****

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP.HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO ĐỒ ÁN CƠ SỞ**

**ĐỀ TÀI:**

**WEBSITE QUẢN LÝ THƯ VIỆN**

Giảng viên hướng dẫn: **Lê Cung Tưởng**

Sinh viên thực hiện:

**Nguyễn Hiền Thục MSSV: 2280603165 Lớp: 22DTHB4**

**Bùi Nhật Huy MSSV: 2280618692 Lớp: 22DTHB4**

**TP. Hồ Chí Minh, tháng 4 năm 2025**

# **NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN**

Giảng Viên: **Lê Cung Tưởng**

Tên đề tài: **WEBSITE QUẢN LÝ THƯ VIỆN**

Nội dung nhận xét:

……………………………..………………………………………………………….…….…………………….…….………………………………………………………..………….………...……………….…………………………………………………..………………..………….…………………………..……………………………………..………………...……………………………………………………………………………………………...…………………………………………………………………..…………………………..………………………………………………………………………………......…………..……………………………………………………………………………………………….……………………………………………………………………………………………...……………..………………………………………………………………………………..…………………………..…………………………………………………………………...………………………………………..…………………………………………………….……………………………..……………………………………………………..…………………………………………………………………………………………………………

Điểm:

Bằng số: ……………………….

Bằng chữ: ……………………...

**GIẢNG VIÊN CHẤM**

(Ký, ghi rõ họ tên)

**Lê Cung Tưởng**

# **LỜI MỞ ĐẦU**

Trong thời đại công nghệ số phát triển mạnh mẽ, thư viện không chỉ là nơi lưu trữ tri thức truyền thống mà còn là trung tâm thông tin, kết nối cộng đồng và thúc đẩy sự phát triển văn hóa đọc. Tuy nhiên, việc quản lý thư viện theo phương pháp truyền thống đang gặp phải nhiều thách thức, từ việc tìm kiếm, mượn trả sách đến thống kê, báo cáo. Để giải quyết vấn đề này, dự án "**Hệ thống Website Quản lý Thư Viện**" được triển khai nhằm xây dựng một nền tảng trực tuyến toàn diện, hiệu quả và thân thiện với người dùng.

Hệ thống này được thiết kế để tối ưu hóa các hoạt động của thư viện, mang đến trải nghiệm tốt hơn cho cả thủ thư và người đọc. Với các tính năng như tìm kiếm nâng cao, quản lý tài khoản người dùng, đặt mượn sách trực tuyến, thống kê báo cáo chi tiết, hệ thống giúp tiết kiệm thời gian, công sức và nâng cao hiệu quả công việc. Đặc biệt, hệ thống còn tích hợp các công cụ tương tác, cho phép người đọc chia sẻ đánh giá, nhận xét về sách, tham gia các diễn đàn thảo luận và tương tác trực tiếp với thư viện.

Chúng em tin rằng, dự án **"Hệ thống Website Quản lý Thư viện**" sẽ là một bước tiến quan trọng trong việc nâng cao chất lượng dịch vụ thư viện, thúc đẩy văn hóa đọc và tạo ra một cộng đồng học tập và nghiên cứu năng động.

Do thời gian và kiến thức của nhóm em còn hạn hẹp nên đề tài không thể tránh khỏi những thiếu sót, nhóm chúng em mong nhận được sự đóng góp của các thầy cô để đề tài được hoàn thiện hơn.

Chúng em xin chân thành cảm ơn!

Sinh viên thực hiện

**Bùi Nhật Huy**

**Nguyễn Hiền Thục**

# **LỜI CẢM ƠN**

Em xin gửi lời cảm ơn chân thành đến Giảng viên: **Lê Cung Tưởng** đã dành thời gian và tâm huyết để hướng dẫn nhóm em hoàn thành đề tài này. Sự hướng dẫn tận tình, những lời khuyên quý báu và những kiến thức sâu sắc của thầy/cô đã giúp em có được những định hướng đúng đắn và vượt qua những khó khăn trong quá trình nghiên cứu.

Em cũng xin gửi lời cảm ơn đến Ban Lãnh đạo Khoa **Công Nghệ Thông Tin** đã tạo điều kiện thuận lợi cho chúng em trong việc thu thập dữ liệu và thực hiện các khảo sát cần thiết. Xin cảm ơn các thầy cô giáo trong khoa đã chia sẻ những kinh nghiệm quý báu và cung cấp những tài liệu tham khảo hữu ích.

Cuối cùng, em xin cảm ơn gia đình và bạn bè đã luôn ở bên cạnh, động viên và ủng hộ em trong suốt quá trình thực hiện đề tài này. Sự quan tâm và động viên của mọi người là nguồn động lực lớn lao giúp em hoàn thành tốt đề tài của mình.".

# **LỜI CAM ĐOAN**

Nhóm chúng em xin cam đoan rằng mọi thông tin và nghiên cứu được trình bày trong đồ án này là trung thực và khách quan, được thu thập và phân tích một cách cẩn thận, dựa trên các nguồn chính thống và đáng tin cậy. Bất kỳ thông tin hoặc ý kiến nào được trích dẫn từ các nguồn khác đều được nêu rõ nguồn gốc và được trích dẫn theo đúng quy định. Chúng em cam đoan rằng không có bất kỳ sự sao chép hoặc sử dụng thông tin không đúng đắn nào từ các nguồn khác. Đồ án này là công trình nghiên cứu của nhóm em, chưa từng được công bố ở bất kỳ nơi nào khác. Chúng em cam đoan rằng đã tuân thủ đầy đủ các quy tắc và quy định, bao gồm cả việc tham khảo và sử dụng dữ liệu cũng như các công cụ nghiên cứu. Chúng em hy vọng rằng đồ án “**Website Quản Lý Thư Viện**” cung cấp một cái nhìn tổng quan rõ ràng và toàn diện về thuật toán, đồng thời đóng góp một phần nhỏ vào việc phát triển các hệ thống tương tự trong tương lai.

Nhóm thực hiện

**Bùi Nhật Huy**

**Nguyễn Hiền Thục**

**TÓM TẮT**

Website hệ thống quản lý thư viện là một giải pháp công nghệ hiện đại được xây dựng nhằm hỗ trợ tối ưu hóa quy trình quản lý sách và phục vụ bạn đọc trong các thư viện truyền thống. Hệ thống cho phép quản trị viên dễ dàng quản lý toàn bộ kho sách, từ việc thêm mới, chỉnh sửa, cập nhật thông tin sách cho đến phân loại theo thể loại, tác giả, nhà xuất bản. Người dùng có thể tra cứu sách trực tuyến thông qua công cụ tìm kiếm thông minh, xem chi tiết thông tin sách, tình trạng còn hoặc đã được mượn, và thực hiện yêu cầu mượn sách nếu có tài khoản đăng nhập. Bên cạnh đó, hệ thống còn tích hợp chức năng quản lý tài khoản người dùng, theo dõi lịch sử mượn – trả sách, gửi thông báo nhắc nhở hạn trả sách, hỗ trợ đăng ký tài khoản và thay đổi thông tin cá nhân. Đối với quản trị viên, hệ thống cung cấp các tính năng thống kê sách, quản lý độc giả, giám sát số lượng mượn trả theo từng thời điểm, và xuất báo cáo định kỳ để đánh giá hiệu quả hoạt động của thư viện. Giao diện website được thiết kế trực quan, dễ sử dụng trên cả máy tính và thiết bị di động, tích hợp các hiệu ứng CSS để nâng cao trải nghiệm người dùng. Về mặt kỹ thuật, hệ thống được phát triển trên nền tảng ASP.NET Core MVC kết hợp với Entity Framework Core, sử dụng SQL Server để lưu trữ dữ liệu, đảm bảo độ ổn định, bảo mật và khả năng mở rộng trong tương lai. Với các tính năng mạnh mẽ và khả năng ứng dụng thực tiễn cao, website hệ thống quản lý thư viện là công cụ hỗ trợ đắc lực cho việc vận hành và hiện đại hóa hoạt động thư viện trong kỷ nguyên số.

# 

# **MỤC LỤC**

[**NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN** 1](#_Toc199601148)

[**LỜI MỞ ĐẦU** 2](#_Toc199601149)

[**LỜI CẢM ƠN** 3](#_Toc199601150)

[**LỜI CAM ĐOAN** 4](#_Toc199601151)

[**MỤC LỤC** 6](#_Toc199601152)

[**DANH MỤC HÌNH** 8](#_Toc199601153)

[**DANH SÁCH CÁC KÝ TỰ, CHỮ VIẾT TẮT** 10](#_Toc199601154)

[**CHƯƠNG 1: TỔNG QUÁT** 12](#_Toc199601155)

[**1.1** **Lý do thực hiện đề tài** 12](#_Toc199601156)

[**1.2** **Mục đích nghiên cứu** 13](#_Toc199601157)

[**1.3** **Đối tượng và phạm vi nghiên cứu** 13](#_Toc199601158)

[**1.4** **Phương pháp nghiên cứu** 14](#_Toc199601159)

[**1.4.2.** **Phương pháp khảo sát thực tế** 14](#_Toc199601160)

[**1.4.3.** **Phương pháp thực nghiệm** 15](#_Toc199601161)

[**1.4.4.** **Phương pháp phân tích và tổng hợp** 15](#_Toc199601162)

[**1.4.5.** **Phương pháp đánh giá và đề xuất** 15](#_Toc199601163)

[**1.4.6.** **Phương pháp phân tích thiết kế hệ thống** 15](#_Toc199601164)

[**1.5** **Đặc điểm của hệ thống quản lý** 17](#_Toc199601165)

[**1.6** **Ý nghĩa khoa học và thực tiễn** 17](#_Toc199601166)

[**CHƯƠNG 2: CƠ SỞ LÝ THUYẾT** 19](#_Toc199601167)

[**2.1 Giới thiệu đồ án** 19](#_Toc199601168)

[**2.2 Công nghệ sử dụng** 25](#_Toc199601169)

[**2.3 Khảo sát hệ thống** 30](#_Toc199601170)

[**2.4 Yêu cầu hệ thống** 32](#_Toc199601171)

[**2.5 Xây dựng sơ đồ Class Diagram** 34](#_Toc199601172)

[**2.6 Sơ đồ Use Case** 35](#_Toc199601173)

[**CHƯƠNG 3: HIỆN THỰC ỨNG DỤNG** 36](#_Toc199601174)

[**3.1 Giao diện đăng ký, đăng nhập** 36](#_Toc199601175)

[**3.2 Giao diện người dùng** 37](#_Toc199601176)

[**3.3 Giao diện quản trị** 40](#_Toc199601177)

[**CHƯƠNG 4: TỔNG KẾT** 55](#_Toc199601178)

[**4.1 Kết quả.** 55](#_Toc199601179)

[**4.2 Hạn chế của đề tài.** 55](#_Toc199601180)

[**4.3 Hướng phát triển.** 55](#_Toc199601181)

[**4.4 Kết luận.** 56](#_Toc199601182)

[**TÀI LIỆU THAM KHẢO** 57](#_Toc199601183)

# **DANH MỤC HÌNH**

[Hình 1: Logo C# 26](#_Toc195141207)

[Hình 2: Logo SQL Server 26](#_Toc195141208)

[Hình 3: Logo Visual Studio 27](#_Toc195141209)

[Hình 4: Logo MVVM, Entity FrameWork, và Repository Pattern 28](#_Toc195141210)

[Hình 5: Logo HTML 29](#_Toc195141211)

[Hình 6: Logo CSS 29](#_Toc195141212)

[Hình 7: Logo JavaScript 30](#_Toc195141213)

[Hình 8: Logo Bootstrap 30](#_Toc195141214)

[Hình 9: Code First 31](#_Toc195141215)

[Hình 10. Sơ đồ RD 34](#_Toc195141216)

[Hình 11: Đăng Ký 35](#_Toc195141217)

[Hình 12: Đăng Nhập 35](#_Toc195141218)

[Hình 13. Giao diện trang chủ 36](#_Toc195141219)

[Hình 14. Giao diện giới thiệu hệ thống thư viện 36](#_Toc195141220)

[Hình 15. Giao diện Sách 37](#_Toc195141221)

[Hình 16. Giao diện liên hệ 37](#_Toc195141222)

[Hình 17. Giao diện Sách chi tiết 38](#_Toc195141223)

[Hình 18. Giao diện quản trị nếu là người dùng 38](#_Toc195141224)

[Hình 19. Giao diện chung 39](#_Toc195141225)

[Hình 20. Giao diện chính quản lý sách 39](#_Toc195141226)

[Hình 21. Giao diện thông tin chi tiết sách 40](#_Toc195141227)

[Hình 22. Giao diện chỉnh sửa, cập nhật sách 40](#_Toc195141228)

[Hình 23. Giao diện xóa sách khỏi danh sách thư viện 41](#_Toc195141229)

[Hình 24. Giao diện thêm sách mới vào thư viện 41](#_Toc195141230)

[Hình 27. Giao diện sửa thể loại 43](#_Toc195141231)

[Hình 28. Giao diện xóa thể loại 43](#_Toc195141232)

[Hình 29. Giao diện chính 44](#_Toc195141233)

[Hình 30. Giao diện đăng ký thẻ thư viện 44](#_Toc195141234)

[Hình 31. Giao diện chỉnh sửa thẻ thư viện 45](#_Toc195141235)

[Hình 32. Giao diện xóa thẻ thư viện 45](#_Toc195141236)

[Hình 33. Giao diện chung của quản lý mượn, trả sách 46](#_Toc195141237)

[Hình 34. Giao diện thêm người dùng mượn sách 46](#_Toc195141238)

[Hình 35. Giao diện chỉnh sửa người mượn 47](#_Toc195141239)

[Hình 36. Giao diện hiển thị chi tiết mượn sách 47](#_Toc195141240)

[Hình 37. Giao diện xóa người dùng mượn sách 48](#_Toc195141241)

[Hình 38. Giao diện chính thông tin phiếu phạt 48](#_Toc195141242)

[Hình 39. Khi nhấn phiếu chưa thanh toán sẽ hiện các phiếu chưa thanh toán 49](#_Toc195141243)

[Hình 40. Giao diện thêm phiếu phạt khi người dùng mượn quá hạn, mất sách 49](#_Toc195141244)

[Hình 41. Sửa thông tin phiếu phạt 50](#_Toc195141245)

[Hình 42. Chi tiết phiếu phạt 50](#_Toc195141246)

[Hình 43. Khi người dùng đã thanh toán phiếu phạt thì sẽ hiện đã thanh toán 51](#_Toc195141247)

[Hình 44. Xóa phiếu phạt khi đã thanh toán 51](#_Toc195141248)

# **DANH SÁCH CÁC KÝ TỰ, CHỮ VIẾT TẮT**

|  |  |  |
| --- | --- | --- |
| **Từ viết tắt** | **Ý nghĩa đầy đủ** | **Mô tả** |
| SQL CF | Cơ sở dữ liệu Code First | để xây dựng và quản lý cơ sở dữ liệu trong Entity Framework mà không cần phải viết mã SQL thủ công |
| DBMS | Cơ sở dữ liệu SQL | cho phép người dùng lưu trữ và quản lý dữ liệu trong các cơ sở dữ liệu quan hệ |
| [ASP.NET](http://ASP.NET) | Active Server Pages .NET | Một khung làm việc để xây dựng các ứng dụng web động sử dụng ngôn ngữ C# hoặc VB.NET. |
| ERD | Entity-Relationship Diagram | Sơ đồ thực thể-quan hệ, một công cụ mô hình hóa dữ liệu để biểu diễn cấu trúc của một cơ sở dữ liệu. |
| UI | User Interface | Giao diện người dùng, phần của một ứng dụng mà người dùng tương tác trực tiếp. |
| MVC | Model-View-Controller | Mô hình kiến trúc phần mềm, chia ứng dụng thành ba phần chính: Model (dữ liệu), View (giao diện) và Controller (điều khiển). |
| C# | C sharp | Một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft, chạy trên nền tảng .NET. |
| .NET Visual Studio | Microsoft Visual Studio | Một IDE phổ biến để phát triển các ứng dụng .NET. |
| Entity Framework | Entity Framework | Một ORM (Object-Relational Mapper) của Microsoft, cho phép các nhà phát triển làm việc với dữ liệu trong cơ sở dữ liệu bằng các đối tượng .NET. |
| HTML | HyperText Markup Language | Là khung xương của một trang web. HTML dùng để tạo và tổ chức nội dung như văn bản, hình ảnh, liên kết, bảng biểu, biểu mẫu |
| CSS | Cascading Style Sheets | CSS dùng để tạo giao diện, màu sắc, bố cục, font chữ... cho trang web. Nói cách khác, CSS giúp "trang điểm" cho phần HTML. |
| JS | JavaScript | Làm cho trang web tương tác, động và thông minh (như: kiểm tra biểu mẫu, hiển thị giờ hiện tại, thay đổi nội dung khi nhấn nút) |
| Bootstrap | Bootstrap | Giúp xây dựng giao diện web hiện đại, đẹp, và phản hồi tốt trên mọi thiết bị (responsive), mà không cần viết CSS từ đầu. |

# **CHƯƠNG 1: TỔNG QUÁT**

* 1. **Lý do thực hiện đề tài**

Trong bối cảnh cách mạng công nghiệp 4.0 đang diễn ra mạnh mẽ, công nghệ thông tin ngày càng đóng vai trò quan trọng trong việc nâng cao hiệu suất làm việc và tối ưu hóa quy trình quản lý trong mọi lĩnh vực của đời sống, đặc biệt là lĩnh vực giáo dục và văn hóa. Thư viện, với vai trò là trung tâm lưu trữ và cung cấp tri thức, cần được đổi mới và hiện đại hóa để đáp ứng nhu cầu ngày càng cao của người sử dụng. Tuy nhiên, trên thực tế, tại nhiều đơn vị như trường học, thư viện công cộng, trung tâm đào tạo,... việc quản lý sách, tài liệu, bạn đọc và quá trình mượn – trả vẫn còn được thực hiện thủ công hoặc sử dụng các công cụ quản lý đơn giản như sổ tay, file Excel. Cách làm này không chỉ mất nhiều thời gian, dễ xảy ra sai sót mà còn gây khó khăn trong việc tra cứu thông tin, thống kê số liệu hay kiểm tra tình trạng mượn – trả sách.

Bên cạnh đó, nhu cầu tìm kiếm, truy cập và sử dụng thông tin của người dùng ngày càng trở nên linh hoạt và yêu cầu cao hơn. Người đọc mong muốn có thể tra cứu thông tin sách mọi lúc, mọi nơi, nắm được tình trạng hiện tại của tài liệu, và dễ dàng đăng ký mượn sách thông qua nền tảng trực tuyến. Điều này đòi hỏi các thư viện cần có một hệ thống quản lý thông minh, tiện lợi và dễ sử dụng, có thể tích hợp các chức năng như tìm kiếm sách, phân loại, thống kê, quản lý người dùng và hỗ trợ mượn trả hiệu quả.

Xuất phát từ thực tế đó, nhóm thực hiện đề tài nhận thấy rằng việc phát triển một **hệ thống quản lý thư viện trực tuyến** là cần thiết và mang tính ứng dụng cao. Hệ thống này không chỉ hỗ trợ cán bộ thư viện quản lý kho sách một cách dễ dàng, giảm thiểu tối đa sai sót và khối lượng công việc thủ công, mà còn mang lại sự thuận tiện cho người sử dụng trong việc tra cứu và mượn tài liệu. Đề tài đồng thời cũng là cơ hội để nhóm áp dụng những kiến thức đã học về lập trình web, thiết kế cơ sở dữ liệu, mô hình MVC, bảo mật và các công nghệ hiện đại như ASP.NET Core MVC, Entity Framework Core vào thực tế. Việc xây dựng một hệ thống có tính thực tiễn cao, vừa đáp ứng nhu cầu quản lý vừa thân thiện với người dùng, là một bước tiến quan trọng trong hành trình rèn luyện kỹ năng và phát triển tư duy lập trình của sinh viên.

Vì những lý do trên, đề tài "Xây dựng hệ thống quản lý thư viện" được lựa chọn với mong muốn tạo ra một sản phẩm không chỉ mang tính học thuật mà còn có khả năng ứng dụng thực tế trong môi trường giáo dục và đào tạo hiện nay. Đề tài kỳ vọng sẽ góp phần nhỏ vào công cuộc hiện đại hóa công tác quản lý thư viện, nâng cao chất lượng phục vụ bạn đọc, và mở rộng khả năng tích hợp công nghệ vào lĩnh vực thư viện trong tương lai.

* 1. **Mục đích nghiên cứu**

Mục đích chính của đề tài "Xây dựng hệ thống quản lý thư viện" là thiết kế và phát triển một website quản lý thư viện hiện đại, tiện ích, có khả năng hỗ trợ tối ưu hóa quy trình quản lý sách, người dùng và hoạt động mượn – trả trong môi trường thư viện. Thông qua đề tài, người thực hiện mong muốn giải quyết những hạn chế còn tồn tại trong các phương thức quản lý thư viện truyền thống, đồng thời tạo ra một hệ thống thông minh, thân thiện với người sử dụng và có tính ứng dụng thực tiễn cao.

Cụ thể, đề tài hướng đến các mục tiêu sau:

* **Xây dựng một hệ thống phần mềm quản lý thư viện dựa trên nền tảng web**, có thể hoạt động trên các trình duyệt phổ biến và tương thích với nhiều thiết bị khác nhau như máy tính, máy tính bảng, điện thoại thông minh.
* **Tích hợp đầy đủ các chức năng quản lý cốt lõi**, bao gồm: thêm, sửa, xóa, tìm kiếm sách; phân loại theo thể loại, tác giả, nhà xuất bản; quản lý thông tin người dùng; chức năng mượn – trả sách và theo dõi tình trạng sách.
* **Tạo giao diện thân thiện, dễ sử dụng**, phù hợp với cả người quản trị (admin) và người đọc (user), giúp người dùng dễ dàng thao tác và tiếp cận thông tin.
* **Ứng dụng các kiến thức đã học** về mô hình MVC, lập trình hướng đối tượng, cơ sở dữ liệu, bảo mật, giao diện người dùng, đồng thời rèn luyện kỹ năng triển khai dự án thực tế bằng các công nghệ hiện đại như ASP.NET Core MVC, Entity Framework Core và SQLCode First.
* **Đảm bảo tính bảo mật và tính mở rộng** của hệ thống, giúp việc quản lý dữ liệu được an toàn, đồng thời dễ dàng nâng cấp, tích hợp thêm các chức năng mới trong tương lai (như chat hỗ trợ, đánh giá sách, xuất file thống kê).
* **Tạo nền tảng cho việc chuyển đổi số trong công tác quản lý thư viện**, góp phần nâng cao chất lượng hoạt động của thư viện trong thời đại công nghệ thông tin.
  1. **Đối tượng và phạm vi nghiên cứu**

Tìm hiểu về .NET Framework khảo sát các quy trình phương pháp xây dựng một ứng dụng quản lý.

**- Đối tượng nghiên cứu:**

Đối tượng nghiên cứu của đề tài là quá trình xây dựng và triển khai một hệ thống quản lý thư viện trực tuyến trên nền tảng ASP.NET Core MVC. Hệ thống này tập trung vào việc số hóa các hoạt động trong thư viện như quản lý sách, người dùng và quy trình mượn – trả sách, nhằm giảm tải công việc cho cán bộ thư viện và nâng cao trải nghiệm của bạn đọc. Trong quá trình thực hiện, đề tài khai thác các khía cạnh kỹ thuật quan trọng như lập trình mô hình MVC, tổ chức cơ sở dữ liệu bằng SQL Server, thao tác dữ liệu với Entity Framework Core, xây dựng giao diện người dùng với HTML, CSS và Bootstrap/Tailwind CSS, cùng với áp dụng các mô hình thiết kế phần mềm như Repository Pattern. Đối tượng hướng đến là cả người quản trị (admin) và người dùng thông thường, với chức năng và quyền hạn phù hợp.

**- Phạm vi nghiên cứu:**

Phạm vi của đề tài được giới hạn trong việc xây dựng một hệ thống thư viện ở quy mô vừa và nhỏ, đảm bảo đầy đủ các chức năng cơ bản như thêm, sửa, xóa, tìm kiếm sách, phân loại sách theo thể loại, quản lý tài khoản người dùng, mượn và trả sách. Hệ thống có hai vai trò người dùng chính: quản trị viên và bạn đọc. Các chức năng nâng cao như thống kê chuyên sâu, đánh giá sách, mã hóa dữ liệu, tích hợp đa thư viện, hoặc xác thực đa lớp sẽ không được triển khai trong phạm vi hiện tại mà có thể được mở rộng trong các giai đoạn sau. Đề tài tập trung vào phần kỹ thuật phát triển phần mềm, chưa đi sâu vào chuẩn nghiệp vụ chuyên biệt của ngành thư viện học.

* 1. **Phương pháp nghiên cứu**

Để thực hiện đề tài "Xây dựng hệ thống quản lý thư viện " các phương pháp nghiên cứu được áp dụng bao gồm:

**1.4.1 Phương pháp nghiên cứu tài liệu**

Phương pháp này tập trung vào việc thu thập và tìm hiểu các tài liệu học thuật, sách chuyên ngành, bài báo, luận văn, cũng như tài nguyên trực tuyến có liên quan đến lĩnh vực quản lý thư viện. Các nội dung nghiên cứu bao gồm: nguyên lý quản lý thư viện, phương pháp tổ chức kho sách, quản lý bạn đọc, quản lý mượn trả, và xu hướng ứng dụng công nghệ trong số hóa thư viện hiện đại. Đặc biệt, các nguồn tài liệu về kiến trúc phần mềm ASP.NET Core MVC, quản lý cơ sở dữ liệu với Entity Framework Core, và các mô hình thiết kế như Repository Pattern cũng được nghiên cứu kỹ lưỡng. Việc nghiên cứu tài liệu giúp tạo nền tảng lý luận vững chắc cho quá trình xây dựng và phát triển hệ thống.

* + 1. **Phương pháp khảo sát thực tế**

Để hệ thống thiết kế ra phù hợp với nhu cầu sử dụng, đề tài tiến hành khảo sát thực tế tại các thư viện trường học, thư viện công cộng hoặc trung tâm học liệu. Thông qua việc phỏng vấn thủ thư, nhân viên quản lý và bạn đọc, nghiên cứu ghi nhận các quy trình nghiệp vụ hiện có, những vướng mắc gặp phải trong quá trình quản lý sách và mượn – trả tài liệu, cũng như mong muốn cải tiến hệ thống. Đồng thời, khảo sát một số hệ thống thư viện đã được triển khai nhằm đánh giá các tính năng ưu việt và hạn chế còn tồn tại. Những dữ liệu thu được sẽ đóng vai trò quan trọng trong việc định hướng thiết kế và phát triển chức năng cho hệ thống.

* + 1. **Phương pháp thực nghiệm**

Sau khi có cơ sở lý luận và dữ liệu khảo sát, nhóm nghiên cứu sẽ tiến hành xây dựng một hệ thống quản lý thư viện mẫu bằng ASP.NET Core MVC. Quá trình thực nghiệm bao gồm việc thiết kế giao diện người dùng thân thiện, tổ chức cơ sở dữ liệu chặt chẽ và lập trình các chức năng như: quản lý sách, quản lý người dùng, mượn và trả sách, tìm kiếm sách, phân loại theo thể loại hoặc tác giả. Việc thực nghiệm nhằm kiểm tra tính khả thi của hệ thống, độ ổn định, độ chính xác của dữ liệu, và tốc độ xử lý trong môi trường mô phỏng thực tế hoặc triển khai thử nghiệm.

* + 1. **Phương pháp phân tích và tổng hợp**

Tất cả dữ liệu và kết quả thu được từ quá trình nghiên cứu tài liệu, khảo sát thực tế và thực nghiệm sẽ được phân tích và tổng hợp nhằm đánh giá chất lượng và hiệu quả của hệ thống. Phương pháp này giúp so sánh hệ thống mới với phương pháp quản lý thủ công hoặc hệ thống cũ (nếu có), đồng thời chỉ ra những cải tiến rõ rệt về mặt chức năng, tốc độ xử lý, độ chính xác, và trải nghiệm người dùng. Việc phân tích tổng hợp cũng giúp rút ra các nhận định khách quan và định hướng cải tiến hệ thống trong tương lai.

* + 1. **Phương pháp đánh giá và đề xuất**

Dựa trên kết quả của quá trình nghiên cứu, khảo sát và thực nghiệm, nhóm thực hiện sẽ tiến hành đánh giá toàn diện hệ thống thư viện mẫu. Việc đánh giá tập trung vào hiệu suất, tính ổn định, khả năng mở rộng và tính tiện ích trong thực tiễn. Những điểm mạnh và hạn chế của hệ thống sẽ được chỉ ra rõ ràng, từ đó đề xuất các hướng cải tiến hợp lý. Đây là bước quan trọng để hệ thống có thể hoàn thiện hơn, dễ dàng triển khai trong các thư viện thực tế và có khả năng phát triển lâu dài.

* + 1. **Phương pháp phân tích thiết kế hệ thống**

Phương pháp này là nền tảng trong việc xây dựng hệ thống, bao gồm:

* **Phân tích yêu cầu**: Thông qua khảo sát và phỏng vấn các đối tượng sử dụng như thủ thư, cán bộ thư viện và người đọc, đề tài xác định được các chức năng chính cần có: quản lý sách (thêm, sửa, xóa, phân loại), quản lý tài khoản người dùng, mượn – trả sách, thống kê và báo cáo. Đồng thời phân tích các yêu cầu phi chức năng như bảo mật, hiệu suất, khả năng mở rộng.
* **Mô hình hóa hệ thống**: Code First là một phương pháp trong **Entity Framework** (EF) để tạo và quản lý cơ sở dữ liệu. Với phương pháp này, bạn bắt đầu bằng cách **định nghĩa các lớp mô hình** (model classes) trong mã nguồn của bạn, và Entity Framework sẽ tự động tạo ra cơ sở dữ liệu từ các lớp này. Phương pháp này giúp phát triển ứng dụng một cách linh hoạt và thuận tiện, đặc biệt trong quá trình thiết kế cơ sở dữ liệu.
* **Thiết kế chức năng**: Dựa trên phân tích yêu cầu, tiến hành thiết kế chi tiết các chức năng như:
  + Quản lý sách: nhập, chỉnh sửa, tìm kiếm, phân loại theo thể loại hoặc tác giả.
  + Quản lý thể loại: Nhập, thêm, xóa, sửa thể loại.
  + Mượn/trả sách: ghi nhận giao dịch, theo dõi tình trạng sách, hạn mượn và trả.
  + Quản lý phiếu phạt: Thên, xóa, sửa theo dõi tình trạng mượn sách, nếu quá hạn thì có thể bị phạt tiền
  + Quản lý thẻ thư viện: Cho phép người dùng đăng ký thẻ thư viện và cấp thời gian tạo thẻ và thời gian hết hạn thẻ để mượn sách.
  + Quản lý người dùng: Cho phép hiển thị theo dõi thông tin tài khoản hiện có trên hệ thống
  + Báo cáo thống kê: Thống kê lại tất cả số liệu như: tổng có bao nhiêu sách,số lượt mượn sách gần đây trong tháng và bảng thống kê số liệu biểu đồ cột theo từng tháng.
* **Thiết kế giao diện người dùng (UI)**: Hệ thống hướng tới giao diện trực quan, dễ sử dụng, tối ưu cho cả người quản lý và bạn đọc. Sử dụng Bootstrap hoặc Tailwind CSS để thiết kế giao diện hiện đại, phản hồi nhanh trên nhiều thiết bị.
* **Thiết kế tổng thể hệ thống**: Hệ thống sử dụng kiến trúc MVC để tách biệt giao diện, logic xử lý và truy xuất dữ liệu. Các yếu tố như bảo mật người dùng, phân quyền và khả năng mở rộng cũng được tích hợp ngay từ khâu thiết kế ban đầu, đảm bảo hệ thống bền vững và dễ nâng cấp.
  1. **Đặc điểm của hệ thống quản lý**

Hệ thống quản lý thư viện có những đặc điểm riêng biệt phản ánh tính chất đặc thù của hoạt động thư viện. Phân cấp quản lý là yếu tố quan trọng, với một cấu trúc tổ chức rõ ràng, phân định trách nhiệm và quyền hạn giữa các cấp quản lý, từ thủ thư, người dùng, Việc phân cấp này đảm bảo mỗi cá nhân hoặc bộ phận đều có quyền kiểm soát và thực hiện các công việc cụ thể trong quy trình quản lý thư viện.

Các luồng thông tin trong hệ thống rất đa dạng, bao gồm thông tin về sách, độc giả, mượn – trả sách, thống kê, báo cáo, v.v. Những luồng thông tin này cần được xử lý và truyền tải một cách nhanh chóng, chính xác và hiệu quả để đảm bảo sự hoạt động liên tục và thông suốt của thư viện. Việc quản lý thông tin phải được thực hiện qua các chức năng như tìm kiếm, cập nhật và báo cáo, giúp tối ưu hóa quy trình vận hành của thư viện.

Mô hình hệ thống thông tin trong quản lý thư viện bao gồm các thành phần chính như cơ sở dữ liệu, phần mềm quản lý, giao diện người dùng và các thiết bị phần cứng. Mô hình này cần được thiết kế linh hoạt, có khả năng mở rộng để đáp ứng nhu cầu phát triển trong tương lai và dễ dàng tích hợp với các hệ thống khác, như hệ thống tìm kiếm tài liệu trực tuyến hoặc các nền tảng học liệu điện tử.

* 1. **Ý nghĩa khoa học và thực tiễn**
* **Ý nghĩa khoa học:**

Nghiên cứu đề tài "Xây dựng hệ thống quản lý thư viện" đóng góp vào kho tàng tri thức trong lĩnh vực quản lý thông tin và công nghệ thông tin. Đề tài không chỉ khảo sát và áp dụng các công nghệ hiện đại như **Entity Framework** và **ASP.NET Core MVC**, mà còn mở rộng hiểu biết về các quy trình quản lý trong môi trường thư viện. Việc nghiên cứu và phát triển các mô hình quản lý mới giúp nâng cao lý thuyết về quản lý thư viện, đồng thời cung cấp các khung lý thuyết có thể được áp dụng trong các nghiên cứu tương lai. Hơn nữa, việc sử dụng các phương pháp phân tích và thiết kế hệ thống giúp hình thành các tiêu chí và tiêu chuẩn cho việc xây dựng các hệ thống quản lý thư viện tương tự, tạo cơ sở cho các nghiên cứu sau có thể phát triển và ứng dụng hiệu quả hơn. Đề tài này sẽ trở thành tài liệu nghiên cứu, tham khảo, dễ hiểu và thiết thực hơn cho sinh viên, các nhà nghiên cứu và những ai quan tâm đến ứng dụng công nghệ trong quản lý thư viện.

* **Ý nghĩa thực tiễn:**

Hệ thống quản lý thư viện được phát triển từ nghiên cứu này có ý nghĩa thực tiễn quan trọng đối với các thư viện, từ thư viện công cộng đến các thư viện trường học. Hệ thống giúp tối ưu hóa quy trình quản lý sách, nâng cao hiệu quả trong việc mượn trả sách, kiểm soát tồn kho và giảm thiểu các lỗi trong quá trình quản lý. Việc áp dụng công nghệ thông tin trong quản lý giúp cải thiện trải nghiệm của độc giả, từ việc tìm kiếm sách nhanh chóng đến việc trả sách và gia hạn sách tiện lợi hơn. Hệ thống còn cung cấp các báo cáo thống kê và phân tích dữ liệu hữu ích, giúp cán bộ thư viện đưa ra quyết định kịp thời và chính xác hơn về việc bổ sung sách mới, quản lý các hoạt động thư viện và nâng cao chất lượng dịch vụ.

Với sự thành công của đề tài này, hệ thống sẽ góp phần giúp các thư viện tối ưu hóa quy trình quản lý, loại bỏ các khó khăn trong việc quản lý tài liệu và dịch vụ. Hệ thống cũng có thể được áp dụng rộng rãi cho nhiều mô hình thư viện khác nhau, từ thư viện trường học đến thư viện công cộng, góp phần thúc đẩy sự chuyển đổi số trong lĩnh vực quản lý thư viện. Hệ thống này không chỉ hỗ trợ các thư viện hiện tại mà còn tạo nền tảng quản lý vững chắc cho các thư viện mới thành lập, giúp họ bắt đầu hoạt động với hệ thống quản lý hiệu quả ngay từ những ngày đầu. Tóm lại, đề tài nghiên cứu không chỉ mang lại giá trị khoa học mà còn có ứng dụng thiết thực trong việc nâng cao chất lượng dịch vụ thư viện và tối ưu hóa quá trình quản lý tài liệu.

# **CHƯƠNG 2: CƠ SỞ LÝ THUYẾT**

## **2.1 Giới thiệu đồ án**

**2.1.1 Sơ lược về đồ án**

Đồ án Hệ thống quản lý thư viện được xây dựng với mục tiêu hỗ trợ công tác quản lý sách, độc giả và hoạt động mượn - trả sách một cách khoa học, tiện lợi và hiệu quả. Hệ thống được phát triển bằng ASP.NET Core MVC – một framework mạnh mẽ của Microsoft cho phép xây dựng các ứng dụng web theo mô hình phân lớp rõ ràng, dễ bảo trì và mở rộng.

Trong quá trình phát triển, mô hình Code First của Entity Framework Core được sử dụng để tạo và quản lý cơ sở dữ liệu. Với cách tiếp cận này, các lớp mô hình (Model) được thiết kế trước, sau đó Entity Framework sẽ tự động tạo cơ sở dữ liệu dựa trên các lớp này. Điều này giúp việc phát triển hệ thống linh hoạt hơn và dễ dàng cập nhật, chỉnh sửa.

Về giao diện người dùng, đồ án sử dụng kết hợp HTML, CSS, JavaScript và Bootstrap để xây dựng các trang web thân thiện, dễ sử dụng và có tính thẩm mỹ cao. Bootstrap giúp bố cục giao diện trở nên chuẩn chỉnh và tương thích với nhiều thiết bị khác nhau, trong khi CSS và JavaScript giúp tạo hiệu ứng động, tăng trải nghiệm người dùng.

Hệ thống cung cấp các chức năng cơ bản như:

* Quản lý thể loại sách (thêm, sửa, xóa, tìm kiếm)
* Quản lý người dùng/độc giả
* Quản lý đăng ký thẻ thư viện
* Quản lý mượn và trả sách
* Quản lý phiếu phạt (mất sách, quá hạn)
* Giao diện thân thiện, dễ sử dụng
* Báo cáo thống kê
* Quản lý danh sách sách

Thông qua việc xây dựng đồ án này, sinh viên không chỉ được rèn luyện kỹ năng lập trình với công nghệ ASP.NET Core MVC, mà còn có cơ hội áp dụng kiến thức về thiết kế giao diện, xử lý dữ liệu và triển khai hệ thống một cách toàn diện.

**2.1.2 Tên đồ án**

“Hệ thống Quản Lý Thư Viện”

**2.1.3 Chức năng**

Đề tài “Xây dựng hệ thống quản lý thư viện” nhằm phát triển một phần mềm quản lý thư viện hiện đại, tiện lợi, hỗ trợ nhân viên và quản trị viên trong việc tổ chức, tra cứu, mượn trả sách và quản lý thông tin người dùng. Hệ thống được xây dựng bằng công nghệ ASP.NET Core MVC, sử dụng mô hình Code First với Entity Framework Core, kết hợp HTML, CSS, JavaScript và Bootstrap để tạo giao diện trực quan, thân thiện với người dùng.

**Đăng Nhập, đăng ký:**

Cho phép đăng ký, đăng nhập và phân quyền theo Admin hoặc người dùng

**Giao diện Độc giả:**

* Trang chủ: Hiển thị thông tin của hệ thống.
* Giới thiệu: Vài nét đôi lược về hệ thống quản lý thư viện
* Sách: Thông tin các sách nổi bật nhất của hệ thống thư viện
* Liên hệ: Form thông tin dùng để liên hệ cho người dùng
* Quản trị: Là phần chỉ có quyền admin mới có thể truy cập để có thể quản lý mọi thứ liên quan tới thư viện
* Sách chi tiết: Hiển thị thông tin những sách có sẵn trong thư viện và các sách sau khi được Admin thêm vào, độc giả có thể nhấn vào và xem chi tiết sách, nếu có nhu cầu đọc thì gửi yêu cầu liên hệ tới Admin bằng form liên hệ.

**Giao diện Admin:**

**• Quản lý sách:**

* + Thêm, sửa, xóa sách: Quản lý toàn bộ thông tin sách trong thư viện như: tên sách, mã sách, tác giả, nhà xuất bản, thể loại, số lượng, vị trí, năm xuất bản...
  + Danh sách sách: Hiển thị sách dưới dạng danh sách có hình ảnh, tiêu đề và thông tin cơ bản.
  + Bộ lọc sách theo danh mục: Lọc theo thể loại sách như: Văn học, Khoa học, Lịch sử, Kinh tế, Công nghệ…
  + Tìm kiếm sách: Tìm kiếm nhanh theo tên sách, mã sách hoặc tên tác giả.
  + Xem chi tiết sách: Khi nhấn vào một cuốn sách sẽ hiển thị chi tiết thông tin, tình trạng mượn, mô tả...

**• Quản lý mượn – trả:**

* Tạo phiếu mượn: Nhập thông tin người mượn, ngày mượn, ngày dự kiến trả và danh sách sách mượn.
* Ghi nhận trả sách: Xác nhận trả sách, cập nhật lại trạng thái sách trong hệ thống.
* Tìm kiếm phiếu mượn/trả: Tra cứu theo mã phiếu, tên người mượn, ngày mượn hoặc hạn trả.
* Chi tiết phiếu: Hiển thị thông tin từng phiếu bao gồm danh sách sách đã mượn và tình trạng từng quyển.

**• Báo Cáo Thống kê (Trang chủ)**

* Thống kê: Tài khoản hiện có trên hệ thống, tổng số sách hiện có trong hệ thống, số lượt mượn sách theo tháng
* Thống kê: Số lượng mượn sách theo tháng bằng biểu đồ cột và thông tin bảng thống kê mượn sách gần nhất.

**• Quản lý người dùng (tài khoản):**

* Thông tin chi tiết thông tin tài khoản hiện có trong dữ liệu người dùng

**• Quản lý đăng ký thẻ thư viện**

* Cung cấp các thông tin cần thiết: Họ tên, ngày sinh, giới tính, địa chỉ, số điện thoại, email, ảnh chân dung (tùy chọn).
* Danh sách thẻ đã đăng ký hiển thị dưới dạng bảng: mã thẻ, tên độc giả, ngày cấp, tình trạng (còn hiệu lực/hết hạn).
* Tìm kiếm: Cho phép tìm kiếm thẻ theo mã thẻ hoặc tên độc giả.
* Cập nhật: Có thể sửa thông tin nếu có sai sót hoặc cập nhật khi độc giả thay đổi địa chỉ, số điện thoại...
* Kiểm tra: Khi thực hiện mượn/trả sách, hệ thống kiểm tra thẻ còn hiệu lực hay không.
* Tích hợp với chức năng mượn/trả sách:
* Mỗi giao dịch mượn/trả sách đều gắn với mã thẻ thư viện để đảm bảo tính chính xác và lịch sử mượn rõ ràng.

**• Quản lý thể loại sách**

* Nhập tên thể loại (ví dụ: Văn học, Lịch sử, Khoa học, Công nghệ, Kinh tế...).
* Có thể bổ sung mô tả chi tiết cho mỗi thể loại để dễ nhận biết và phân biệt.
* Danh sách tất cả thể loại được hiển thị dưới dạng bảng: mã thể loại, tên thể loại, mô tả (nếu có), số lượng sách thuộc thể loại đó.
* Hỗ trợ tìm kiếm nhanh thể loại theo tên.
* Cho phép sửa đổi tên hoặc mô tả của thể loại trong trường hợp cần thay đổi.
* Có thể xóa thể loại nếu không còn sử dụng.
* Hệ thống kiểm tra nếu thể loại đang có sách liên quan thì không cho phép xóa để đảm bảo tính toàn vẹn dữ liệu.
* Mỗi cuốn sách khi thêm vào hệ thống sẽ được gắn với một thể loại cụ thể.
* Hệ thống cho phép lọc, thống kê sách theo thể loại.

**• Quản lý phiếu phạt**

* Bạn đọc trả sách quá hạn, sách bị mất hoặc hư hỏng khi trả lại.
* Hiển thị thông tin gồm: Mã phiếu phạt, tên người mượn, ngày lập phiếu, lý do, số tiền phạt, trạng thái.
* Cập nhật: Thêm, xóa, sửa phiếu phạt

**• Thoát**

* Khi Admin ở chế độ quản lý bấm thoát sẽ trở về trang chính

**2.1.4 Hoạt động nghiệp vụ**

**1) Quản lý Sách:**

* Quản lý sách trong hệ thống thư viện bao gồm các hoạt động quản lý thông tin chi tiết về từng cuốn sách như tên, tác giả, nhà xuất bản, năm xuất bản, số lượng sách có sẵn, và trạng thái của sách (đang mượn, có sẵn). Các hoạt động chính bao gồm:
* Thêm sách: Nhân viên thư viện có thể thêm sách mới vào hệ thống, cung cấp đầy đủ thông tin như tên sách, tác giả, nhà xuất bản, năm xuất bản, thể loại, số lượng, và mã vạch.
* Cập nhật thông tin sách: Cập nhật thông tin khi có sự thay đổi, ví dụ như thay đổi giá, bổ sung sách mới, hoặc cập nhật trạng thái của sách.
* Xóa sách: Xóa các sách không còn trong kho hoặc không được sử dụng nữa.
* Tìm kiếm sách: Cung cấp chức năng tìm kiếm sách theo nhiều tiêu chí như tên sách, tác giả, thể loại, trạng thái mượn, v.v.

**2) Quản lý Thể loại:**

Quản lý thể loại giúp phân loại sách thành các nhóm dựa trên các tiêu chí nhất định (ví dụ: khoa học, văn học, công nghệ, lịch sử…). Việc này giúp thư viện dễ dàng tìm kiếm và phân loại sách theo từng thể loại cụ thể. Các hoạt động chính bao gồm:

* Thêm thể loại: Cung cấp chức năng tạo mới các thể loại sách như: văn học, khoa học, lịch sử, kỹ thuật, v.v.
* Cập nhật thể loại: Cập nhật tên thể loại hoặc thông tin liên quan nếu cần thiết.
* Xóa thể loại: Xóa các thể loại không còn được sử dụng hoặc không còn sách nào thuộc thể loại đó.
* Gán thể loại cho sách: Mỗi cuốn sách có thể được phân loại theo một hoặc nhiều thể loại để dễ dàng tra cứu.

**3) Quản lý Thẻ Thư viện:**

Thẻ thư viện là công cụ để xác nhận danh tính của độc giả khi mượn sách. Quản lý thẻ thư viện bao gồm việc cấp phát, cập nhật và thu hồi thẻ của độc giả. Các hoạt động chính bao gồm:

* Cấp thẻ thư viện: Khi một người muốn sử dụng dịch vụ thư viện, hệ thống sẽ cấp cho họ một thẻ thư viện với mã thẻ duy nhất.
* Cập nhật thông tin thẻ: Cập nhật thông tin cá nhân của người dùng như tên, địa chỉ, số điện thoại khi có sự thay đổi.
* Xóa hoặc thu hồi thẻ: Xóa thẻ khi độc giả không còn sử dụng dịch vụ hoặc thẻ bị hỏng.
* Kiểm tra trạng thái thẻ: Kiểm tra các thẻ đang hoạt động, đã hết hạn hoặc bị khóa.

**4) Quản lý Mượn Trả Sách:**

Quản lý mượn trả sách là một phần quan trọng trong việc theo dõi sách đã được mượn và trả lại. Các hoạt động chính bao gồm:

* Mượn sách: Khi độc giả có thẻ thư viện hợp lệ, họ có thể mượn sách. Thông tin mượn bao gồm mã thẻ, mã sách, ngày mượn và ngày trả dự kiến.
* Trả sách: Khi độc giả trả sách, hệ thống sẽ ghi nhận ngày trả, tính toán phí trễ hạn nếu có, và cập nhật trạng thái của sách (có sẵn để mượn).
* Gia hạn mượn: Nếu độc giả cần thêm thời gian, hệ thống có thể cho phép gia hạn thời gian mượn, miễn là sách không có người yêu cầu mượn.
* Theo dõi tình trạng sách: Giám sát trạng thái sách có sẵn hoặc đang mượn, từ đó giúp thư viện quản lý tốt hơn các cuốn sách trong kho.

**5) Quản lý Phiếu Phạt:**

Quản lý phiếu phạt xử lý các trường hợp vi phạm quy định của thư viện, như việc trả sách trễ hạn, hỏng sách. Các hoạt động chính bao gồm:

* Lập phiếu phạt: Khi độc giả trả sách trễ hạn, hệ thống sẽ tự động lập phiếu phạt, ghi nhận thông tin như mã thẻ, sách bị phạt, số tiền phạt, lý do phạt (trễ hạn, làm hỏng sách, mất sách, v.v.).
* Cập nhật phiếu phạt: Nếu có thay đổi liên quan đến phiếu phạt (ví dụ: giảm phí phạt hoặc hủy bỏ), hệ thống sẽ cập nhật lại thông tin.
* Thu tiền phạt: Khi độc giả thanh toán phí phạt, hệ thống sẽ ghi nhận số tiền đã thanh toán và cập nhật tình trạng phiếu phạt.
* Lịch sử phạt: Lưu trữ lịch sử các phiếu phạt của từng độc giả để có thể dễ dàng tra cứu và theo dõi các vi phạm trước đó.

**6) Quản lý người dùng:**

* Tìm kiếm: Cho phép tìm kiếm người dùng theo email hoặc tên để giúp truy vấn dữ liệu nhanh hơn, đỡ tốn thời gian.
* Thêm người dùng mới: Cho phép thêm người dùng vào hệ thống và cấp quyền cho tài khoản.

**7) Báo cáo thống kê:**

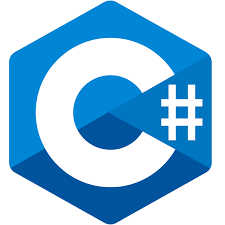
* Tổng quan số liệu: Giao diện hiển thị các số liệu tổng hợp giúp quản trị viên nhanh chóng nắm bắt tình hình hoạt động của thư viện
* Thống kê mượn sách theo tháng: Biểu đồ cột trực quan hóa số lượt mượn sách trong các tháng gần đây, Giúp phân tích xu hướng mượn sách, đánh giá mức độ sử dụng thư viện theo thời gian.

## **2.2 Công nghệ sử dụng**

**2.2.1 Giới thiêu về C#**

C# (C Sharp) là một ngôn ngữ lập trình hướng đối tượng được phát triển bởi Microsoft, nằm trong khuôn khổ .NET Framework. Ngôn ngữ này không chỉ dễ học mà còn mạnh mẽ và linh hoạt, cho phép lập trình viên phát triển một loạt các ứng dụng khác nhau, từ ứng dụng desktop cho đến web và mobile. C# hỗ trợ nhiều tính năng lập trình hiện đại như lập trình bất đồng bộ (asynchronous programming), LINQ (Language Integrated Query), và các nguyên tắc của lập trình hướng đối tượng như kế thừa, đa hình và trừu tượng, giúp tăng cường khả năng tái sử dụng mã và bảo trì ứng dụng.

C# thường được sử dụng để phát triển các ứng dụng yêu cầu hiệu suất cao và tính ổn định, như các ứng dụng doanh nghiệp. Trong bối cảnh xây dựng hệ thống quản lý bán hàng cho cửa hàng tiện lợi, C# sẽ được sử dụng để phát triển các logic nghiệp vụ của ứng dụng, xử lý dữ liệu và giao diện người dùng, cũng như tích hợp với cơ sở dữ liệu. Sự dễ đọc, cú pháp rõ ràng và khả năng bảo trì cao của C# làm cho nó trở thành một sự lựa chọn lý tưởng cho các dự án phát triển phần mềm phức tạp, giúp đội ngũ phát triển tiết kiệm thời gian và nâng cao hiệu quả công việc.



Hình : Logo C#

**2.2.2 Giới thiệu về SQL Server**

SQL Server là một hệ quản trị cơ sở dữ liệu (DBMS) phát triển bởi Microsoft, được thiết kế để quản lý, lưu trữ và truy xuất dữ liệu một cách hiệu quả. SQL Server hỗ trợ nhiều tính năng mạnh mẽ như khả năng bảo mật cao, hiệu suất tối ưu, và khả năng mở rộng cho các ứng dụng quy mô lớn. Nó có thể xử lý khối lượng dữ liệu lớn và cung cấp khả năng phục hồi sau sự cố, đảm bảo rằng dữ liệu luôn an toàn và có sẵn.

Hệ thống này sử dụng ngôn ngữ truy vấn SQL (Structured Query Language) để thực hiện các thao tác trên dữ liệu, bao gồm thêm, sửa, xóa và truy vấn dữ liệu. Trong hệ thống quản lý bán hàng cho cửa hàng tiện lợi, SQL Server sẽ được sử dụng để lưu trữ và quản lý tất cả các dữ liệu liên quan đến sản phẩm, khách hàng, giao dịch bán hàng, và nhân viên. Việc sử dụng SQL Server không chỉ đảm bảo tính toàn vẹn và bảo mật cho dữ liệu mà còn cho phép thực hiện các báo cáo và phân tích dữ liệu dễ dàng. Điều này sẽ giúp quản lý đưa ra các quyết định kinh doanh kịp thời và chính xác, từ đó tối ưu hóa quy trình bán hàng và nâng cao hiệu suất làm việc.



Hình : Logo SQL Server

**2.2.3 Giới thiệu về Visual Studio**

Visual Studio là một môi trường phát triển tích hợp (IDE) mạnh mẽ do Microsoft phát triển, hỗ trợ lập trình viên trong việc phát triển ứng dụng trên nền tảng .NET. Visual Studio cung cấp một loạt các công cụ và tính năng như trình biên dịch, gỡ lỗi (debugger), trình quản lý dự án, và nhiều mẫu dự án sẵn có, giúp lập trình viên nhanh chóng bắt đầu và triển khai ứng dụng. Nó cũng hỗ trợ tích hợp với các công cụ kiểm soát phiên bản như Git, giúp quản lý mã nguồn hiệu quả hơn.

Môi trường này hỗ trợ nhiều ngôn ngữ lập trình, bao gồm C#, VB.NET, và F#, cho phép lập trình viên phát triển các ứng dụng web, desktop, và mobile trong cùng một nền tảng. Với giao diện người dùng thân thiện và nhiều tính năng hỗ trợ, Visual Studio giúp tiết kiệm thời gian và công sức cho lập trình viên. Trong dự án xây dựng hệ thống quản lý bán hàng cho cửa hàng tiện lợi, Visual Studio sẽ được sử dụng để phát triển giao diện người dùng, xử lý các logic nghiệp vụ và kết nối với cơ sở dữ liệu SQL Server. Sự tích hợp tốt giữa C# và SQL Server trong Visual Studio giúp tăng cường hiệu quả phát triển và giảm thiểu thời gian triển khai ứng dụng, đồng thời tạo điều kiện thuận lợi cho việc gỡ lỗi và kiểm tra.



Hình : Logo Visual Studio

**2.2.4 Giới thiệu về MVC và mô hình thiết kế Repository Pattern**

MVVM, Entity Framework và Repository Pattern: Bộ ba hoàn hảo

MVVM (Model-View-ViewModel):

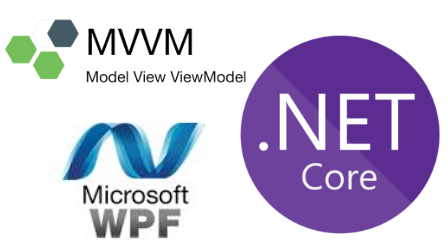
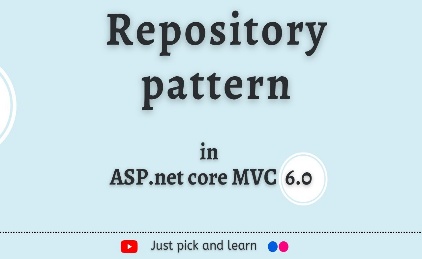
- Model: Đại diện cho dữ liệu của ứng dụng, thường được ánh xạ với các bảng trong cơ sở dữ liệu, thực hiện các thao tác với dữ liệu như thêm, sửa, xóa, tìm kiếm.

- View: Là giao diện người dùng, nơi người dùng tương tác. View chỉ chịu trách nhiệm hiển thị và nhận các sự kiện từ người dùng, không chứa logic xử lý dữ liệu.

- ViewModel: Là lớp trung gian, kết nối Model và View, xử lý các logic liên quan đến giao diện và thực hiện các lệnh (commands). ViewModel cung cấp dữ liệu từ Model cho View và thực hiện các thao tác thay đổi dữ liệu qua Model.

- Entity Framework: Entity Framework là một ORM (Object-Relational Mapper) giúp bạn làm việc với cơ sở dữ liệu một cách trực quan bằng cách sử dụng các đối tượng C#. Nó tự động ánh xạ giữa các đối tượng C# (entities) và các bảng trong cơ sở dữ liệu, giúp giảm thiểu việc viết mã SQL thủ công và dễ dàng thực hiện các thao tác CRUD.

- Repository Pattern: Repository Pattern là một mẫu thiết kế được sử dụng để cách ly logic truy cập dữ liệu khỏi phần còn lại của ứng dụng. Thay vì trực tiếp tương tác với Entity Framework trong các lớp ViewModel hoặc Controller, bạn tạo ra một lớp Repository để thực hiện tất cả các thao tác CRUD và quản lý dữ liệu. Điều này giúp mã nguồn trở nên dễ bảo trì và dễ kiểm thử hơn.



Hình : Logo MVVM, Entity FrameWork, và Repository Pattern

**2.2.5 Giới thiệu về HTML**

HTML (Hypertext Markup Language) là ngôn ngữ đánh dấu được sử dụng để tạo ra các trang web. Nó cung cấp cấu trúc cơ bản cho các tài liệu web và cho phép trình duyệt hiểu và hiển thị nội dung cho người dùng. HTML sử dụng các thẻ (tags) để xác định các phần tử khác nhau trên trang web như văn bản, hình ảnh, liên kết, bảng, và các thành phần giao diện khác.



Hình 5: Logo HTML

**2.2.6 Giới thiệu về Css**

CSS (Cascading Style Sheets) là ngôn ngữ được sử dụng để định dạng và trang trí giao diện của các trang web. CSS cho phép bạn kiểm soát cách thức hiển thị của các phần tử HTML trên trang, bao gồm màu sắc, phông chữ, bố cục, khoảng cách, và các hiệu ứng đặc biệt khác.



Hình 6: Logo CSS

**2.2.7 Giới thiệu về JavaScript**

JavaScript (JS) là một ngôn ngữ lập trình phổ biến và mạnh mẽ được sử dụng để phát triển các trang web động và tương tác. Nó là một trong ba công nghệ chính (HTML, CSS và JavaScript) được sử dụng để xây dựng và phát triển trang web, với HTML định nghĩa cấu trúc và CSS định dạng giao diện, trong khi JavaScript cung cấp khả năng tương tác và động cho các trang web.

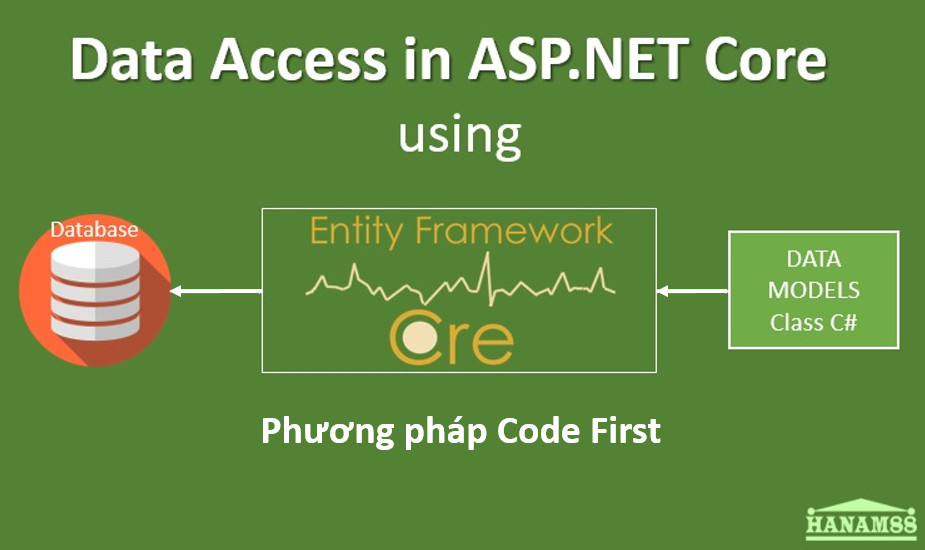


Hình 7: Logo JavaScript

**2.2.8 Giới thiệu về Bootstrap**

Bootstrap là một framework mã nguồn mở được phát triển bởi Twitter và hiện nay là một trong những công cụ phổ biến nhất giúp xây dựng giao diện web nhanh chóng và dễ dàng. Nó cung cấp một bộ sưu tập các công cụ và thành phần sẵn có để tạo giao diện người dùng đẹp mắt, đáp ứng, và tương thích với tất cả các loại thiết bị (responsive design).

Hình 8: Logo Bootstrap

**2.2.9 Giới thiệu về SQL Code First**

Hình 9: Code First

## **2.3 Khảo sát hệ thống**

Khảo sát hệ thống là bước đầu tiên và rất quan trọng trong quá trình phân tích và thiết kế hệ thống quản lý thư viện. Mục tiêu của khảo sát này là hiểu rõ các yêu cầu, quy trình nghiệp vụ hiện tại và các vấn đề mà hệ thống hiện tại đang gặp phải. Dưới đây là những khía cạnh chính được khảo sát trong hệ thống quản lý thư viện:

* **Xác định yêu cầu người dùng:**

Để phát triển một hệ thống đáp ứng nhu cầu thực tế, việc phỏng vấn và thu thập ý kiến từ các bên liên quan, bao gồm quản lý thư viện, thủ thư, nhân viên, và bạn đọc là rất cần thiết. Những câu hỏi sẽ tập trung vào các chức năng mà họ mong muốn từ hệ thống, cũng như những khó khăn mà họ đang gặp phải với các phương pháp quản lý hiện tại.

Thông qua khảo sát, người dùng có thể đưa ra những mong muốn cụ thể về các chức năng như quản lý sách, quản lý thẻ thư viện, mượn trả sách, phiếu phạt, và báo cáo thống kê. Việc này giúp đảm bảo rằng hệ thống phát triển sẽ phục vụ đúng nhu cầu thực tế của người dùng và cải thiện trải nghiệm quản lý thư viện.

* **Đánh giá hệ thống hiện tại:**

Khảo sát sẽ tiến hành đánh giá hiệu suất của hệ thống hiện tại (nếu có), xác định những điểm mạnh và điểm yếu của nó. Các vấn đề thường gặp có thể bao gồm quy trình mượn trả sách thủ công, thiếu chính xác trong việc quản lý sách, và khó khăn trong việc truy xuất dữ liệu khi cần báo cáo thống kê.

Việc phân tích các vấn đề này sẽ giúp xác định các tính năng cần cải thiện trong hệ thống mới, đồng thời cung cấp một cái nhìn tổng quan về các quy trình làm việc hiện tại, từ đó xác định những thay đổi cần thiết để nâng cao hiệu quả hoạt động và quản lý.

* **Phân tích quy trình nghiệp vụ:**

Khảo sát sẽ tiếp tục phân tích các quy trình nghiệp vụ cụ thể của thư viện, chẳng hạn như quy trình mượn sách, trả sách, gia hạn thẻ, phát phiếu phạt, và kiểm kê sách. Điều này giúp hiểu rõ cách mà các quy trình này đang hoạt động và phát hiện các điểm nghẽn hoặc thiếu sót có thể tồn tại.

Mỗi quy trình nghiệp vụ sẽ được ghi chép chi tiết, từ các bước thực hiện, các bên liên quan, cho đến các tài liệu hỗ trợ cần thiết. Điều này sẽ tạo ra cơ sở dữ liệu cho việc thiết kế lại quy trình trong hệ thống mới, nhằm nâng cao hiệu quả và tiết kiệm thời gian.

* **Khảo sát công nghệ và cơ sở hạ tầng hiện tại:**

Một phần quan trọng trong khảo sát hệ thống là đánh giá công nghệ hiện tại mà thư viện đang sử dụng, bao gồm phần mềm quản lý thư viện, phần cứng, và mạng lưới. Việc xác định các công nghệ hiện có và khả năng tích hợp của chúng với hệ thống mới sẽ giúp lên kế hoạch cho việc triển khai và đào tạo.

Đánh giá cơ sở hạ tầng công nghệ thông tin hiện tại cũng giúp xác định xem có cần đầu tư thêm thiết bị hoặc phần mềm nào để hỗ trợ cho hệ thống quản lý thư viện mới hay không.

* **Xác định các yêu cầu phi chức năng:**

Ngoài các yêu cầu chức năng, khảo sát cũng cần xem xét các yêu cầu phi chức năng như hiệu suất, khả năng mở rộng, tính bảo mật và khả năng sử dụng. Những yêu cầu này có thể ảnh hưởng lớn đến thiết kế hệ thống và trải nghiệm của người dùng.

Ví dụ, cần xác định mức độ bảo mật cần thiết để bảo vệ thông tin người đọc và dữ liệu giao dịch, cũng như yêu cầu về thời gian phản hồi của hệ thống khi xử lý các giao dịch mượn trả sách hoặc cập nhật dữ liệu.

## **2.4 Yêu cầu hệ thống**

**2.4.1 Yêu cầu chức năng**

a. Quản lý thông tin sách: Hệ thống phải cho phép thêm, sửa, xóa thông tin chi tiết của sách, bao gồm tên sách, tác giả, thể loại, mô tả, và số lượng sách có sẵn trong thư viện. Hệ thống cần hỗ trợ tìm kiếm sách theo nhiều tiêu chí như tên sách, tác giả, thể loại.

b. Quản lý mượn và trả sách: Hệ thống phải cho phép ghi nhận thông tin mượn sách, bao gồm tên người mượn, danh sách sách mượn, ngày mượn và ngày dự kiến trả. Tự động cập nhật tình trạng sách và số lượng sách mượn. Hệ thống cần quản lý quy trình trả sách, ghi nhận thông tin về ngày trả, tình trạng sách khi trả và nhân viên thực hiện giao dịch. Cập nhật lại số lượng sách có sẵn trong thư viện.

c. Quản lý khách hàng và thẻ thư viện: Hệ thống phải cho phép tạo và quản lý thông tin khách hàng (thư viện viên), bao gồm tên, địa chỉ, số điện thoại, email và lịch sử mượn sách. Quản lý thẻ thư viện cho khách hàng, bao gồm việc cấp thẻ mới, gia hạn thẻ, và theo dõi lịch sử sử dụng thẻ. Hệ thống cần cung cấp các chương trình khuyến mãi, ưu đãi cho khách hàng thân thiết (ví dụ: miễn phí gia hạn thẻ cho người dùng lâu năm).

d. Quản lý phiếu phạt: Hệ thống phải ghi nhận các phiếu phạt khi khách hàng trả sách muộn hoặc làm hỏng sách. Bao gồm thông tin về sách vi phạm, số tiền phạt, và lý do phạt. Hệ thống cần tự động tính toán và thông báo số tiền phạt khi khách hàng vi phạm quy định mượn sách.

e. Báo cáo và phân tích: Hệ thống cần cung cấp các báo cáo tổng hợp về số lượng sách mượn, sách trả muộn, phiếu phạt và tình hình mượn trả sách theo ngày, tuần, và tháng. Cung cấp các báo cáo chi tiết về các sách đang được mượn nhiều nhất, các khách hàng sử dụng dịch vụ thường xuyên và hiệu suất sử dụng thư viện.

**2.4.2 Yêu cầu phi chức năng**

a. Hiệu suất: Hệ thống phải đảm bảo thời gian phản hồi nhanh chóng khi người dùng thực hiện các thao tác như tìm kiếm sách, mượn trả sách, tạo báo cáo, và xử lý phiếu phạt. Thời gian phản hồi không được vượt quá 2 giây cho các thao tác cơ bản.

b. Tính bảo mật: Hệ thống cần bảo vệ thông tin nhạy cảm của khách hàng (thư viện viên) và dữ liệu sách bằng cách sử dụng các phương thức mã hóa dữ liệu và xác thực người dùng. Phân quyền truy cập theo vai trò để đảm bảo chỉ những người có quyền mới có thể truy cập và thay đổi dữ liệu quan trọng, như quản lý sách và phiếu phạt.

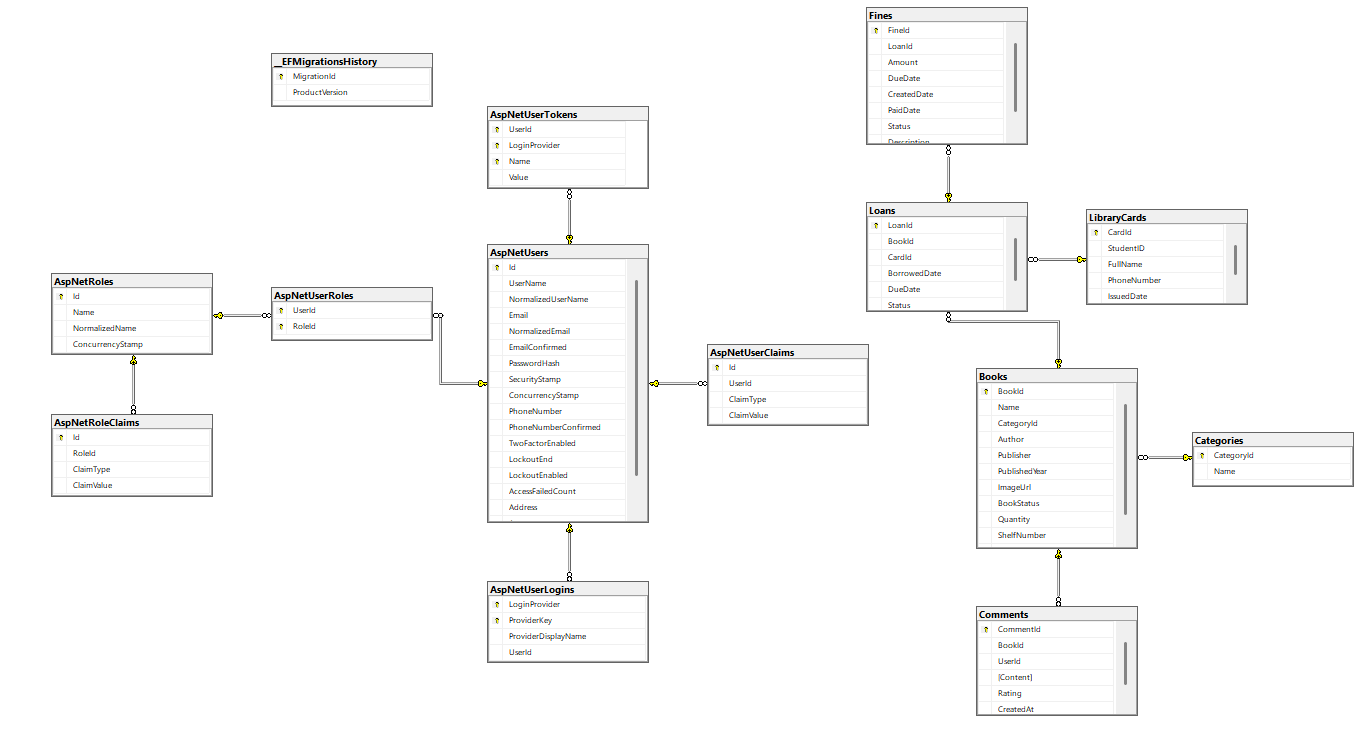
c. Khả năng mở rộng: Hệ thống cần có khả năng mở rộng để đáp ứng nhu cầu gia tăng trong tương lai, chẳng hạn như số lượng sách và người dùng. Cần thiết kế hệ thống sao cho dễ dàng cập nhật và thêm mới các chức năng mà không ảnh hưởng đến hiệu suất tổng thể, ví dụ: bổ sung thêm tính năng cho phép mượn sách trực tuyến.

d. Tính khả dụng: Hệ thống phải đảm bảo tính khả dụng cao, với thời gian hoạt động (uptime) đạt ít nhất 99%. Cần có kế hoạch sao lưu và phục hồi dữ liệu để giảm thiểu rủi ro mất mát dữ liệu trong trường hợp sự cố xảy ra.

e. Tính sử dụng: Giao diện người dùng cần được thiết kế đơn giản, trực quan và thân thiện, giúp người dùng dễ dàng thực hiện các thao tác mà không cần đào tạo nhiều. Hệ thống cũng nên cung cấp hướng dẫn sử dụng và hỗ trợ kỹ thuật khi cần.

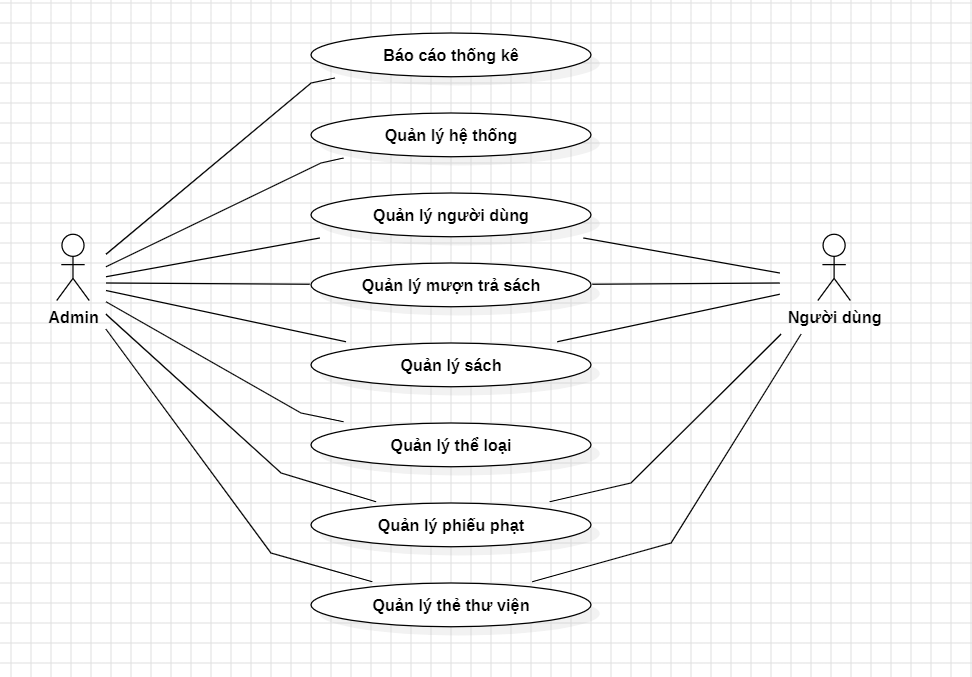
f. Khả năng tích hợp: Hệ thống phải có khả năng tích hợp với các phần mềm khác (nếu cần), chẳng hạn như hệ thống quản lý khách hàng hoặc các hệ thống học tập trực tuyến (nếu có), để cải thiện quy trình làm việc và tối ưu hóa dữ liệu giữa các hệ thống.

## **2.5 Xây dựng sơ đồ Class Diagram**

****

Hình 10. Sơ đồ RD

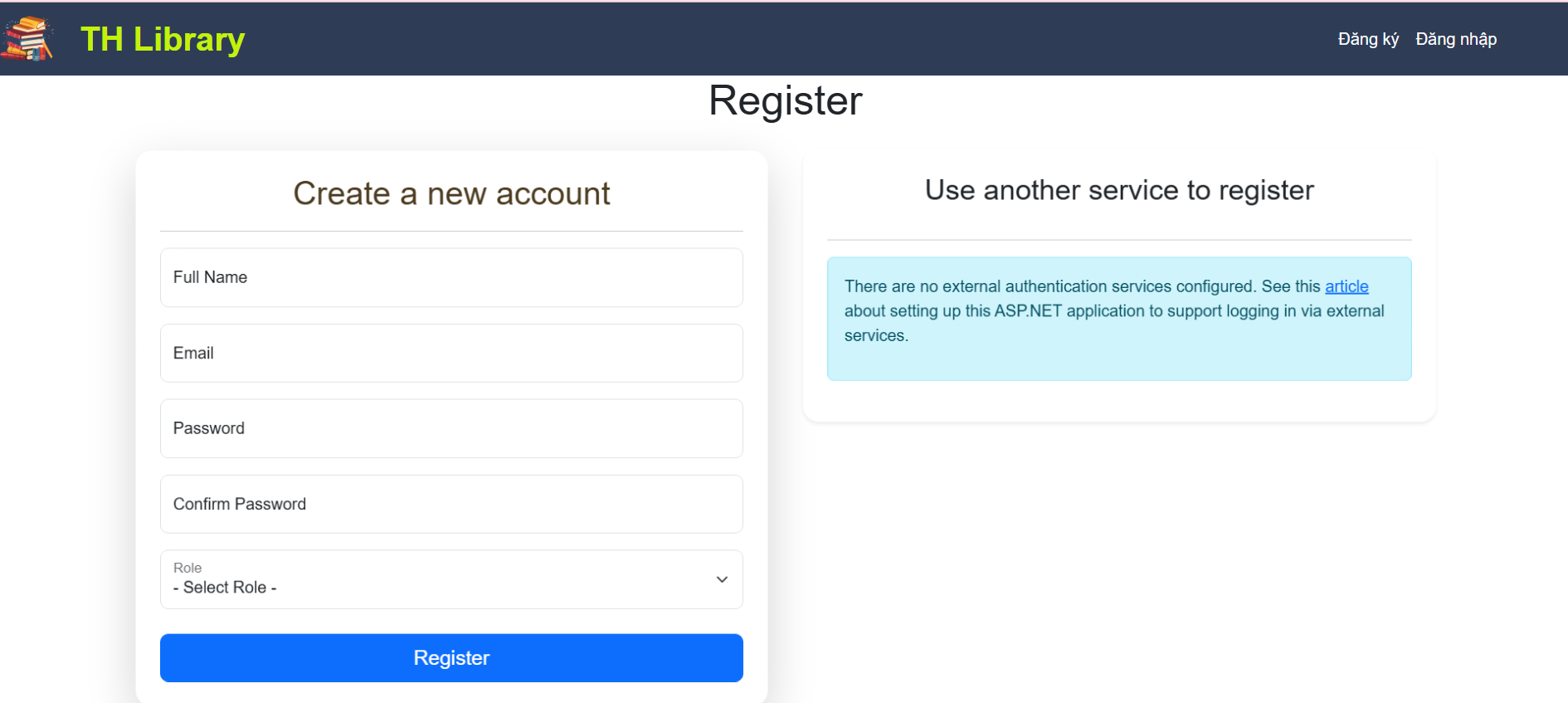
## **2.6 Sơ đồ Use Case**

****

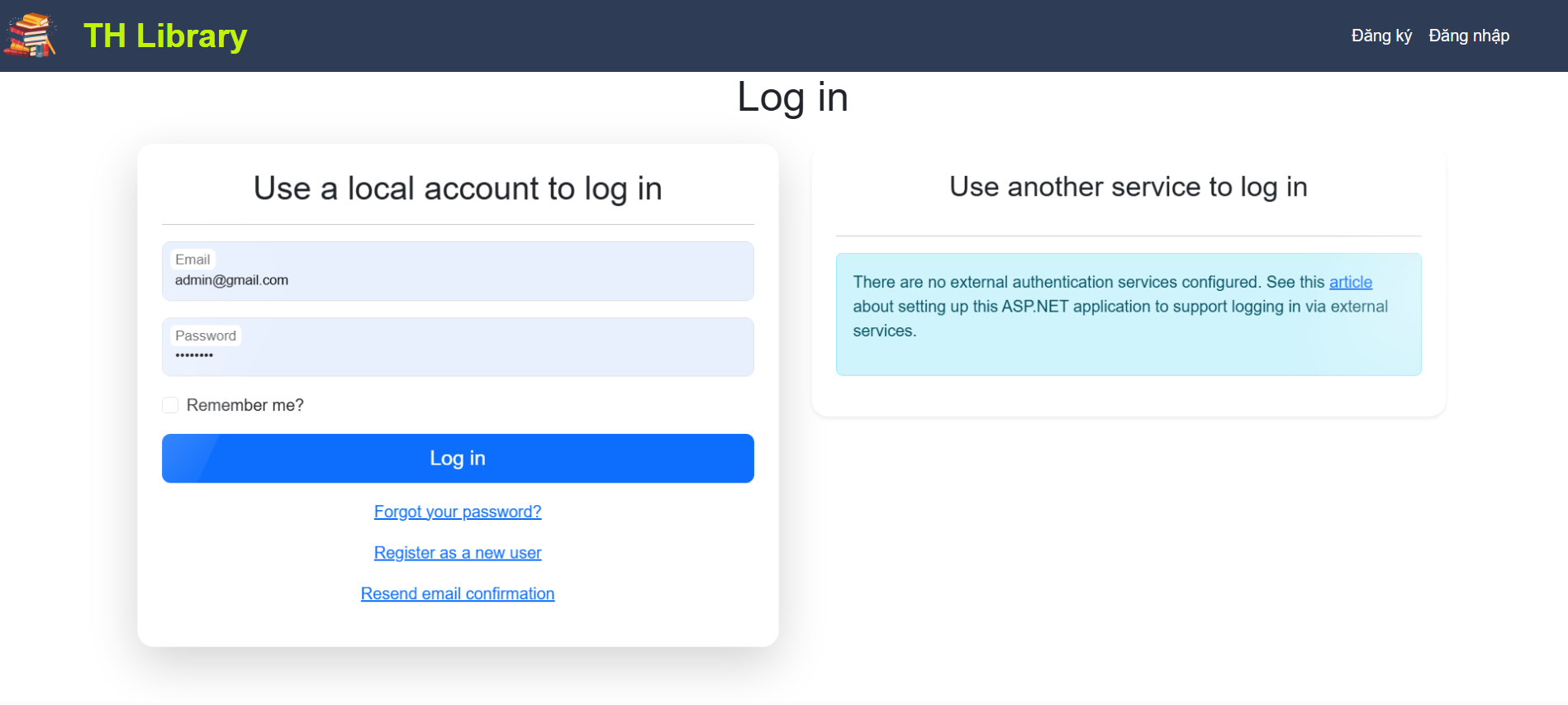
# **CHƯƠNG 3: HIỆN THỰC ỨNG DỤNG**

## **3.1 Giao diện đăng ký, đăng nhập**

Chức năng giao diện: Cho phép đăng ký, đăng nhập tài khoản theo phân quyền Admin hoặc người dùng.



Hình 11: Đăng Ký



Hình 12: Đăng Nhập

## **3.2 Giao diện người dùng**

**3.2.1 Trang chủ**

Chức năng giao diện: Hiện thị thông tin toàn bộ trang chủ của người dùng



Hình 13. Giao diện trang chủ

**3.2.1 Giới thiệu**

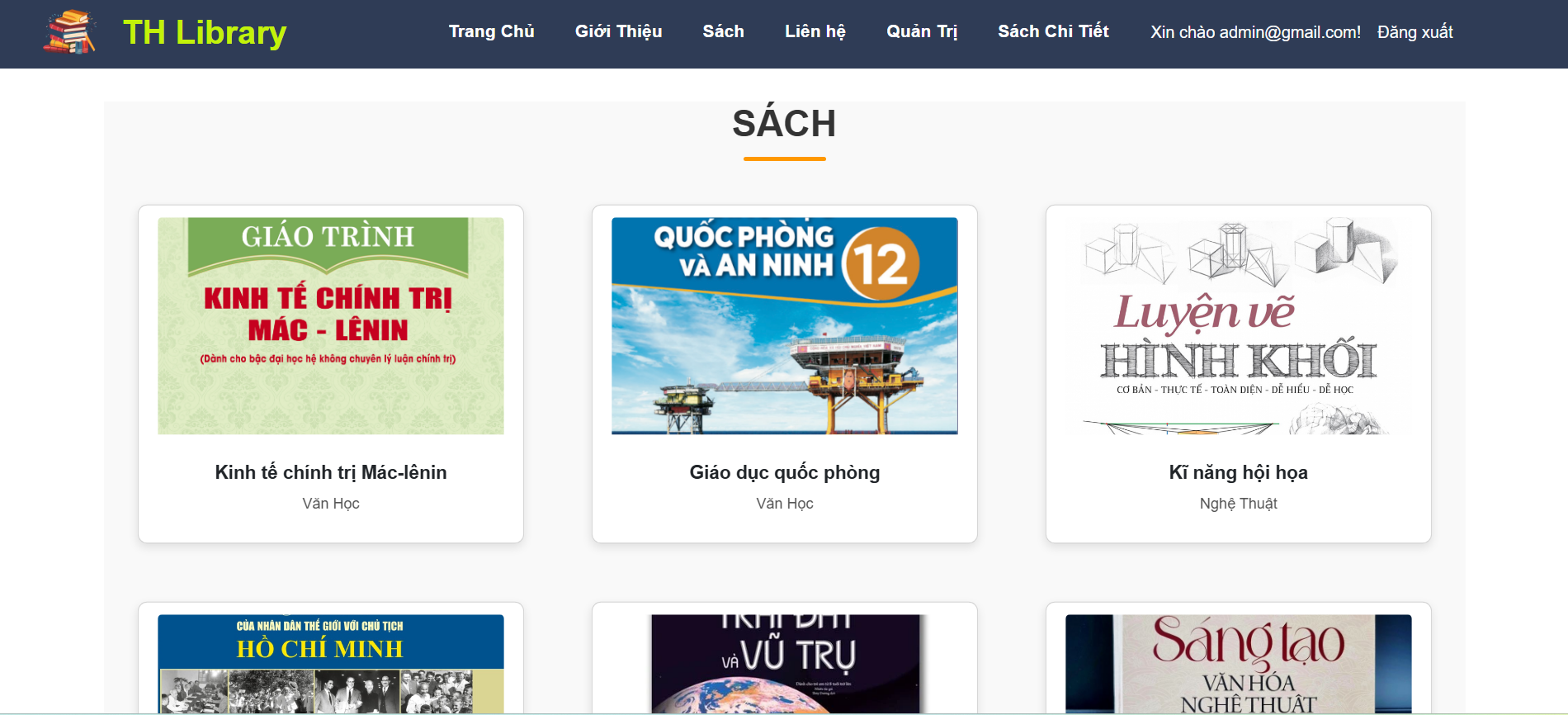
Chức năng giao diện: Giới thiệu sơ bộ về hệ thống quản lý thư viện

****

Hình 14. Giao diện giới thiệu hệ thống thư viện

**3.2.2 Sách**

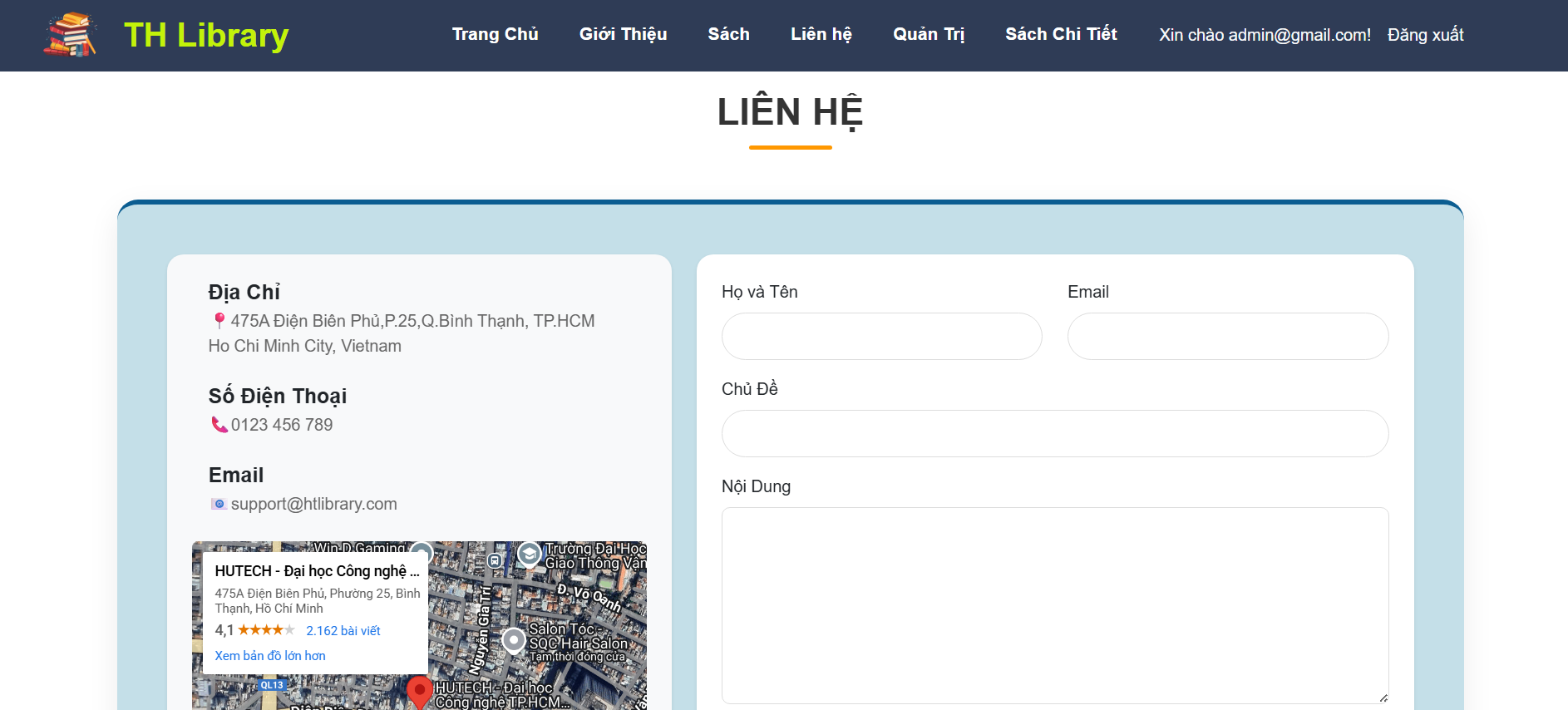
Chức năng giao diện: Hiện thị thông tin những cuốn sách nổi bật được nhiều lượt đọc nhất, được trưng bày.



Hình 15. Giao diện Sách

**3.2.3 Liên hệ**

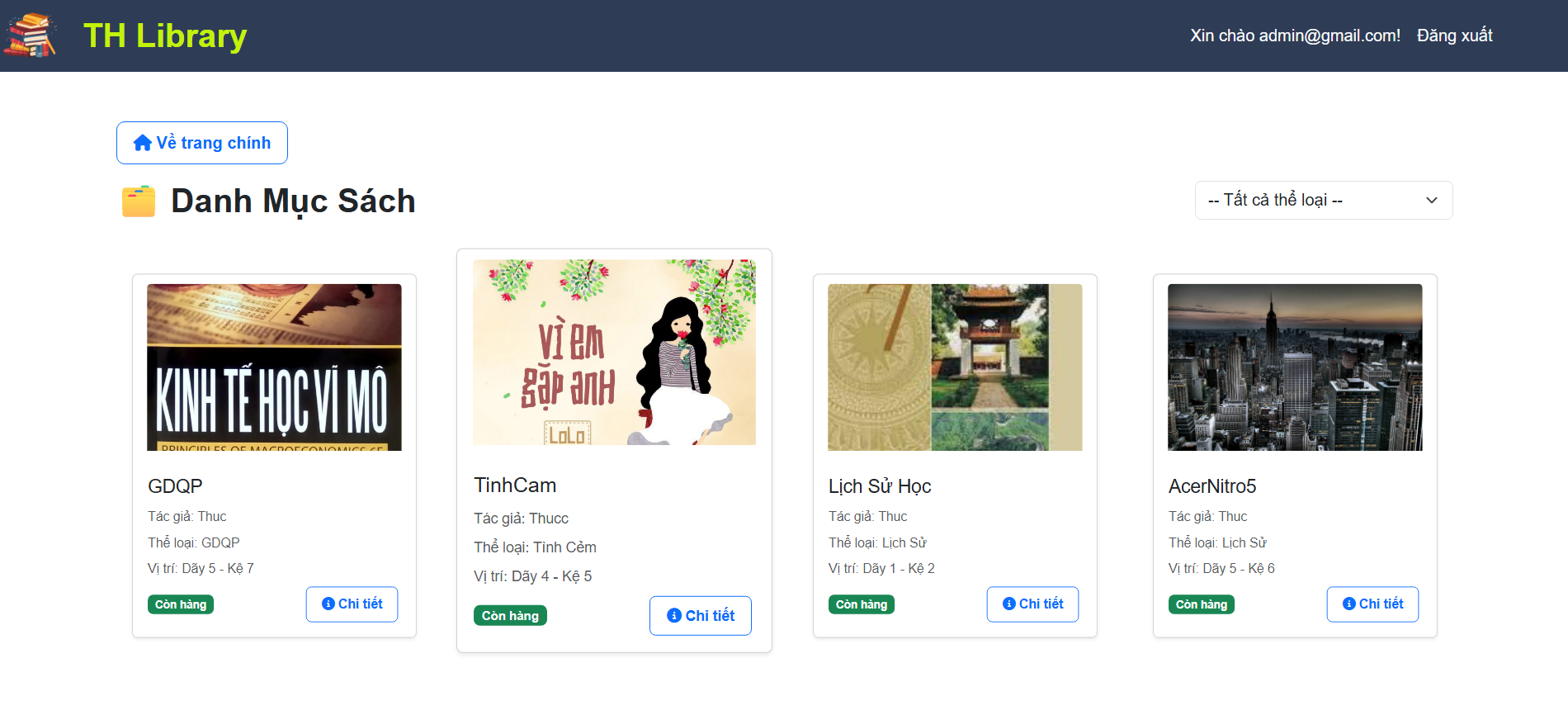
Chức năng giao diện: Giúp người dùng có thể liên hệ với Admin nếu muốn mượn sách

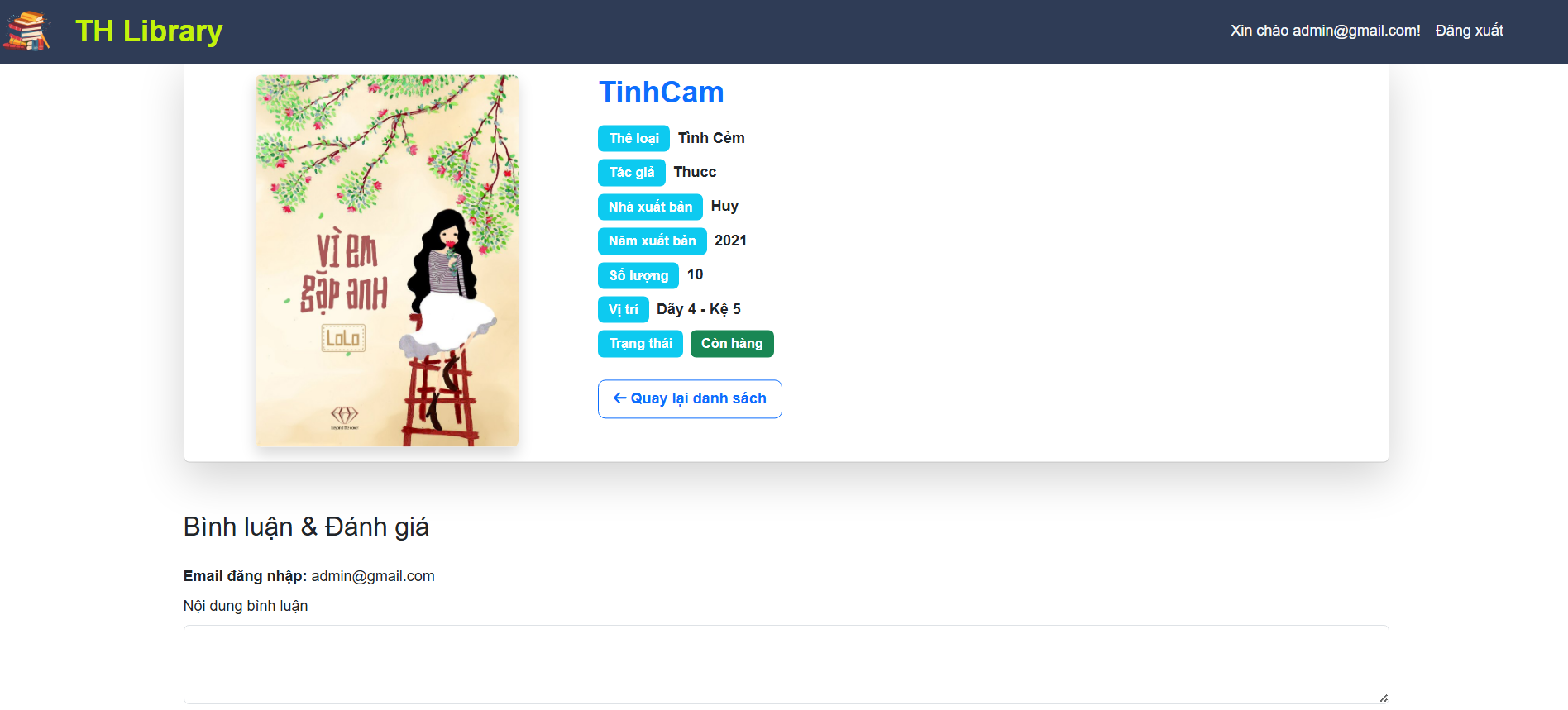
****

Hình 16. Giao diện liên hệ

**3.2.4 Sách chi tiết**

Chức năng giao diện: Liệt kê toàn bộ sách đang có ở thư viện, cho biết sách đang mượn hay chưa được mượn, và cho phép tìm kiếm theo thể loại sách, cho phép người dùng đánh giá sách.





Hình 17. Giao diện Sách chi tiết

**3.2.5 Quản trị**

Chức năng giao diện: Chỉ duy nhất Admin mới có thể truy cập quyền này

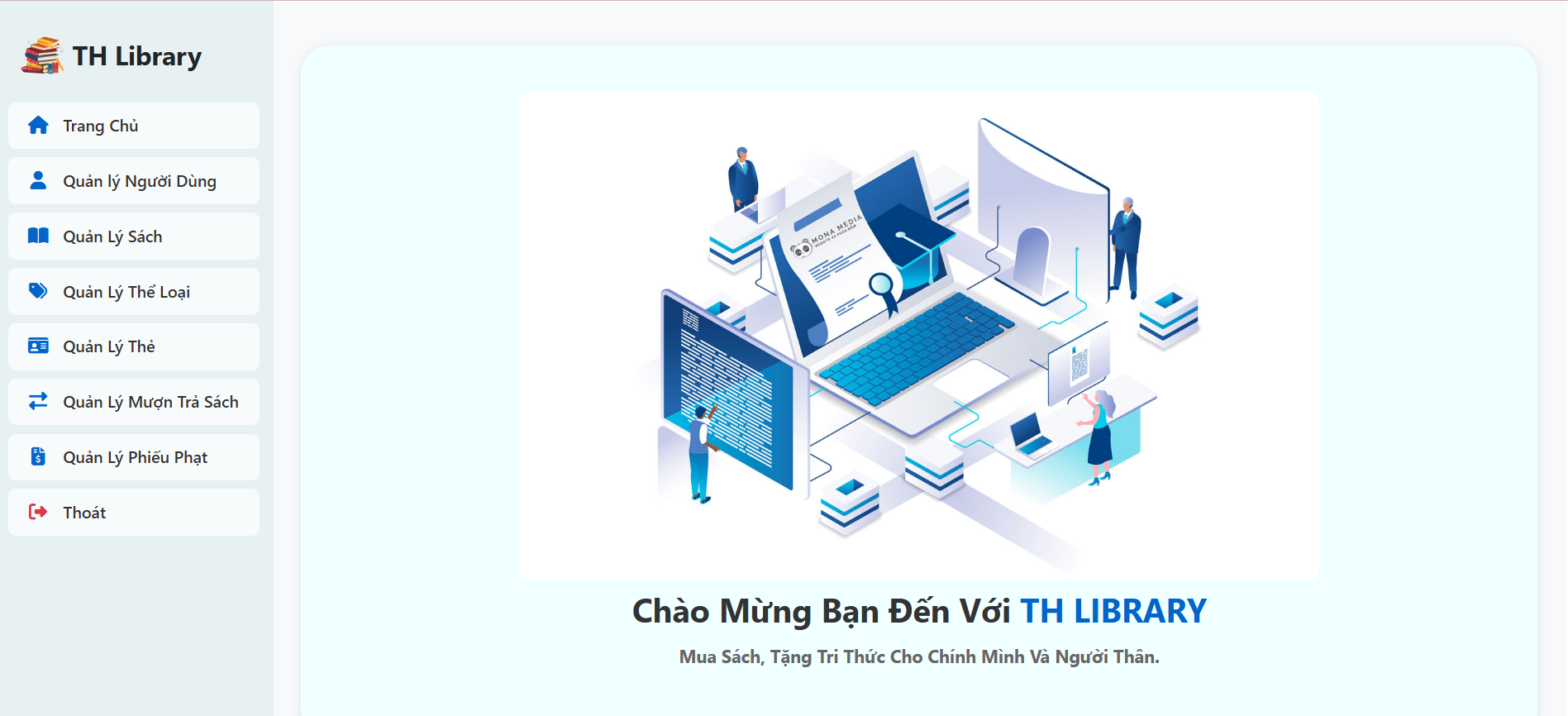


Hình 18. Giao diện quản trị nếu là người dùng

## **3.3 Giao diện quản trị**

**3.3.1 Trang chủ**

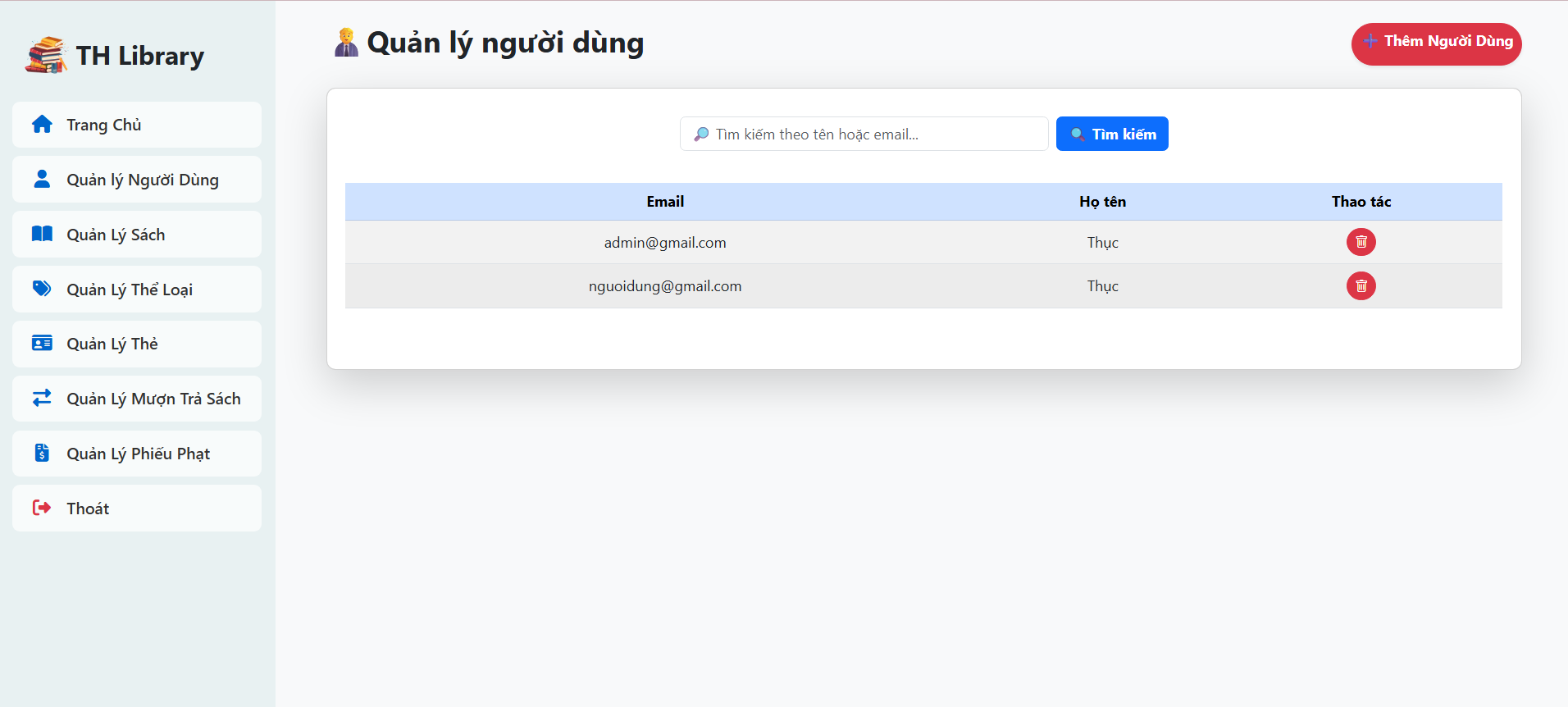
Chức năng giao diện: Tổng hợp các chức năng của hệ thống





Hình 19. Giao diện chung

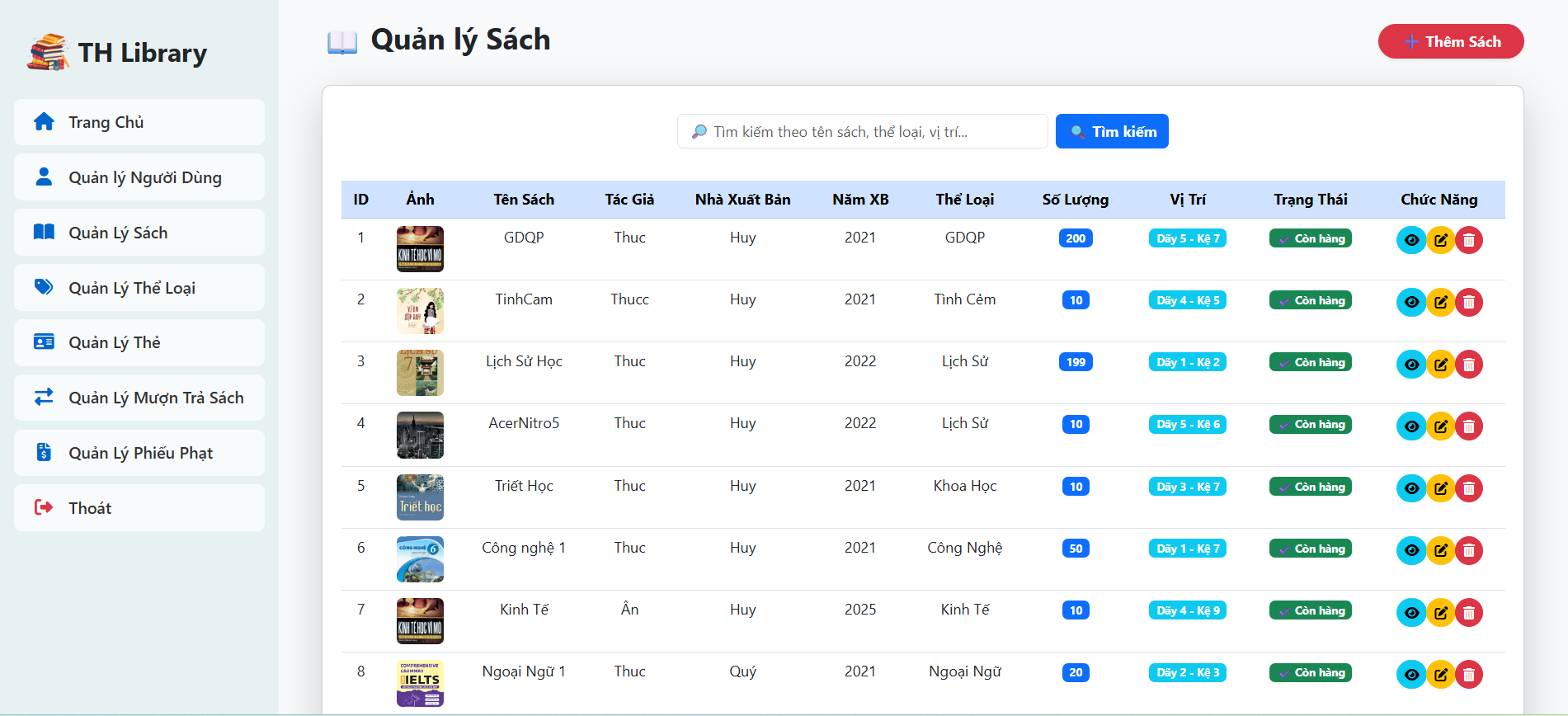
**3.3.2 Quản lý người dùng**

****

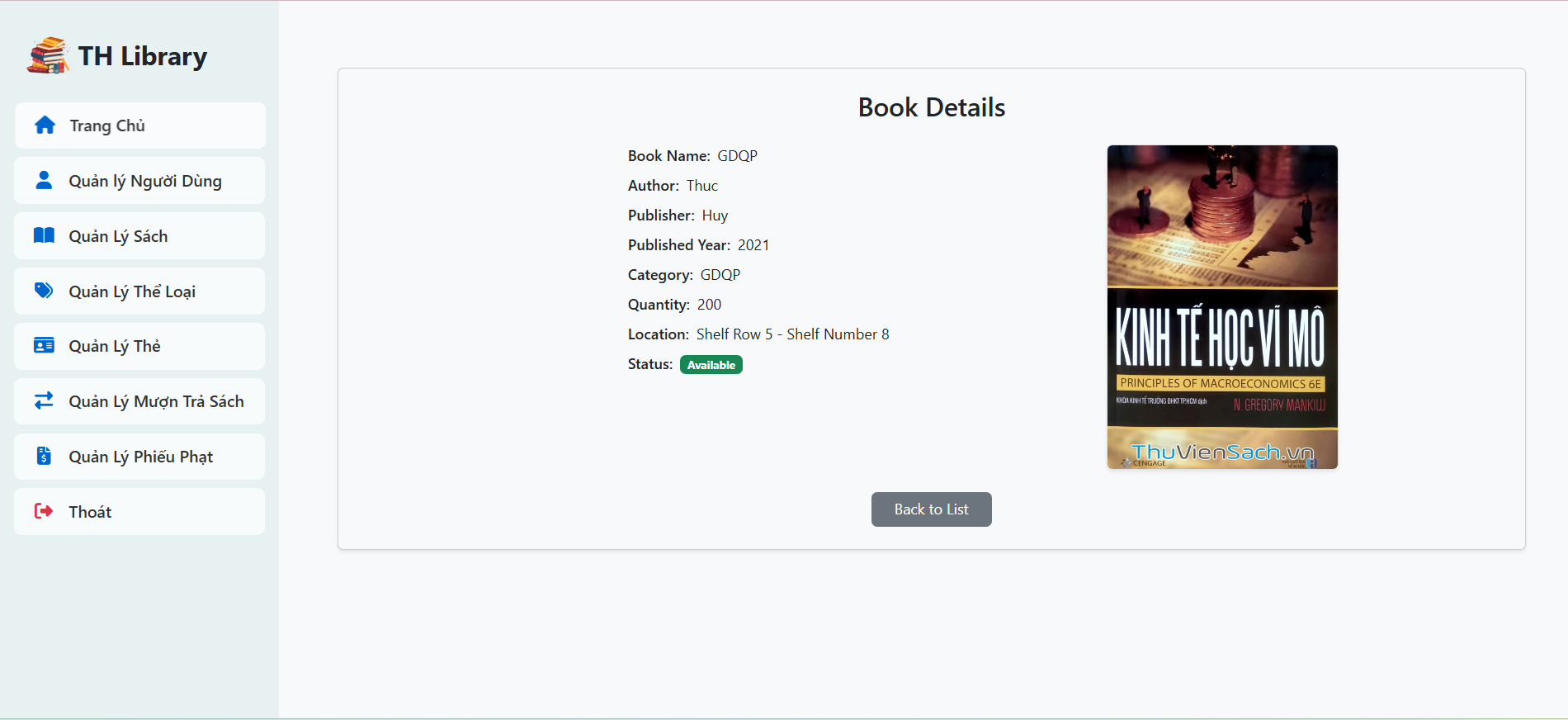
Hình 20. Quản lý người dùng

**3.3.3 Quản lý sách**

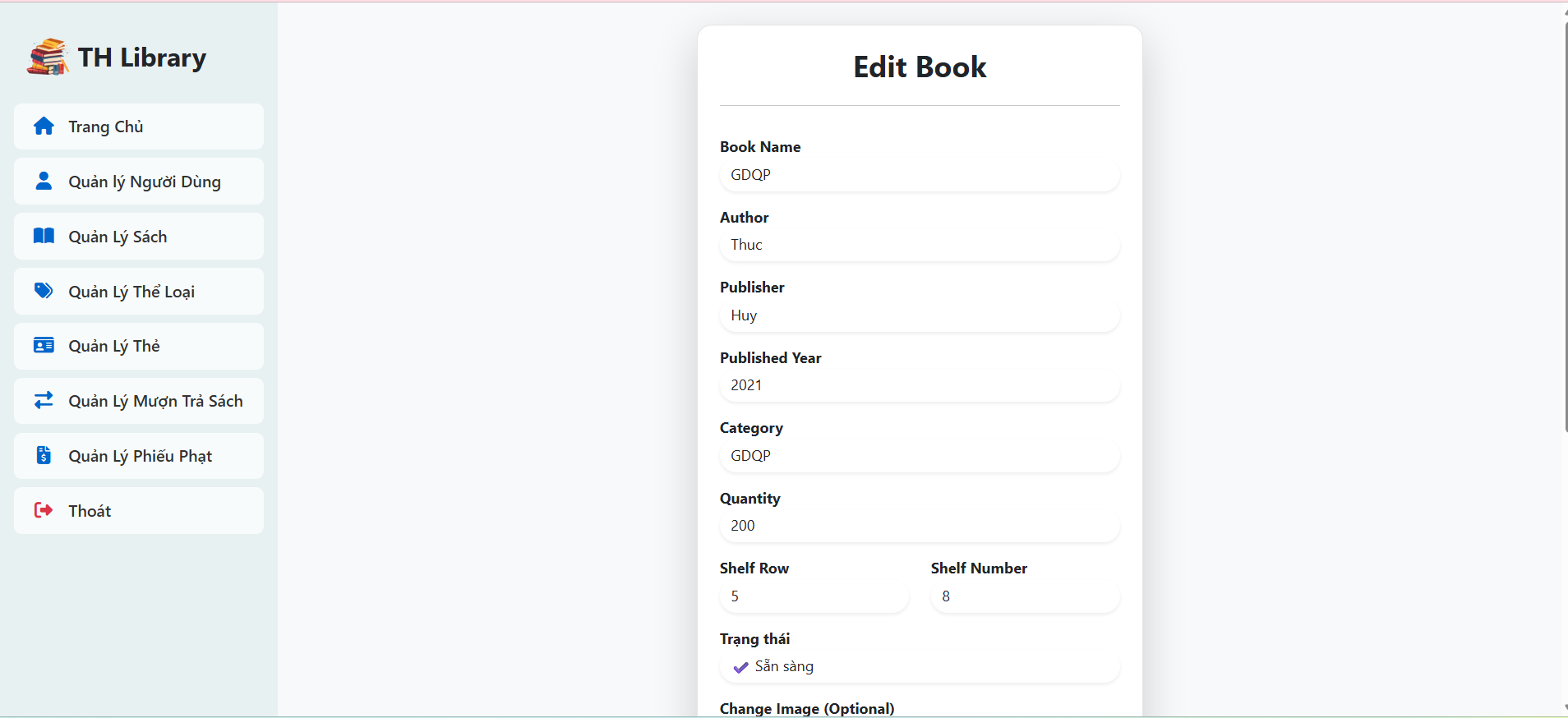
Chức năng giao diện: Hiển thị tông tin sách, thêm, xóa, sửa, cập nhật sách.



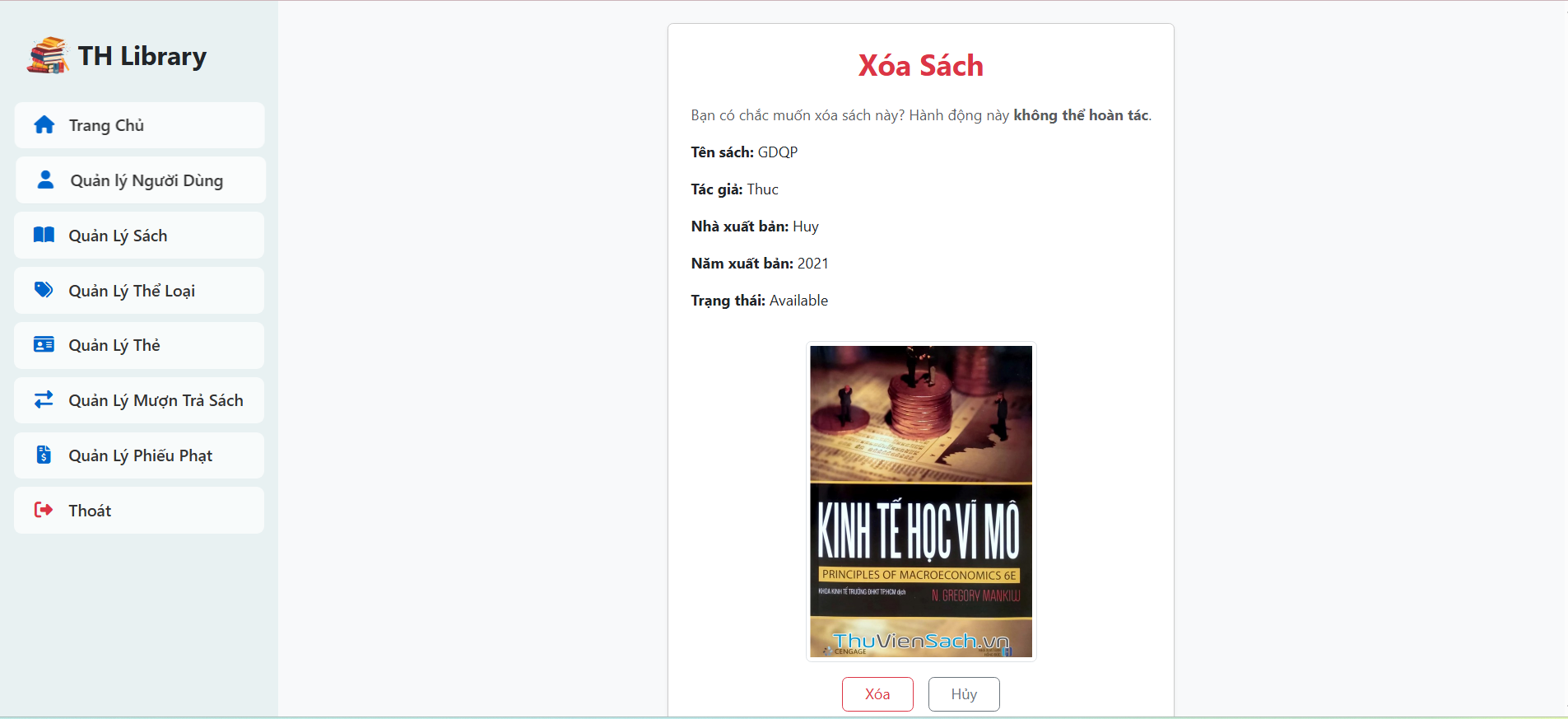
Hình 20. Giao diện chính quản lý sách



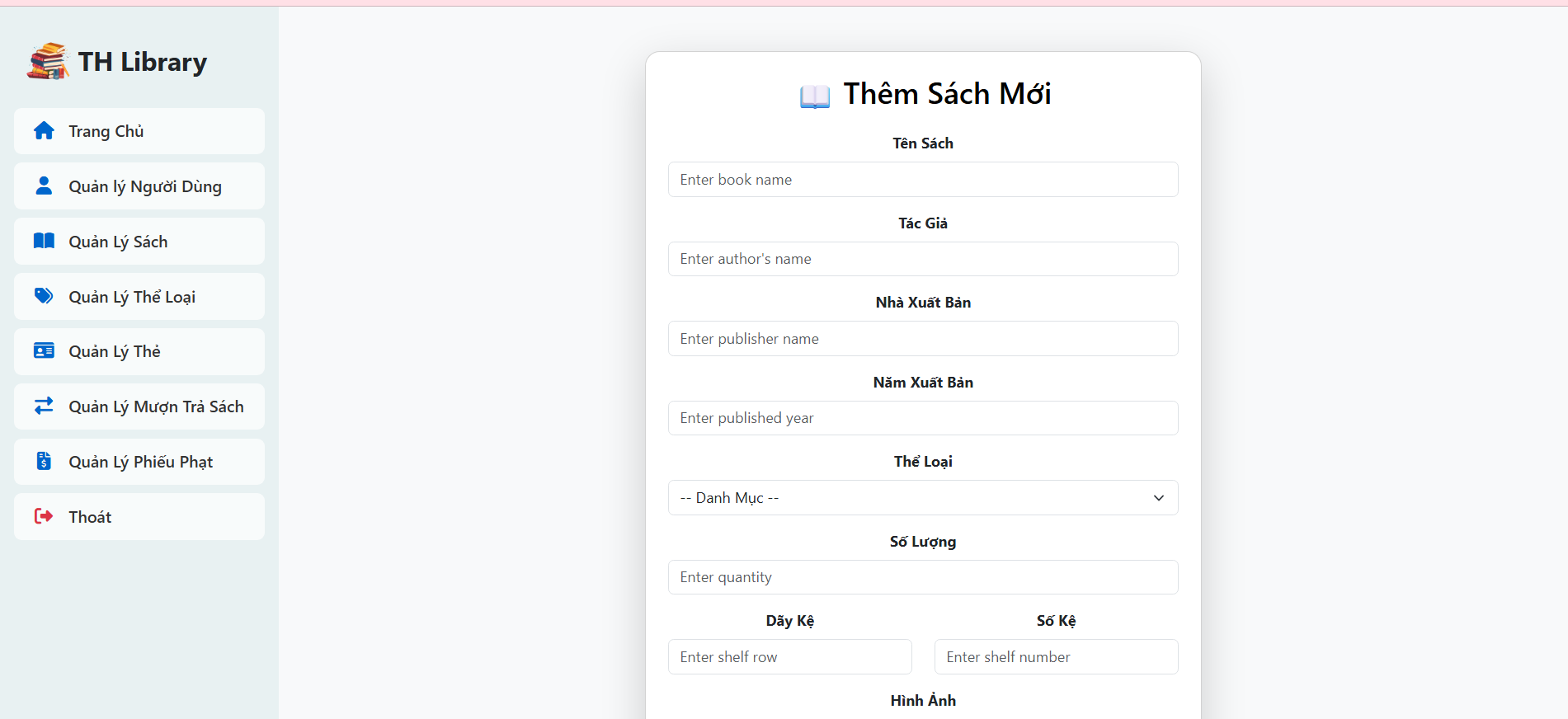
Hình 21. Giao diện thông tin chi tiết sách



Hình 22. Giao diện chỉnh sửa, cập nhật sách



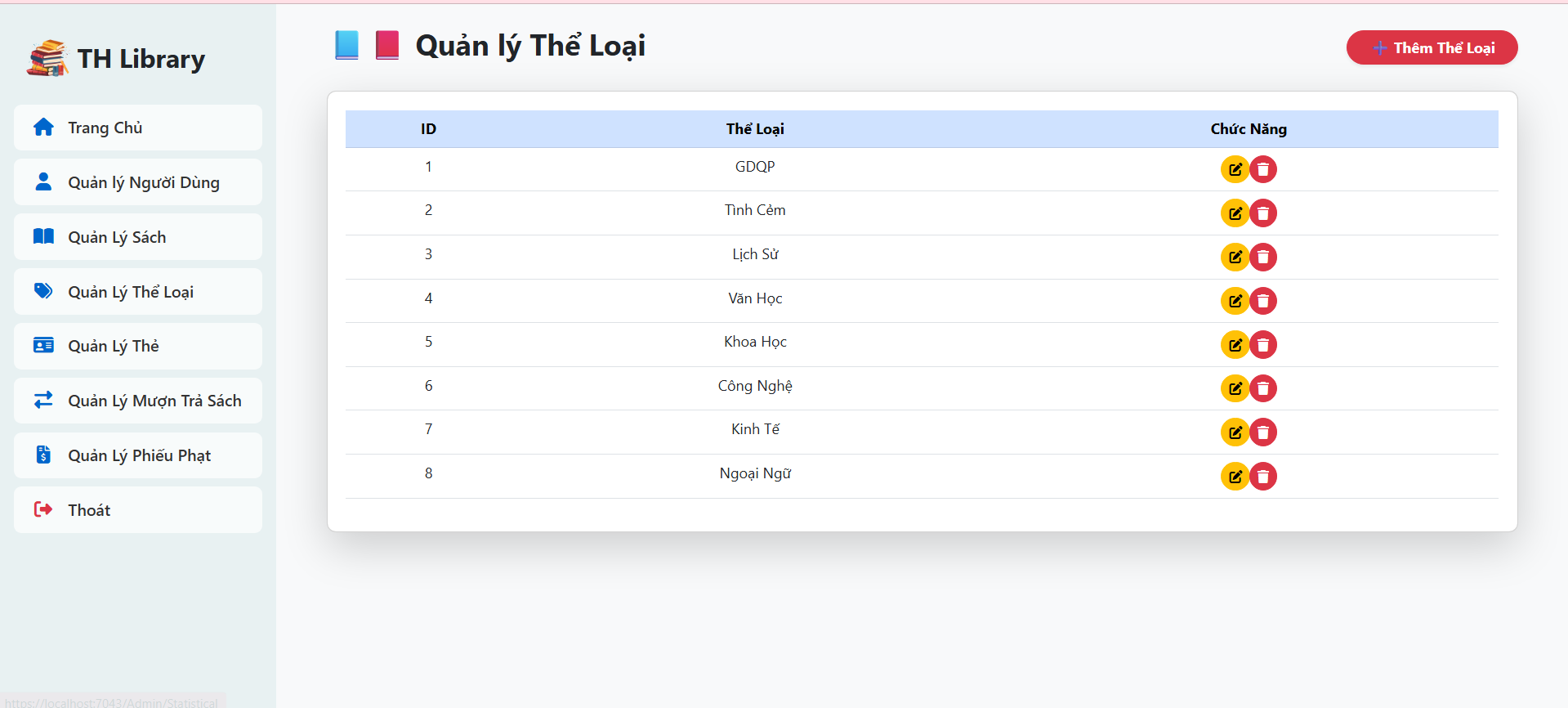
Hình 23. Giao diện xóa sách khỏi danh sách thư viện



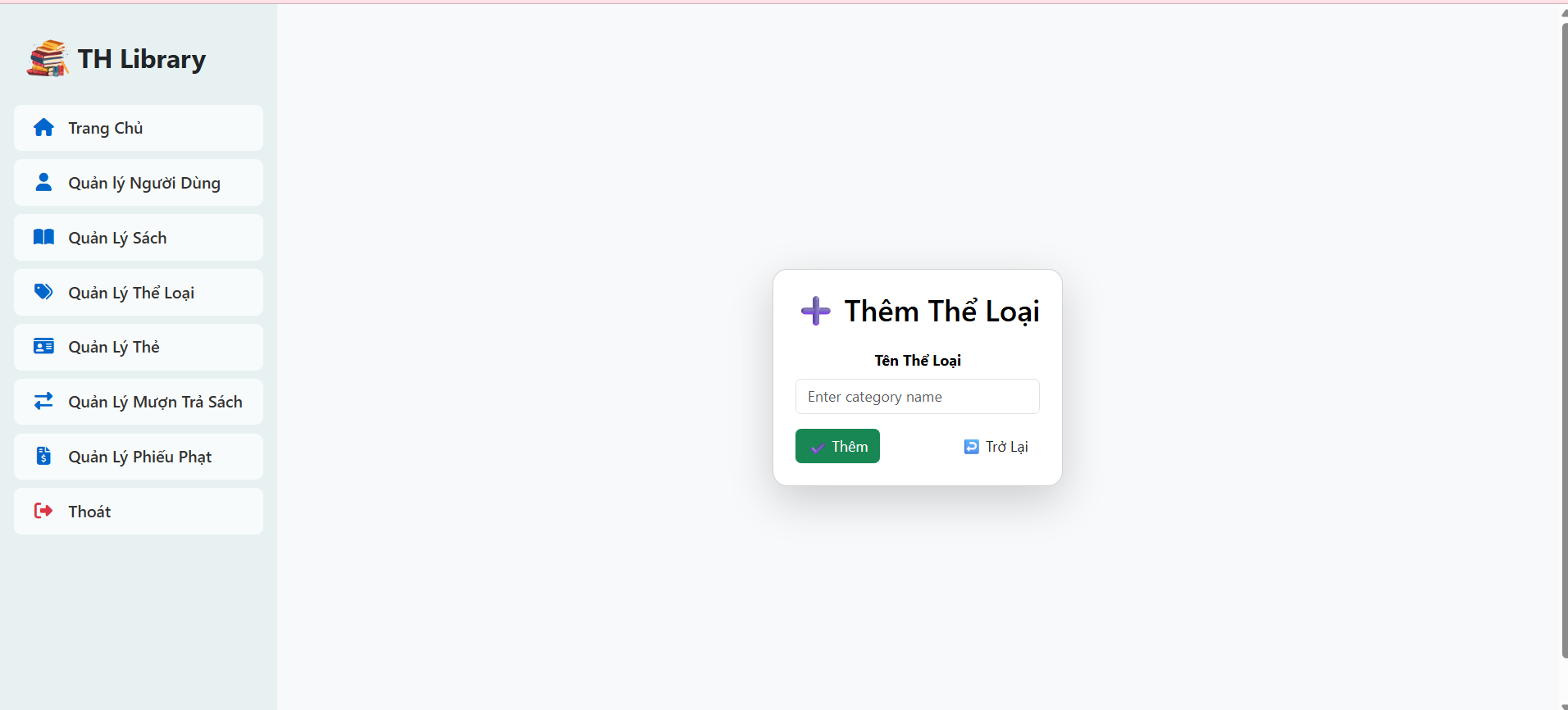
Hình 24. Giao diện thêm sách mới vào thư viện

**3.3.3 Quản lý thể loại**

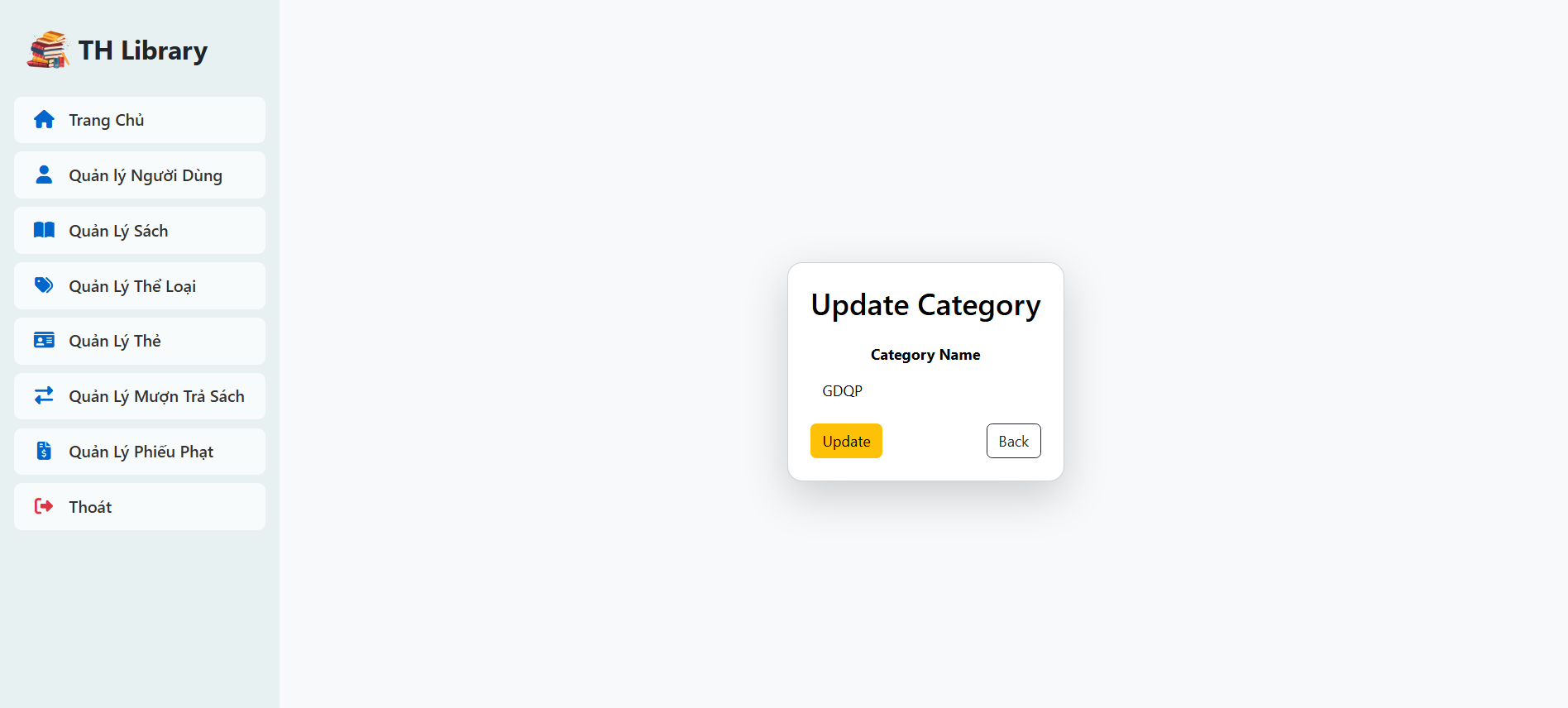
Chức năng giao diện: Giao diện chính của quản lý thể loại thể hiện các thông tin như thêm, xóa, sửa thể loại sách.



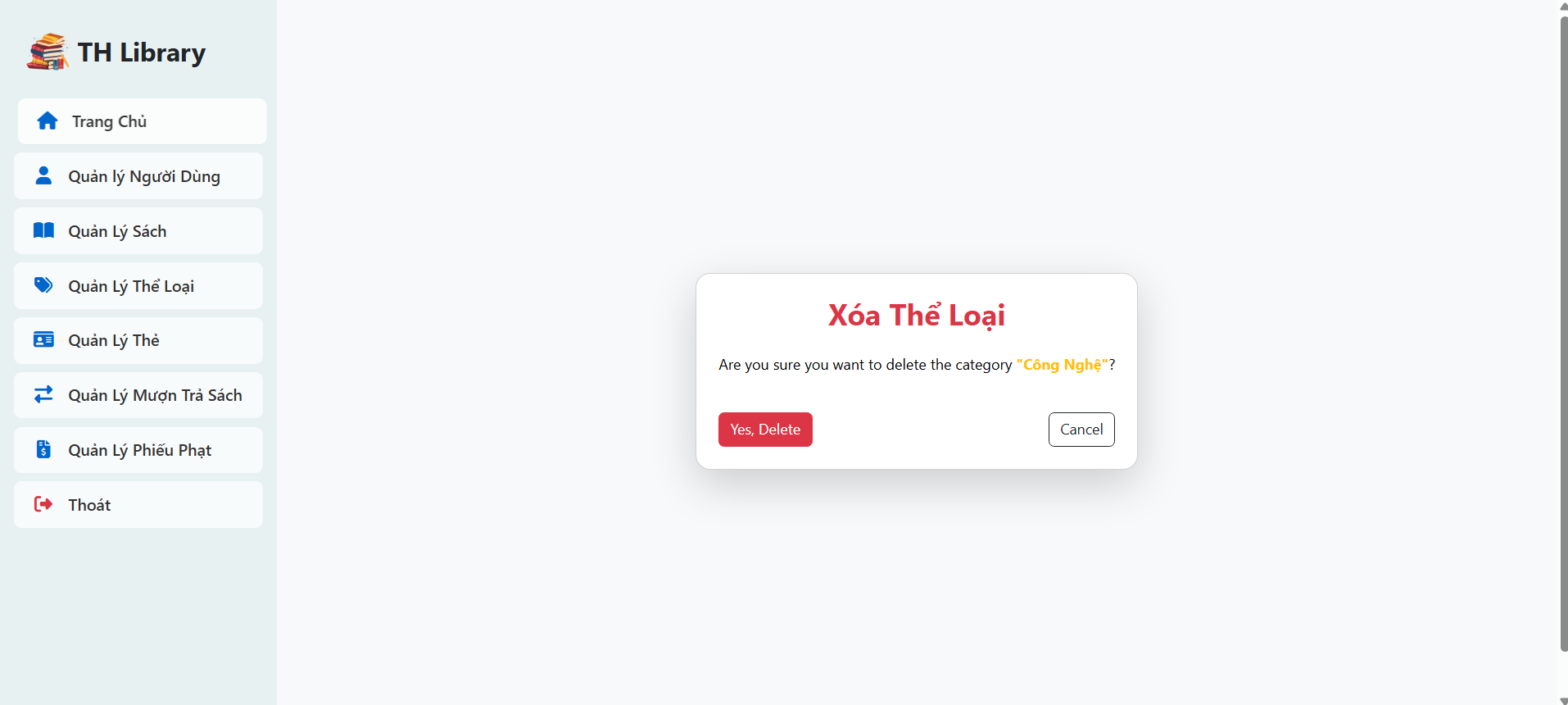
Hình 25. Giao diện chính quản lý thể loại



Hình 26. Giao diện thêm thể loại sách



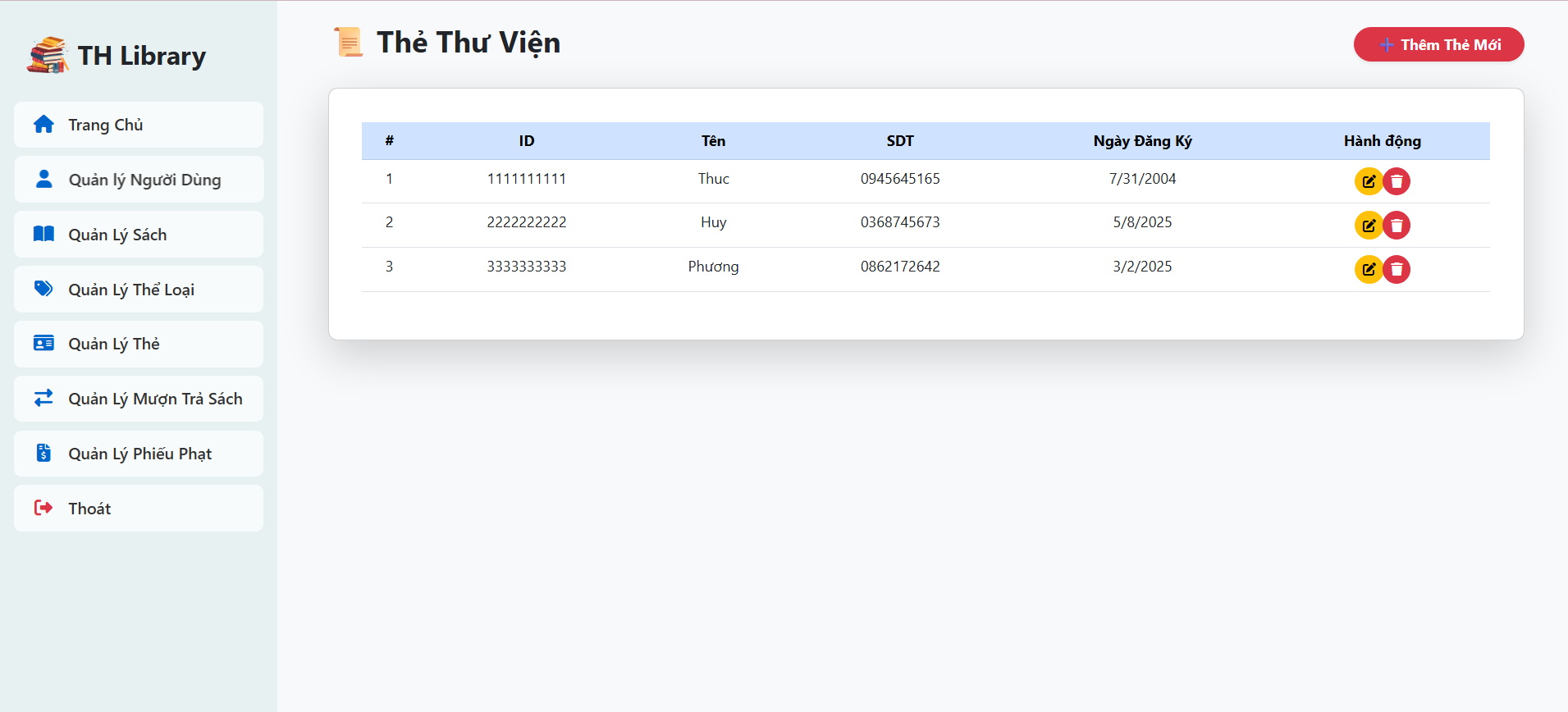
Hình 27. Giao diện sửa thể loại



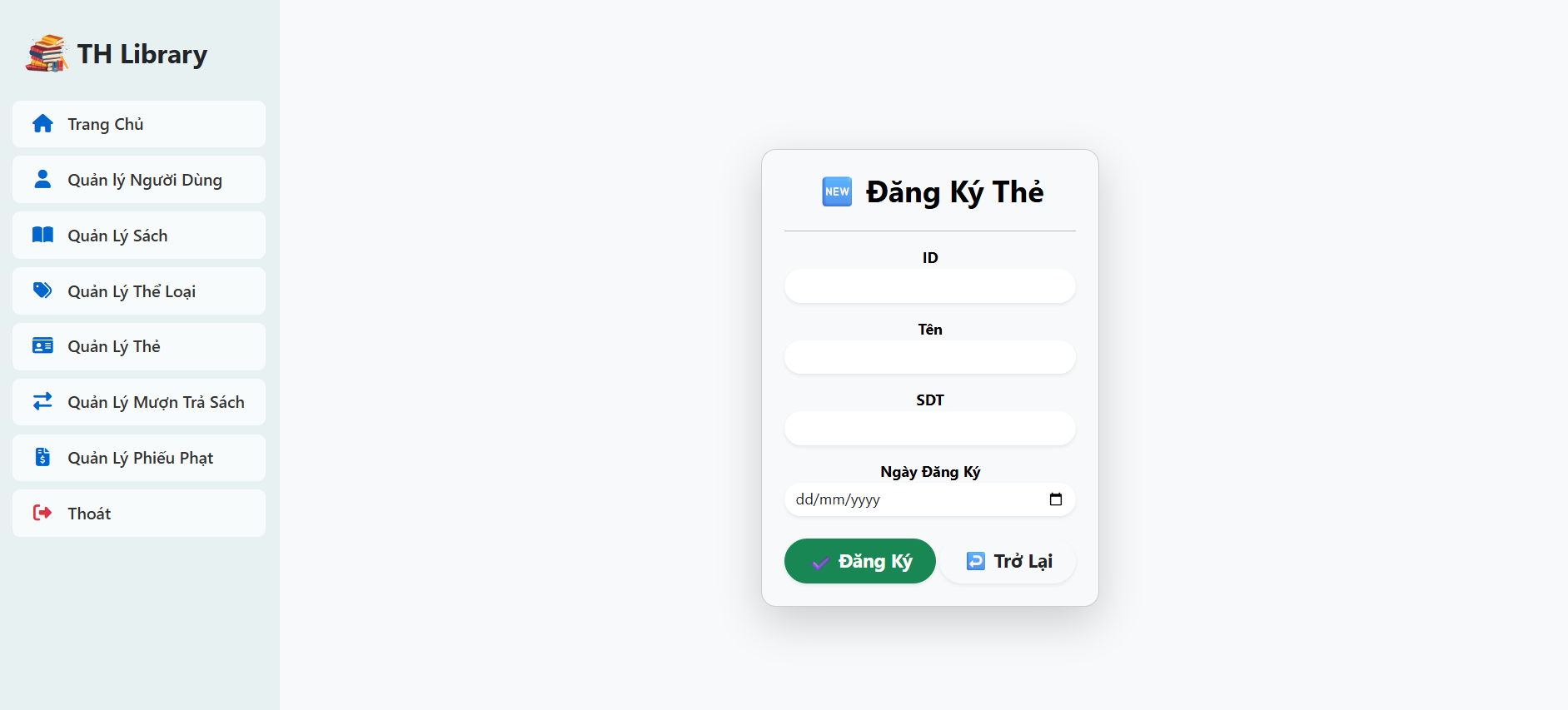
Hình 28. Giao diện xóa thể loại

**3.3.4 Quản lý thẻ**

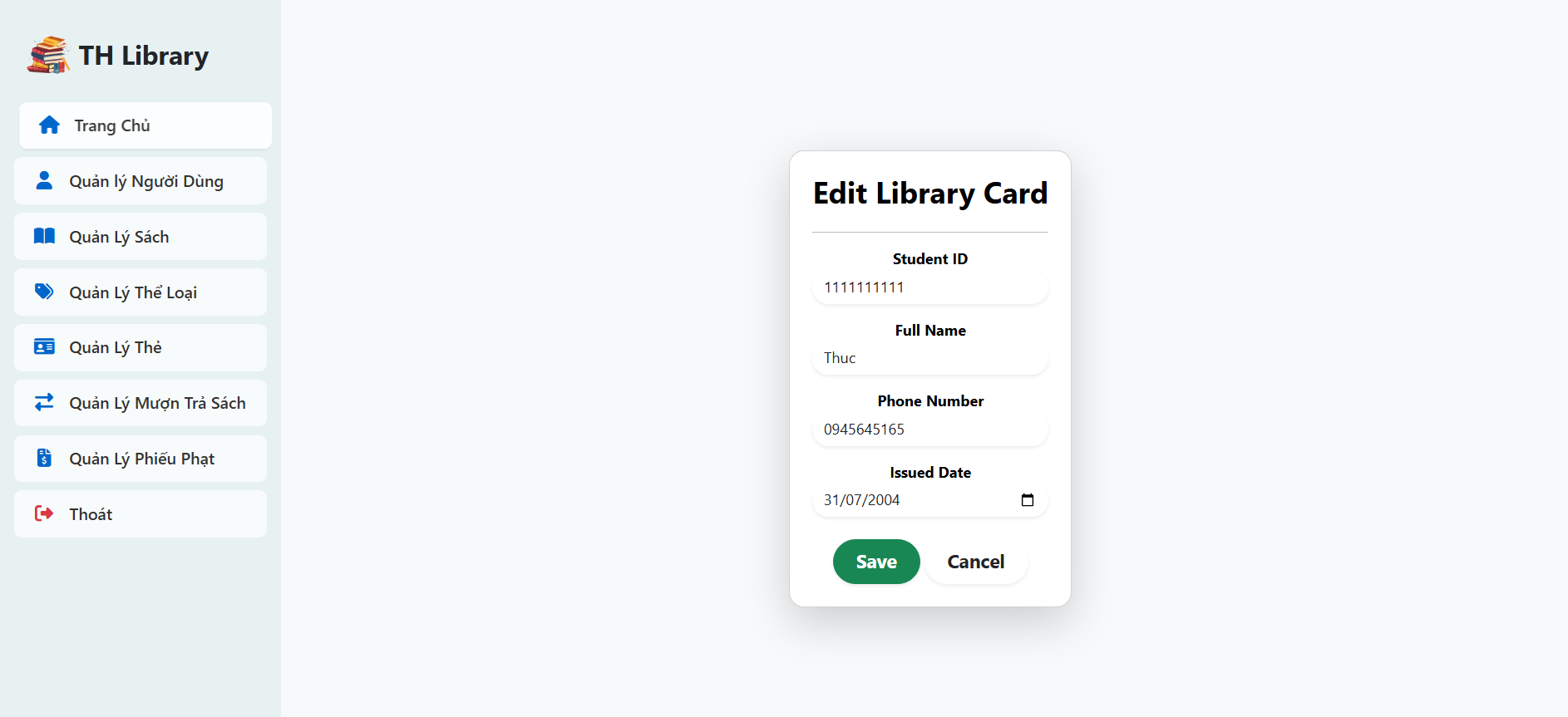
Chức năng giao diện: Giao diện chính của quản lý thẻ thư viện



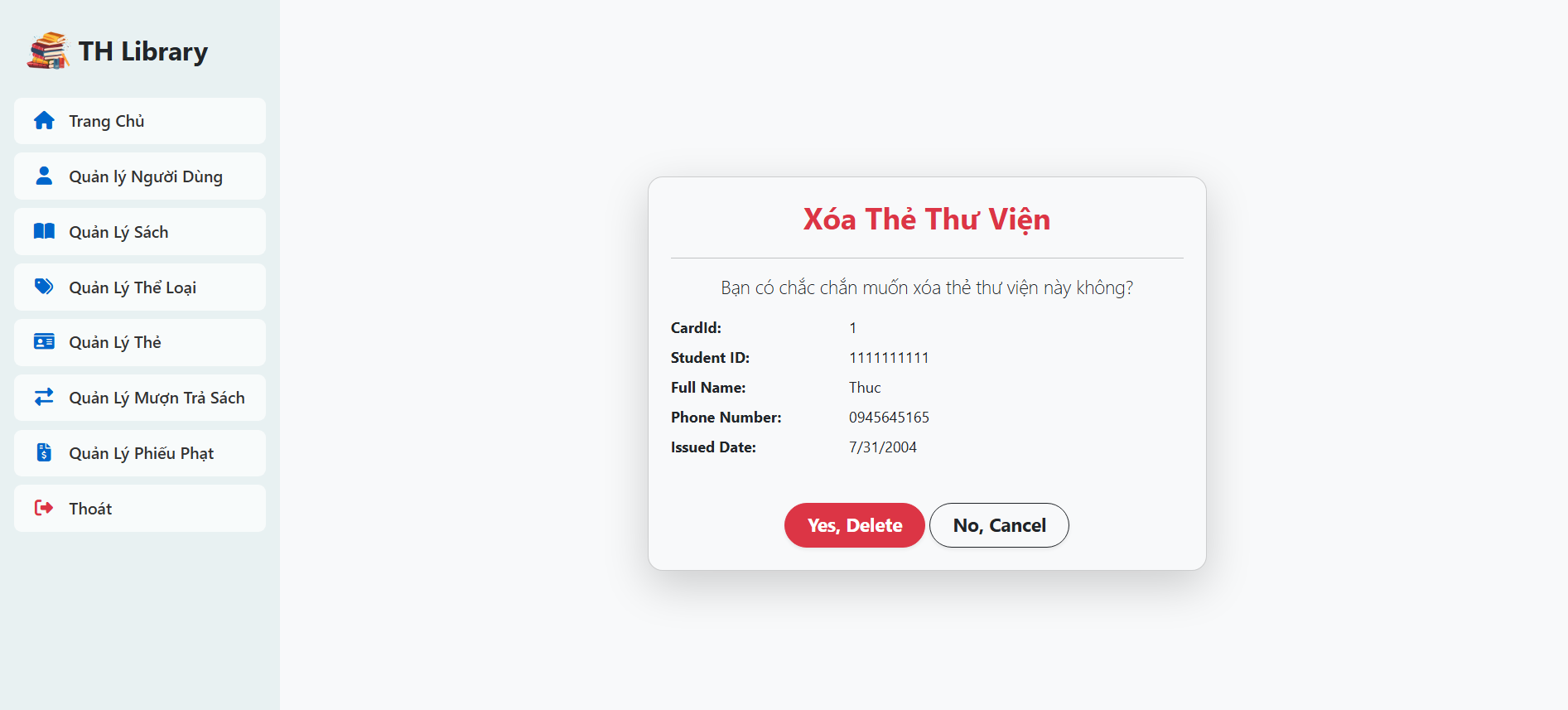
Hình 29. Giao diện chính



Hình 30. Giao diện đăng ký thẻ thư viện



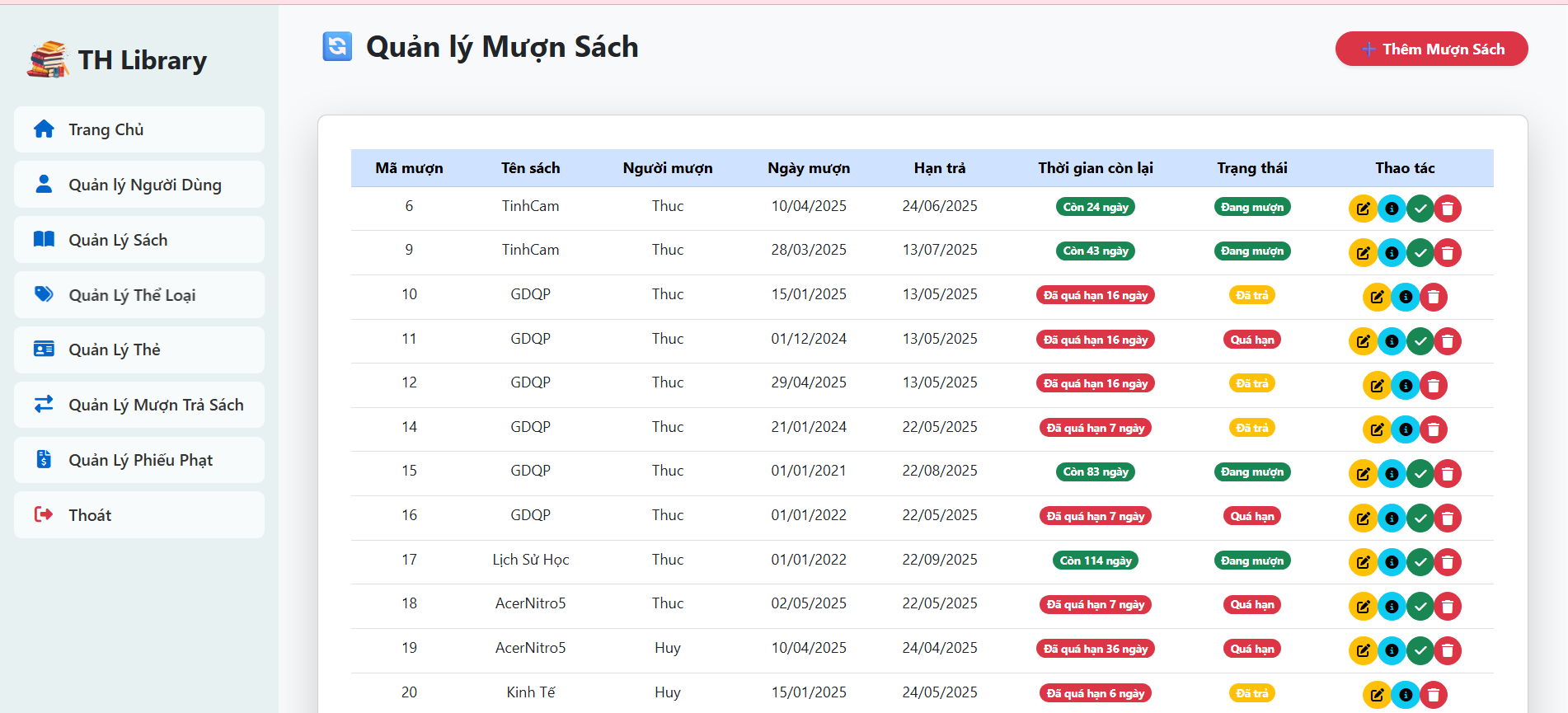
Hình 31. Giao diện chỉnh sửa thẻ thư viện



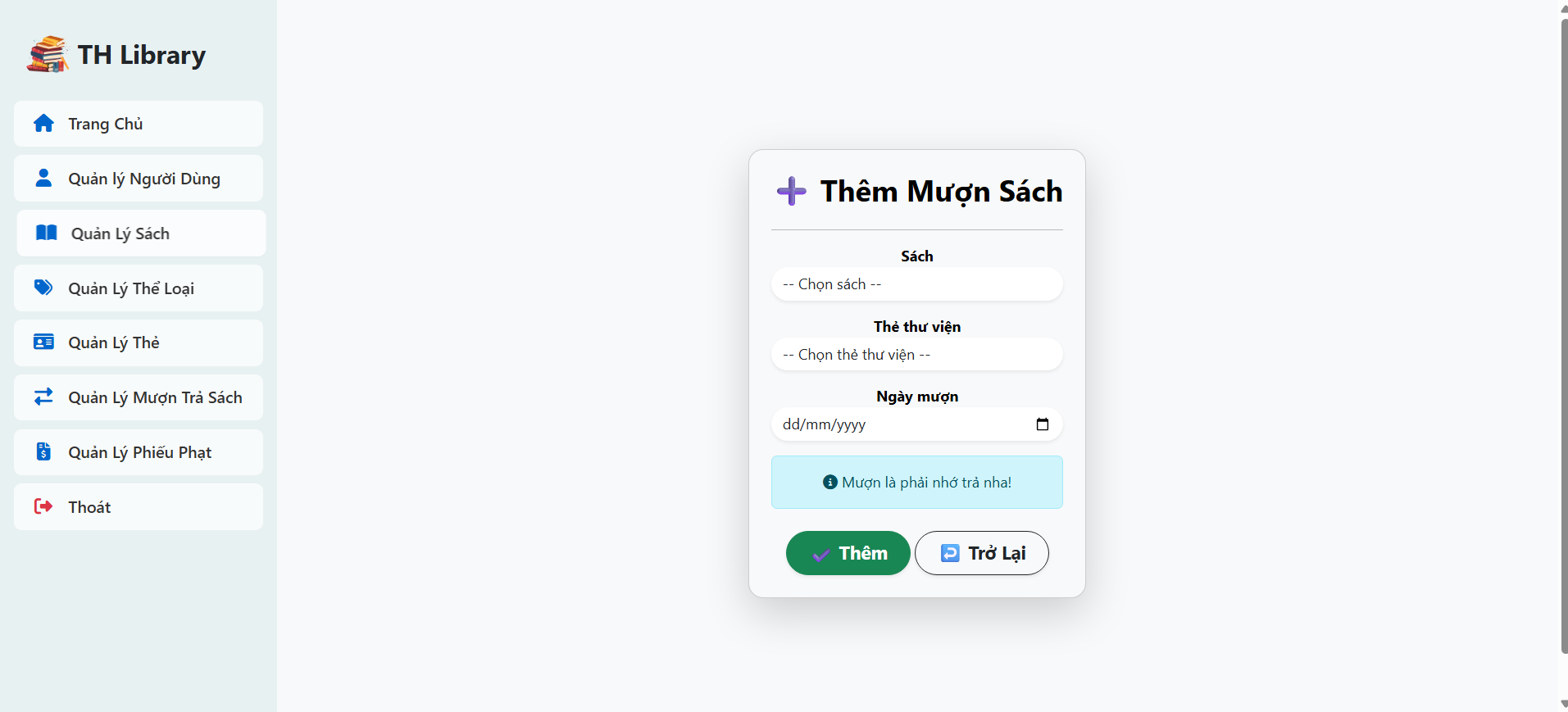
Hình 32. Giao diện xóa thẻ thư viện

**3.3.5 Quản lý mượn, trả sách**

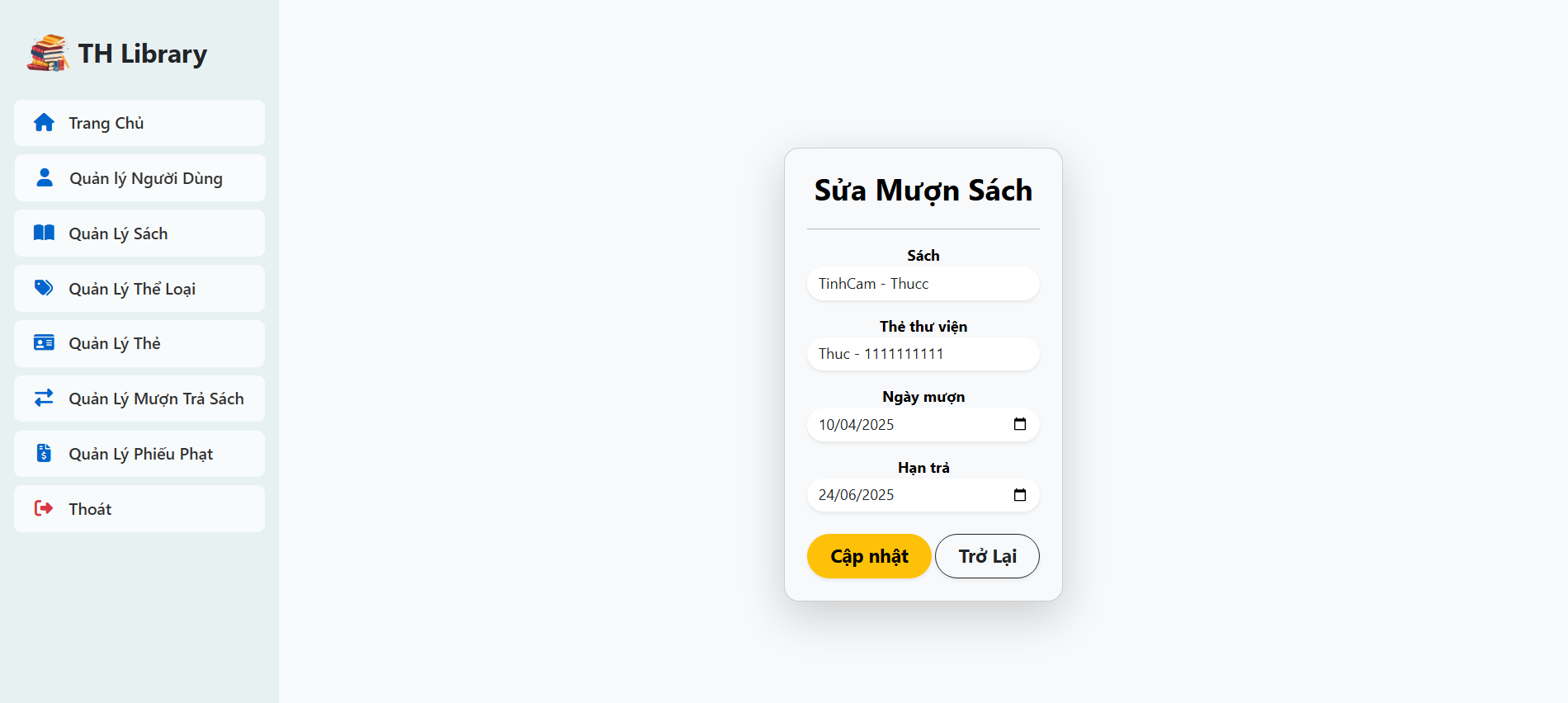
Chức năng giao diện: Hiện thị thông tin người mượn….



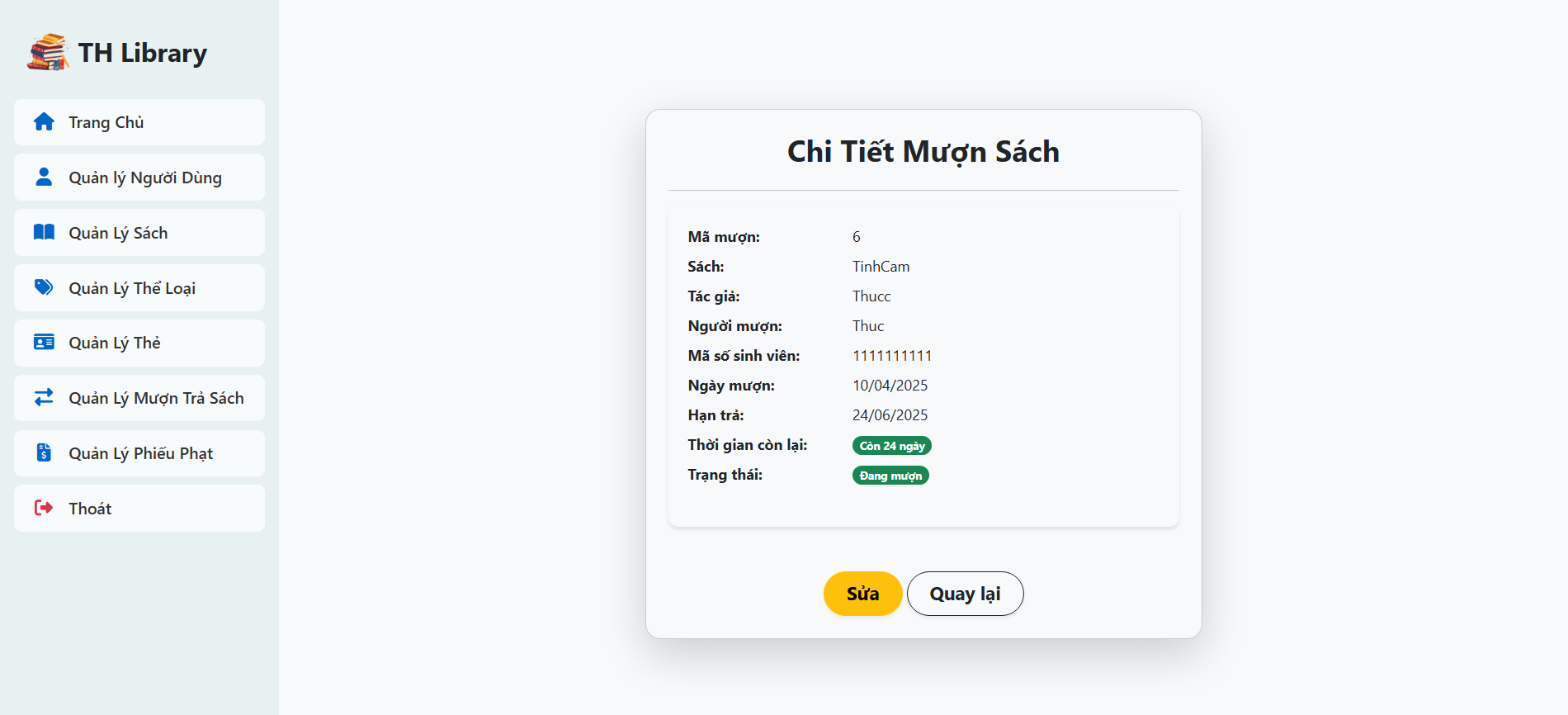
Hình 33. Giao diện chung của quản lý mượn, trả sách



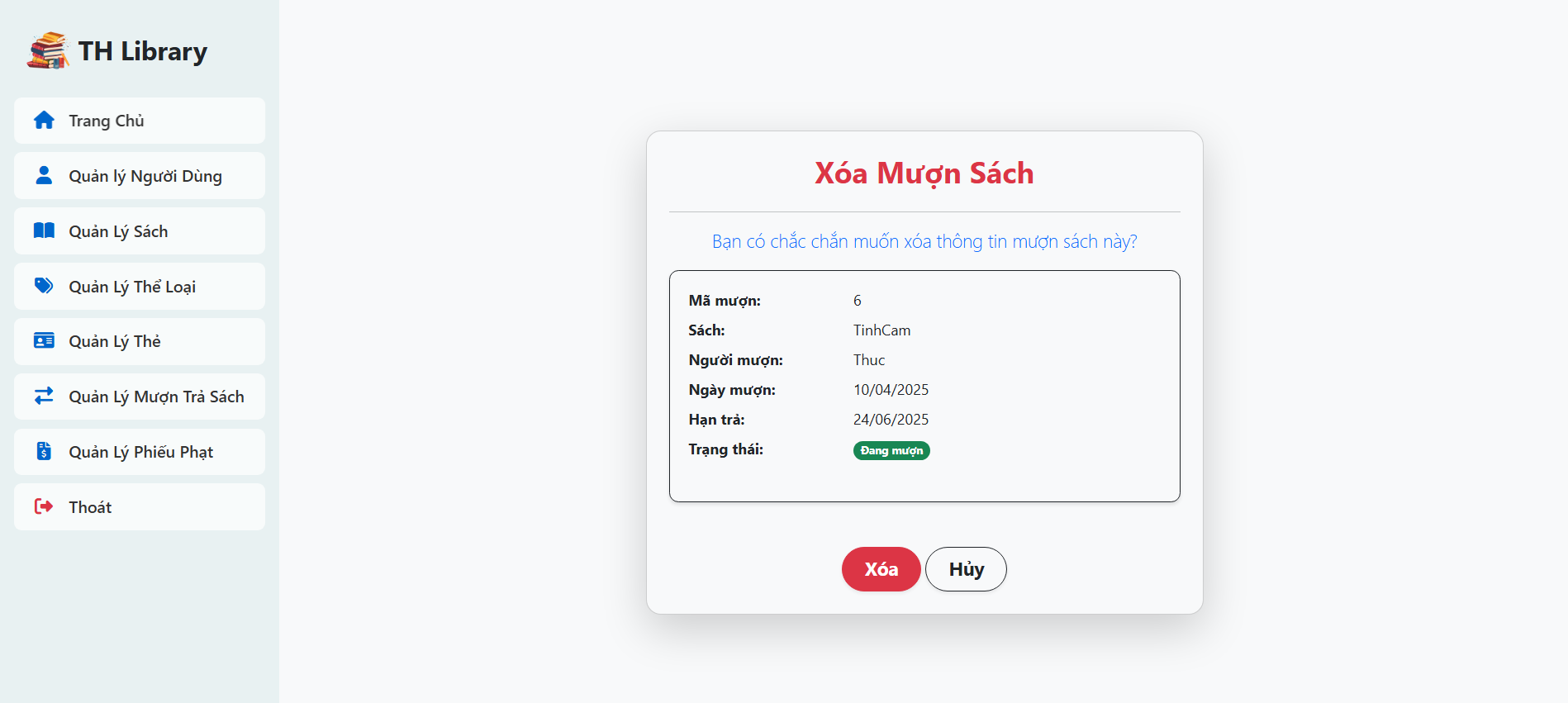
Hình 34. Giao diện thêm người dùng mượn sách

****

Hình 35. Giao diện chỉnh sửa người mượn



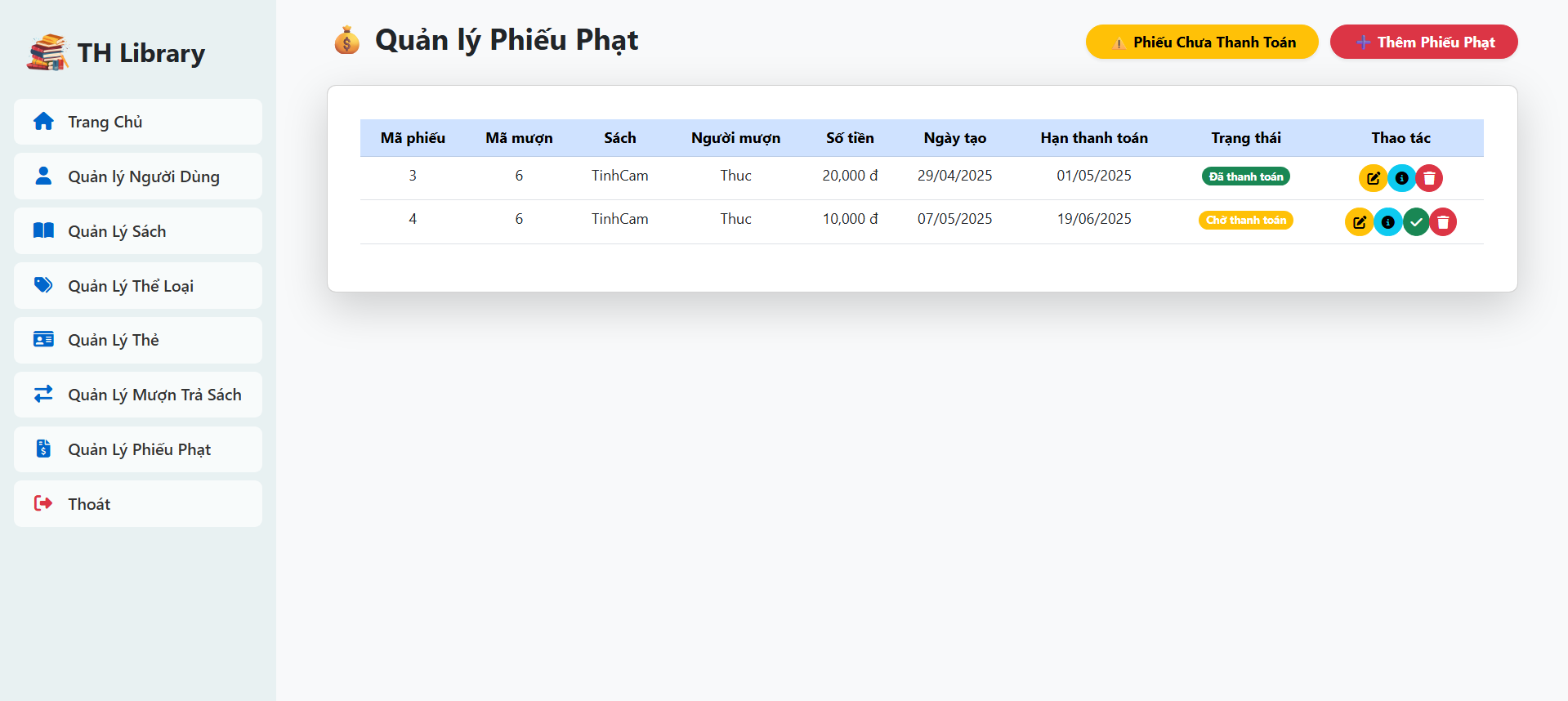
Hình 36. Giao diện hiển thị chi tiết mượn sách

****

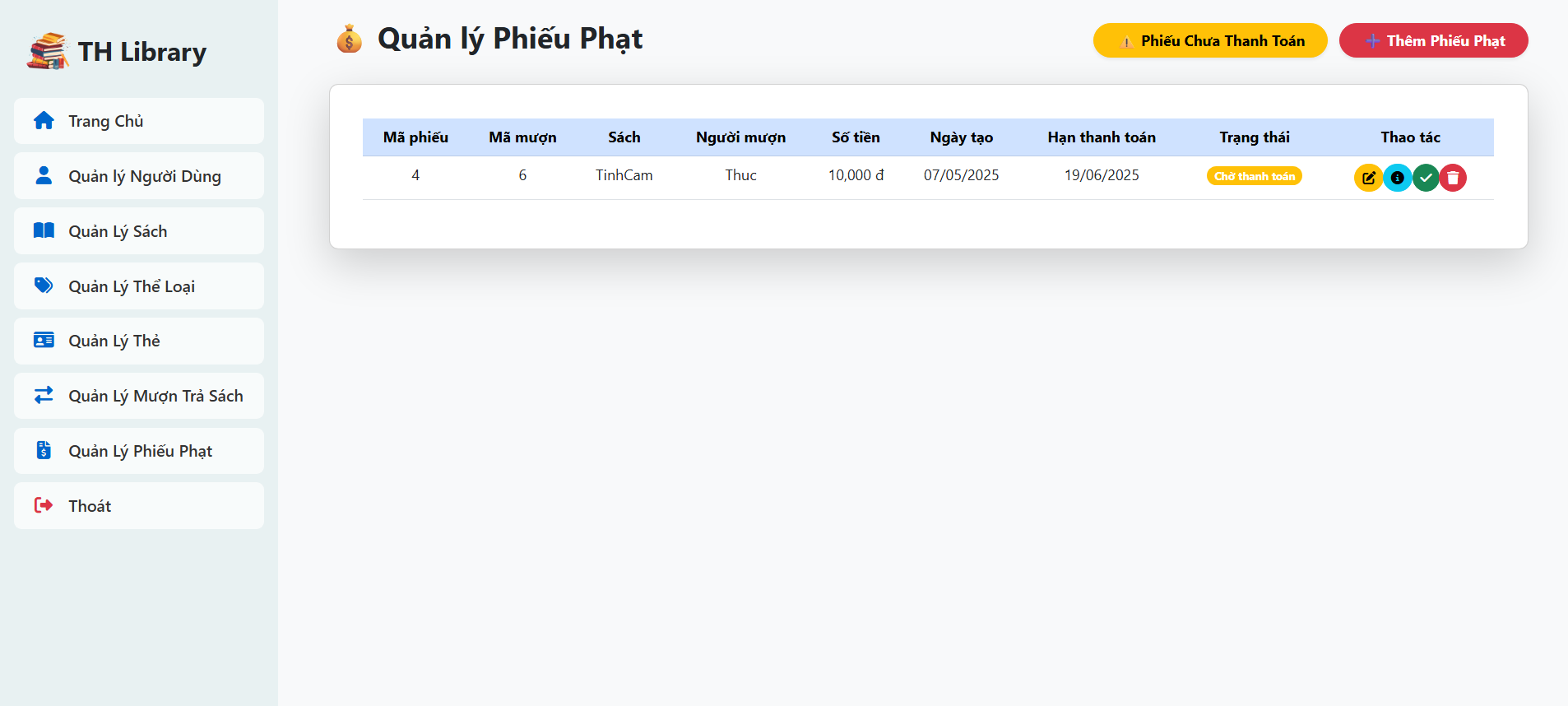
Hình 37. Giao diện xóa người dùng mượn sách

**3.3.6 Quản lý phiếu phạt**

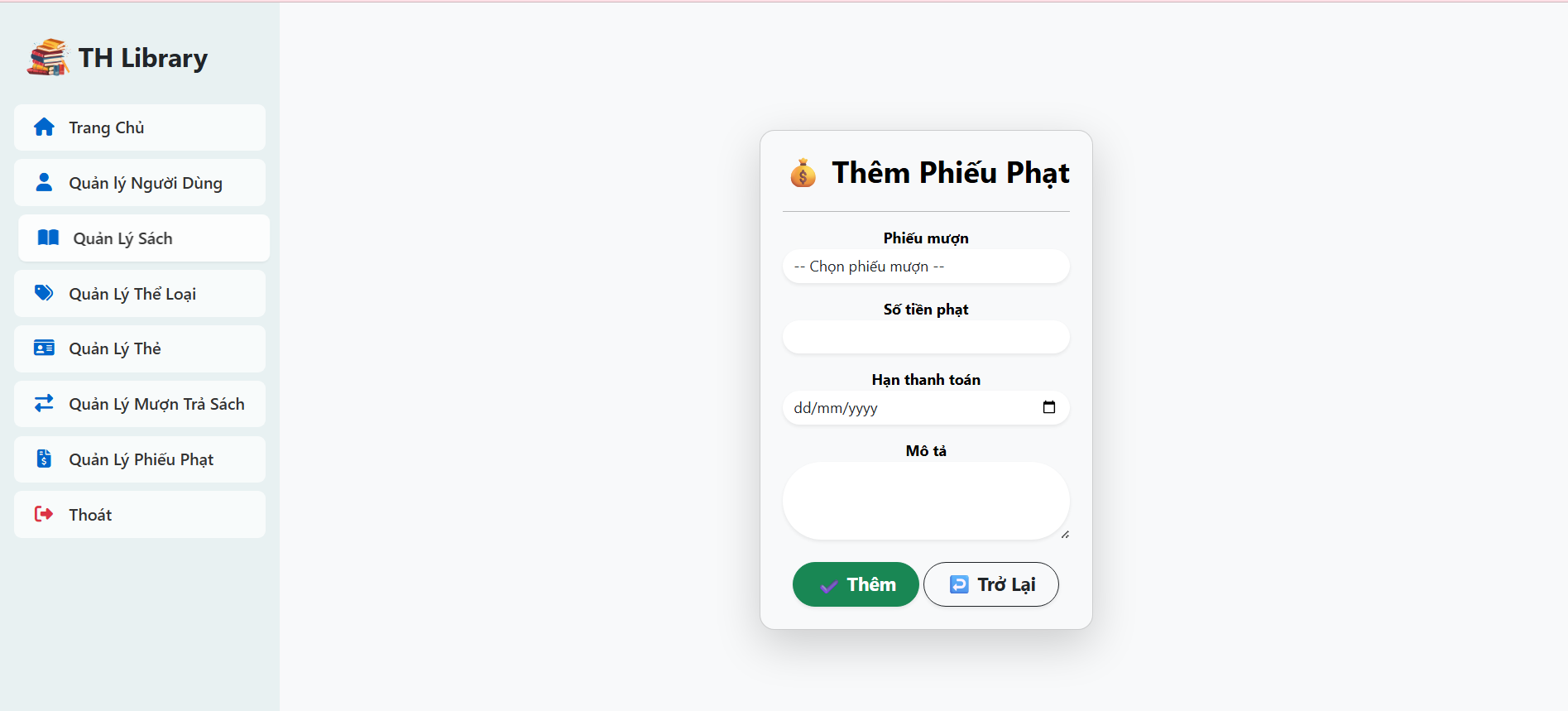
Chức năng giao diện: Hiển thị thông tin chi tiết phiếu phạt…



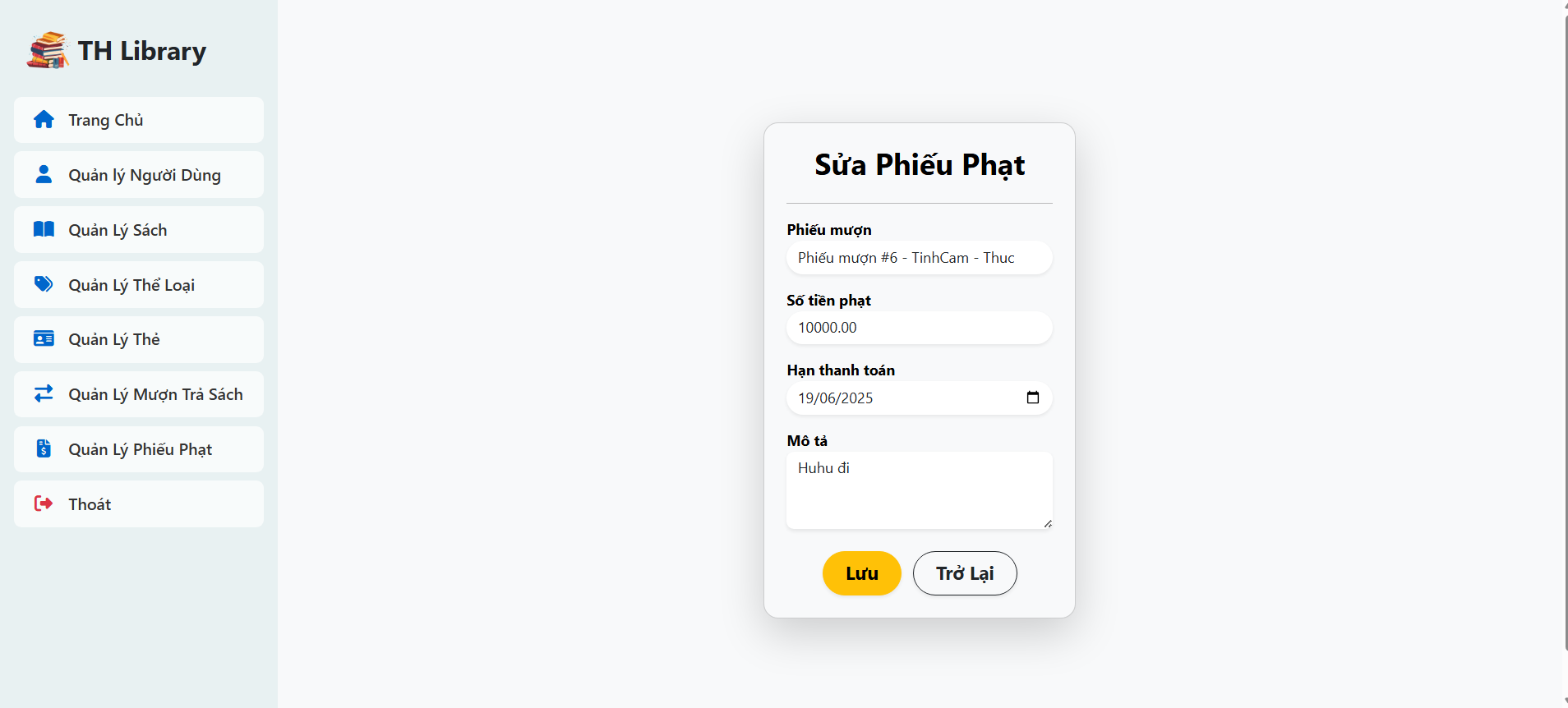
Hình 38. Giao diện chính thông tin phiếu phạt



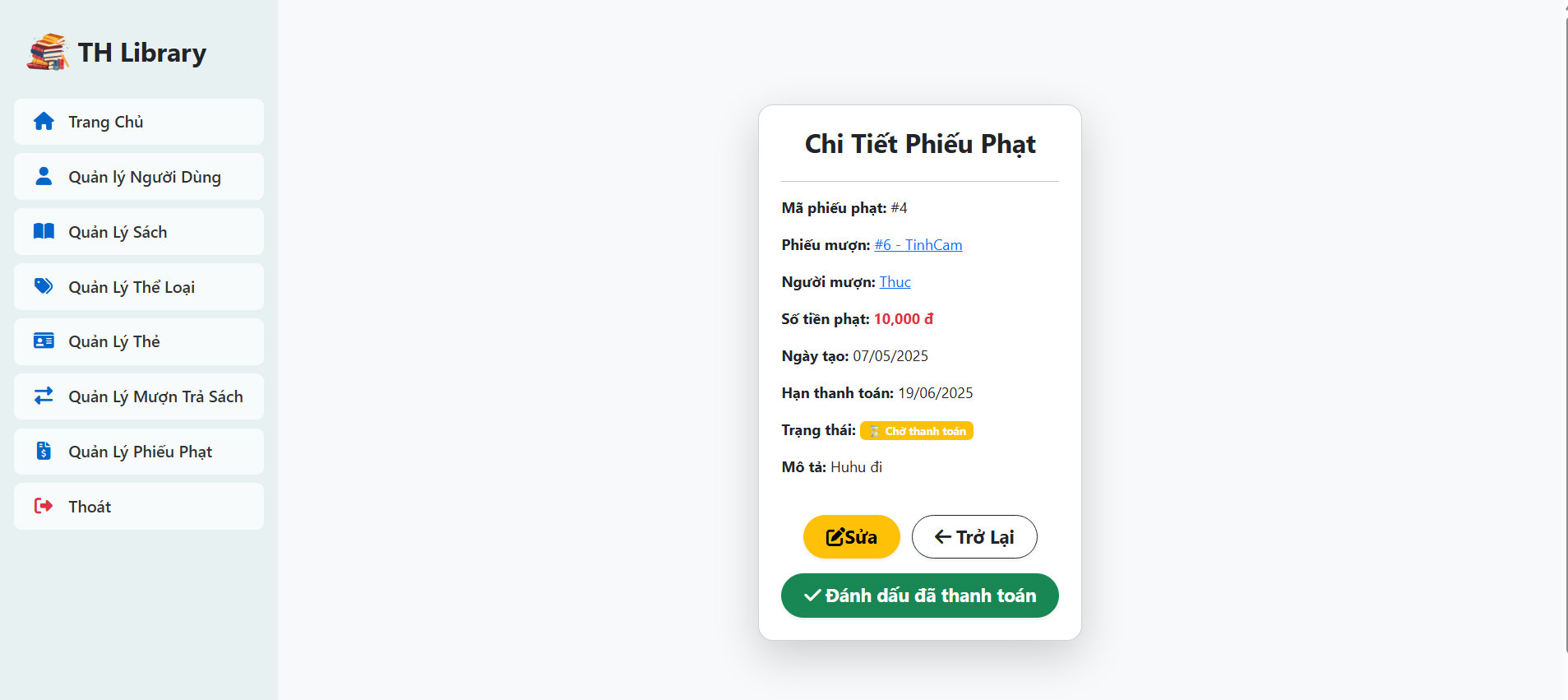
Hình 39. Khi nhấn phiếu chưa thanh toán sẽ hiện các phiếu chưa thanh toán



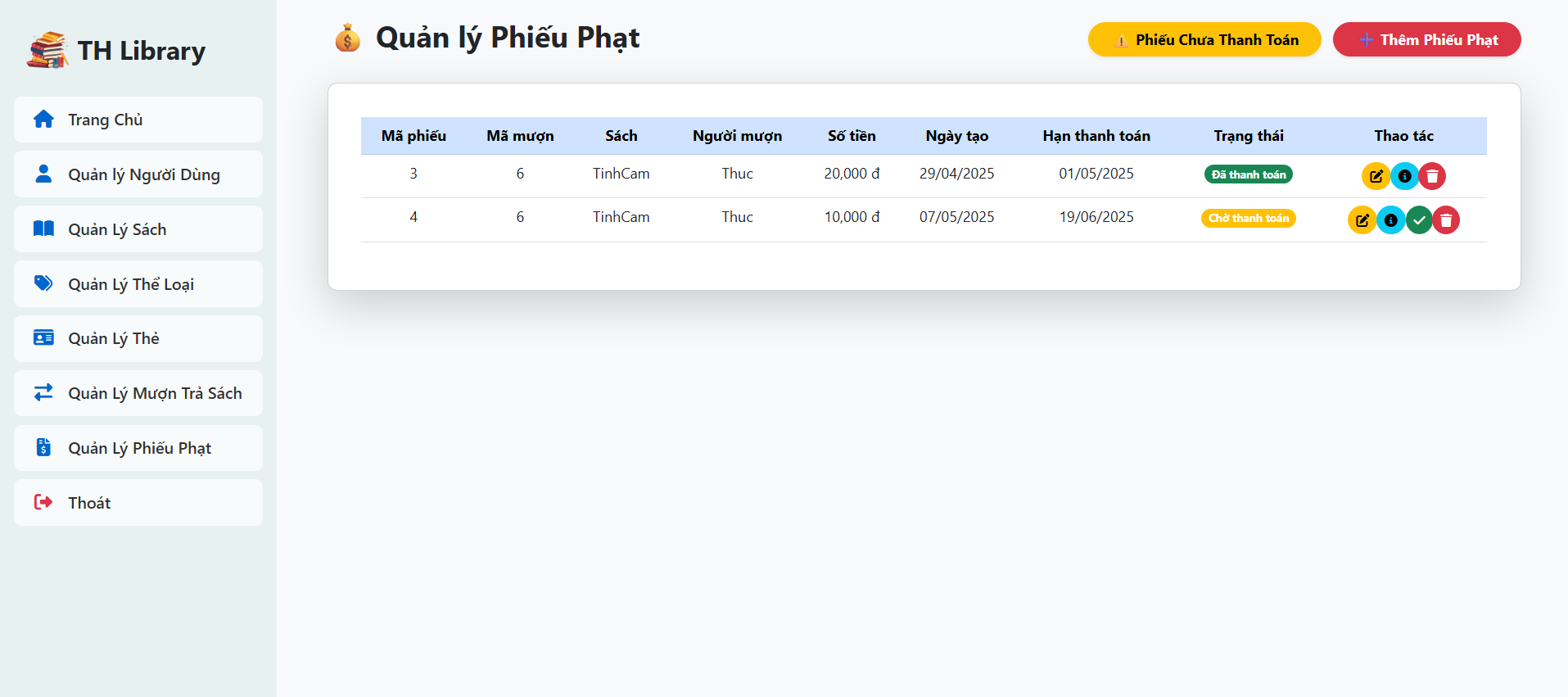
Hình 40. Giao diện thêm phiếu phạt khi người dùng mượn quá hạn, mất sách



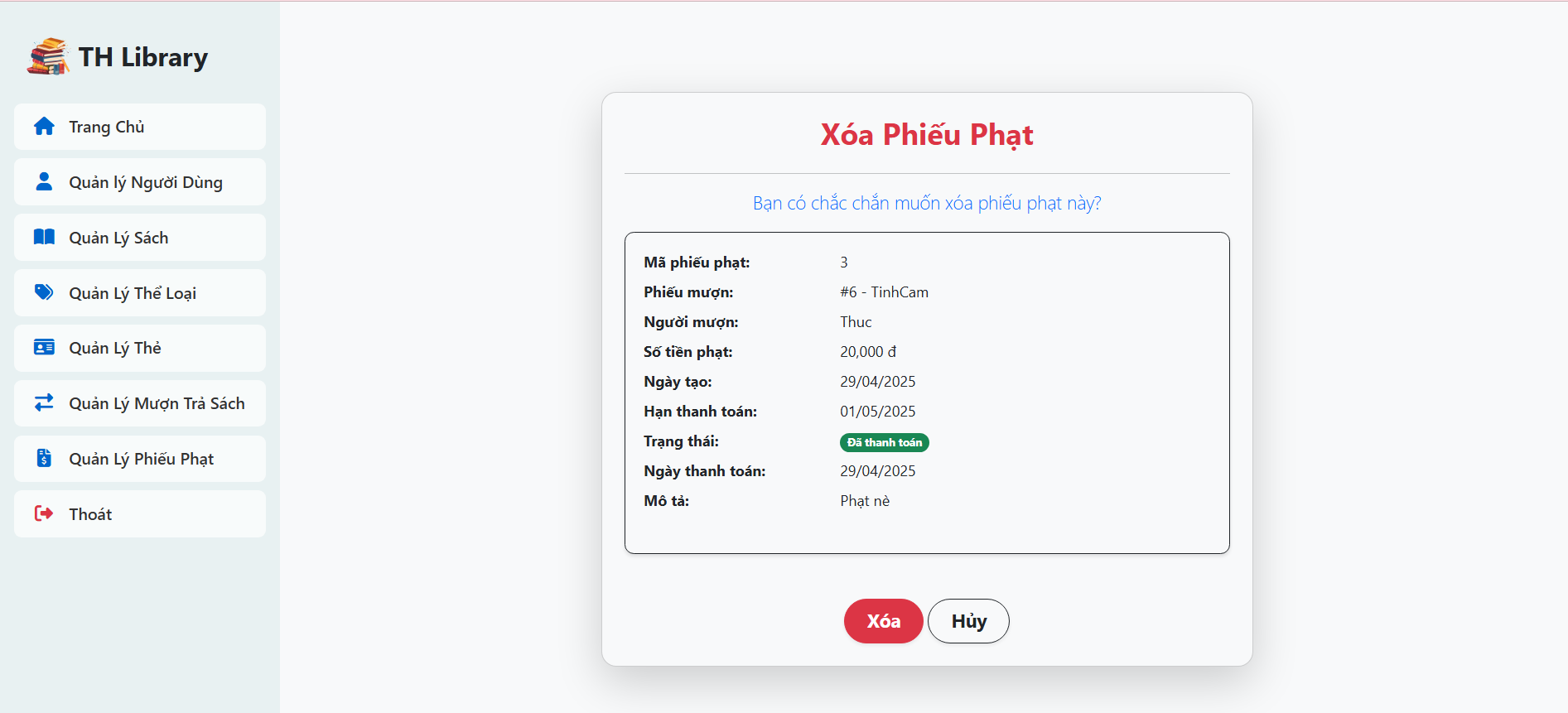
Hình 41. Sửa thông tin phiếu phạt



Hình 42. Chi tiết phiếu phạt



Hình 43. Khi người dùng đã thanh toán phiếu phạt thì sẽ hiện đã thanh toán



Hình 44. Xóa phiếu phạt khi đã thanh toán

**3.3.7 Thoát**

Sẽ thoát ra khỏi trang quản trị và về trang người dùng

# **CHƯƠNG 4: TỔNG KẾT**

## **4.1 Kết quả.**

Trong quá trình phát triển hệ thống quản lý thư viện, nhóm đã đạt được nhiều kết quả tích cực. Trước hết, hệ thống đã được thiết kế và triển khai thành công, đáp ứng đầy đủ các yêu cầu chức năng như quản lý sách, quản lý độc giả, mượn-trả sách, và tra cứu thông tin. Hệ thống hỗ trợ hiệu quả việc lưu trữ, cập nhật và theo dõi thông tin tài liệu cũng như trạng thái mượn trả, giúp quá trình quản lý trở nên chính xác và thuận tiện hơn.

Việc áp dụng công nghệ hiện đại đã góp phần nâng cao hiệu suất làm việc của cán bộ thư viện, giảm thời gian xử lý thủ công, đồng thời mang đến trải nghiệm thân thiện cho người dùng trong việc tra cứu và mượn sách. Các tính năng như tìm kiếm thông minh, phân loại sách theo danh mục, hay hiển thị thông tin sách nổi bật giúp bạn đọc dễ dàng tiếp cận nguồn tài liệu phù hợp với nhu cầu.

Bên cạnh đó, hệ thống còn hỗ trợ tạo các báo cáo thống kê như số lượt mượn sách, lượng sách tồn kho, độc giả mượn nhiều nhất,… giúp ban quản lý có cái nhìn tổng quan và đưa ra các chiến lược phát triển thư viện hợp lý. Ngoài ra, chức năng quản lý tài khoản người dùng và hỗ trợ độc giả theo dõi lịch sử mượn trả cũng góp phần tăng tính tiện lợi và sự gắn kết giữa thư viện và người đọc.

## **4.2 Hạn chế của đề tài.**

Mặc dù đã đạt được nhiều kết quả khả quan, hệ thống vẫn còn một số hạn chế cần khắc phục. Đầu tiên, việc làm quen và sử dụng hệ thống có thể gây khó khăn cho một số cán bộ thư viện lớn tuổi hoặc chưa quen với công nghệ, dẫn đến hiệu suất chưa tối ưu trong giai đoạn đầu vận hành.

Thứ hai, hệ thống hiện tại chưa hỗ trợ tích hợp với một số phần mềm khác như phần mềm kế toán hoặc quản lý tài chính, điều này khiến việc đồng bộ dữ liệu và lập kế hoạch tài nguyên thư viện còn hạn chế.

Ngoài ra, khi số lượng người dùng truy cập và tra cứu tăng cao trong các đợt cao điểm (đăng ký học kỳ mới, thi cử), hệ thống có thể gặp tình trạng chậm trễ hoặc quá tải. Cuối cùng, hệ thống cần được cập nhật định kỳ để bắt kịp với xu hướng công nghệ mới và đáp ứng những yêu cầu ngày càng cao của người dùng.

## **4.3 Hướng phát triển.**

Để hệ thống quản lý thư viện ngày càng hoàn thiện hơn, cần triển khai một số hướng phát triển trong tương lai. Trước hết, có thể tích hợp các công nghệ mới như trí tuệ nhân tạo (AI) nhằm đề xuất tài liệu phù hợp cho từng người dùng, hoặc sử dụng học máy (machine learning) để phân tích hành vi đọc và tối ưu hóa trải nghiệm người dùng.

Tiếp theo, phát triển phiên bản ứng dụng di động hoặc giao diện thân thiện với thiết bị di động sẽ giúp bạn đọc dễ dàng tra cứu sách, theo dõi tình trạng mượn trả, hoặc đăng ký mượn sách trực tuyến mọi lúc mọi nơi. Đồng thời, tối ưu hóa cơ sở dữ liệu và quy trình xử lý để đảm bảo tốc độ và hiệu suất hoạt động của hệ thống, đặc biệt trong các thời điểm truy cập cao.

Về mặt bảo mật, hệ thống cần được nâng cấp các tính năng như mã hóa dữ liệu, phân quyền truy cập và xác thực hai lớp nhằm bảo vệ thông tin người dùng và tài liệu thư viện khỏi các mối đe dọa mạng.

Cuối cùng, việc tổ chức các buổi hướng dẫn sử dụng và xây dựng bộ phận hỗ trợ kỹ thuật thường trực sẽ giúp nhân viên và người dùng tiếp cận, khai thác hiệu quả hệ thống hơn.

## **4.4 Kết luận.**

Tóm lại, hệ thống quản lý thư viện không chỉ mang lại hiệu quả trong công tác quản lý tài nguyên và phục vụ bạn đọc mà còn là nền tảng công nghệ có tiềm năng phát triển mạnh mẽ. Những thành quả đã đạt được, cùng với việc nhận diện rõ những hạn chế và định hướng cải tiến hợp lý, sẽ giúp hệ thống ngày càng hoàn thiện và trở thành công cụ không thể thiếu trong quá trình chuyển đổi số ngành thư viện.

# **TÀI LIỆU THAM KHẢO**

<https://luanvan.net.vn/luan-van/de-tai-xay-dung-website-quan-ly-thu-vien-truong-dai-hoc-dien-luc-63978/>

<https://fr.slideshare.net/slideshow/thit-k-xy-dng-phn-mm-qun-l-th-vindoc/262534462>

<https://adsmo.vn/thiet-ke-website-quan-ly-thu-vien/>

<https://web4s.vn/thiet-ke-website-quan-ly-thu-vien.html?srsltid=AfmBOop1kM2tdIPGfiIeUx_NUKMhkDRTfcKKs3z4yTTEN0LcqphPTXSF>

<https://giaiphapweb.vn/thiet-ke-website-quan-ly-thu-vien/>

<https://www.w3schools.com/Html/>

<http://w3schools.com/css/>

<https://www.w3schools.com/js/>

<https://getbootstrap.com/>