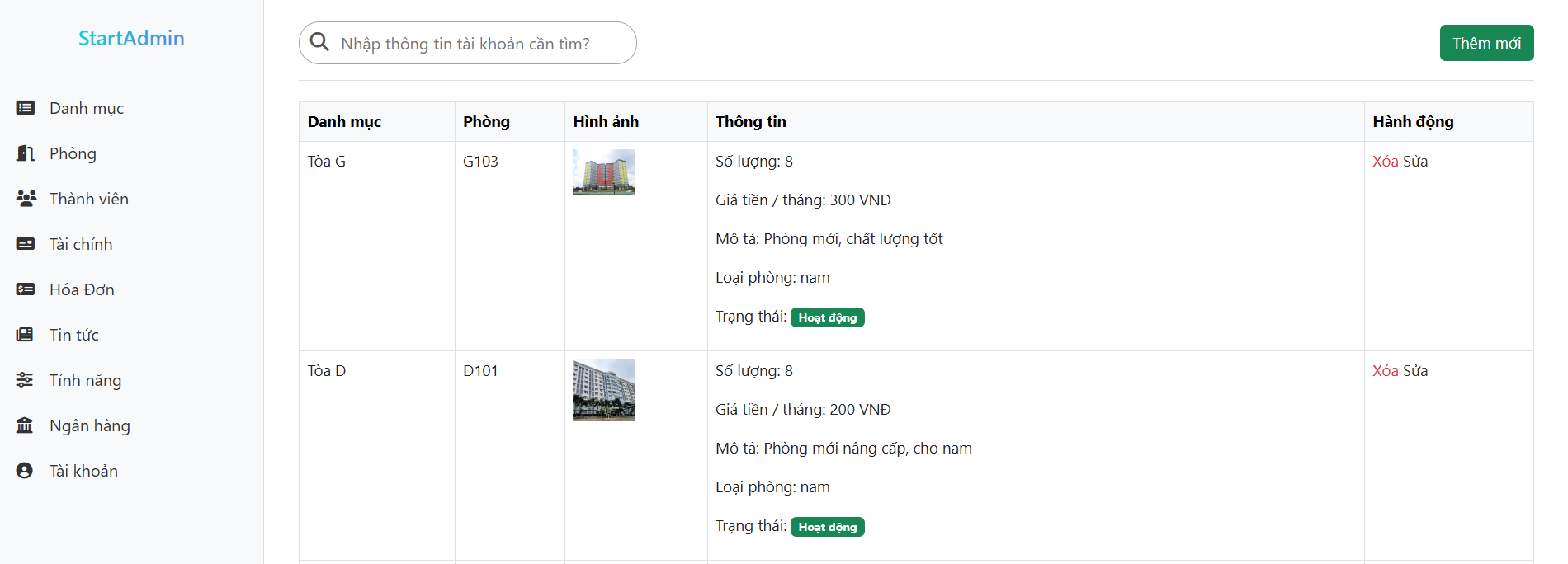


Hình 3.14. Trang thêm danh mục

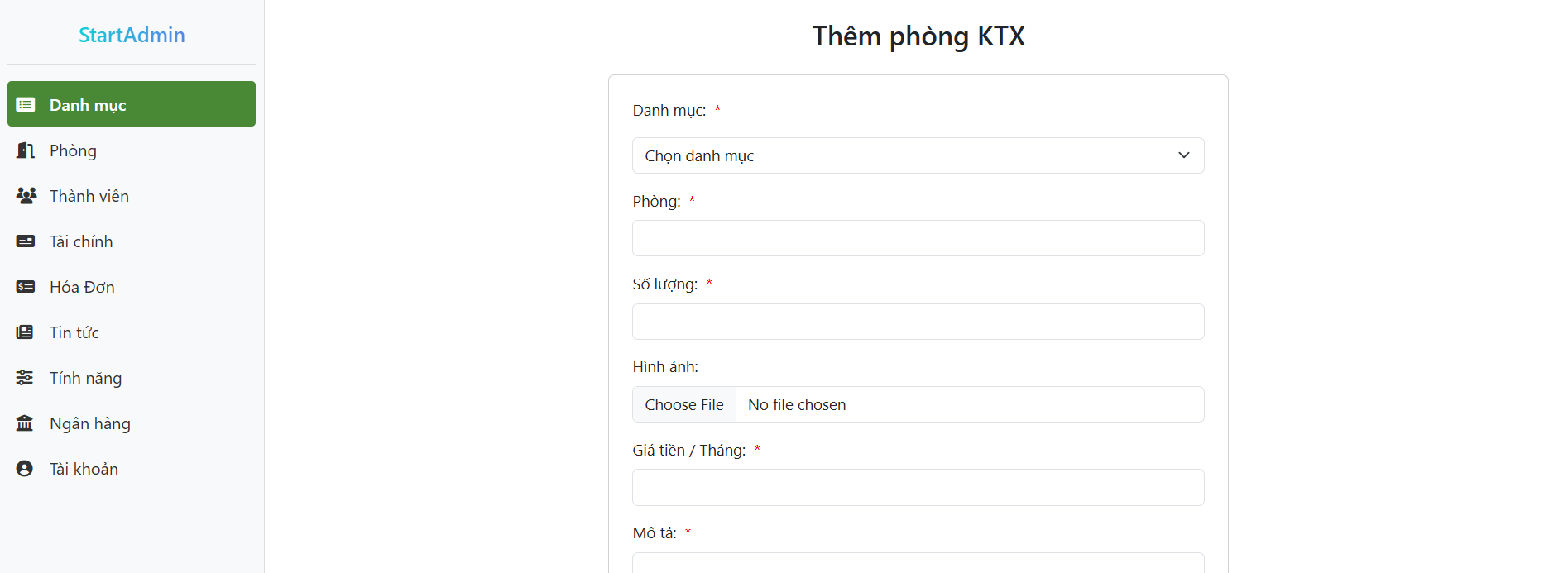


Hình 3.15. Trang sửa danh mục

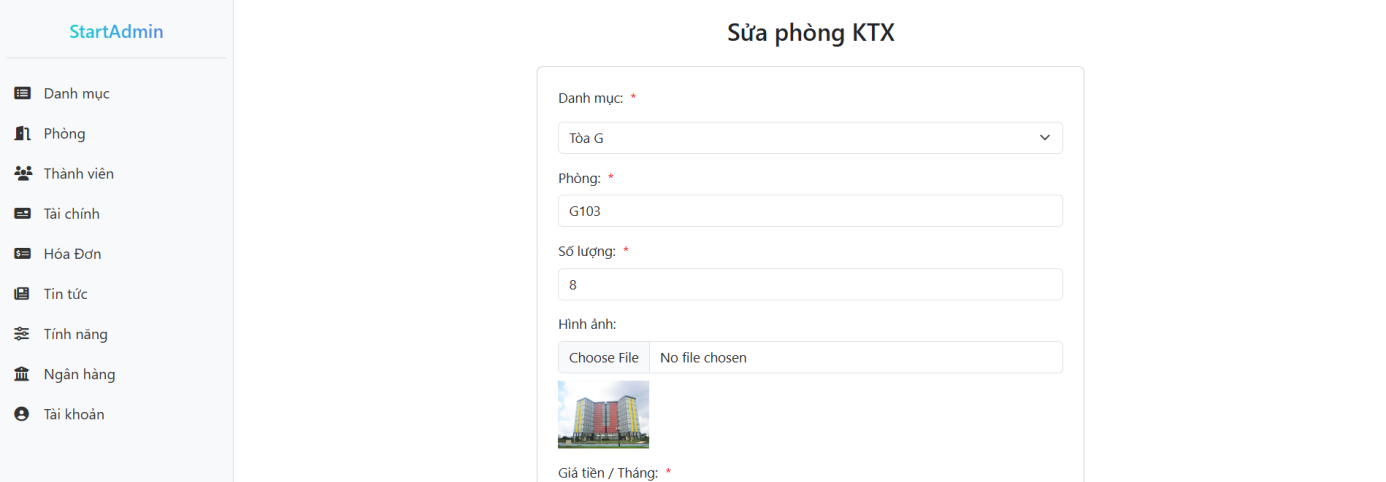
### *3.2.3. Trang quản lý phòng ở*



Hình 3.16. Trang quản lý ở KTX

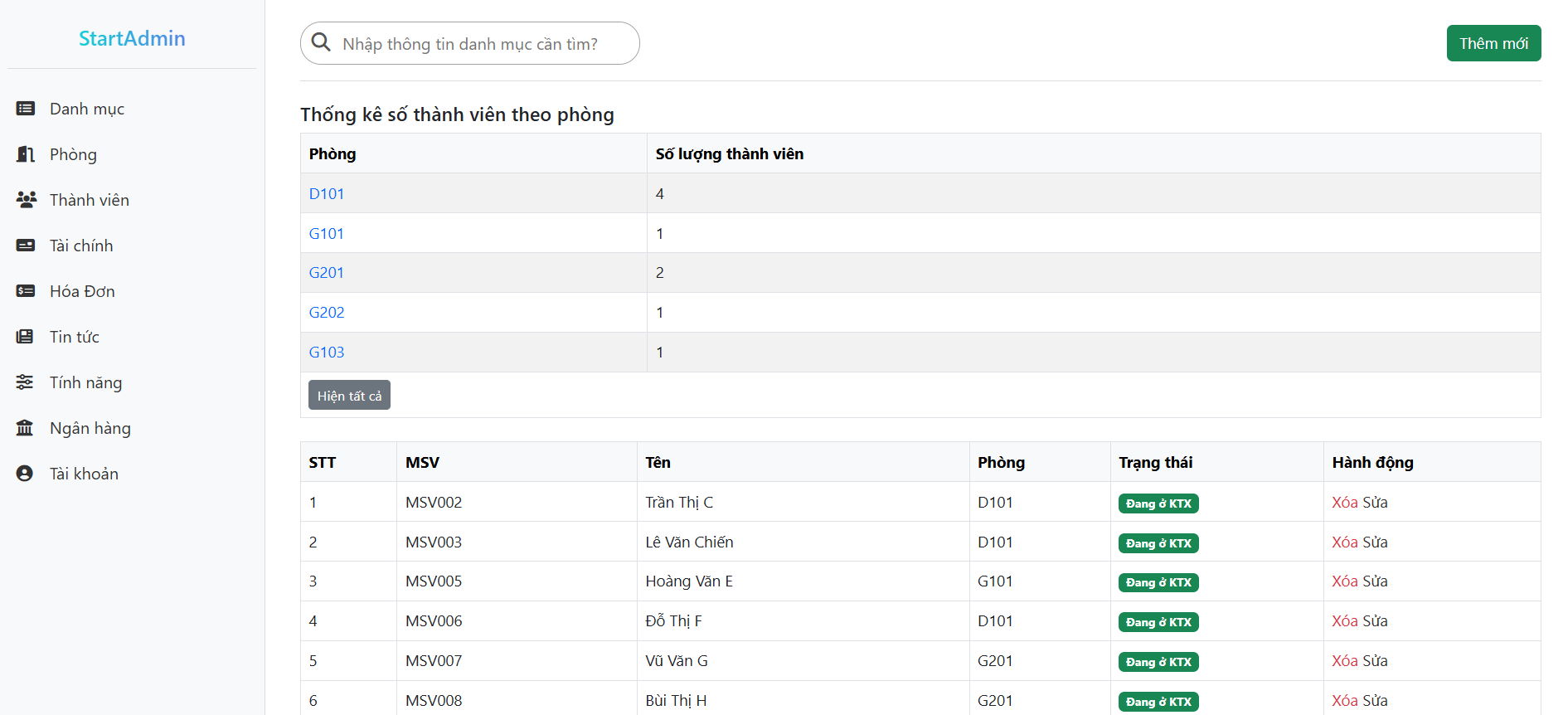


Hình 3.17. Trang thêm phòng KTX

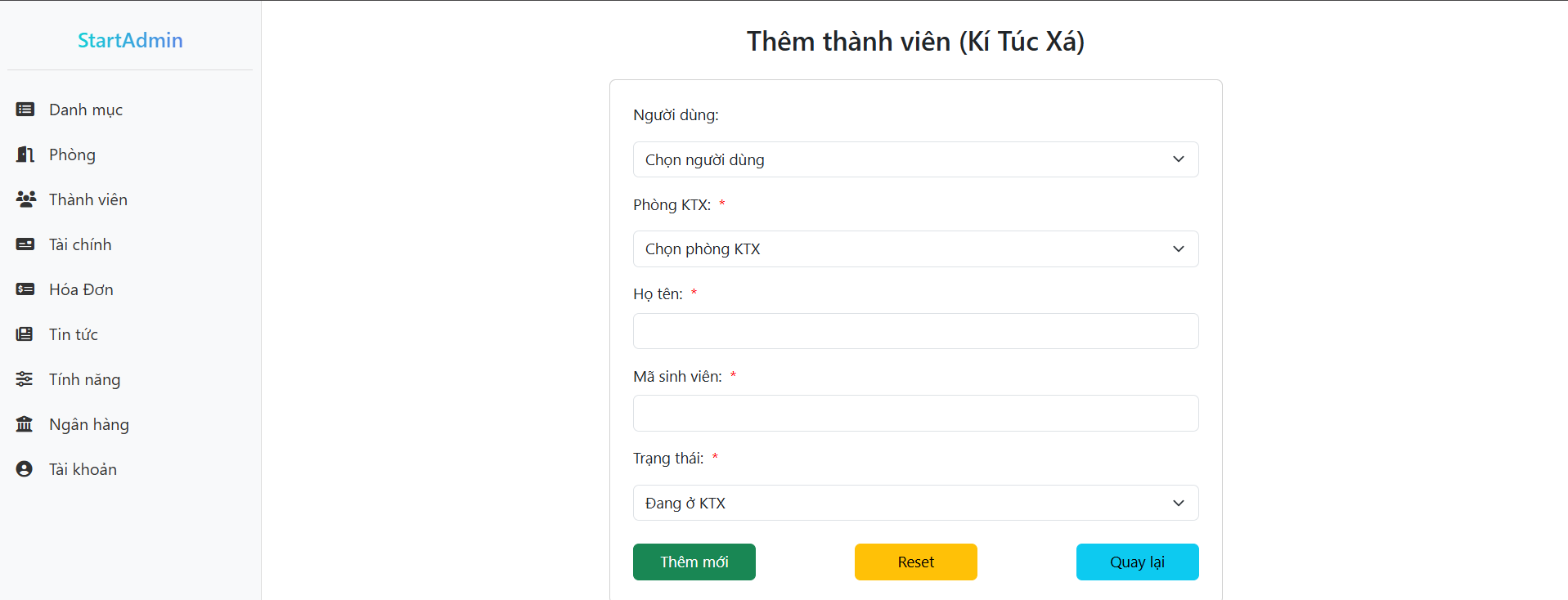
****

Hình 3.18. Trang sửa phòng KTX

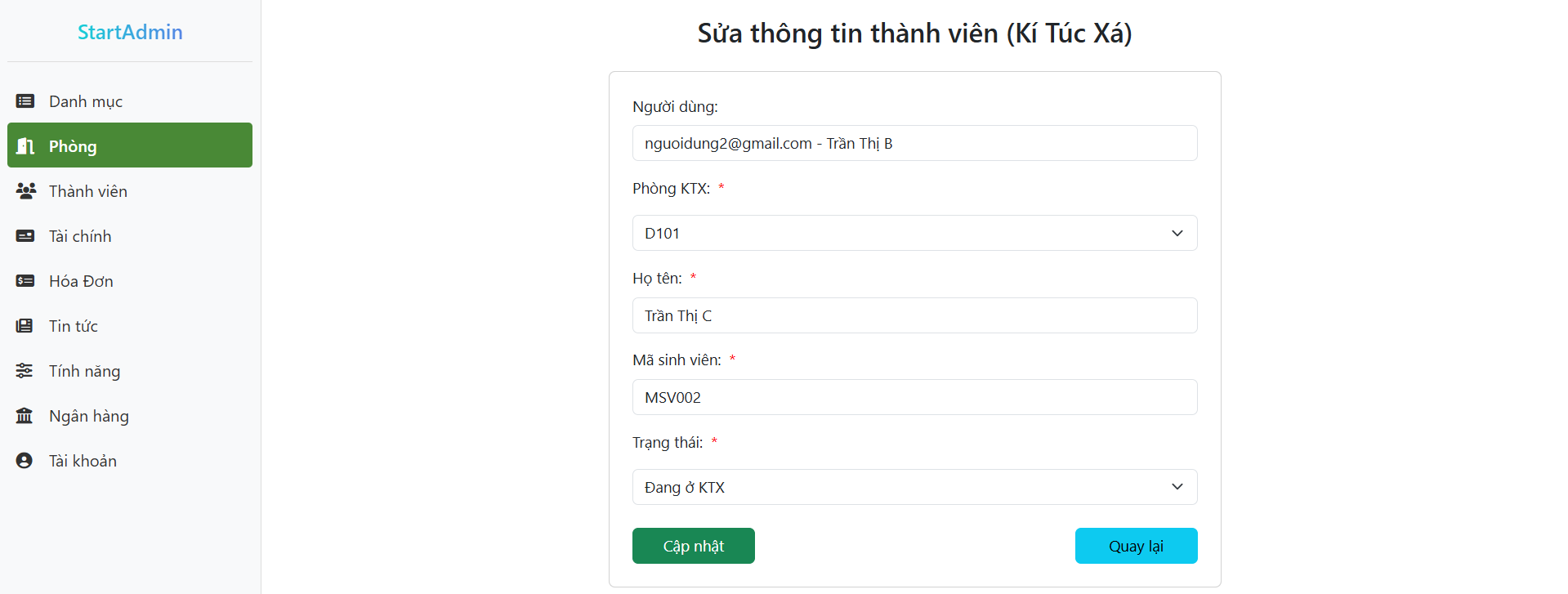
### *3.2.4. Trang quản lý thành viên*



Hình 3.19. Trang quản lý thành viên KTX

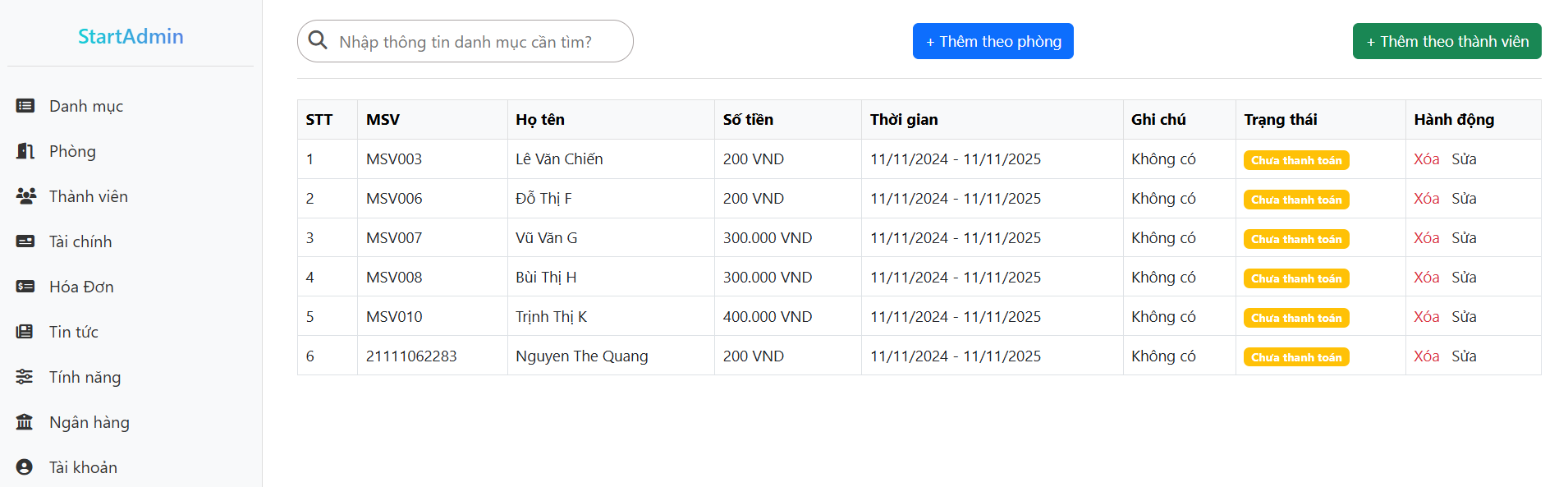


Hình 3.20. Trang thêm thành viên KTX

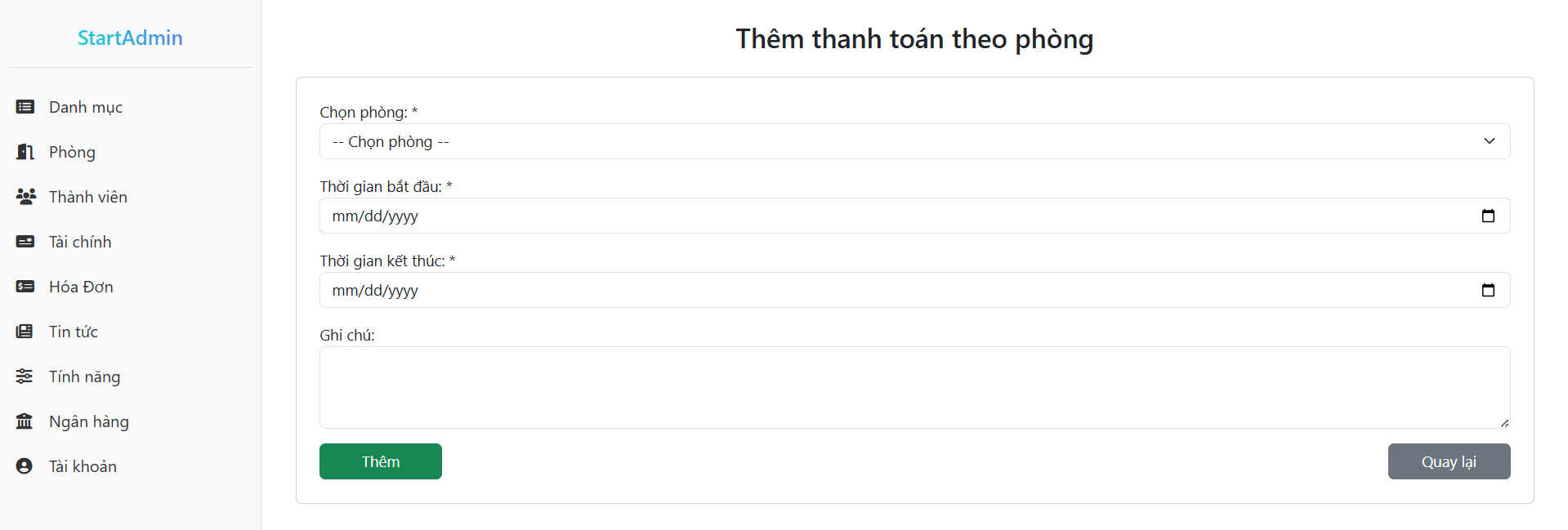
**

Hình 3.21. Trang sửa thông tin thành viên KTX

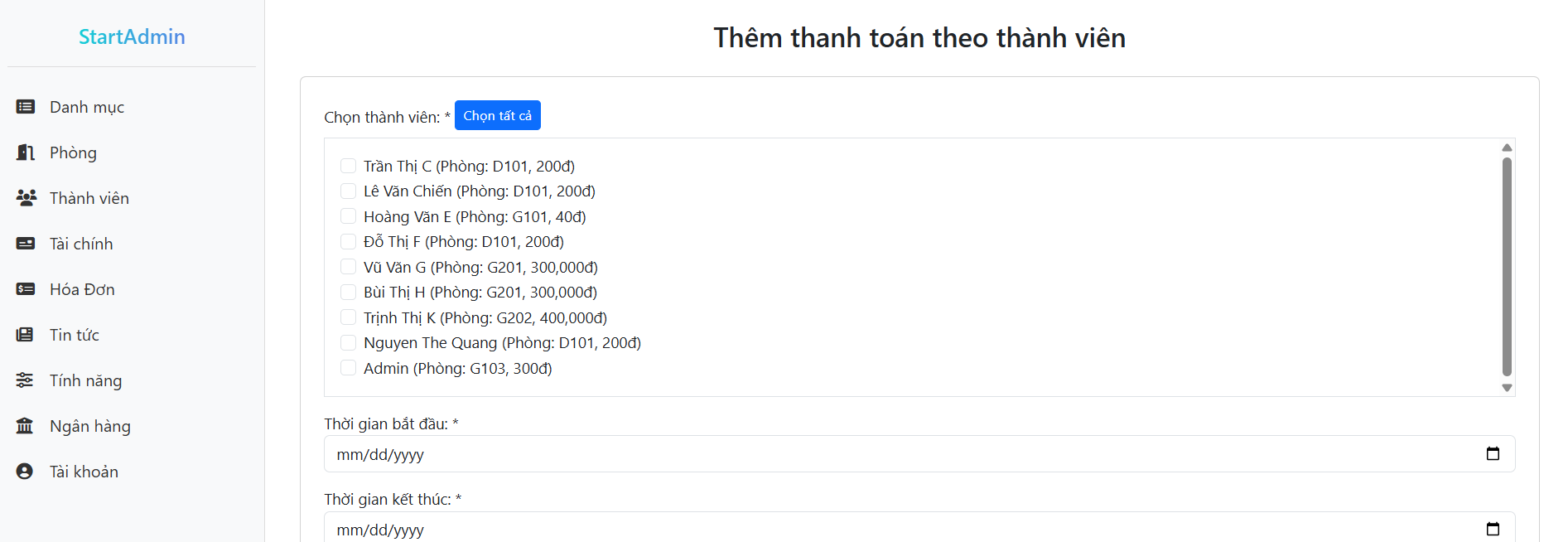
### *3.2.5. Trang quản lý tài chính*



Hình 3.22. Trang quản lý tài chính

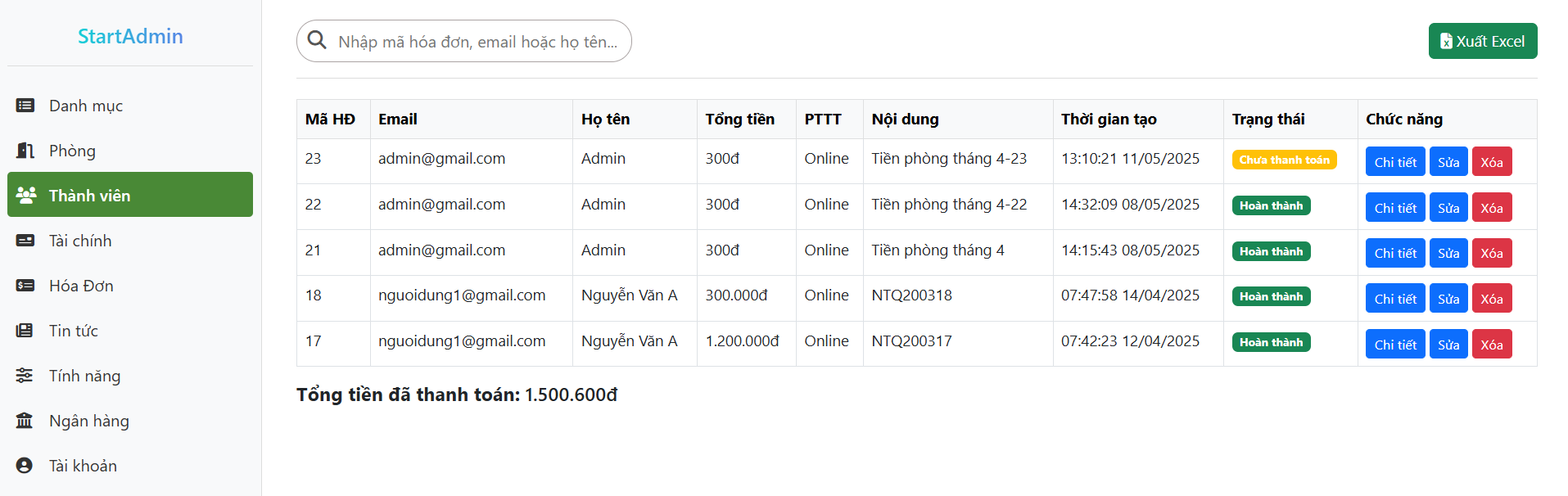


Hình 3.23. Trang thêm thanh toán theo phòng

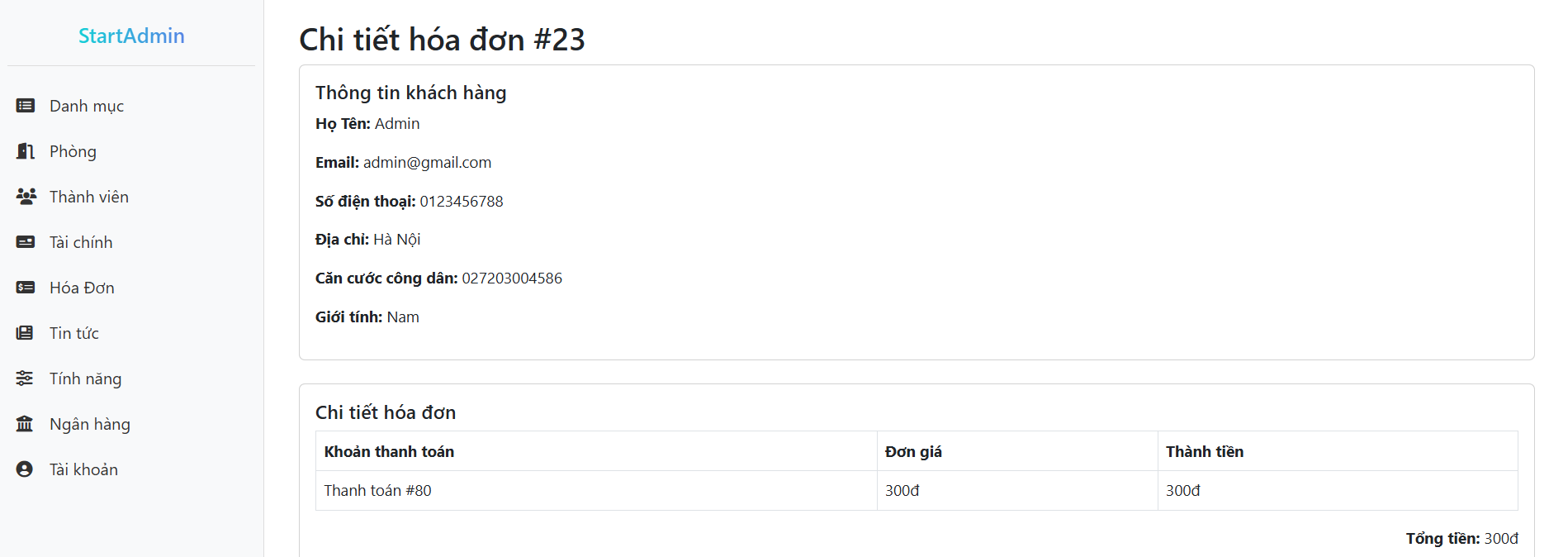


Hình 3.24. Trang thêm thanh toán theo thành viên

### *3.2.6. Trang quản lý hoá đơn*



Hình 3.25. Trang quản lý hóa đơn

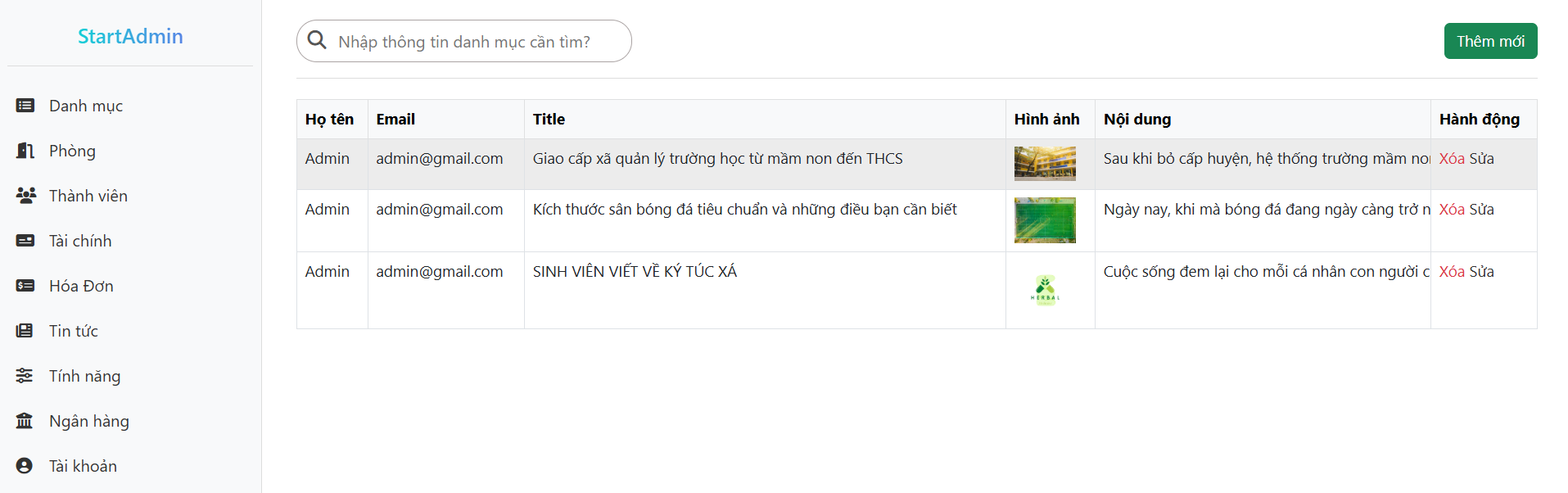


Hình 3.26. Trang chi tiết hóa đơn

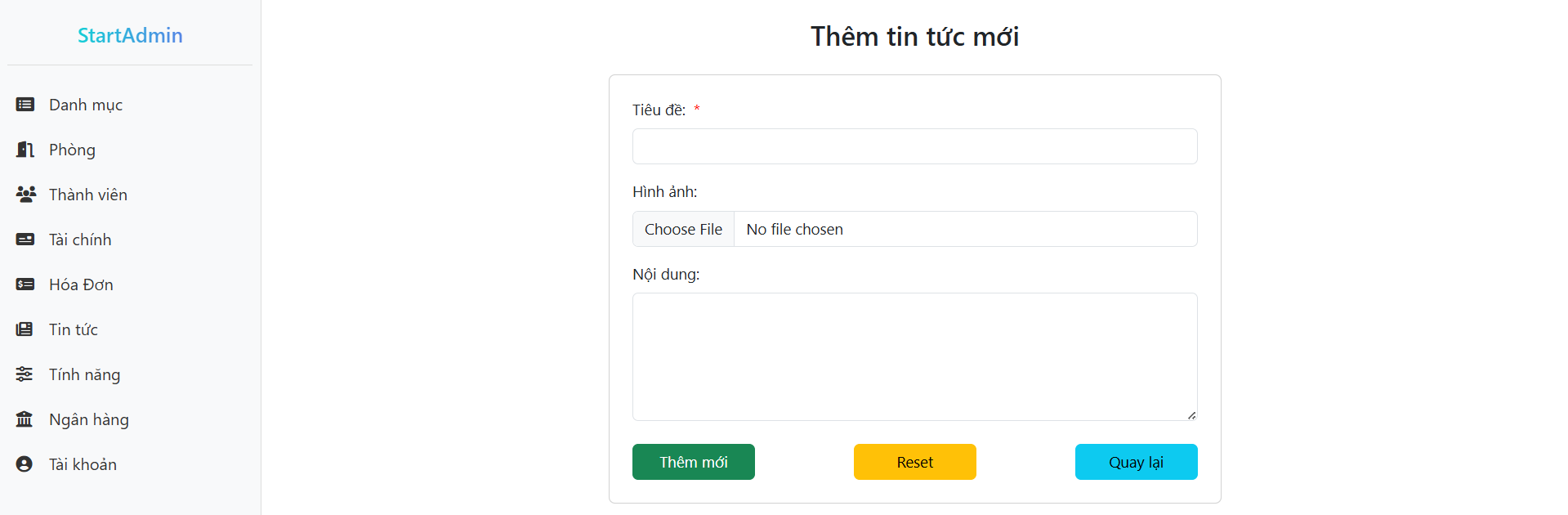


Hình 3.27. Trang sửa trạng thái hóa đơn

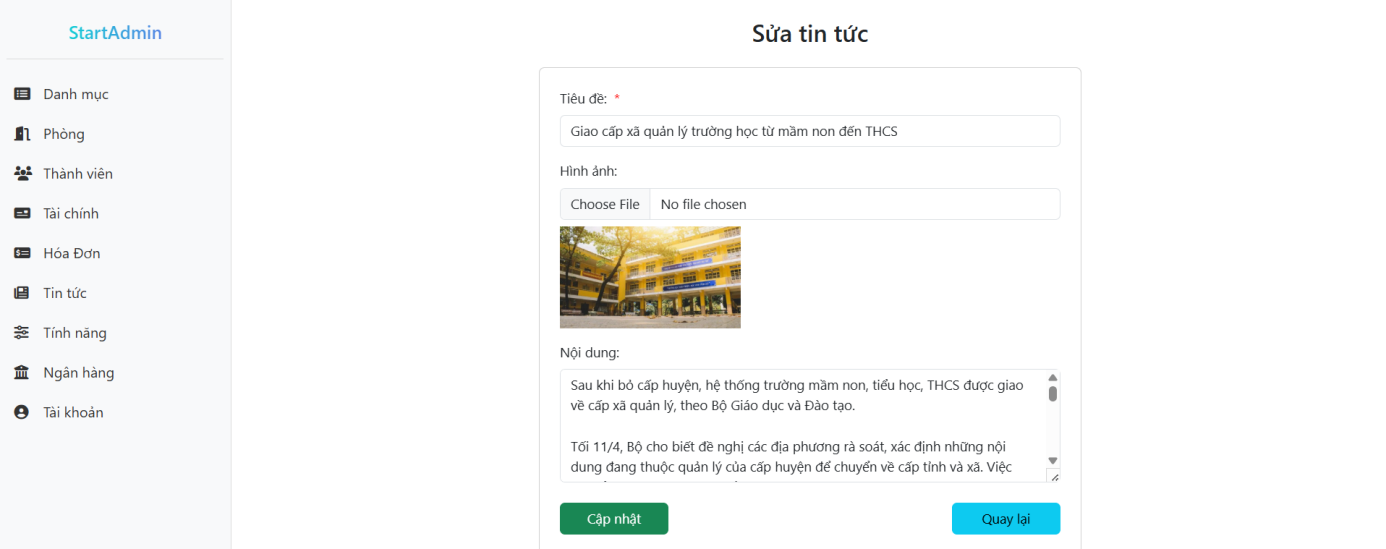
### *3.2.7. Trang quản lý tin tức*



Hình 3.28. Trang quản lý tin tức

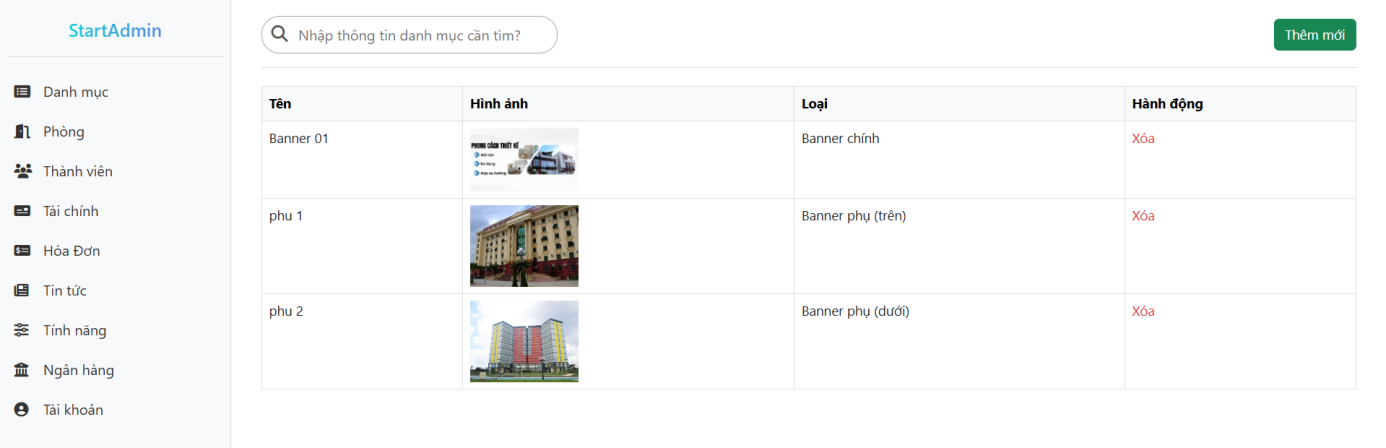


Hình 3.29. Trang thêm tin tức



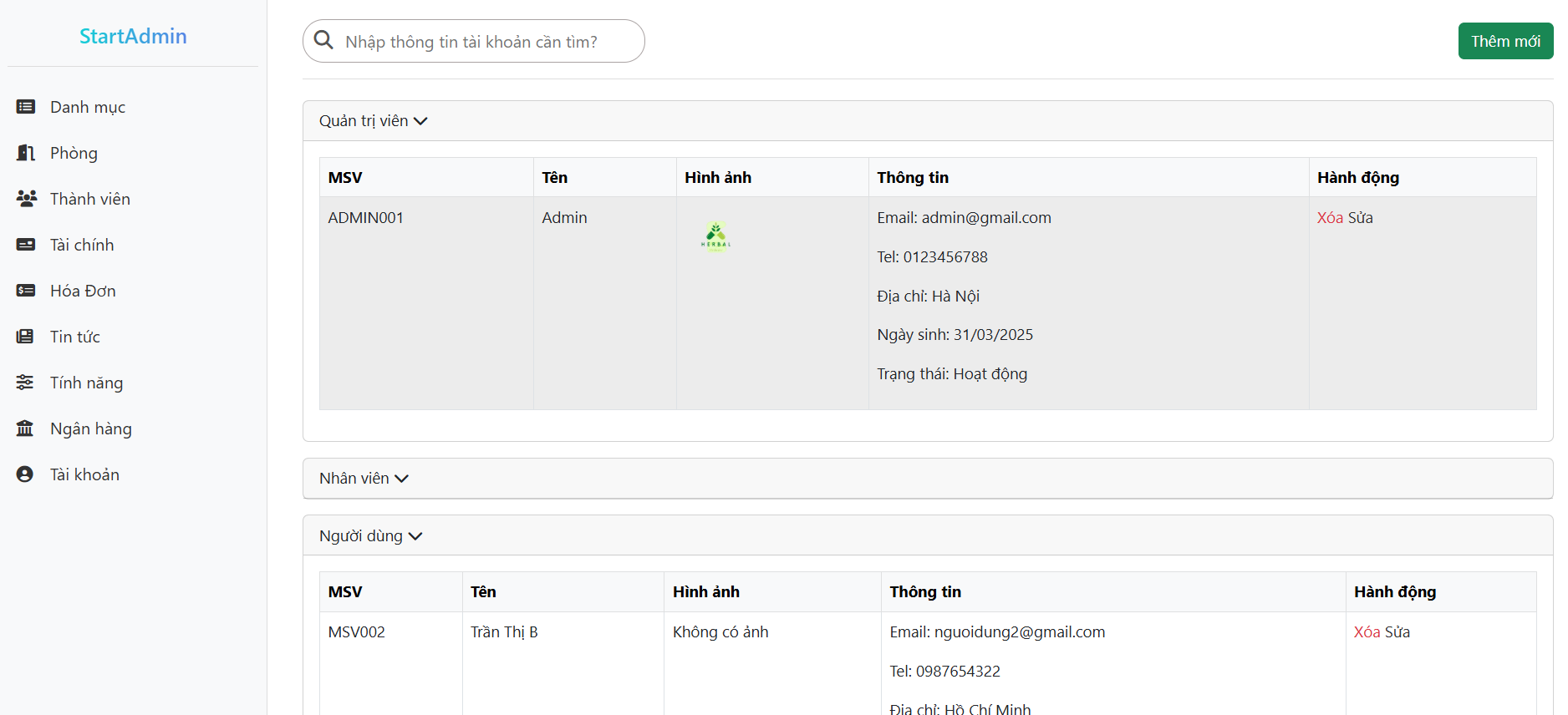
Hình 3.30. Trang sửa tin tức

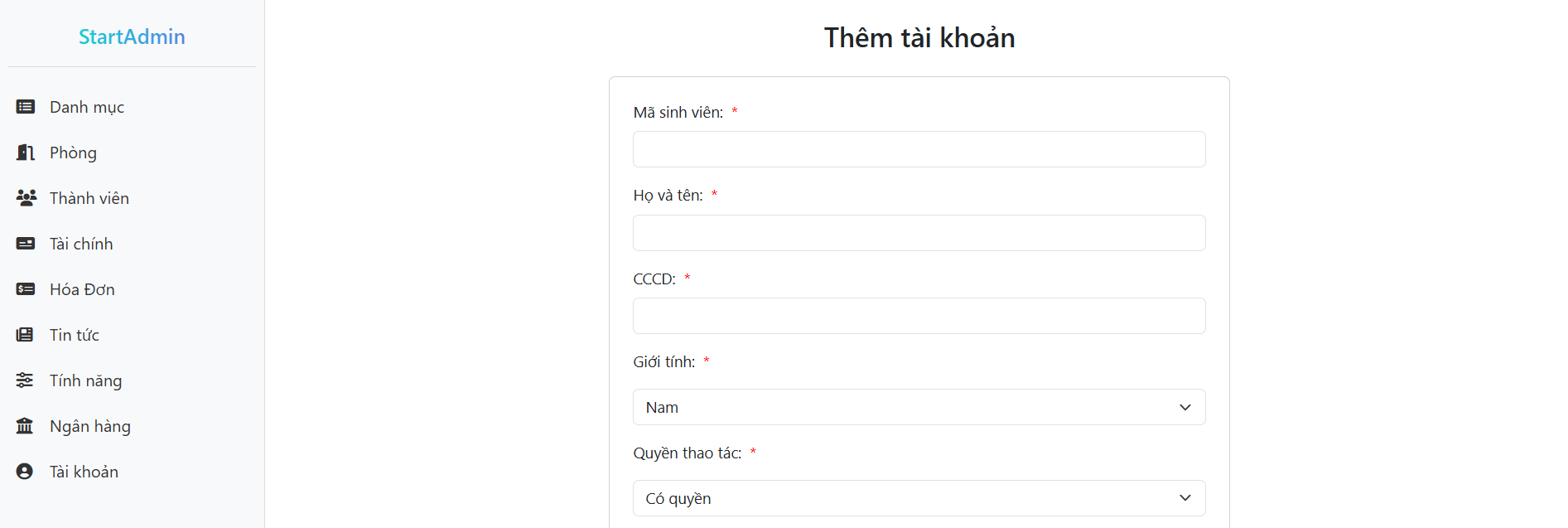
### *3.2.8. Trang quản lý tính năng*



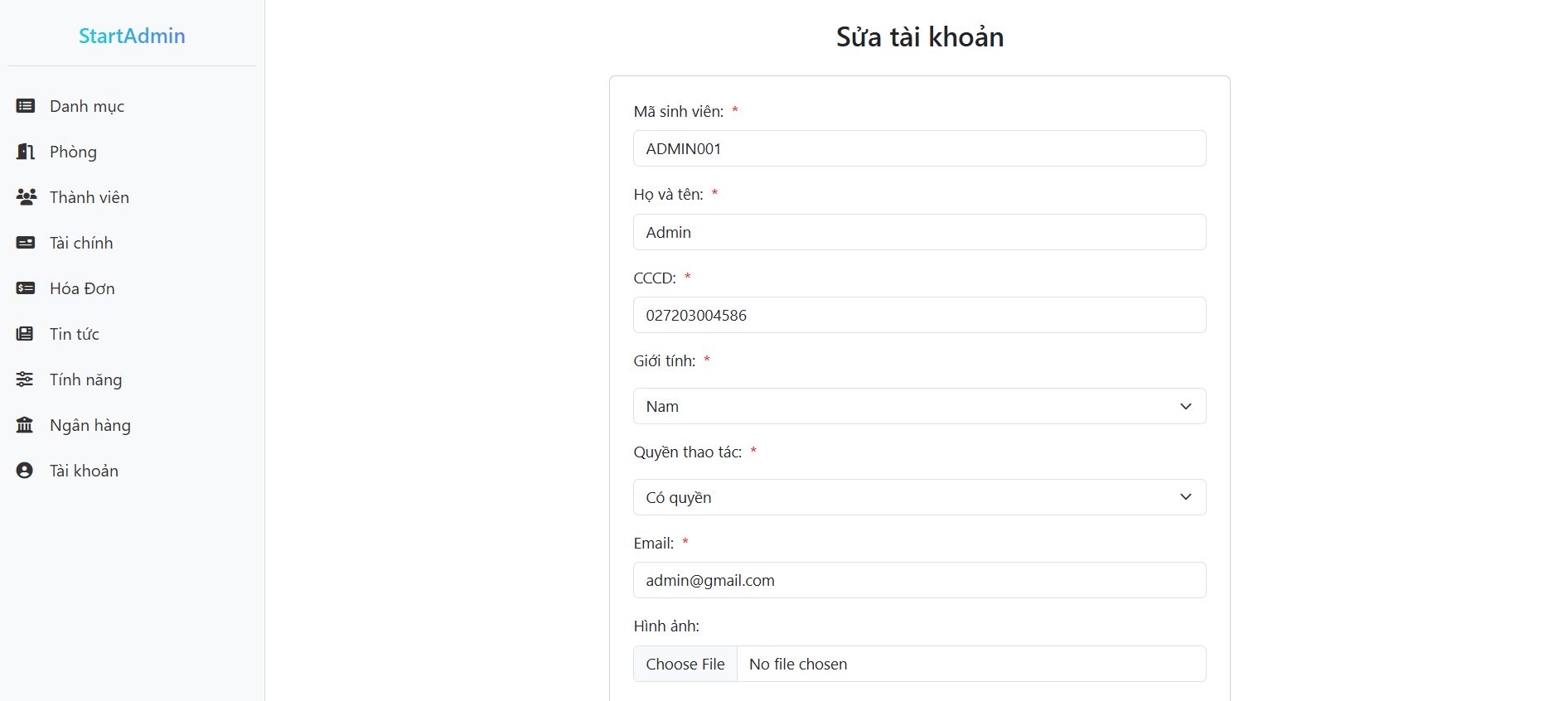
Hình 3.31. Trang quản lý tính năng

### *3.2.9. Trang quản lý tài khoản*

Hình 3.32. Trang quản lý tài khoản



Hình 3.33. Trang thêm tài khoản



Hình 3.34. Trang sửa tài khoản

**KẾT LUẬN**

Trong quá trình thực hiện đề tài "Xây dựng ứng dụng quản lý ký túc xá trên nền web sử dụng công nghệ Laravel" cho báo cáo tốt nghiệp, em đã dành nhiều thời gian và công sức để nghiên cứu, phát triển và triển khai một hệ thống quản lý ký túc xánhằm đáp ứng nhu cầu của người dùng và thị trường hiện nay.

Mặc dù vẫn còn một số hạn chế và thiếu sót trong đề tài, nhưng qua quá trình này, em đã đạt được nhiều thành tựu và học hỏi được nhiều kinh nghiệm quý báu trong lĩnh vực phát triển web và quản lý ký túc xá.Hướng phát triển và nghiên cứu trong tương lai của tôi bao gồm:

* Tiếp tục tìm hiểu sâu hơn về ngôn ngữ Javascript để có thể đáp ứng nhiều hơn nữa nhu cầu của người sử dụng, phát triển và tối ưu hóa hệ thống.
* Kết hợp ngôn ngữ Javascript với các hệ quản trị CSDL như mysql để xây dựng ứng dụng có tính tương tác cao và hiệu suất tốt.
* Nâng cao kiến thức về các công nghệ và công cụ phát triển web mới nhằm tạo ra giao diện đồ họa đẹp mắt và thân thiện hơn với người dùng.
* Thực hiện sản phẩm khóa luận xây dựng các trang web quy mô lớn hơn với nhiều ứng dụng và chức năng phức tạp để thử nghiệm và phát triển kỹ năng của mình.

Với những kế hoạch và mục tiêu này, em hy vọng sẽ tiếp tục phát triển sự nghiệp trong lĩnh vực công nghệ thông tin và đóng góp vào sự phát triển của cộng đồng công nghệ.

**TÀI LIỆU THAM KHẢO**

**1. Tài liệu tiếng việt**

1. Trịnh Thanh Tùng, *Thiết kế và lập trình web,* Nhà xuất bản Đại học Bách

khoa Hà Nội

[2]. Nguyễn Đắc Nghĩa, *Cấu trúc dữ liệu và thuật toán*, Nhà xuất bản Đại học

Bách khoa Hà Nội.

[3]. Trần Đình Quế (2018), *Phân tích và thiết kế hệ thống thông tin,* Nhà xuất

bản Học viện công nghệ Bưu chính Viễn thông.

[4]. Đậu Quang Tuấn (2007), *Tự học thiết kế trang Web bằng HTML,* Nhà xuất

bản Giao Thông Vận Tải, Hà Nội.

**Tài liệu website**

[5]. https://www.php.net/manual/en/

[6]. <https://laravel.com/docs/11.x>

[7]. <https://www.w3schools.com/mysql/default.asp>

[8]. <https://www.w3schools.com/bootstrap5/index.php>

# LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của riêng tôi. Những tài liệu trong đồ án tốt nghiệp là hoàn toàn trung thực. Các kết quả nghiên cứu do chính tôi thực hiện dưới sự chỉ đạo của giáo viên hướng dẫn.