

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  
**KHOA CÔNG NGHỆ THÔNG TIN**

-----



**BÁO CÁO ĐỒ ÁN CHUYÊN NGÀNH**  
**MÃ HỌC PHẦN: IT6122**

**Đề tài: Xây dựng website quản lý mượn trả sách cho thư viện**  
**Trường ĐHCNHN**

**GVHD:** TS. Phạm Văn Hiệp

**Nhóm sinh viên thực hiện:**

- |                      |                   |
|----------------------|-------------------|
| 1. Nguyễn Đức Thắng  | Mã SV: 2021604798 |
| 2. Phạm Đức Thắng    | Mã SV: 2021604984 |
| 3. Lê Công Thành     | Mã SV: 2021607917 |
| 4. Nguyễn Xuân Thành | Mã SV: 2021603021 |

**Mã lớp:** 20241IT6122001    **Khóa:** 16    **Nhóm:** 21

**Hà Nội – Năm 2024**

## LỜI NÓI ĐẦU

Trong bối cảnh các hoạt động quản lý ngày càng phức tạp và yêu cầu cao về tính chính xác, việc ứng dụng công nghệ thông tin vào quản lý là xu hướng tất yếu. Đặc biệt, với những thư viện có số lượng tài nguyên lớn như tại Trường Đại học Công nghiệp Hà Nội, việc số hóa quy trình quản lý mượn trả sách không chỉ giúp tối ưu hóa hiệu suất làm việc mà còn đảm bảo tính minh bạch và thuận tiện cho cả người dùng lẫn cán bộ thư viện.

Với mục tiêu giải quyết những hạn chế của hệ thống quản lý thủ công và mang lại một công cụ quản lý hiện đại, nhóm chúng em đã lựa chọn thực hiện đề tài: ***“Xây dựng website quản lý mượn trả sách cho thư viện Trường ĐHCNHN”***. Sản phẩm không chỉ tập trung vào cải thiện quy trình quản lý mà còn cung cấp trải nghiệm dễ sử dụng, hỗ trợ tìm kiếm thông tin nhanh chóng và nâng cao hiệu quả vận hành.

Hành trình thực hiện đề tài là khoảng thời gian ý nghĩa, trong đó chúng em nhận được sự hướng dẫn tận tình từ TS. Phạm Văn Hiệp và sự hỗ trợ từ khoa CNTT Trường Đại học Công nghiệp Hà Nội. Chúng em xin gửi lời cảm ơn chân thành đến quý Thầy Cô, những người đã đồng hành và giúp đỡ nhóm trong quá trình thực hiện.

Nhóm chúng em xin chân thành cảm ơn!

## MỤC LỤC

<b>LỜI NÓI ĐẦU .....</b>	<b>i</b>
<b>MỤC LỤC.....</b>	<b>ii</b>
<b>DANH MỤC HÌNH VẼ .....</b>	<b>iii</b>
<b>DANH MỤC BẢNG BIỂU .....</b>	<b>v</b>
<b>MỞ ĐẦU .....</b>	<b>1</b>
<b>CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN .....</b>	<b>4</b>
1.1. Giới thiệu chung .....	4
1.2. Nghiên cứu hiện trạng và giải quyết vấn đề.....	5
1.3. Công nghệ, ngôn ngữ được sử dụng .....	10
<b>CHƯƠNG 2: KẾT QUẢ NGHIÊN CỨU.....</b>	<b>15</b>
2.1. Phân tích hệ thống .....	15
2.2. Các tác nhân của hệ thống.....	16
2.3. Biểu đồ usecase .....	17
2.4. Mô tả chi tiết các usecase.....	22
2.5. Biểu đồ trình tự các usecase.....	34
2.6. Thiết kế cơ sở dữ liệu .....	41
2.7. Thiết kế giao diện .....	45
2.8. Kết quả giao diện.....	51
2.9. Kiểm thử ứng dụng.....	55
2.10. Cài đặt và triển khai ứng dụng.....	56
<b>CHƯƠNG 3: KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM .....</b>	<b>67</b>
3.1. Kiến thức đã học được.....	67
3.2. Bài học kinh nghiệm.....	67
3.3. Tính khả thi của đề tài .....	68
3.4. Những thuận lợi và khó khăn .....	68
<b>TÀI LIỆU THAM KHẢO .....</b>	<b>69</b>

## DANH MỤC HÌNH VẼ

Hình 1.1: Cơ cấu tổ chức thư viện Trường ĐHCNHN .....	8
Hình 1.2: Giới thiệu về ASP.NET Core API .....	11
Hình 1.3: Mô hình hoạt động của framework Next.js .....	12
Hình 2.1: Sơ đồ hệ thống .....	15
Hình 2.2: Sơ đồ luồng dữ liệu .....	16
Hình 2.3: Biểu đồ usecase tổng quát.....	18
Hình 2.4: Biểu đồ usecase Quản lý sách.....	19
Hình 2.5: Biểu đồ usecase Quản lý phiếu mượn/trả sách .....	19
Hình 2.6: Biểu đồ usecase Quản lý thành viên .....	20
Hình 2.7: Biểu đồ usecase Quản lý nhân viên .....	20
Hình 2.8: Biểu đồ usecase Quản lý tủ sách.....	21
Hình 2.9: Biểu đồ usecase Xem báo cáo thống kê.....	21
Hình 2.10: Biểu đồ trình tự usecase Đăng nhập .....	35
Hình 2.11: Biểu đồ trình tự usecase Đăng ký .....	35
Hình 2.12: Biểu đồ trình tự usecase Quản lý sách.....	36
Hình 2.13: Biểu đồ trình tự usecase Quản lý phiếu mượn trả sách .....	37
Hình 2.14: Biểu đồ trình tự usecase Quản lý tủ sách.....	38
Hình 2.15: Biểu đồ trình tự usecase Quản lý thành viên .....	39
Hình 2.16: Biểu đồ trình tự usecase Quản lý danh mục .....	40
Hình 2.17: Sơ đồ dữ liệu quan hệ.....	41
Hình 2.18: Hình dung màn hình giao diện Quản lý sách.....	45
Hình 2.19: Hình dung màn hình giao diện Thêm sách mới.....	46
Hình 2.20: Hình dung màn hình giao diện Xóa sách.....	46
Hình 2.21: Hình dung màn hình giao diện Quản lý phiếu mượn .....	47
Hình 2.22: Hình dung màn hình giao diện Thêm phiếu mượn.....	47
Hình 2.23: Hình dung màn hình giao diện Xóa phiếu mượn.....	48

Hình 2.24: Hình dung màn hình giao diện Quản lý danh mục .....	48
Hình 2.25: Hình dung màn hình giao diện Thêm danh mục mới .....	49
Hình 2.26: Hình dung màn hình giao diện Quản lý tử/ngăn sách.....	49
Hình 2.27: Hình dung màn hình giao diện Thêm tử/ngăn sách mới.....	50
Hình 2.28: Hình dung màn hình giao diện Quản lý nhân sự .....	50
Hình 2.29: Hình dung màn hình giao diện Thêm nhân sự.....	51
Hình 2.30: Hình dung màn hình giao diện Quản lý thành viên .....	51
Hình 2.31: Kết quả giao diện màn hình Trang chủ.....	52
Hình 2.32: Kết quả giao diện màn hình Quản lý phiếu mượn.....	52
Hình 2.33: Kết quả giao diện màn hình Quản lý tiền quỹ .....	53
Hình 2.34: Kết quả giao diện màn hình Quản lý sách .....	53
Hình 2.35: Kết quả giao diện màn hình Quản lý tử/ngăn sách .....	54
Hình 2.36: Kết quả giao diện màn hình Quản lý danh mục.....	54
Hình 2.37: Kết quả giao diện màn hình Quản lý nhân sự.....	55
Hình 2.38: Kết quả giao diện màn hình Quản lý thành viên .....	55

**DANH MỤC BẢNG BIỂU**

Bảng 2.1: Bảng sách.....	42
Bảng 2.2: Bảng ngăn sách.....	42
Bảng 2.3: Bảng tủ sách.....	43
Bảng 2.4: Bảng danh mục .....	43
Bảng 2.5: Bảng người dùng .....	43
Bảng 2.6: Bảng lịch sử hoạt động người dùng.....	44
Bảng 2.7: Bảng token đăng nhập .....	44
Bảng 2.8: Bảng phiếu mượn sách .....	44
Bảng 2.9: Bảng chi tiết phiếu mượn .....	45
Bảng 2.10: Bảng kế hoạch kiểm thử .....	55

## MỞ ĐẦU

### 1. Lý do chọn đề tài:

Việc quản lý mượn trả sách trong các thư viện truyền thống thường gặp nhiều khó khăn, đặc biệt là khi số lượng đầu sách và người dùng lớn. Hệ thống thủ công không chỉ tốn thời gian mà còn dễ xảy ra sai sót trong việc kiểm soát tình trạng sách, theo dõi thời hạn mượn trả, và quản lý người dùng.

Vì vậy, nhóm quyết định chọn đề tài “Xây dựng website quản lý mượn trả sách cho thư viện Trường ĐHCNHN” nhằm tạo ra một giải pháp số hóa quy trình quản lý thư viện. Hệ thống sẽ giúp tối ưu hóa việc quản lý mượn trả sách, đảm bảo tính chính xác, minh bạch và thuận tiện hơn cho cả người dùng lẫn cán bộ thư viện. Đồng thời, đề tài cũng đáp ứng được nhu cầu thực tiễn tại thư viện Trường ĐHCNHN, nơi lượng sách lớn và nhiều sinh viên cần truy cập hệ thống thường xuyên.

### 2. Mục đích nghiên cứu:

- Tạo ra một hệ thống trực tuyến cho phép quản lý toàn bộ quá trình mượn trả sách một cách dễ dàng, nhanh chóng và chính xác.
- Giảm thiểu sai sót và những vấn đề tồn đọng của quy trình quản lý truyền thống, như thất lạc sách, kiểm soát thời hạn mượn sách và lưu trữ dữ liệu người dùng.
- Nâng cao trải nghiệm của người dùng và cán bộ thư viện, với một giao diện thân thiện, dễ sử dụng, và khả năng truy cập mọi lúc mọi nơi.
- Cung cấp công cụ hỗ trợ quản lý giúp cán bộ thư viện dễ dàng theo dõi số lượng sách, tình trạng sách, và số liệu thống kê về mượn trả, từ đó nâng cao hiệu quả quản lý và phục vụ tốt hơn nhu cầu của người dùng.

### 3. Đối tượng nghiên cứu:

- Hệ thống quản lý mượn trả sách: Các chức năng quản lý sách, tủ sách, người dùng, quy trình mượn trả và theo dõi hạn trả.
- Người dùng hệ thống: Quản trị viên (Admin), thủ thư và người mượn sách (Sinh viên, Giảng viên).
- Cơ sở dữ liệu: Tổ chức và lưu trữ dữ liệu mượn trả, thông tin sách và người dùng bằng SQL Server.
- Công nghệ phát triển: ASP.NET Core Web API, Next.js và ShadcnUI

#### **4. Phạm vi nghiên cứu:**

- Thời gian: Từ ngày 11/09/2024 đến 22/12/2024.
- Địa điểm: Trường Đại học Công nghiệp Hà Nội.
- Nội dung: Nghiên cứu chủ yếu tập trung vào việc xây dựng hệ thống quản lý mượn trả sách trong môi trường học đường, với phạm vi ứng dụng tại thư viện của trường.

#### **5. Kết quả mong muốn:**

- Xây dựng một website quản lý mượn trả sách hoàn chỉnh cho thư viện Trường ĐHCNHN, với giao diện thân thiện và dễ sử dụng.
- Cung cấp chức năng quản lý sách, quản lý người dùng, và quy trình mượn trả sách rõ ràng, hiệu quả.
- Hỗ trợ tính năng tìm kiếm, phân loại sách, và theo dõi tình trạng mượn trả theo thời gian thực.
- Xây dựng hệ thống phân quyền cho quản trị viên và người mượn sách, đảm bảo an toàn và bảo mật thông tin.
- Đảm bảo hệ thống vận hành ổn định, có khả năng mở rộng và bảo trì dễ dàng trong tương lai.

#### **6. Cấu trúc báo cáo:**



Báo cáo thực tập gồm 3 chương:

- Chương 1: Giới thiệu tổng quan.
- Chương 2: Kết quả nghiên cứu.
- Chương 3: Kết luận và bài học kinh nghiệm.

## CHƯƠNG 1: GIỚI THIỆU TỔNG QUAN

### 1.1. Giới thiệu chung

Thư viện là trung tâm tri thức, đóng vai trò quan trọng trong việc hỗ trợ học tập, nghiên cứu và phát triển văn hóa đọc cho sinh viên và giảng viên. Tại Trường Đại học Công nghiệp Hà Nội, thư viện không chỉ cung cấp nguồn tài liệu học thuật phong phú mà còn là nơi tổ chức các hoạt động như hội thảo, chia sẻ học thuật, và hướng dẫn kỹ năng học tập. Tuy nhiên, cùng với sự gia tăng số lượng sinh viên, giảng viên và nhu cầu sử dụng thư viện, việc quản lý tài nguyên một cách hiệu quả đã trở thành một thách thức lớn.

Quy trình quản lý mượn trả sách truyền thống của nhiều thư viện tại Việt Nam thường dựa vào hệ thống ghi chép thủ công hoặc sử dụng phần mềm cơ bản, chưa đáp ứng tốt nhu cầu ngày càng cao về tính tiện lợi, chính xác và khả năng truy cập từ xa. Điều này không chỉ gây khó khăn cho việc kiểm soát số lượng sách, tình trạng mượn trả, mà còn làm mất nhiều thời gian của người dùng và nhân viên thư viện.

Với sự phát triển mạnh mẽ của công nghệ thông tin, việc số hóa các hoạt động quản lý thư viện đã trở thành một xu thế tất yếu. Một hệ thống quản lý mượn trả sách trực tuyến không chỉ giúp tự động hóa các quy trình, mà còn tạo ra trải nghiệm người dùng tối ưu, hỗ trợ sinh viên và giảng viên trong việc tìm kiếm tài liệu nhanh chóng, đồng thời giúp cán bộ thư viện dễ dàng giám sát và quản lý nguồn tài nguyên một cách hiệu quả.

Website quản lý mượn trả sách dành riêng cho thư viện Trường Đại học Công nghiệp Hà Nội được thiết kế với mục tiêu:

- Cải thiện quy trình vận hành: Tự động hóa các khâu đăng ký mượn trả sách, kiểm tra tình trạng sách và gửi thông báo đến người dùng.

- Tối ưu trải nghiệm người dùng: Cho phép sinh viên, giảng viên tra cứu và đặt sách trực tuyến mọi lúc, mọi nơi.
- Nâng cao hiệu quả quản lý: Hỗ trợ cán bộ thư viện theo dõi lịch sử mượn trả, thống kê sách được sử dụng nhiều nhất, và kiểm kê tài sản dễ dàng.

Ngoài ra, hệ thống này còn góp phần hỗ trợ nhà trường trong việc thúc đẩy phong trào đọc sách, nghiên cứu và ứng dụng công nghệ thông tin vào quản lý giáo dục, hướng tới mục tiêu xây dựng một môi trường học tập hiện đại, năng động.

## **1.2. Nghiên cứu hiện trạng và giải quyết vấn đề**

### ***1.2.1. Hiện trạng của thư viện***

#### ***1.2.1.1. Quy mô và chức năng hiện tại của thư viện:***

Thư viện Trường ĐHCNHN hiện là một trung tâm tri thức phục vụ hàng ngàn sinh viên và giảng viên mỗi năm. Với hàng chục ngàn đầu sách bao gồm giáo trình, tài liệu nghiên cứu, sách tham khảo và tài liệu số, thư viện đóng vai trò quan trọng trong việc hỗ trợ giảng dạy và học tập.

Một số đặc điểm chính của thư viện:

- Cơ sở vật chất: Thư viện được trang bị các khu vực đọc sách, phòng tự học, và các kho sách vật lý lưu trữ số lượng lớn tài liệu.
- Nguồn tài nguyên: Tài liệu phong phú bao gồm sách giáo khoa, tài liệu chuyên ngành, tạp chí khoa học và tài nguyên số như e-books, cơ sở dữ liệu học thuật.
- Dịch vụ hỗ trợ: Cung cấp mượn trả sách, hướng dẫn tra cứu tài liệu, và hỗ trợ nghiên cứu.

#### ***1.2.1.2. Quy trình quản lý hiện tại***

Hiện nay, việc quản lý tài nguyên và hoạt động mượn trả sách tại thư viện chủ yếu được thực hiện qua hình thức truyền thống, với một số công cụ hỗ trợ cơ bản. Quy trình cụ thể như sau:

- Đăng ký mượn sách: Sinh viên và giảng viên cần trực tiếp đến quầy dịch vụ, cung cấp thông tin cá nhân và mã số sách để thực hiện mượn sách.
- Trả sách: Người dùng cần đến thư viện đúng thời hạn để hoàn trả sách, tránh bị phạt quá hạn.
- Quản lý thông tin: Thư viện sử dụng Excel hoặc phần mềm quản lý offline để lưu trữ dữ liệu về người mượn, tình trạng sách, và lịch sử mượn trả.

#### *1.2.1.3. Những hạn chế hiện tại*

Dù đóng vai trò quan trọng trong việc hỗ trợ sinh viên và giảng viên, hệ thống quản lý thư viện hiện nay vẫn tồn tại nhiều hạn chế:

- Quản lý thủ công kém hiệu quả:
  - + Quy trình ghi nhận thông tin mượn trả sách thủ công dễ gây ra sai sót, mất mát dữ liệu hoặc ghi nhận không đầy đủ.
  - + Việc kiểm kê kho sách và tình trạng sách mất nhiều thời gian, đặc biệt với khối lượng tài liệu lớn.
- Hạn chế trong tra cứu và tìm kiếm tài liệu:
  - + Sinh viên và giảng viên phải đến trực tiếp thư viện để tìm kiếm sách, không có khả năng tra cứu online.
  - + Việc tra cứu tại chỗ đôi khi mất nhiều thời gian vì tài liệu không được sắp xếp hoặc lưu trữ hợp lý.
- Thời gian xử lý lâu:

- + Thời gian mượn trả sách thường kéo dài do việc nhập liệu thủ công tại quầy dịch vụ.
- + Khi có số lượng lớn người dùng cùng lúc, tình trạng chờ đợi xảy ra thường xuyên.
- Khó khăn trong việc thống kê, báo cáo:
  - + Nhân viên thư viện phải tổng hợp dữ liệu thủ công để lập báo cáo định kỳ, dễ xảy ra sai sót và tốn công sức.

#### *1.2.1.4. Công nghệ hỗ trợ hiện tại*

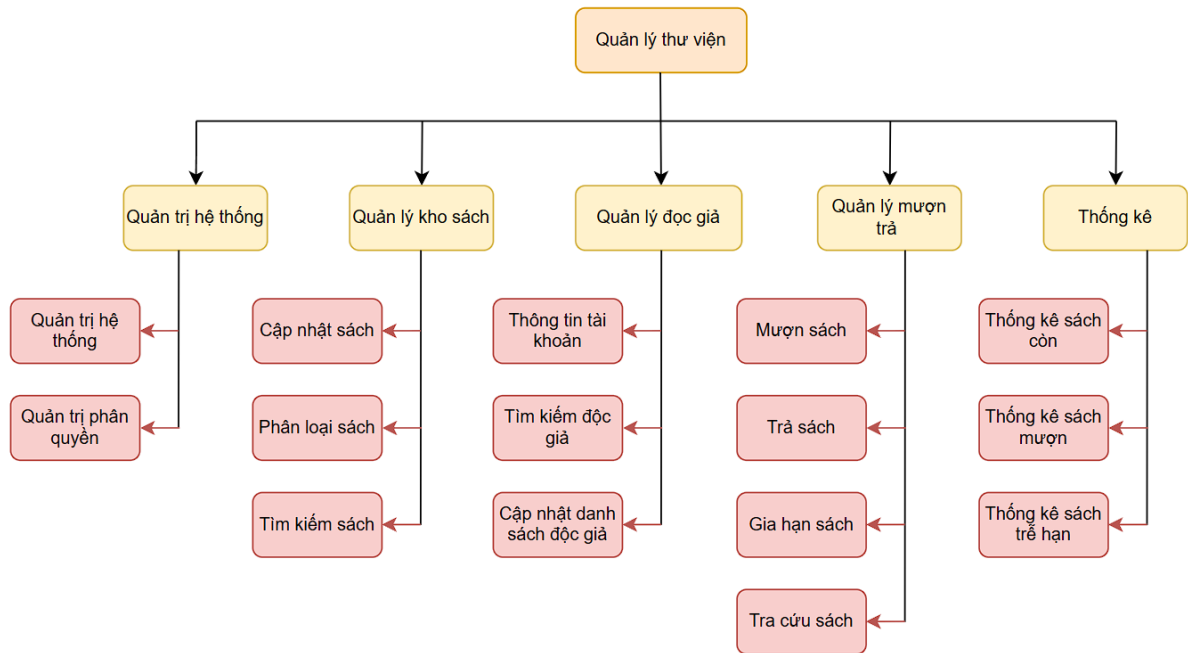
Mặc dù thư viện đã triển khai một số công cụ công nghệ, nhưng các giải pháp này chưa đáp ứng tốt yêu cầu hiện đại hóa:

- Phần mềm quản lý cơ bản: Một số phần mềm nội bộ được sử dụng nhưng không có khả năng tích hợp với các hệ thống khác.
- Thiếu nền tảng trực tuyến: Người dùng không thể tra cứu sách hoặc thực hiện mượn trả trực tuyến, gây ra sự bất tiện lớn.

#### *1.2.1.5. Nhu cầu và thách thức hiện tại*

- Tăng số lượng người dùng: Số lượng sinh viên và giảng viên ngày càng tăng khiến nhu cầu sử dụng thư viện trở nên quá tải.
- Cần hiện đại hóa: Một hệ thống quản lý mượn trả sách trực tuyến là yêu cầu cấp thiết để cải thiện hiệu quả vận hành và tăng trải nghiệm người dùng.
- Khả năng tiếp cận từ xa: Nhu cầu tra cứu tài liệu từ xa ngày càng tăng, đặc biệt trong bối cảnh học tập và nghiên cứu trực tuyến trở nên phổ biến.

#### *1.2.2. Cơ cấu tổ chức*



*Hình 1.1: Cơ cấu tổ chức thư viện Trường ĐHCNH*

### **1.2.3. Hoạt động của các bộ phận**

Các hoạt động nghiệp vụ của một thư viện sách thông thường có thể được tóm tắt như sau:

- Thư viện tạo các thông tin cuốn sách gồm các thông tin: tên tiêu đề sách, tên tác giả, danh mục, tên nhà xuất bản, năm xuất bản, số trang, số lượng hiện có, giá tiền, mô tả. Các đầu sách có thể được phân theo chuyên ngành hoặc tài liệu.
- Độc giả muốn mượn sách thì tra cứu các cuốn sách rồi ghi vào phiếu mượn, hoặc có thể đặt lịch hẹn trước trên website.
- Sau khi kiểm tra đầy đủ thông tin phiếu mượn, thủ thư kiểm tra điều kiện mượn của độc giả và xác nhận cho phép mượn sách. Một số thông tin trong phiếu mượn được lưu lại để quản lý, phiếu mượn sẽ được gài vào chỗ sách được lấy đi, sách được giao cho độc giả.

- Khi độc giả trả sách: xác nhận thẻ độc giả, xác định phiếu mượn, việc trả sách được ghi nhận vào dòng ngày trả và trạng thái phiếu. Phiếu mượn được lưu lại để quản lý và theo dõi.
- Sách bị trả trễ lịch hẹn hoặc hỏng sẽ bị phạt.

#### ***1.2.4. Xác định yêu cầu***

- Yêu cầu chức năng:
  - + Giúp độc giả tra cứu theo loại sách, theo tên sách, theo tác giả, theo danh mục,... trên website.
  - + Cung cấp cho thủ thư thông tin về các đầu sách một độc giả đang mượn và hạn trả, và các cuốn sách còn đang được mượn.
  - + Thống kê hàng tháng số sách cho mượn, số phiếu đang mượn, số phiếu đã trả, số phiếu quá hạn.
  - + Hỗ trợ thủ thư cập nhật thông tin sách, xác nhận cho mượn sách và nhận lại sách khi độc giả trả sách.
  - + Hỗ trợ quản lý các thông tin về độc giả dựa trên tài khoản độc giả, thông tin phiếu mượn.
  - + Hỗ trợ chức năng quản trị chung hệ thống, trong đó người quản trị chung có thể thay đổi thông tin hoặc thêm bớt các thủ thư.
- Yêu cầu phi chức năng:
  - + Độc giả có thể tra cứu thông tin sách trên website của thư viện, có thể yêu cầu phiếu mượn trước. Tuy nhiên, việc mượn và trả sách phải thực hiện trực tiếp trên thư viện. Thủ thư sử dụng hệ thống để cập nhật và quản lý quá trình mượn trả sách.
  - + Thông tin thống kê phải đảm bảo tính chính xác, khách quan. Các hình thức phạt đối với độc giả quá hạn sẽ được lưu lại và thông báo cho độc giả biết.

### 1.3. Công nghệ, ngôn ngữ được sử dụng

#### 1.3.1. Giới thiệu về ASP.NET Core API

**ASP.NET Core API** là một framework phát triển ứng dụng mạnh mẽ, được Microsoft thiết kế dành cho việc xây dựng các dịch vụ web và API hiện đại. Với kiến trúc đa nền tảng, ASP.NET Core API cho phép các nhà phát triển triển khai ứng dụng trên Windows, macOS và Linux, giúp tối ưu hóa khả năng tích hợp với nhiều môi trường làm việc khác nhau.

ASP.NET Core bao gồm các thành phần theo hướng module nhằm tối thiểu tài nguyên và chi phí phát triển, như vậy có thể giữ lại được sự mềm dẻo trong việc xây dựng giải pháp. Điểm nổi bật của ASP.NET Core API là hiệu suất cao, khả năng mở rộng dễ dàng và sự hỗ trợ đầy đủ cho các tiêu chuẩn hiện đại như RESTful, giúp xây dựng các dịch vụ API linh hoạt và dễ dàng quản lý.

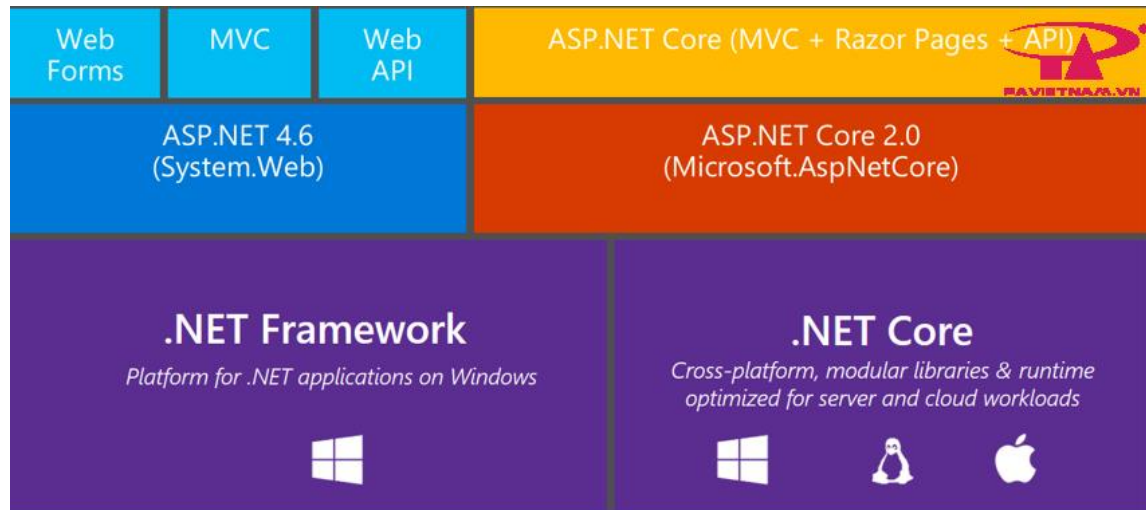
Ngoài ra, ASP.NET Core API còn tích hợp các công nghệ như Dependency Injection, Middleware và Entity Framework Core để tăng cường tính linh hoạt và hiệu quả trong phát triển. Hệ thống bảo mật mạnh mẽ với sự hỗ trợ tích hợp cho các phương thức xác thực như OAuth, JWT và Identity Server giúp các ứng dụng API dễ dàng quản lý quyền truy cập và bảo mật dữ liệu. Nhờ vào khả năng mở rộng và hiệu suất vượt trội, ASP.NET Core API trở thành lựa chọn lý tưởng cho các dự án từ nhỏ đến lớn, đáp ứng mọi nhu cầu của doanh nghiệp và nhà phát triển hiện đại.

Các thành phần của ASP.NET Core:

- Ngôn ngữ: ASP.NET sử dụng nhiều ngôn ngữ lập trình khác nhau như VB.NET và C#.
- Thư viện: ASP.NET có bộ thư viện chuẩn bao gồm các giao diện, các lớp và kiểu giá trị. Bộ thư viện này có thể sử dụng lại cho quá trình phát triển ASP.NET và xây dựng các chức năng của hệ thống.



- Thời gian chạy ngôn ngữ chung (CLR): CLR – Common Language Runtime được sử dụng để thực hiện các hoạt động mã. Các hoạt động này sẽ thực hiện xử lý các ngoại lệ và thu gom rác.



Hình 1.2: Giới thiệu về ASP.NET Core API

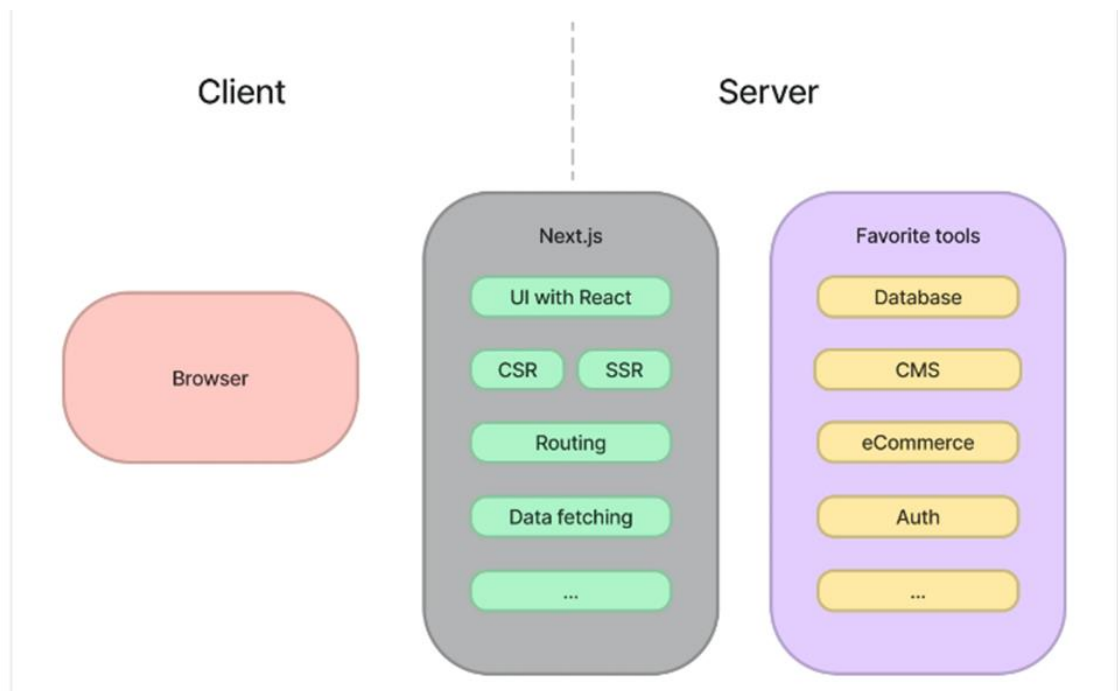
### 1.3.2. Giới thiệu về Next.js và ShadcnUI

#### 1.3.2.1. Next.js:

**Next.js** là một framework mã nguồn mở dựa trên React, giúp xây dựng các ứng dụng web hiện đại, tối ưu hóa hiệu suất và nâng cao trải nghiệm người dùng. Dưới đây là những điểm nổi bật của Next.js:

- **Server-Side Rendering (SSR):** Next.js hỗ trợ việc render trang web phía server, giúp cải thiện hiệu suất tải trang và tối ưu SEO cho các ứng dụng web.
- **Static Site Generation (SSG):** Hỗ trợ tạo các trang tĩnh một cách nhanh chóng và hiệu quả, lý tưởng cho các ứng dụng cần tối ưu hóa thời gian tải và SEO.
- **Incremental Static Regeneration (ISR):** Cho phép cập nhật các trang tĩnh mà không cần phải build lại toàn bộ ứng dụng, tiết kiệm thời gian và tài nguyên hệ thống.

- **Tính năng tự động phân trang:** Next.js tự động phân trang và tối ưu hóa các trang web, giúp tăng hiệu suất ứng dụng.
- **API Routes:** Hỗ trợ dễ dàng xây dựng các API ngay trong cùng một ứng dụng, giúp đơn giản hóa việc phát triển full-stack.
- **Tính linh hoạt trong phát triển:** Các nhà phát triển có thể chọn giữa Server-Side Rendering (SSR) và Client-Side Rendering (CSR) tùy vào từng tình huống.



*Hình 1.3: Mô hình hoạt động của framework Next.js*

Với các tính năng vượt trội này, Next.js giúp các nhà phát triển xây dựng các ứng dụng web hiệu quả, dễ dàng duy trì và mở rộng, đồng thời cung cấp một trải nghiệm người dùng mượt mà và nhanh chóng.

#### 1.3.2.2. ShadcnUI:

**ShadcnUI** là một thư viện giao diện người dùng (UI) dành cho các ứng dụng React, cung cấp các thành phần UI đẹp mắt và linh hoạt, giúp các nhà phát triển

dễ dàng tạo ra giao diện trực quan cho ứng dụng. Các đặc điểm chính của ShadcnUI bao gồm:

- **Thư viện UI dễ sử dụng:** ShadcnUI cung cấp một bộ các thành phần giao diện phổ biến như modal, tooltip, button, card, và form, giúp các nhà phát triển không phải tốn thời gian tạo ra các thành phần từ đầu.
- **Tính tương thích với React và Next.js:** Thư viện được thiết kế để dễ dàng tích hợp vào các dự án React và Next.js, giúp việc phát triển giao diện trở nên dễ dàng và nhanh chóng.
- **Tùy chỉnh linh hoạt:** Các thành phần UI trong ShadcnUI có thể dễ dàng tùy chỉnh về màu sắc, kích thước và kiểu dáng, giúp đáp ứng nhu cầu thiết kế riêng của ứng dụng.
- **Tính nhất quán trong thiết kế:** ShadcnUI cung cấp các thành phần được thiết kế sẵn với phong cách hiện đại, đồng nhất, giúp tạo ra một giao diện người dùng mượt mà và dễ nhìn.
- **Hỗ trợ hiệu suất cao:** Các thành phần của ShadcnUI được tối ưu hóa cho hiệu suất, đảm bảo trải nghiệm người dùng mượt mà ngay cả khi sử dụng trên các thiết bị di động.

Với ShadcnUI, các nhà phát triển có thể tạo ra giao diện người dùng đẹp và linh hoạt mà không cần phải lo lắng về việc thiết kế và tối ưu hóa từng thành phần, giúp tiết kiệm thời gian và công sức trong quá trình phát triển ứng dụng.

### 1.3.3. Giới thiệu về SQL Server:

**SQL Server** là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mạnh mẽ và phổ biến do Microsoft phát triển. Được thiết kế để lưu trữ và quản lý dữ liệu, SQL Server hỗ trợ các thao tác truy vấn dữ liệu mạnh mẽ thông qua ngôn ngữ SQL (Structured Query Language), giúp các nhà phát triển dễ dàng thao tác với cơ sở

dữ liệu. SQL Server cung cấp các tính năng như bảo mật cao, khả năng phục hồi, sao lưu và phục hồi dữ liệu, cùng với khả năng xử lý giao dịch với tính toàn vẹn và tính nhất quán dữ liệu.

SQL Server hỗ trợ nhiều tính năng cao cấp như:

- Phân tích dữ liệu (data warehousing).
- Báo cáo (reporting)
- Các công cụ quản lý cơ sở dữ liệu tích hợp (SQL Server Management Studio).

Nó cũng hỗ trợ các kỹ thuật như:

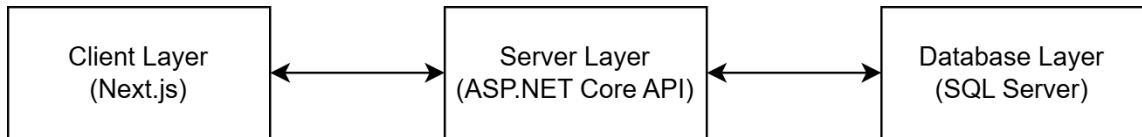
- Replication.
- Full-text search.
- Các chỉ mục toàn văn (full-text indexing) giúp cải thiện hiệu suất và khả năng mở rộng cho các ứng dụng có lượng dữ liệu lớn.

Với các tính năng mở rộng và khả năng tích hợp tốt với các sản phẩm của Microsoft như .NET, SQL Server là một trong những lựa chọn hàng đầu cho các ứng dụng doanh nghiệp và các hệ thống cơ sở dữ liệu phức tạp.

## CHƯƠNG 2: KẾT QUẢ NGHIÊN CỨU

### 2.1. Phân tích hệ thống

#### 2.1.1. Sơ đồ hệ thống

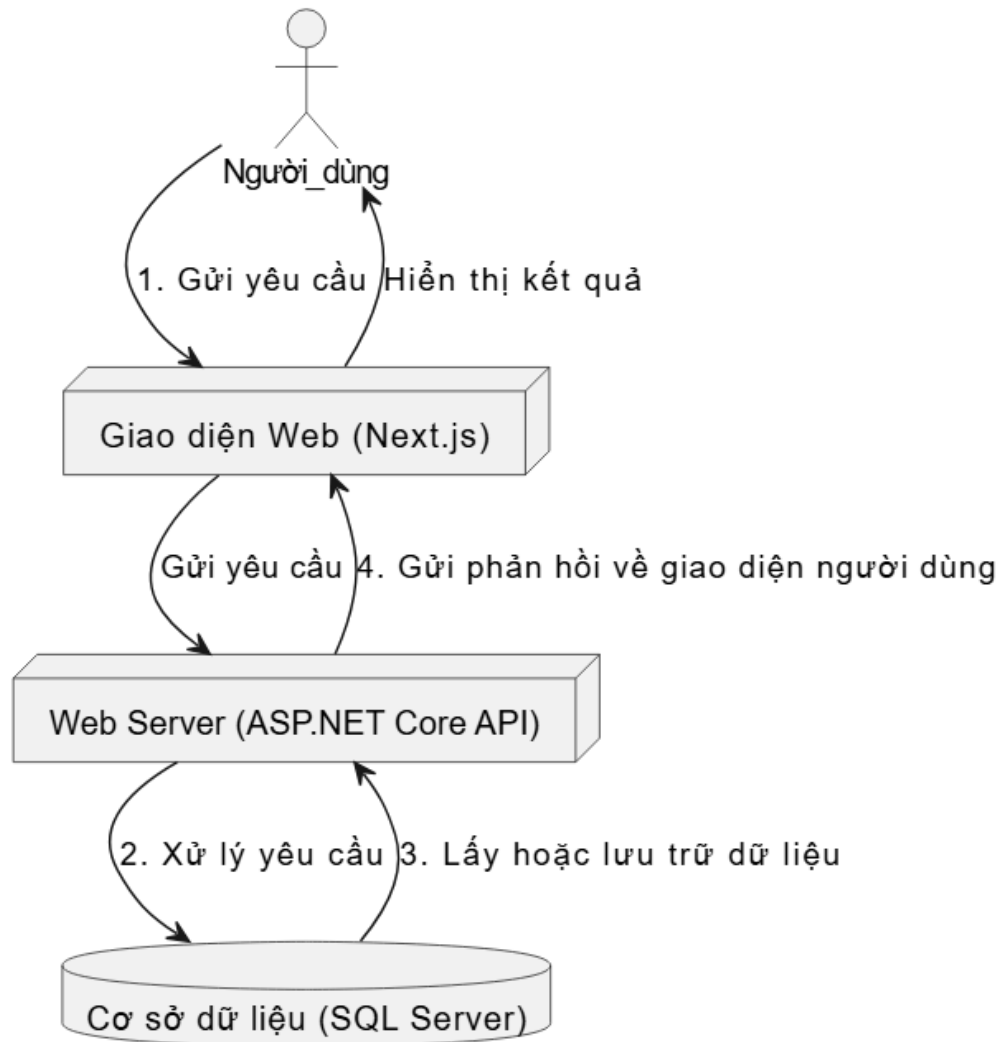


*Hình 2.1: Sơ đồ hệ thống*

- Client Layer (Lớp giao diện người dùng):
  - + Phát triển bằng Next.js và ShadcnUI.
  - + Cho phép người dùng tương tác với ứng dụng để thực hiện các chức năng như quản lý tài khoản, mượn/trả sách và tra cứu thông tin.
  - + Gửi yêu cầu đến Server Layer và nhận phản hồi dưới dạng JSON hoặc HTML để hiển thị dữ liệu.
- Server Layer (Lớp máy chủ):
  - + Phát triển bằng ASP.NET Core API, chịu trách nhiệm xử lý các yêu cầu từ Client Layer.
  - + Thực hiện logic nghiệp vụ, đảm bảo tính toàn vẹn của dữ liệu và các quy tắc nghiệp vụ, ví dụ: kiểm tra quyền hạn trước khi thực hiện hành động.
  - + Tích hợp xác thực và phân quyền dựa trên vai trò (Người dùng, Thủ thư và Quản trị viên).
  - + Kết nối và giao tiếp với Database Layer để truy vấn, lưu trữ dữ liệu cần thiết.
- Database Layer (Lớp CSDL):

- + Sử dụng SQL Server để lưu trữ dữ liệu về người dùng, sách, ngân sách, phiếu mượn/trả, và các thông tin liên quan.
- + Đảm bảo tính nhất quán và an toàn của dữ liệu thông qua các ràng buộc và quy tắc trong cơ sở dữ liệu.
- + Phục vụ các truy vấn nhanh chóng để đáp ứng yêu cầu từ Server Layer.

### 2.1.2. Sơ đồ luồng dữ liệu



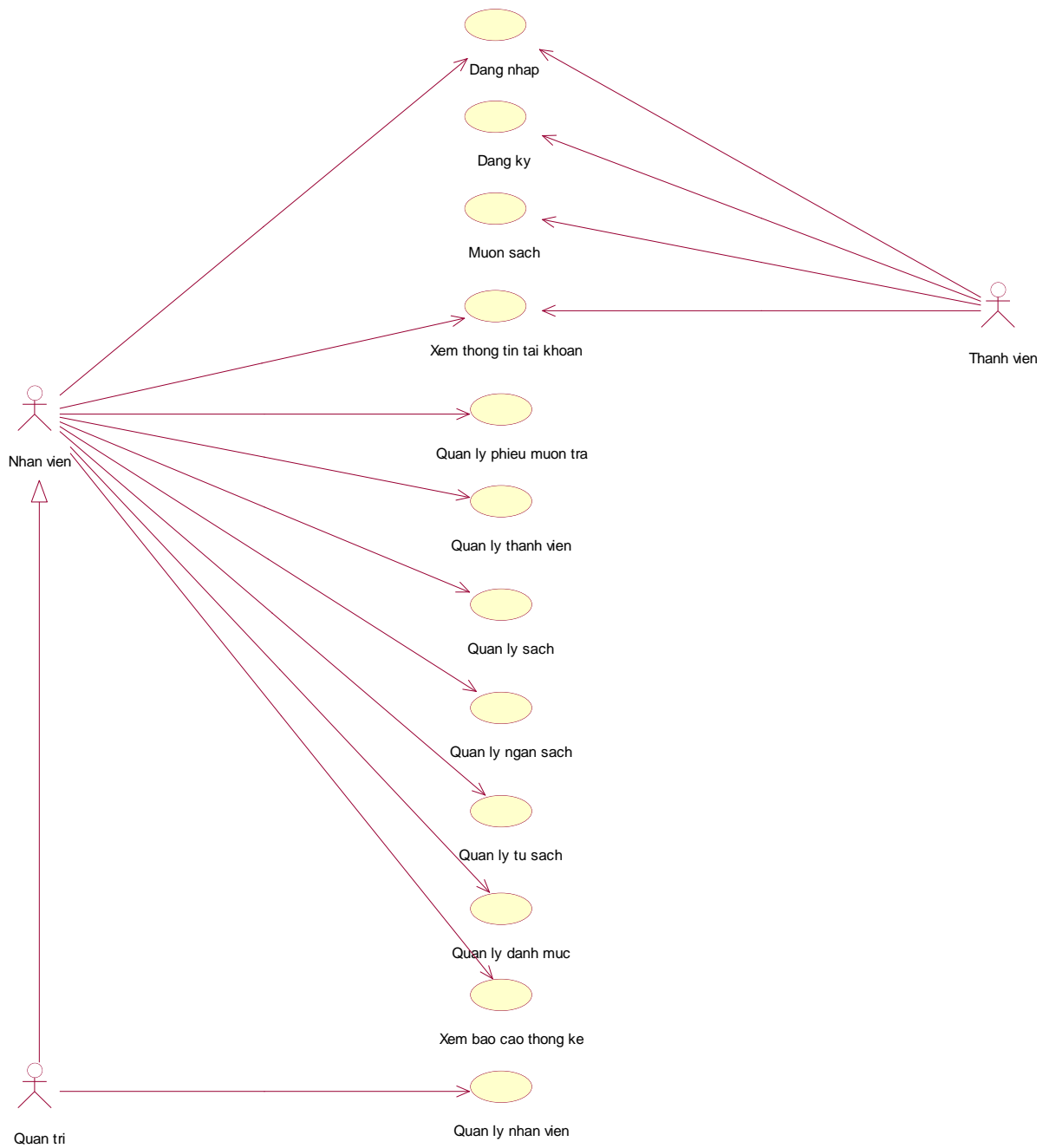
Hình 2.2: Sơ đồ luồng dữ liệu

## 2.2. Các tác nhân của hệ thống

- Thành viên (Member): Là những người dùng cuối sử dụng hệ thống để thực hiện các hoạt động mượn/trả sách. Các chức năng chính:
  - + Đăng ký tài khoản, đăng nhập.
  - + Xem và chỉnh sửa thông tin cá nhân.
  - + Tìm kiếm sách và đăng ký mượn sách.
- Thủ thư (Librarian): Là những người quản lý các hoạt động vận hành của thư viện. Các chức năng chính:
  - + Quản lý sách, ngăn sách, và tủ sách.
  - + Quản lý danh mục sách.
  - + Quản lý thông tin thành viên (Member).
  - + Theo dõi và xử lý phiếu mượn/trả.
  - + Xem báo cáo và thống kê.
- Quản trị (Admin): Là những người có quyền cao nhất trong hệ thống, chịu trách nhiệm quản lý toàn bộ hệ thống thư viện. Các chức năng chính:
  - + Thực hiện tất cả các chức năng của Librarian.
  - + Quản lý thông tin của Librarian.
  - + Xem và phân tích báo cáo, thống kê toàn diện.
- Mối quan hệ giữa các tác nhân:
  - + Thành viên tương tác chủ yếu với giao diện hệ thống để thực hiện các thao tác liên quan đến mượn/trả sách.
  - + Thủ thư và Quản trị quản lý dữ liệu và đảm bảo hoạt động của hệ thống thông qua các chức năng nâng cao.
  - + Các quyền hạn của tác nhân được xác định rõ ràng và phân chia phù hợp với vai trò trong hệ thống.

### 2.3. Biểu đồ usecase

### 2.3.1. Biểu đồ usecase tổng quát

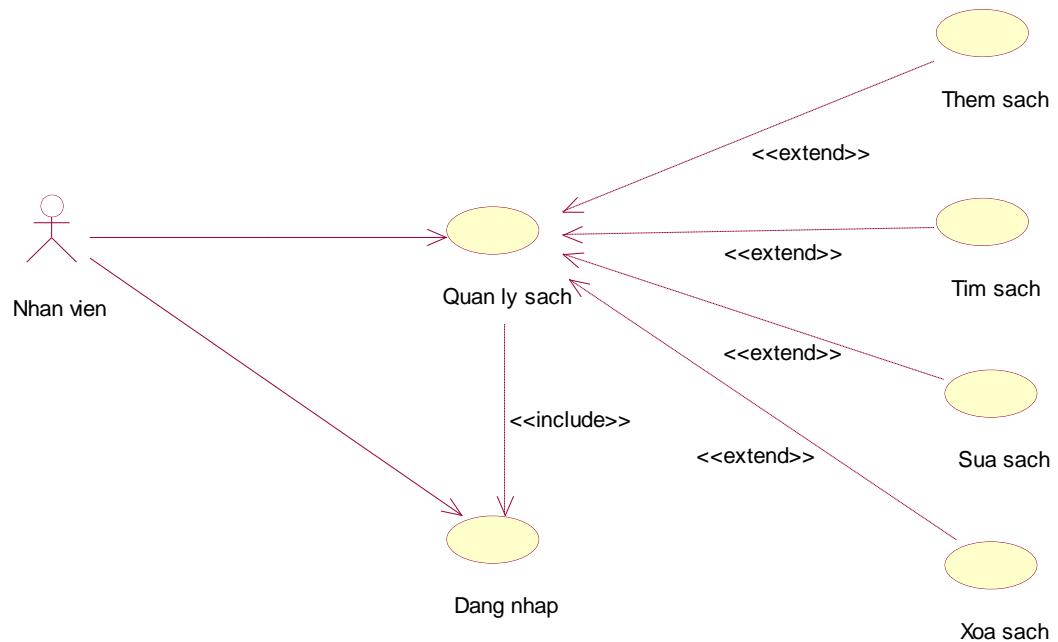


Hình 2.3: Biểu đồ usecase tổng quát

### 2.3.2. Biểu đồ usecase phân rã

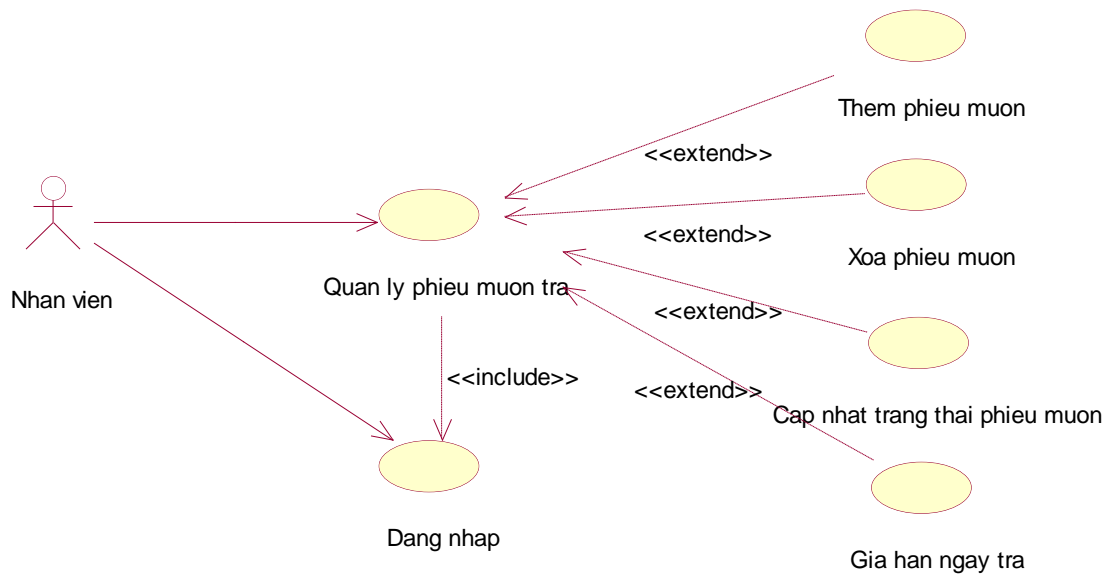
- Biểu đồ usecase Quản lý sách:





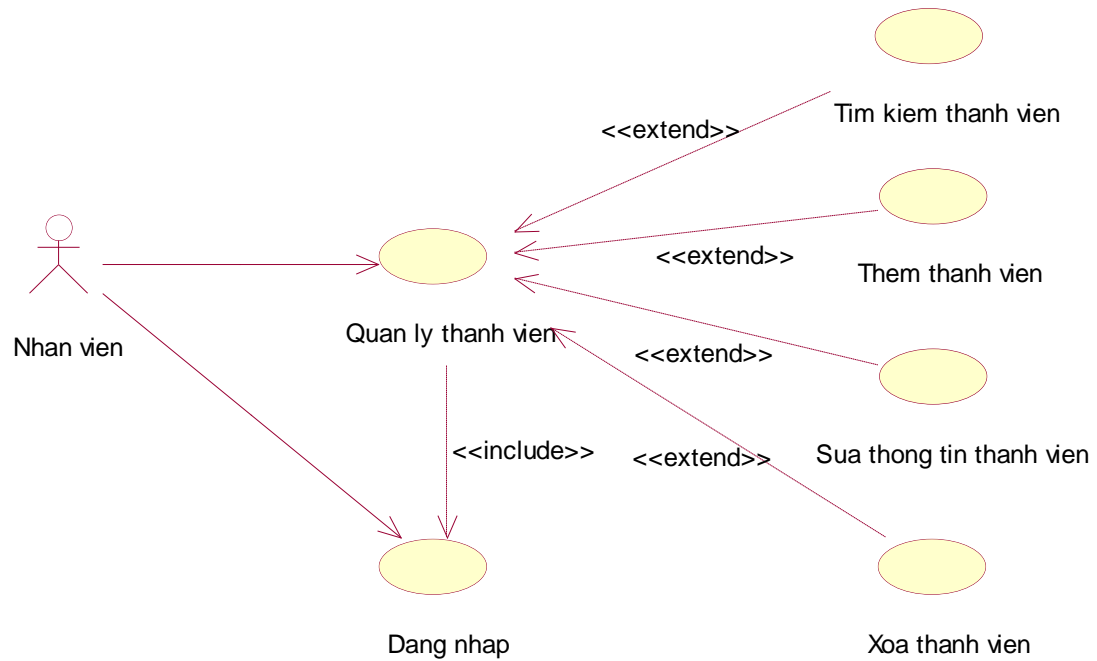
Hình 2.4: Biểu đồ usecase Quản lý sách

- Biểu đồ usecase Quản lý phiếu mượn/trả sách:



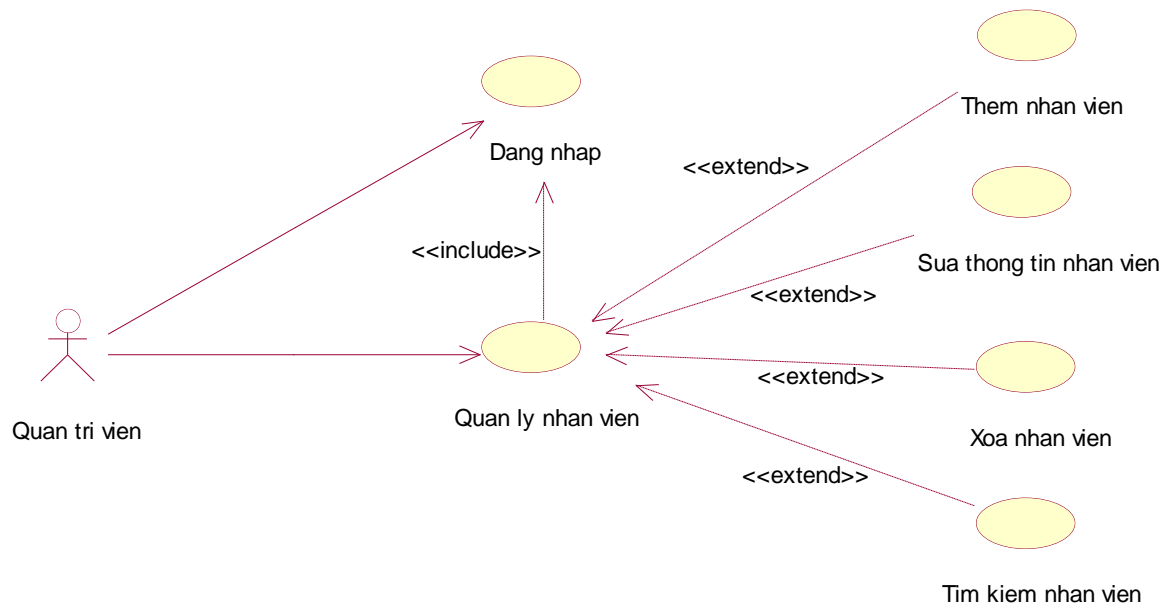
Hình 2.5: Biểu đồ usecase Quản lý phiếu mượn/trả sách

- Biểu đồ usecase Quản lý thành viên:



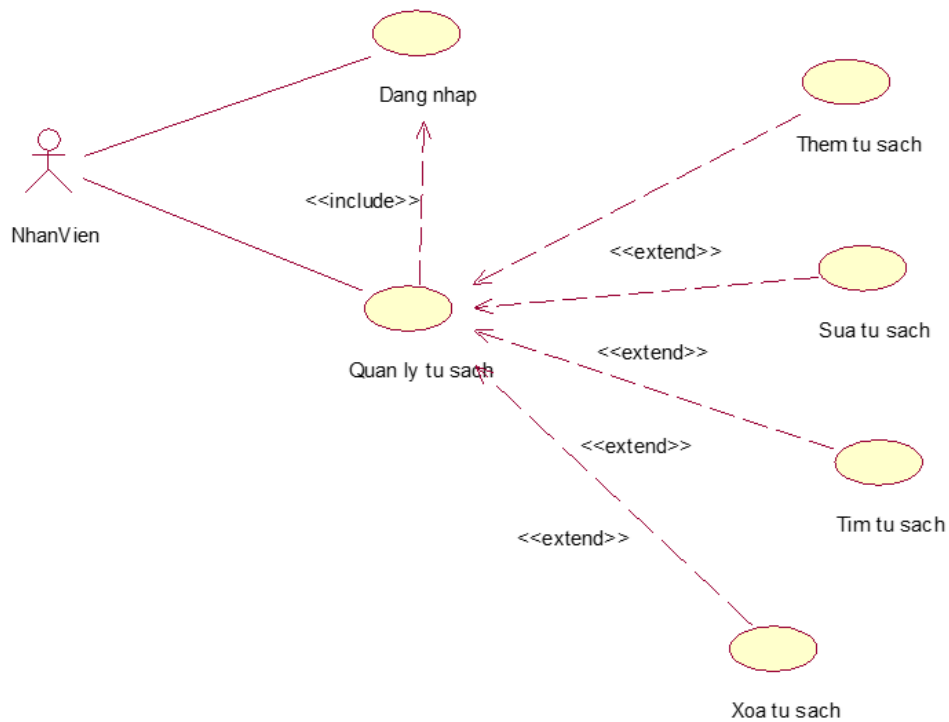
Hình 2.6: Biểu đồ usecase Quản lý thành viên

– Biểu đồ usecase Quản lý nhân viên:



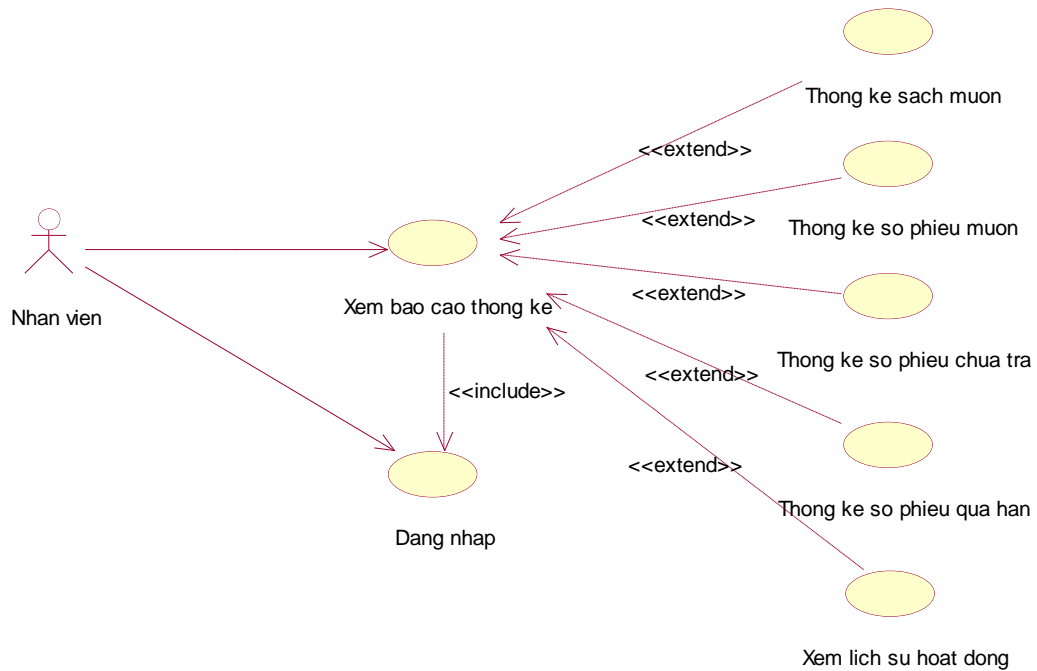
Hình 2.7: Biểu đồ usecase Quản lý nhân viên

- Biểu đồ usecase Quản lý tủ sách:



Hình 2.8: Biểu đồ usecase Quản lý tủ sách

- Biểu đồ usecase Xem báo cáo thống kê:



Hình 2.9: Biểu đồ usecase Xem báo cáo thống kê

## 2.4. Mô tả chi tiết các usecase

### 2.4.1. Mô tả chi tiết usecase Đăng nhập:

- Tên Use case: Đăng nhập
- Mô tả: Use case này cho phép tất cả các vai trò (Thành viên, Nhân viên, Quản trị) đăng nhập vào hệ thống để truy cập các chức năng tương ứng.
- Luồng sự kiện:
  - + Luồng cơ bản:
    - 1) Người dùng chọn tính năng Đăng nhập trên giao diện.
    - 2) Hệ thống hiển thị biểu mẫu đăng nhập, bao gồm các trường: Số điện thoại và Mật khẩu.
    - 3) Người dùng nhập thông tin và nhấn nút Đăng nhập.
    - 4) Hệ thống kiểm tra thông tin: Nếu thông tin hợp lệ, chuyển đến giao diện chính tương ứng với vai trò của người dùng. Nếu thông tin không hợp lệ, hiển thị thông báo lỗi.
  - + Luồng rẽ nhánh:
    - 1) Sai thông tin đăng nhập: Hệ thống thông báo lỗi và yêu cầu nhập lại thông tin.
    - 2) Lỗi kết nối cơ sở dữ liệu: Hệ thống hiển thị thông báo lỗi và kết thúc use case.
- Các yêu cầu đặc biệt: Người dùng phải có tài khoản hợp lệ trong hệ thống.
- Tiền điều kiện: Hệ thống hoạt động bình thường và có kết nối với cơ sở dữ liệu.
- Hậu điều kiện: Người dùng đăng nhập thành công và được điều hướng đến trang phù hợp với vai trò.

- Điểm mở rộng: Hiện tại không có điểm mở rộng nào.

#### **2.4.2. Mô tả chi tiết usecase Đăng ký:**

- Tên Use case: Đăng ký
- Mô tả: Use case này cho phép thành viên đăng ký tài khoản trong hệ thống với vai trò mặc định là Member.
- Luồng sự kiện:
  - + Luồng cơ bản:
    - 1) Người dùng truy cập màn hình đăng ký.
    - 2) Người dùng nhập các thông tin cần thiết: Họ tên, Số điện thoại, Mã sinh viên/giảng viên, Email (tùy chọn), Mật khẩu và Nhập lại mật khẩu.
    - 3) Người dùng nhấn nút "Đăng ký".
    - 4) Hệ thống kiểm tra tính hợp lệ của thông tin đã nhập.
    - 5) Nếu thông tin hợp lệ, tài khoản mới sẽ được tạo với vai trò Member và lưu vào cơ sở dữ liệu.
    - 6) Hệ thống hiển thị thông báo đăng ký thành công.
  - + Luồng rẽ nhánh:
    - 1) Nếu thông tin không hợp lệ (ví dụ: số điện thoại không đúng định dạng, mật khẩu quá ngắn,...), hệ thống sẽ hiển thị thông báo lỗi và yêu cầu người dùng sửa thông tin.
    - 2) Nếu số điện thoại đã được sử dụng, hệ thống sẽ thông báo rằng số điện thoại đã tồn tại.
    - 3) Nếu không thể kết nối cơ sở dữ liệu, hệ thống sẽ hiển thị thông báo lỗi và không thể thực hiện đăng ký.
- Các yêu cầu đặc biệt:
  - + Số điện thoại phải hợp lệ và chưa được sử dụng trong hệ thống.

- + Mật khẩu phải đáp ứng yêu cầu bảo mật (ví dụ: tối thiểu 6 ký tự).
- Tiền điều kiện: Người dùng chưa có tài khoản trong hệ thống.
- Hậu điều kiện: Nếu use case thành công, tài khoản mới sẽ được tạo và lưu trong cơ sở dữ liệu.
- Điểm mở rộng: Không có.

#### ***2.4.3. Mô tả chi tiết usecase Xem thông tin tài khoản:***

- Tên Use case: Xem thông tin tài khoản
- Mô tả vắn tắt: Use case này cho phép người dùng xem các thông tin chi tiết của tài khoản cá nhân đã được lưu trong hệ thống.
- Luồng sự kiện:
  - + Luồng cơ bản:
    - 1) Người dùng đăng nhập vào hệ thống.
    - 2) Người dùng chọn chức năng "Xem thông tin tài khoản" trên giao diện.
    - 3) Hệ thống truy xuất thông tin tài khoản của người dùng từ cơ sở dữ liệu.
    - 4) Hệ thống hiển thị các thông tin tài khoản trên màn hình, bao gồm:
      - Họ tên
      - Email (nếu có)
      - Số điện thoại
      - Địa chỉ (nếu có)
      - Ngày sinh (nếu có)
      - Vai trò
      - Ngày tạo tài khoản
      - Mã Sinh viên/Giảng viên (nếu người dùng là Member).
  - + Luồng rẽ nhánh:

- 1) Nếu hệ thống không thể truy xuất thông tin tài khoản do lỗi cơ sở dữ liệu, hệ thống sẽ hiển thị thông báo lỗi.
- Các yêu cầu đặc biệt: Người dùng phải đăng nhập hợp lệ trước khi thực hiện chức năng này.
  - Tiền điều kiện: Người dùng đã có tài khoản trong hệ thống và đã đăng nhập thành công.
  - Hậu điều kiện: Thông tin tài khoản của người dùng được hiển thị đầy đủ trên màn hình.
  - Điểm mở rộng: Không có.

#### ***2.4.4. Mô tả chi tiết usecase Quản lý sách***

- Tên Use case: Quản lý sách trên hệ thống.
- Mô tả vắn tắt: Use case này cho phép người quản trị (nhân viên thư viện) thực hiện các chức năng thêm, sửa, xóa, tìm kiếm thông tin sách trong hệ thống thư viện.
- Luồng sự kiện:
  - + Luồng cơ bản:
    - 2) Use case này bắt đầu khi người quản trị kích vào nút “Quản lý Sách” trên menu quản trị. Hệ thống hiển thị bảng chứa tất cả sách trong kèm theo các tùy chọn: Thêm sách mới, Cập nhật thông tin sách, Tìm kiếm sách, Xóa sách.
    - 3) Thêm sách mới:
      - a. Người quản trị kích vào nút “Thêm sách mới”. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho sản phẩm gồm tên sách, tác giả, danh mục, năm xuất bản, số lượng, vị trí kệ sách, mô tả và hình ảnh.

- b. Người quản trị nhập thông tin của tên sách, tác giả, danh mục, năm xuất bản, số lượng, vị trí kệ sách, mô tả và hình ảnh. Hệ thống kiểm tra tính hợp lệ của dữ liệu và lưu trữ sách mới vào bảng BOOKS và hiển thị xác nhận sách đã thêm thành công. Use case kết thúc.

4) Cập nhật thông tin sách:

- a. Người quản trị kích vào icon “Cập nhật” ở cột cuối cùng trong bảng. Hệ thống sẽ lấy thông tin cũ của sách và các bảng liên quan cần thiết được chọn hiển thị ra màn hình.
- b. Người quản trị chọn/nhập thông tin mới cho sách gồm: tên sách, tác giả, danh mục, năm xuất bản, số lượng, vị trí kệ sách, mô tả và hình ảnh và sau đó kích vào nút “Cập nhật”. Hệ thống sẽ cập nhật thông tin của sách được chọn trong bảng BOOKS và hiển thị danh sách sách đã được cập nhật. Use case kết thúc.

5) Xóa sản phẩm:

- a. Người quản trị kích vào biểu tượng “Xóa” trên một dòng sách. Hệ thống sẽ hiển thị một thông báo yêu cầu xác nhận xóa.
- b. Người quản trị kích vào nút “Đồng ý”. Hệ thống sẽ xóa sách được chọn => Sau đó hệ thống sẽ hiển thị thông tin danh sách sau khi xóa. Use case kết thúc.

6) Tìm kiếm sách:

- a. Người quản trị nhập từ khóa để tìm kiếm (có thể nhập theo tên sách, tác giả, nhà xuất bản), hoặc có thể chọn bộ lọc theo Danh mục, ngăn sách.
- b. Hệ thống hiển thị danh sách của sách gồm: tên sách, nhà xuất bản, năm xuất bản, số lượng có thể cho mượn, vị trí lưu trữ phù hợp với từ khóa tìm kiếm lên màn hình.



- c. Người quản trị có thể chọn một sách từ kết quả để xem chi tiết hoặc chỉnh sửa thông tin. Usecase kết thúc.

+ Luồng rẽ nhánh:

- 1) Tại bước 2b hoặc 3c trong luồng cơ bản nếu người quản trị nhập thông tin sách không hợp lệ thì hệ thống sẽ hiển thị thông báo lỗi yêu cầu nhập lại. Người quản trị có thể nhập lại để kết thúc phiên làm việc hiện tại.
  - 2) Tại bước 2b hoặc 3b trong luồng cơ bản nếu người quản trị kích vào nút “Hủy bỏ” hệ thống sẽ bỏ qua thao tác sửa và hiển thị lại danh sách các sách.
  - 3) Tại bước 4b trong luồng cơ bản nếu người quản trị kích vào nút “Hủy” thì hệ thống sẽ bỏ qua thao tác xóa và hiển thị lại danh sách các sách.
  - 4) Tại bước 5b trong luồng cơ bản nếu hệ thống thông tìm thấy sách trùng với kết quả tìm kiếm thì hiển thị thông báo “Không tìm thấy sách trong thư viện”. Người quản trị có thể nhập lại để kết thúc phiên làm việc.
- Các yêu cầu đặc biệt: Use case này chỉ cho phép một số vai trò như nhân viên thư viện và quản trị.
  - Tiền điều kiện: Người quản trị cần đăng nhập với vai trò quản trị hoặc nhân viên trước khi có thể thực hiện usecase.
  - Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về các sách (hàng) sẽ được cập nhật trong cơ sở dữ liệu.
  - Điểm mở rộng: Không có.

#### ***2.4.5. Mô tả chi tiết usecase Quản lý phiếu mượn/trả sách:***

- Tên Use case: Quản lý mượn trả sách

- Mô tả vắn tắt: Use case này cho phép nhân viên hoặc quản trị viên thực hiện các hoạt động quản lý phiếu mượn/trả sách như thêm, xóa, cập nhật trạng thái phiếu mượn.
- Luồng sự kiện:
  - + Luồng cơ bản:
    - 1) Thành viên đến thư viện và yêu cầu mượn sách.
    - 2) Thủ thư chọn tính năng "Quản lý phiếu mượn/trả" trên menu.
    - 3) Hệ thống hiển thị bảng thông tin về các phiếu mượn, kèm các nút chức năng như Thêm phiếu mượn, Xóa phiếu mượn, Cập nhật trạng thái phiếu mượn.
    - 4) Thủ thư nhấn nút Thêm phiếu mượn: Hệ thống chuyển qua màn hình Thêm phiếu mượn. Thủ thư chọn thành viên mượn sách và thêm các sách từ danh sách thư viện vào phiếu mượn. Thủ thư xác định ngày trả sách và nhấn nút Xác nhận để hoàn tất giao dịch.
    - 5) Thành viên trả sách khi hết hạn mượn.
    - 6) Thủ thư xác nhận việc trả sách, cập nhật tình trạng sách trong hệ thống (Đã trả hoặc Quá hạn).
  - + Luồng rẽ nhánh:
    - 1) Giao dịch mượn bị hủy: Thành viên hoặc thủ thư quyết định hủy giao dịch mượn trước khi xác nhận. Hệ thống không lưu lại giao dịch.
    - 2) Sách bị quá hạn: Nếu sách bị quá hạn, hệ thống sẽ thông báo cho thủ thư và áp dụng biện pháp xử lý.
    - 3) Tại bất kỳ thời điểm nào trong quá trình thực hiện usecase nếu không kết nối được với cơ sở dữ liệu thì hệ thống hiển thị thông báo lỗi và usecase kết thúc.
- Các yêu cầu đặc biệt:

- + Thành viên phải có tài khoản hợp lệ trong hệ thống.
- + Sách được mượn phải có trong kho và chưa bị mượn hết.
- Tiền điều kiện: Người quản trị cần đăng nhập với vai trò quản trị viên hoặc nhân viên trước khi có thể thực hiện use case.
- Hậu điều kiện: Nếu use case kết thúc thành công thì thông tin về các phiếu mượn sách sẽ được cập nhật trong cơ sở dữ liệu.
- Điểm mở rộng: Không có.

#### ***2.4.6. Mô tả chi tiết usecase Quản lý thành viên:***

- Tên Use case: Quản lý thành viên
- Mô tả: Cho phép nhân viên (hay quản trị) tìm, thêm, cập nhật hay xóa thông tin của thành viên.
- Luồng sự kiện:
  - + Luồng cơ bản:
    - 1) Use case này bắt đầu khi nhân viên kích vào nút “Quản lý thành viên” trên menu quản trị. Hệ thống lấy thông tin chi tiết của thành viên trong CSDL và hiển thị danh sách các thành viên lên màn hình.
    - 2) Nhân viên kích vào nút “Thêm mới” trên cửa sổ danh sách độc giả. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho thành viên. Nhân viên nhập thông tin của thành viên và kích vào nút “Tạo”. Hệ thống sẽ tạo một thành viên mới và hiển thị danh sách các thành viên đã được cập nhật.
    - 3) Nhân viên kích vào nút “Sửa” trên một dòng độc giả. Hệ thống sẽ lấy thông tin cũ của thành viên được chọn và hiển thị lên màn hình. Nhân viên nhập thông tin mới cho thành viên và kích vào nút “Cập nhật”. Hệ

thống sẽ sửa thông tin của thành viên được chọn và hiển thị danh sách độc giả đã cập nhật.

- 4) Nhân viên có thể nhập một trong các thông tin của thành viên như họ tên, số điện thoại, Email, địa chỉ vào ô “Tìm kiếm”. Hệ thống sẽ hiển thị danh sách các thành viên khớp với điều kiện tìm kiếm.
- 5) Nhân viên kích vào nút “Xóa” trên một dòng thành viên. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa. Nhân viên kích vào nút “Đồng ý”. Hệ thống sẽ xóa thành viên được chọn và hiển thị danh sách các thành viên đã cập nhật. Use case kết thúc.

+ Luồng rẽ nhánh:

- 1) Tại thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với CSDL thì hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc

- Các yêu cầu đặc biệt: Không có
- Tiền điều kiện: Để thực hiện quản lý thành viên, người dùng cần đăng nhập với tài khoản nhân viên (hoặc quản trị) của hệ thống.
- Hậu điều kiện: Không có
- Điểm mở rộng: Không có

#### ***2.4.7. Mô tả chi tiết usecase Quản lý nhân viên:***

- Tên Use Case: Quản lý nhân viên
- Mô tả: Cho phép quản trị viên tìm, thêm, cập nhật hay xóa thông tin của nhân viên.
- Luồng sự kiện:

+ Luồng cơ bản:

- 1) Use case này bắt đầu khi quản trị viên kích vào nút “Quản lý nhân sự” trên menu quản trị. Hệ thống lấy thông tin chi tiết của nhân viên trong CSDL và hiển thị danh sách các nhân viên lên màn hình.
- 2) Quản trị viên kích vào nút “Thêm mới” trên cửa sổ danh sách nhân viên. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho nhân viên. Quản trị viên nhập thông tin của nhân viên và kích vào nút “Tạo”. Hệ thống sẽ tạo một nhân viên mới và hiển thị danh sách các nhân viên đã được cập nhật.
- 3) Quản trị viên kích vào nút “Sửa” trên một dòng nhân viên. Hệ thống sẽ lấy thông tin cũ của nhân viên được chọn và hiển thị lên màn hình. Quản trị viên nhập thông tin mới cho nhân viên và kích vào nút “Cập nhật”. Hệ thống sẽ sửa thông tin của nhân viên được chọn và hiển thị danh sách nhân viên đã cập nhật.
- 4) Quản trị viên điền vào ô “Tìm kiếm” một trong các thông tin của thành viên như họ tên, số điện thoại, Email, địa chỉ vào ô “Tìm kiếm”. Hệ thống sẽ hiển thị danh sách các nhân viên khớp với điều kiện tìm kiếm.
- 5) Quản trị viên kích vào nút “Xóa” trên một dòng nhân viên. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa. Quản trị viên kích vào nút “Đồng ý”. Hệ thống sẽ xóa nhân viên được chọn và hiển thị danh sách các nhân viên đã cập nhật. Use case kết thúc.

+ Luồng rẽ nhánh:

- 1) Tại thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với CSDL thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc

– Các yêu cầu đặc biệt: Không có

- Tiền điều kiện: Để thực hiện quản lý nhân viên, người dùng cần đăng nhập với tài khoản Quản trị của hệ thống.
- Hậu điều kiện: Không có
- Điều mở rộng: Không có

#### **2.4.8. Mô tả chi tiết usecase Quản lý tủ sách:**

- Tên usecase: Quản lý Tủ Sách
- Mô tả: Cho phép nhân viên (hoặc quản trị viên) tìm, thêm, cập nhật hay xóa thông tin của các tủ sách.
- Luồng sự kiện:
  - + Luồng cơ bản:
    - 1) Bắt đầu: Use case bắt đầu khi nhân viên kích vào nút “Tủ sách” trên menu quản trị. Hệ thống lấy thông tin chi tiết của các tủ sách trong CSDL và hiển thị danh sách các tủ sách lên màn hình.
    - 2) Thêm mới: Nhân viên kích vào nút “Thêm mới” trên cửa sổ danh sách tủ sách. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho tủ sách. Nhân viên nhập thông tin của tủ sách và kích vào nút “Tạo”. Hệ thống sẽ tạo một tủ sách mới và hiển thị danh sách các tủ sách đã được cập nhật.
    - 3) Sửa: Nhân viên kích vào nút “Sửa” trên một dòng tủ sách. Hệ thống sẽ lấy thông tin cũ của tủ sách được chọn và hiển thị lên màn hình. Nhân viên nhập thông tin mới cho tủ sách và kích vào nút “Cập nhật”. Hệ thống sẽ sửa thông tin của tủ sách được chọn và hiển thị danh sách tủ sách đã cập nhật.

- 4) Tìm kiếm: Nhân viên kích vào nút “Tìm kiếm”. Hệ thống sẽ hiển thị một bảng nhập, nhân viên nhập id của tủ sách và kích vào nút “Tìm”. Hệ thống hiển thị thông tin của tủ sách vừa nhập.
- 5) Xóa: Nhân viên kích vào nút “Xóa” trên một dòng tủ sách. Hệ thống sẽ hiển thị một màn hình yêu cầu xác nhận xóa. Nhân viên kích vào nút “Đồng ý”. Hệ thống sẽ xóa tủ sách được chọn và hiển thị danh sách các tủ sách đã cập nhật. Use case kết thúc.

+ Luồng rẽ nhánh:

- 1) Tại thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với CSDL thì hệ thống sẽ hiển thị một thông báo lỗi và use case kết thúc.
- Các yêu cầu đặc biệt: Không có.
  - Tiền điều kiện: Để thực hiện quản lý tủ sách, nhân viên cần đăng nhập với tài khoản nhân viên của hệ thống.
  - Hậu điều kiện: Không có.
  - Điểm mở rộng: Không có.

#### **2.4.9. Mô tả chi tiết usecase Xem báo cáo thống kê:**

- Tên usecase: Xem báo cáo thống kê
- Mô tả: Cho phép nhân viên (hoặc quản trị viên) xem báo cáo thống kê về hoạt động của thư viện, bao gồm số lượng sách mượn, sách trả, và các hoạt động liên quan.
- Luồng sự kiện:

+ Luồng cơ bản:

- 1) Bắt đầu: Usecase bắt đầu khi nhân viên truy cập vào trang chủ. Hệ thống sẽ hiển thị danh sách các loại báo cáo có sẵn.

- 2) Xem báo cáo: Hệ thống tạo báo cáo dựa trên các tiêu chí và hiển thị kết quả dưới dạng bảng hoặc biểu đồ. Nhân viên có thể xem chi tiết hoặc tải xuống báo cáo nếu cần.
- 3) Kết thúc: Usecase kết thúc khi nhân viên hoàn thành việc xem hoặc tải xuống báo cáo.

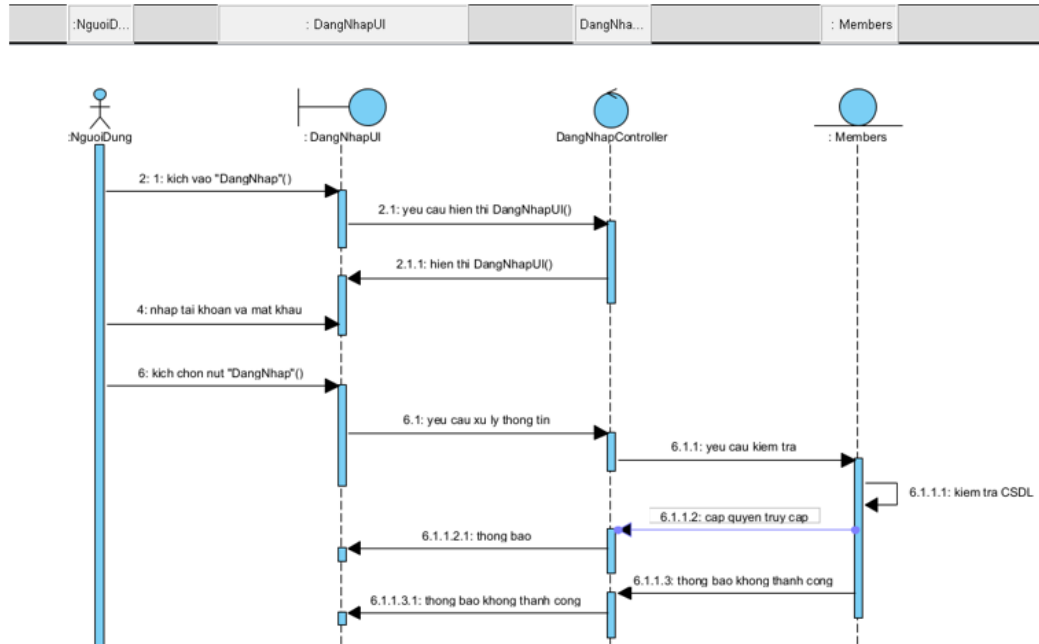
+ Luồng rẽ nhánh:

- 1) Nếu trong quá trình tạo báo cáo, hệ thống không thể lấy dữ liệu từ CSDL, hệ thống sẽ hiển thị thông báo lỗi và usecase kết thúc.
  - 2) Nếu không có dữ liệu nào phù hợp với các tiêu chí đã nhập, hệ thống sẽ hiển thị thông báo không có dữ liệu.
- Các yêu cầu đặc biệt: Báo cáo phải hiển thị rõ ràng, chính xác và có thể tải xuống dưới các định dạng phổ biến như PDF, Excel.
  - Tiền điều kiện: Cần đăng nhập với tài khoản nhân viên (hoặc quản trị viên) và có quyền truy cập vào chức năng xem báo cáo của hệ thống.
  - Hậu điều kiện: Không có.
  - Điểm mở rộng: Không có.

## **2.5. Biểu đồ trình tự các usecase**

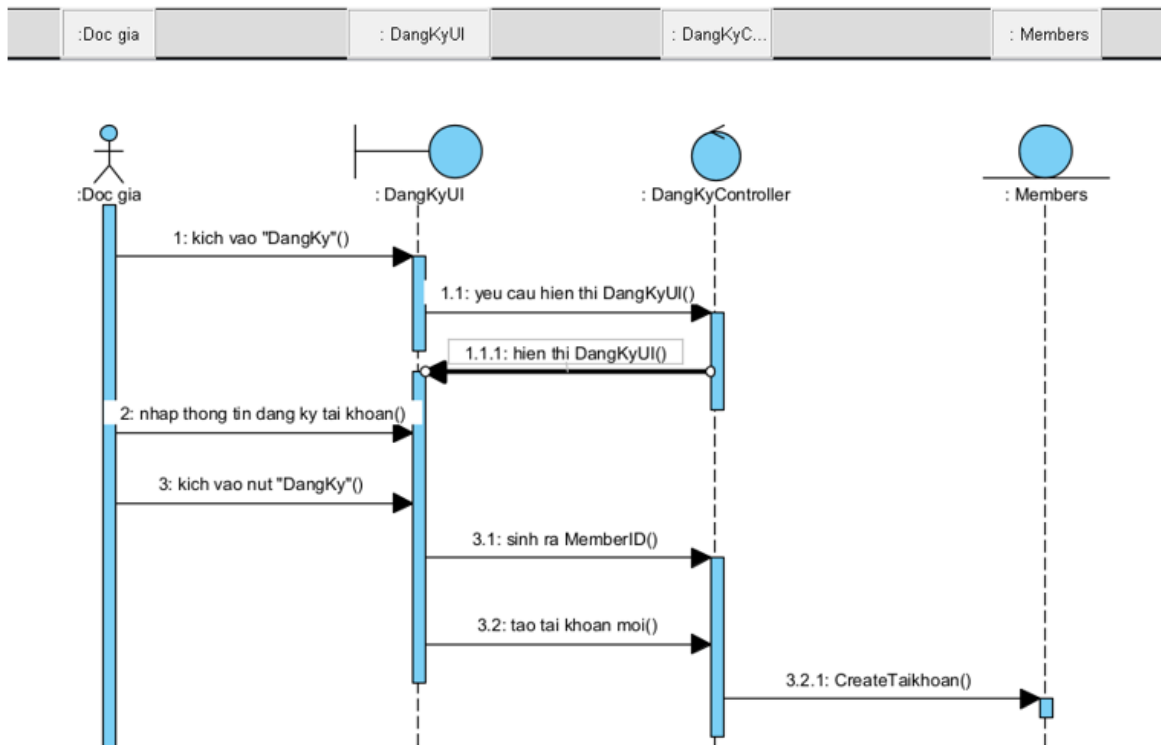
### ***2.5.1. Biểu đồ trình tự usecase Đăng nhập***





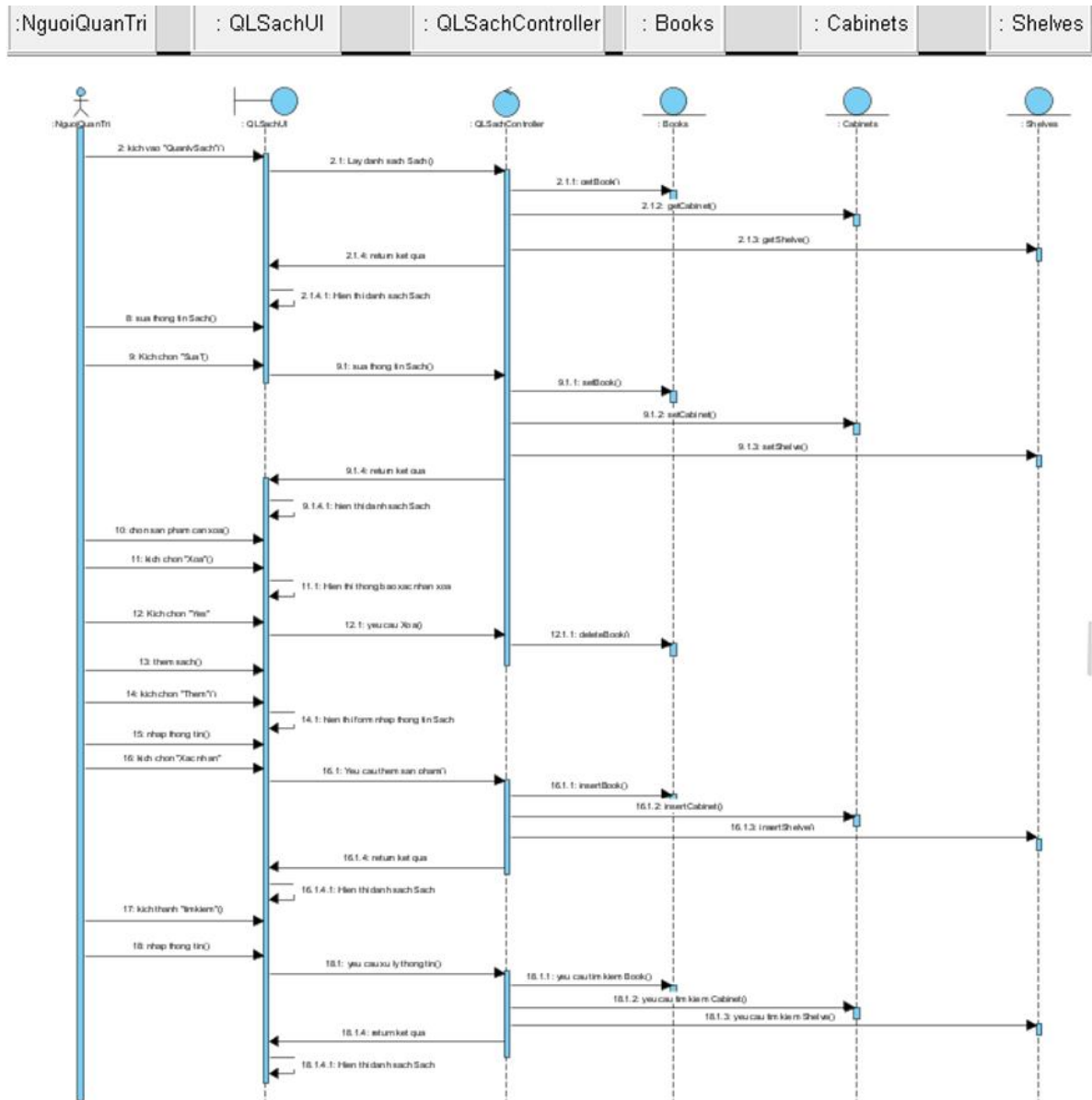
Hình 2.10: Biểu đồ trình tự usecase Đăng nhập

### 2.5.2. Biểu đồ trình tự usecase Đăng ký:



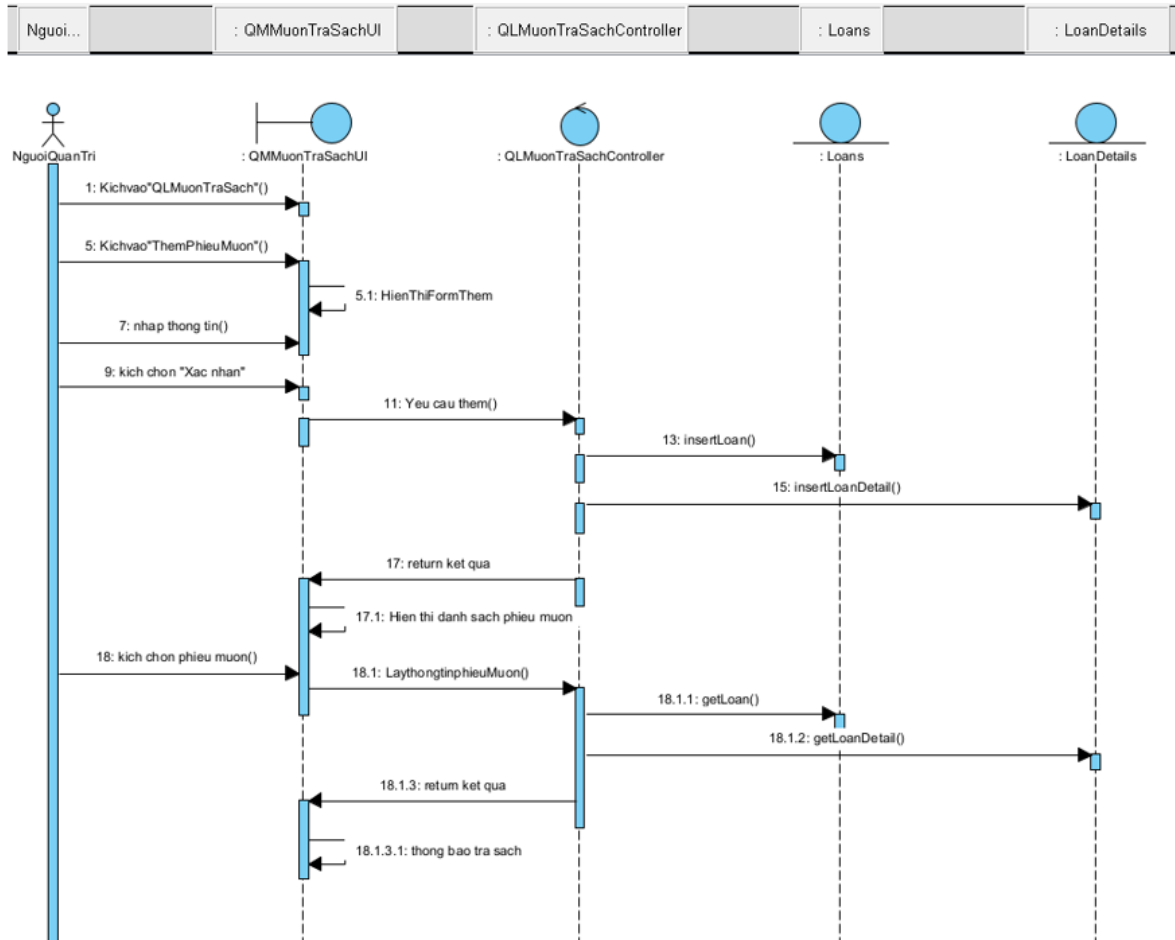
Hình 2.11: Biểu đồ trình tự usecase Đăng ký

### 2.5.3. Biểu đồ trình tự usecase Quản lý sách:



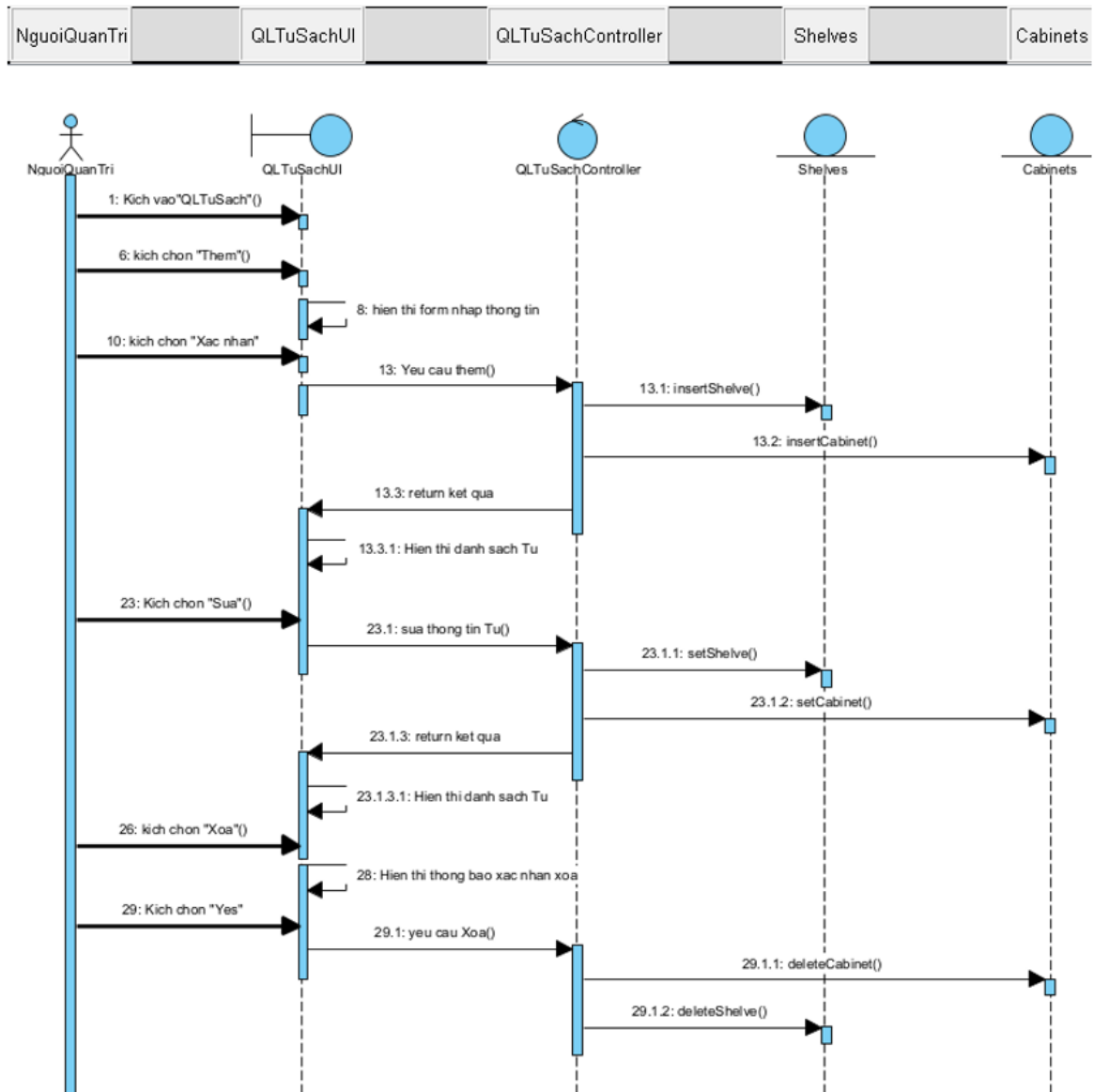
Hình 2.12: Biểu đồ trình tự usecase Quản lý sách

#### 2.5.4. Biểu đồ trình tự usecase Quản lý phiếu mượn trả sách:



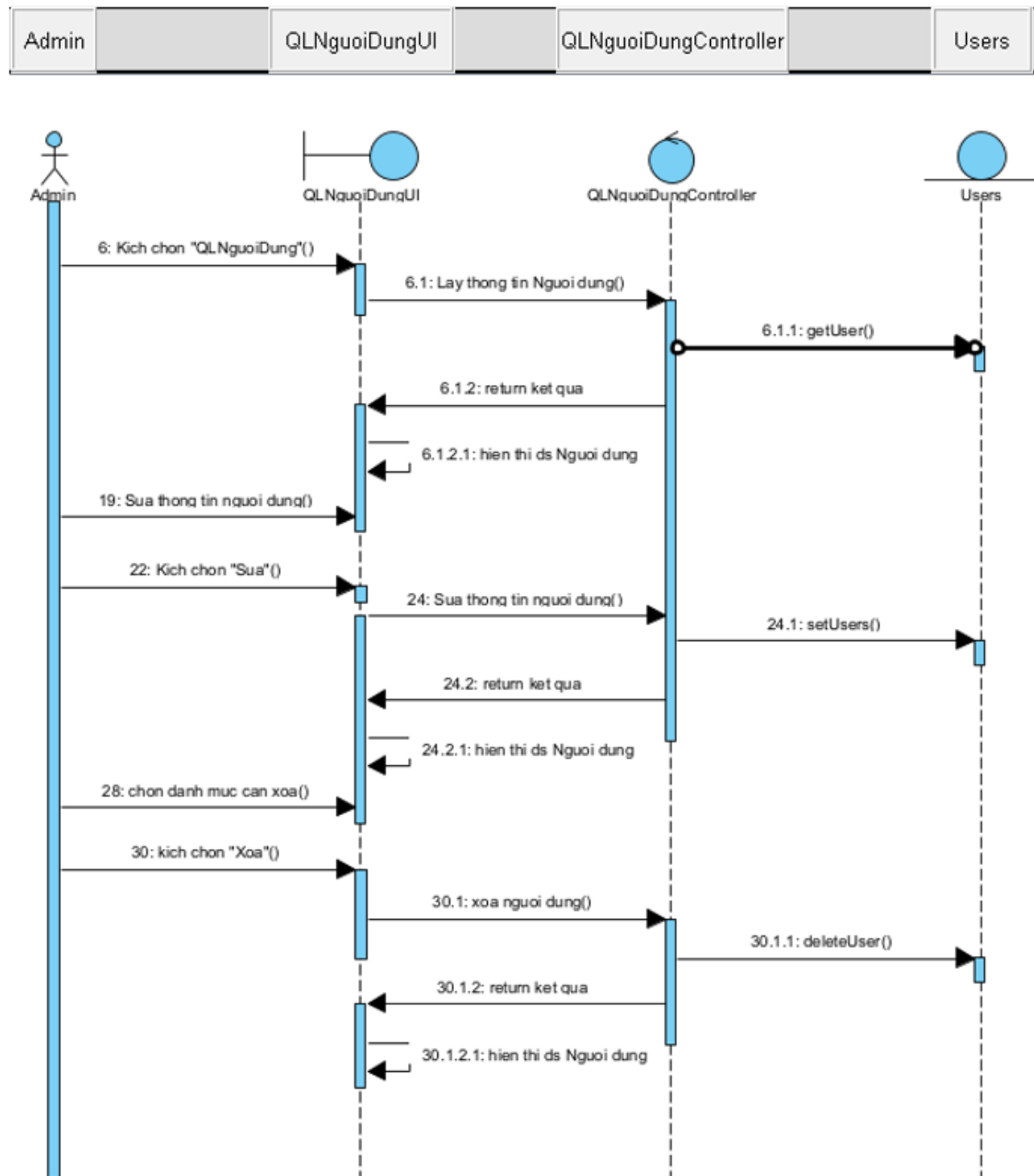
Hình 2.13: Biểu đồ trình tự usecase Quản lý phiếu mượn trả sách

### 2.5.5. Biểu đồ trình tự usecase Quản lý tủ sách:



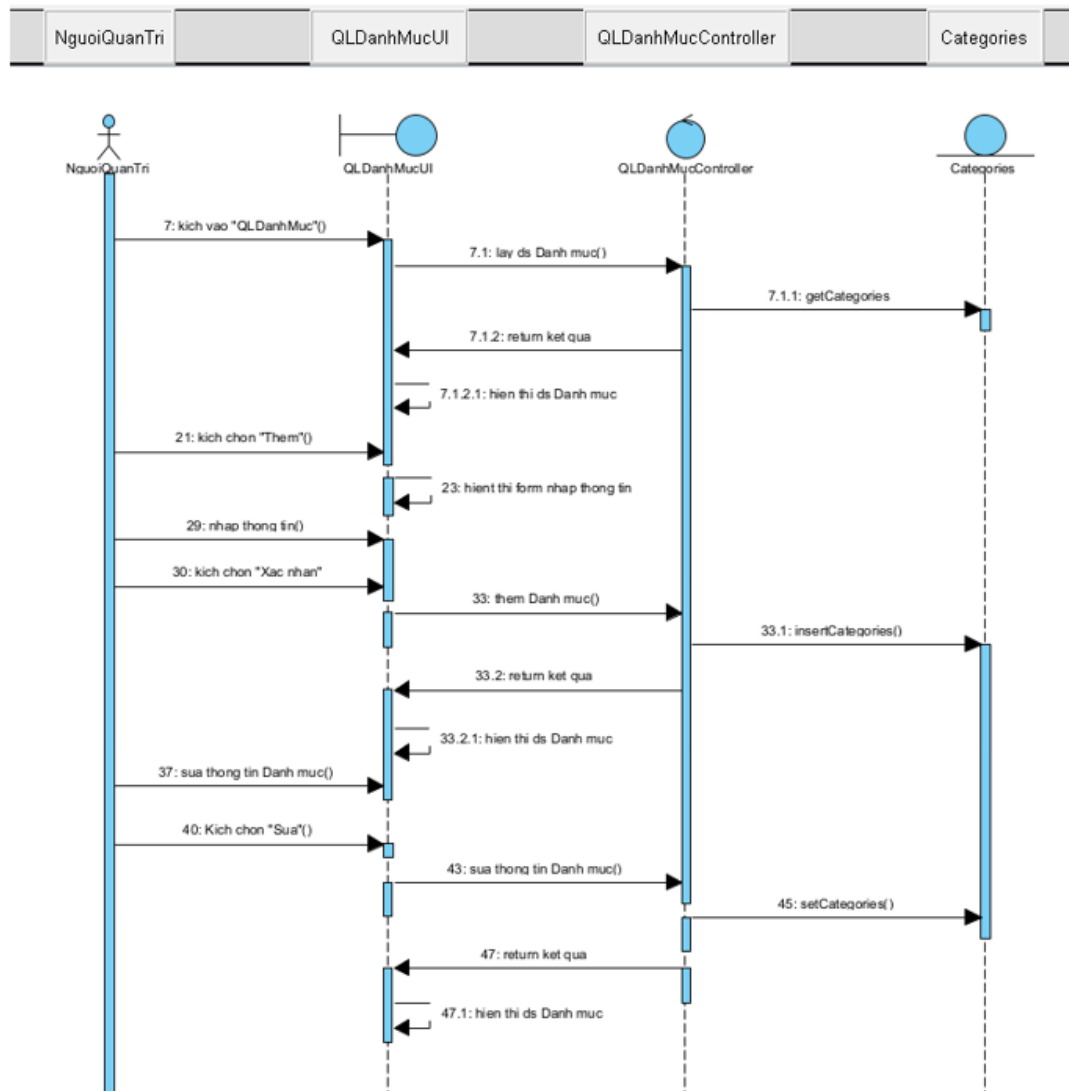
Hình 2.14: Biểu đồ trình tự usecase Quản lý tủ sách

### 2.5.6. Biểu đồ trình tự usecase Quản lý thành viên:



Hình 2.15: Biểu đồ trình tự usecase Quản lý thành viên

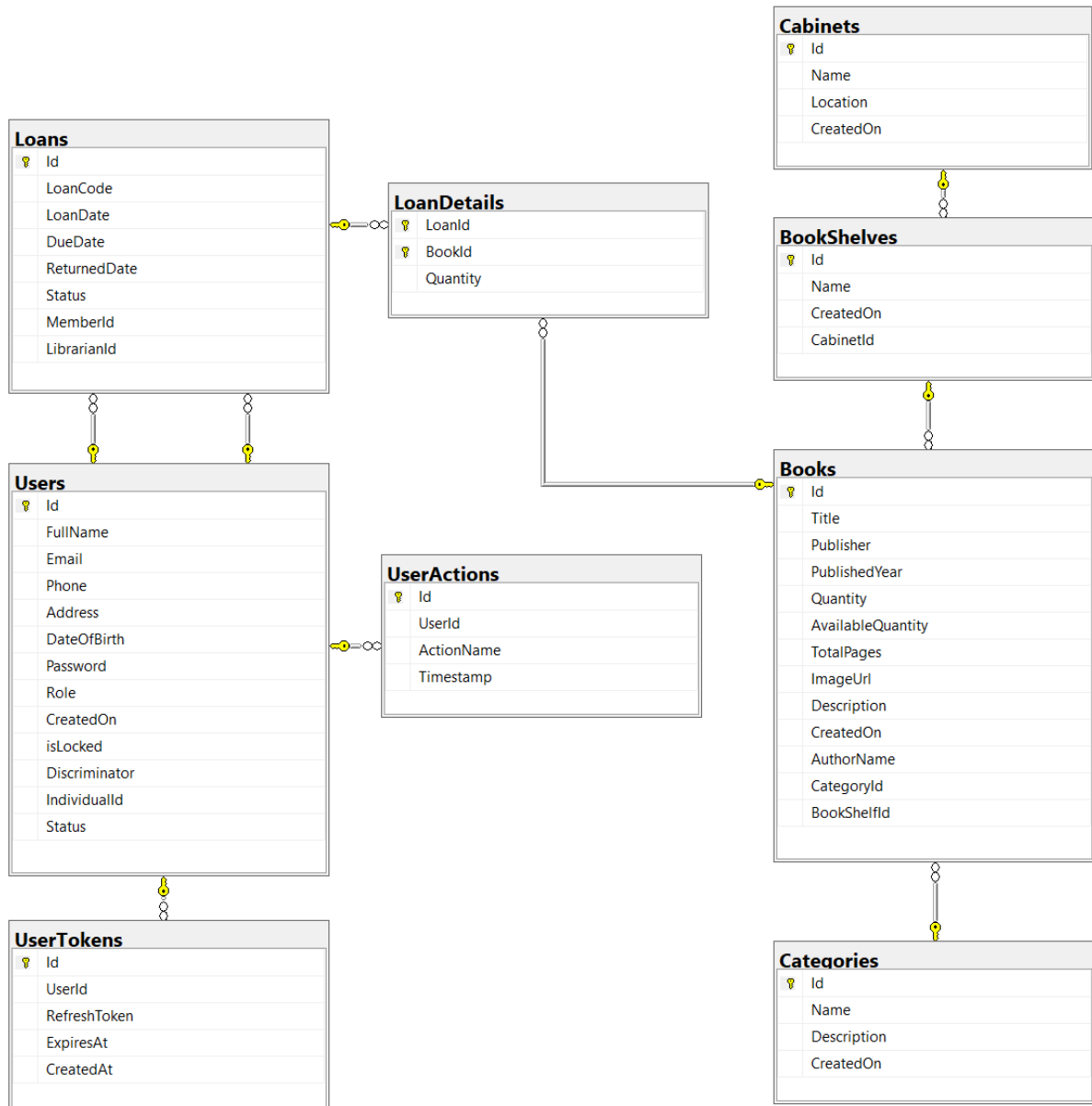
### 2.5.7. Biểu đồ trình tự usecase Quản lý danh mục:



Hình 2.16: Biểu đồ trình tự usecase Quản lý danh mục

## 2.6. Thiết kế cơ sở dữ liệu

### 2.6.1. Sơ đồ dữ liệu quan hệ




Hình 2.17: Sơ đồ dữ liệu quan hệ

### 2.6.2. Chi tiết các bảng dữ liệu


#### a. Bảng sách:

Bảng 2.1: Bảng sách

Books			
	Column Name	Data Type	Allow Nulls
	Id	uniqueidentifier	<input type="checkbox"/>
	Title	nvarchar(255)	<input type="checkbox"/>
	Publisher	nvarchar(100)	<input checked="" type="checkbox"/>
	PublishedYear	int	<input checked="" type="checkbox"/>
	Quantity	int	<input type="checkbox"/>
	AvailableQuantity	int	<input type="checkbox"/>
	TotalPages	int	<input type="checkbox"/>
	ImageUrl	nvarchar(512)	<input type="checkbox"/>
	Description	nvarchar(2000)	<input checked="" type="checkbox"/>
	CreatedOn	datetime2(7)	<input type="checkbox"/>
	AuthorName	nvarchar(MAX)	<input checked="" type="checkbox"/>
	CategoryId	uniqueidentifier	<input checked="" type="checkbox"/>
	BookShelfId	uniqueidentifier	<input checked="" type="checkbox"/>
			<input type="checkbox"/>


#### b. Bảng ngăn sách:

Bảng 2.2: Bảng ngăn sách


BookShelves			
	Column Name	Data Type	Allow Nulls
	Id	uniqueidentifier	<input type="checkbox"/>
	Name	nvarchar(50)	<input type="checkbox"/>
	CreatedOn	datetime2(7)	<input type="checkbox"/>
	CabinetId	uniqueidentifier	<input type="checkbox"/>
			<input type="checkbox"/>




**c. Bảng tủ sách:***Bảng 2.3: Bảng tủ sách*

Cabinets			
	Column Name	Data Type	Allow Nulls
	Id	uniqueidentifier	<input type="checkbox"/>
	Name	nvarchar(50)	<input type="checkbox"/>
	Location	nvarchar(50)	<input checked="" type="checkbox"/>
	CreatedOn	datetime2(7)	<input type="checkbox"/>

**d. Bảng danh mục:***Bảng 2.4: Bảng danh mục*


Categories			
	Column Name	Data Type	Allow Nulls
	Id	uniqueidentifier	<input type="checkbox"/>
	Name	nvarchar(50)	<input type="checkbox"/>
	Description	nvarchar(2000)	<input checked="" type="checkbox"/>
	CreatedOn	datetime2(7)	<input type="checkbox"/>

**e. Bảng người dùng:***Bảng 2.5: Bảng người dùng*

Users			
	Column Name	Data Type	Allow Nulls
	Id	uniqueidentifier	<input type="checkbox"/>
	FullName	nvarchar(100)	<input type="checkbox"/>
	Email	nvarchar(255)	<input checked="" type="checkbox"/>
	Phone	nvarchar(12)	<input type="checkbox"/>
	Address	nvarchar(255)	<input checked="" type="checkbox"/>
	DateOfBirth	datetime2(7)	<input checked="" type="checkbox"/>
	Password	nvarchar(MAX)	<input type="checkbox"/>
	Role	int	<input type="checkbox"/>
	CreatedOn	datetime2(7)	<input type="checkbox"/>
	isLocked	bit	<input type="checkbox"/>
	Discriminator	nvarchar(MAX)	<input type="checkbox"/>
	IndividualId	nvarchar(10)	<input checked="" type="checkbox"/>
	Status	int	<input checked="" type="checkbox"/>


**f. Bảng lịch sử hoạt động người dùng:**

*Bảng 2.6: Bảng lịch sử hoạt động người dùng*

UserActions			
	Column Name	Data Type	Allow Nulls
	Id	uniqueidentifier	<input type="checkbox"/>
	UserId	uniqueidentifier	<input type="checkbox"/>
	ActionName	nvarchar(MAX)	<input type="checkbox"/>
	Timestamp	datetime2(7)	<input type="checkbox"/>


**g. Bảng token đăng nhập:**

*Bảng 2.7: Bảng token đăng nhập*

UserTokens			
	Column Name	Data Type	Allow Nulls
	Id	uniqueidentifier	<input type="checkbox"/>
	UserId	uniqueidentifier	<input type="checkbox"/>
	RefreshToken	nvarchar(MAX)	<input type="checkbox"/>
	ExpiresAt	datetime2(7)	<input type="checkbox"/>
	CreatedAt	datetime2(7)	<input type="checkbox"/>

**h. Bảng phiếu mượn sách:**

*Bảng 2.8: Bảng phiếu mượn sách*

Loans			
	Column Name	Data Type	Allow Nulls
	Id	uniqueidentifier	<input type="checkbox"/>
	LoanCode	nvarchar(450)	<input type="checkbox"/>
	LoanDate	datetime2(7)	<input type="checkbox"/>
	DueDate	datetime2(7)	<input type="checkbox"/>
	ReturnedDate	datetime2(7)	<input checked="" type="checkbox"/>
	Status	int	<input type="checkbox"/>
	MemberId	uniqueidentifier	<input type="checkbox"/>
	LibrarianId	uniqueidentifier	<input type="checkbox"/>

## i. Bảng chi tiết phiếu mượn:

Bảng 2.9: Bảng chi tiết phiếu mượn

LoanDetails			
	Column Name	Data Type	Allow Nulls
🔑	LoanId	uniqueidentifier	<input type="checkbox"/>
🔑	BookId	uniqueidentifier	<input type="checkbox"/>
	Quantity	int	<input type="checkbox"/>
			<input type="checkbox"/>

## 2.7. Thiết kế giao diện

Hệ thống

- Tổng quát
- Quản lý mượn trả
- Quản lý phiếu mượn
- Sách
- Quản lý sách
- Quản lý từ/ngân sách
- Quản lý danh mục
- Nhân sự
- Quản lý nhân sự
- Quản lý thành viên

Quản lý thư viện Docs

Quản lý sách

**Quản lý sách**

Danh sách

Tất cả cuốn sách trong thư viện

🔍 Tìm kiếm sách + Danh mục < + Thêm sách mới

<input type="checkbox"/> #	Tiêu đề	Ảnh	Số lượng	Hiện có	Số trang	Tác giả	Danh mục	Ngân sách	Nhà xuất bản	Năm XB	Thời gian tạo	Tùy chọn
<input type="checkbox"/> 1.	XXX		XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	
<input type="checkbox"/> 2.	XXX		XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	

Hình 2.18: Hình dung màn hình giao diện Quản lý sách

Hệ thống

Tổng quát

Quản lý mượn trả

Quản lý phiếu mượn

Sách

Quản lý sách

Quản lý tài/ngân sách

Quản lý danh mục

Nhân sự

Quản lý nhân sự

Quản lý thành viên

Quản lý thư viện

Docs

Quản lý sách

Quản lý sách > Tạo mới sách

Điền các thông tin của sách

Lưu ý các trường có dấu\* là bắt buộc

Ảnh bìa

Xóa ảnh

Chọn ảnh

Tiêu đề\* sách \*

Số\* trang \*

Số\*lượng \*

Nhà xuất bản

Năm xuất bản

Tác giả

Danh mục

Ngân sách

Mô tả

Hình 2.19: Hình dung màn hình giao diện Thêm sách mới

Hệ thống

- Tổng quát

Quản lý mượn trả

- Quản lý phiếu mượn

Sách

- Quản lý sách
- Quản lý tài/ngân sách
- Quản lý danh mục

Nhân sự

- Quản lý nhân sự
- Quản lý thành viên

Quản lý thư việnDCCA

> Quản lý sách

Quản lý sách

Danh sách  
Tất cả cuốn sách trong thư viện

+ Danh mục

< + Thêm sách mới

<input type="checkbox"/> #	Tiêu đề ↕	Nhà xuất bản	Năm XB ↕	Thời gian tạo ↕	Tùy chọn						
<input type="checkbox"/> 1.	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	✎	🗑
<input type="checkbox"/> 2.	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	✎	🗑

Hình 2.20: Hình dung màn hình giao diện Xóa sách

Hệ thống

Tổng quát

Quản lý mượn trả

Quản lý phiếu mượn

Sách

Quản lý sách

Quản lý tài/nguồn sách

Quản lý danh mục

Nhân sự

Quản lý nhân sự

Quản lý thành viên

**Quản lý thư viện**
Docs

---

>
Quản lý phiếu mượn

## Quản lý phiếu mượn

**Danh sách**

Tất cả phiếu mượn của thành viên

+ Trạng thái phiếu
< (+) Thêm phiếu mới

<input type="checkbox"/> #	Mã phiếu	Trạng thái	Người mượn	Số điện thoại	Ngày mượn	Ngày hạn	Ngày trả	Sách	Nhân viên	Tùy chọn
<input type="checkbox"/> 1.	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	...
<input type="checkbox"/> 2.	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	

< Trước
1
Sau >

Tùy chọn

☒ Duyệt phiếu

☐ Cho mượn

☐ Xóa phiếu

Hình 2.21: Hình dung màn hình giao diện Quản lý phiếu mượn

Hệ thống

Tổng quát

Quản lý mượn trả

Quản lý phiếu mượn

Sách

Quản lý sách

Quản lý tài/ngân sách

Quản lý danh mục

Nhân sự

Quản lý nhân sự

Quản lý thành viên

Quản lý thư viện

Docs

Thông tin phiếu mượn

Biểu mẫu

Điền thông tin phiếu mượn

Lưu ý! Các trường có dấu \* là bắt buộc

Số điện thoại \*

Họ tên

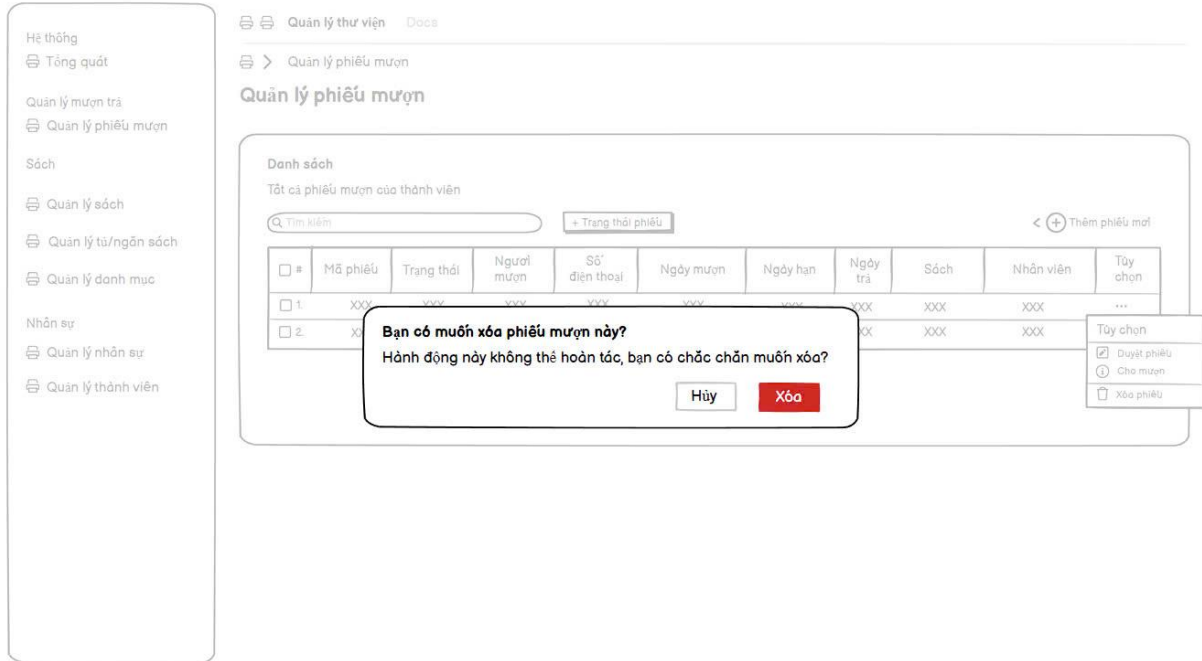
Ngày trả \*

Các sách đã chọn

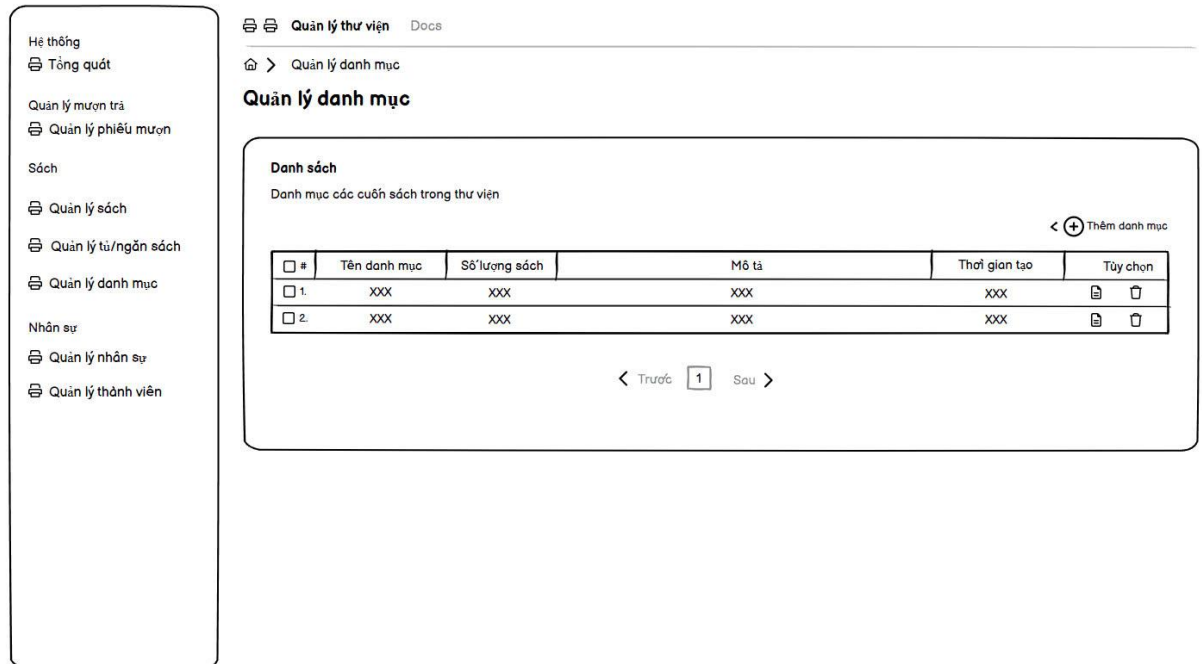
<input type="checkbox"/> #	Tiêu đề	Ảnh	Hiện có	Số trang	Tác giả	Danh mục	Ngân sách	Nhà xuất bản	Năm XB	Số lượng	Bỏ chọn
<input type="checkbox"/> 1.	XXX	<div></div>	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX	XXX
Tổng											XXX

Xác nhận

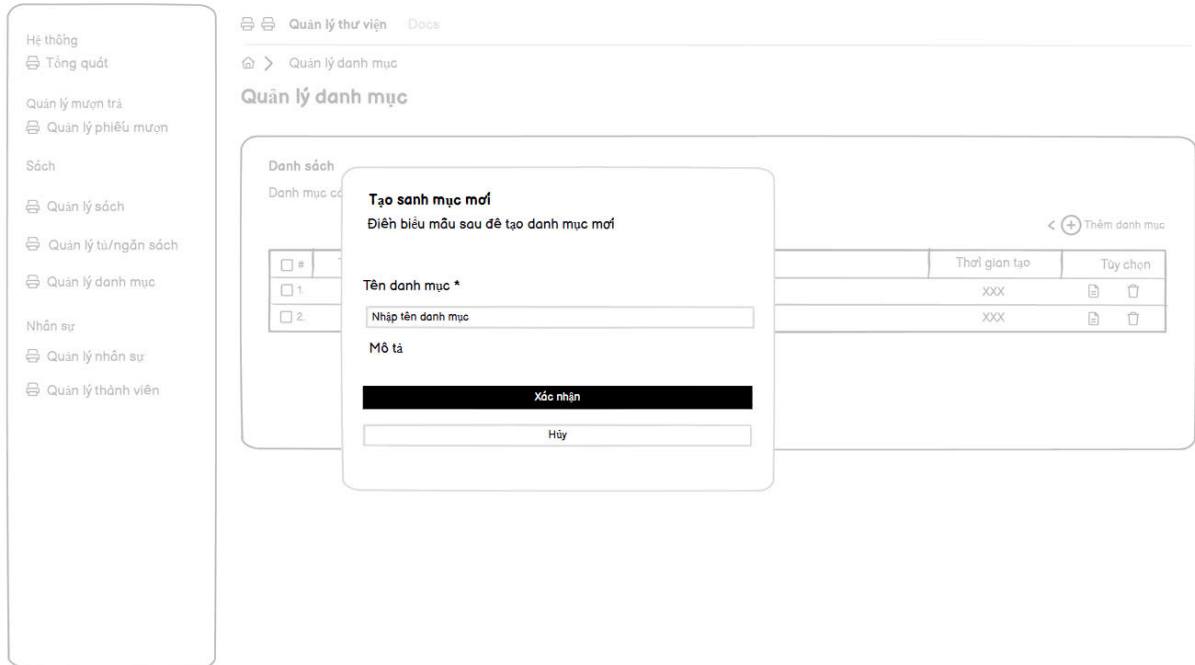
Hình 2.22: Hình dung màn hình giao diện Thêm phiếu mượn



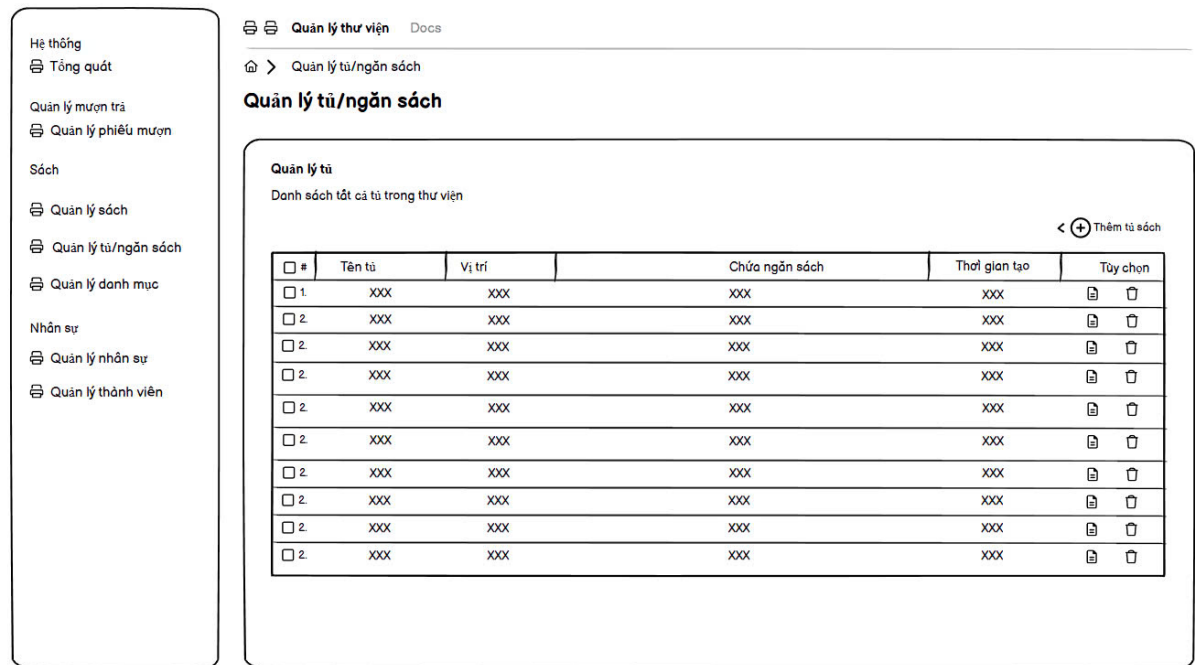
Hình 2.23: Hình dung màn hình giao diện Xóa phiếu mượn



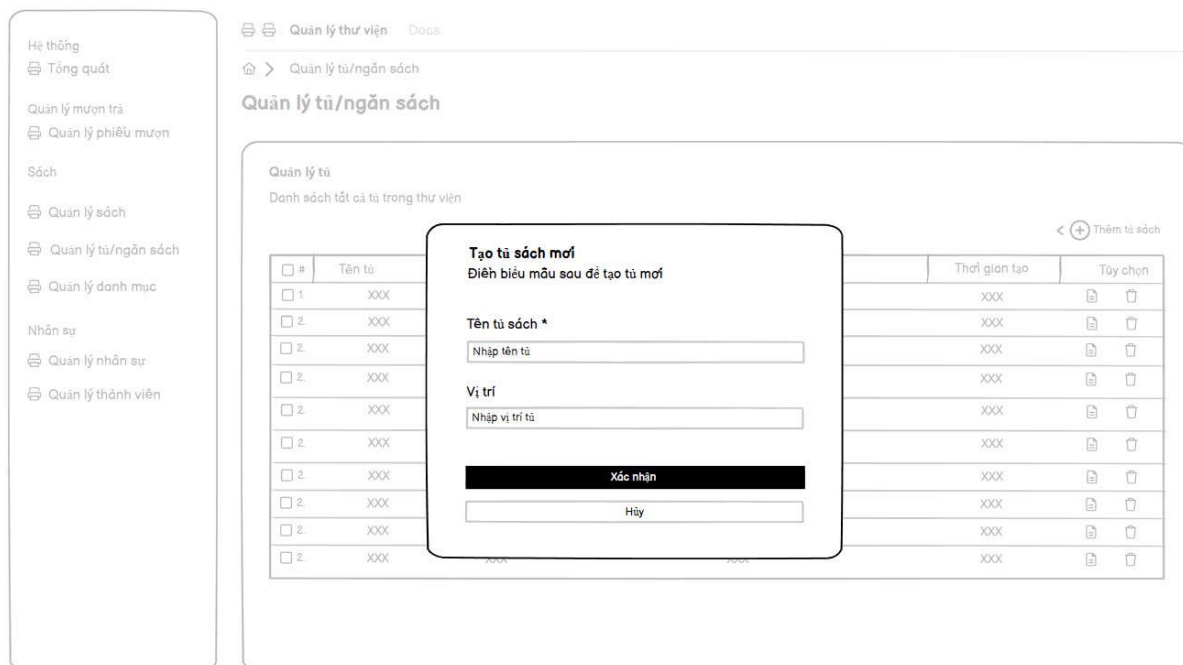
Hình 2.24: Hình dung màn hình giao diện Quản lý danh mục



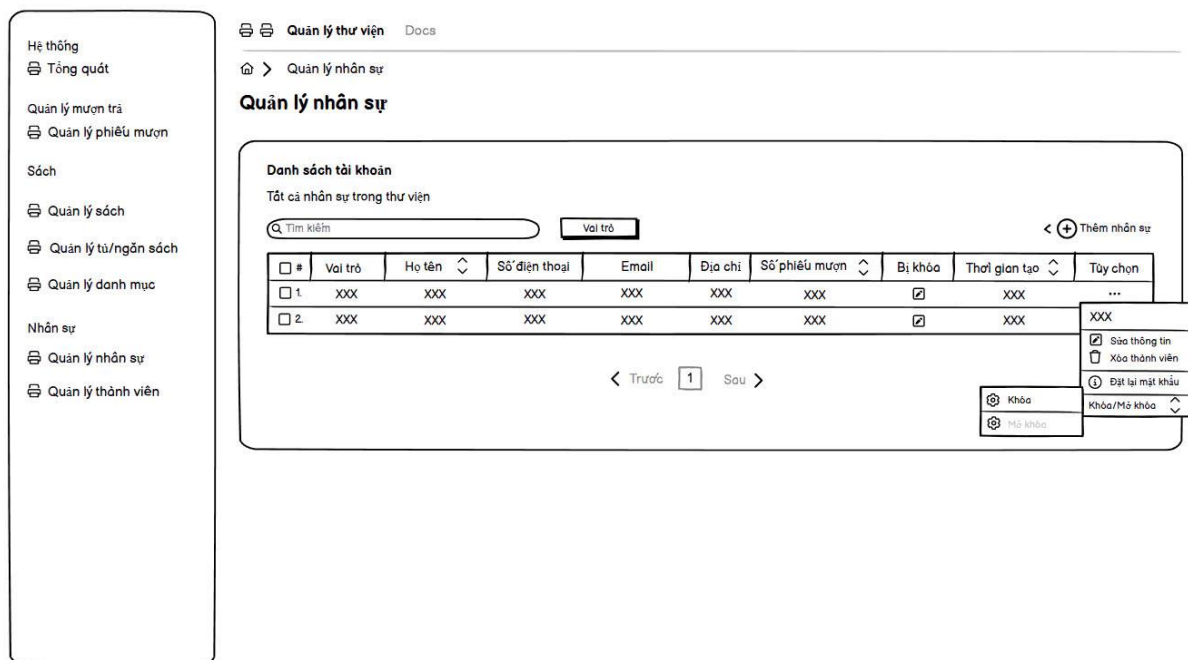
Hình 2.25: Hình dung màn hình giao diện Thêm danh mục mới



Hình 2.26: Hình dung màn hình giao diện Quản lý tủ/ngăn sách

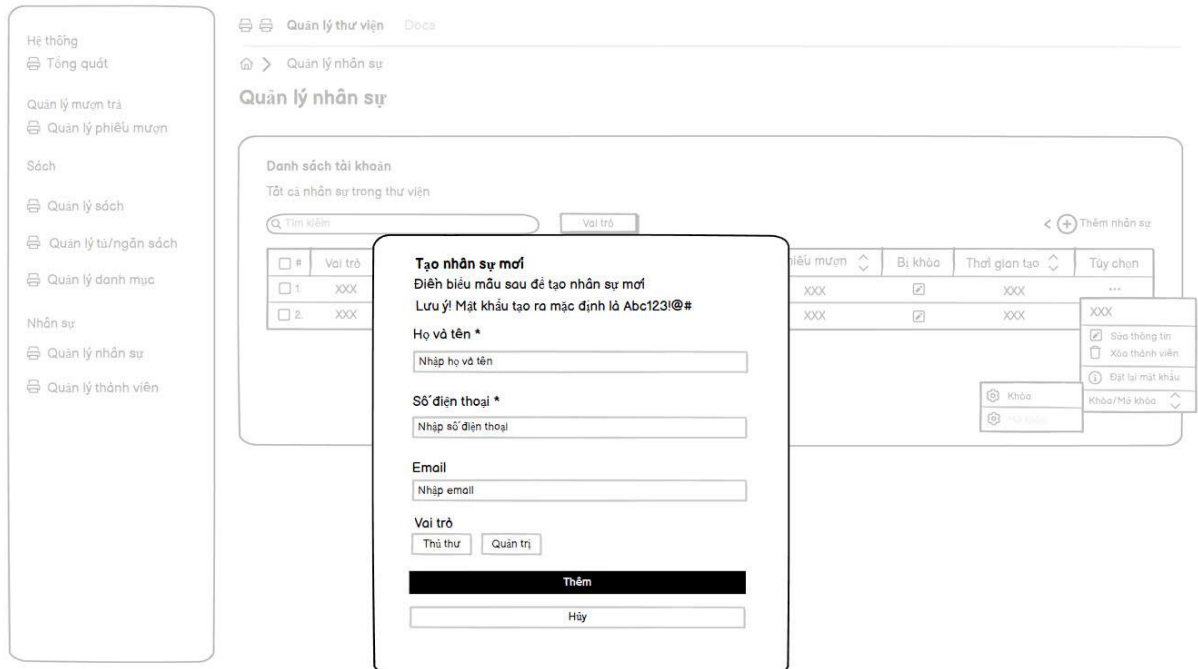


Hình 2.27: Hình dung màn hình giao diện Thêm từ/ngắn sách mới

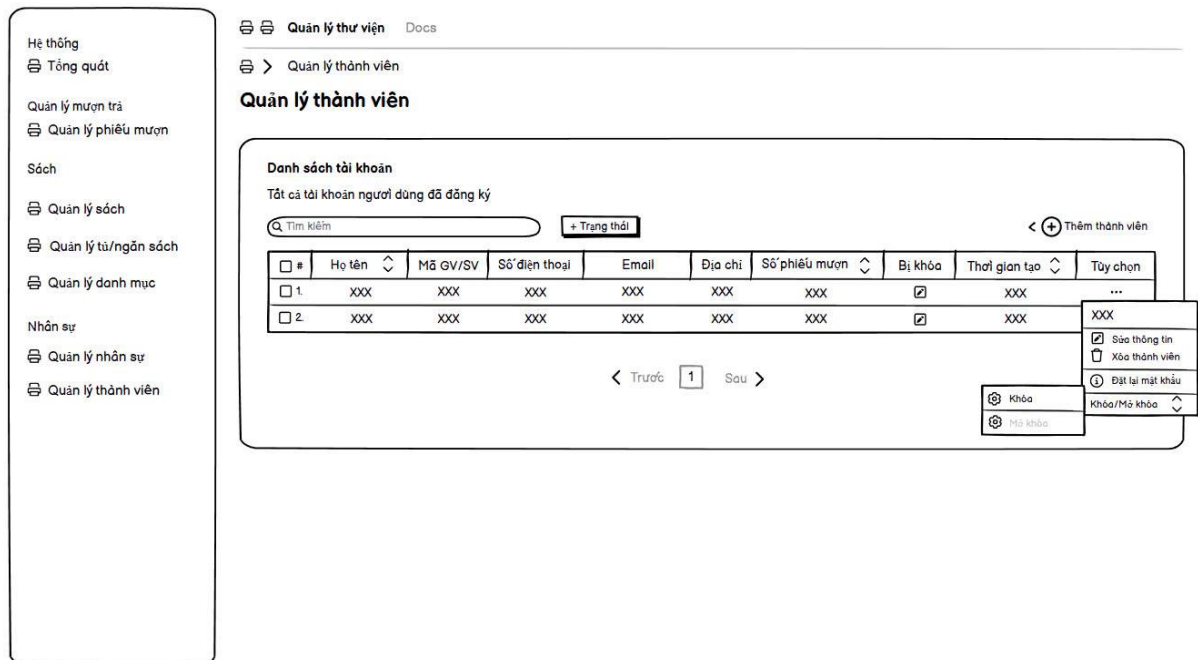


Hình 2.28: Hình dung màn hình giao diện Quản lý nhân sự



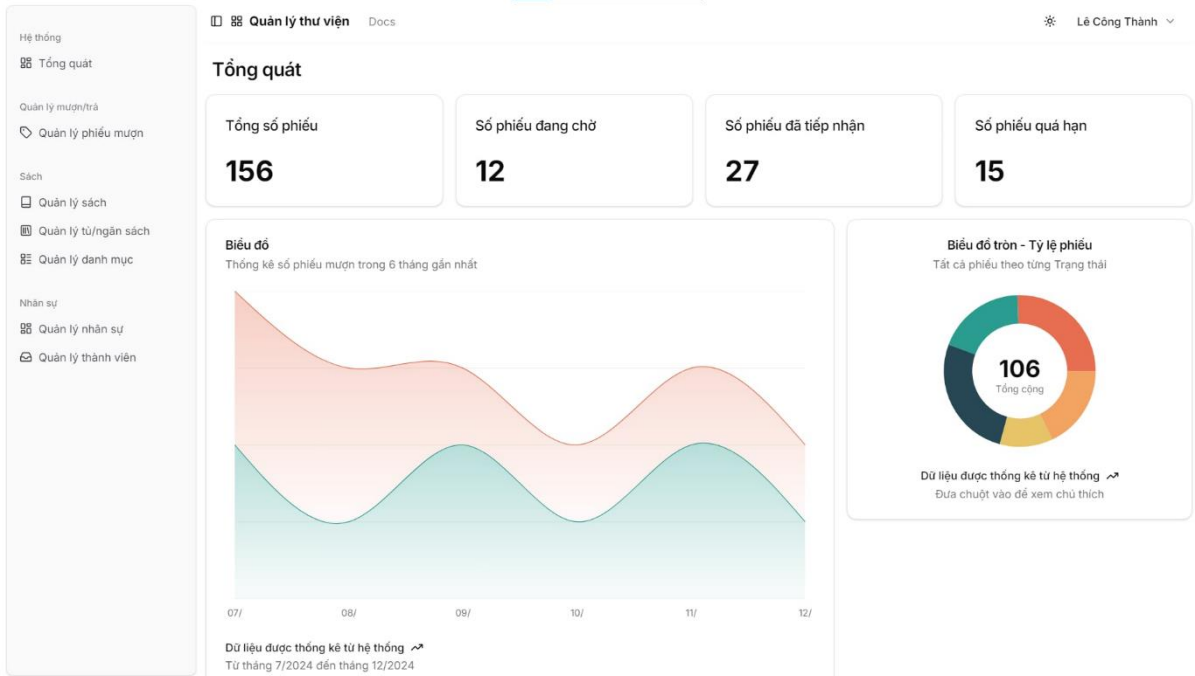


Hình 2.29: Hình dung màn hình giao diện Thêm nhân sự



Hình 2.30: Hình dung màn hình giao diện Quản lý thành viên

## 2.8. Kết quả giao diện



Hình 2.31: Kết quả giao diện màn hình Trang chủ

**Quản lý phiếu mượn**

Danh sách  
Tất cả phiếu mượn của thành viên

Tìm kiếm... Trạng thái phiếu + Thêm phiếu mới

#	Mã phiếu	Trạng thái	Người mượn	Số điện thoại	Ngày mượn	Hạn trả	Ngày trả	Sách	Tiền cọc	Nhân viên	Tùy chọn
1.	ACBXY123	Quá hạn	Lê Thị B	0834361813	00:00 - 17/12/2024	00:00 - 20/12/2024		Tý Quay - Tập 11, Tuổi thơ dẫu đời - Tập 1	96.000 đ	Nguyễn Văn A	...
2.	HCGDT845	Đã trả	Nguyễn Đức Thắng	0834361821	00:00 - 17/11/2024	00:00 - 20/11/2024	00:00 - 21/11/2024	Tuổi thơ dẫu đời - Tập 2	80.000 đ	Nguyễn Văn A	...
3.	HCGDT669	Đã trả	Trần Văn C	0834361814	00:00 - 10/11/2024	00:00 - 15/11/2024	00:00 - 12/11/2024	Hacker lược sử, SGK Tiếng Việt Lớp 1 - Tập một	171.000 đ	Nguyễn Văn A	...
4.	JGIRU857	Đã trả	Phạm Đức Thắng	0834361834	00:00 - 10/10/2024	00:00 - 17/10/2024	00:00 - 15/10/2024	Văn hóa nghệ thuật cũng là một mặt trận	95.000 đ	Lê Công Thành	...
5.	CNFJT094	Đã trả	Phạm Đức Thắng	0834361834	00:00 - 10/09/2024	00:00 - 18/09/2024	00:00 - 15/09/2024	Tập viết chữ Hán - phiên bản mới 2018	85.000 đ	Lê Công Thành	...
6.	KFODY748	Đã trả	Nguyễn Xuân Thành	0834361844	00:00 - 05/09/2024	00:00 - 12/09/2024	00:00 - 13/09/2024	Luật an ninh mạng	25.000 đ	Nguyễn Văn A	...

< Trước 1 Sau >

Hình 2.32: Kết quả giao diện màn hình Quản lý phiếu mượn

Hệ thống

Tổng quát

Quản lý mượn/trả

Quản lý phiếu mượn

Quỹ cọc sách

Sách

Quản lý sách

Quản lý từ/ngân sách

Quản lý danh mục

Tài khoản

Quản lý nhân sự

Quản lý thành viên

Quản lý thư viện

Trường ĐHCN Hà Nội

Khoa CNTT

Lê Công Thành

Quản lý tiền quỹ

Tiền quỹ cọc sách

Tổng tiền quỹ

100.000 đ

Tổng lượt giao dịch

12 lượt

Danh sách

Tất cả giao dịch đối với tiền

+

Thêm tiền vào

-

Rút tiền ra

#	Loại giao dịch	Số tiền	Nhân viên	Chi tiết	Thời gian	Tùy chọn
1.	Thêm vào	4.000 đ	Lê Công Thành	test	15:24 - 30/12/2024	
2.	Cọc phiếu	96.000 đ	Nguyễn Văn A	Lê Thị B đã cọc phiếu ACBXY123	00:00 - 17/12/2024	
3.	Hoàn cọc	80.000 đ	Nguyễn Văn A	Nguyễn Đức Thắng đã lấy lại cọc phiếu HCGDT845	00:00 - 21/11/2024	
4.	Cọc phiếu	80.000 đ	Nguyễn Văn A	Nguyễn Đức Thắng đã cọc phiếu HCGDT845	00:00 - 17/11/2024	
5.	Hoàn cọc	171.000 đ	Nguyễn Văn A	Trần Văn C đã lấy lại cọc phiếu HCGDT669	00:00 - 12/11/2024	
6.	Cọc phiếu	171.000 đ	Nguyễn Văn A	Trần Văn C đã cọc phiếu HCGDT669	00:00 - 10/11/2024	
7.	Hoàn cọc	95.000 đ	Lê Công Thành	Phạm Đức Thắng đã lấy lại cọc phiếu JGIRU857	00:00 - 15/10/2024	

Hình 2.33: Kết quả giao diện màn hình Quản lý tiền quỹ

Hệ thống

Tổng quát

Quản lý mượn/trả

Quản lý phiếu mượn

Quỹ cọc sách

Sách

Quản lý sách

Quản lý từ/ngân sách

Quản lý danh mục

Tài khoản

Quản lý nhân sự

Quản lý thành viên

Quản lý thư viện

Trường ĐHCN Hà Nội

Khoa CNTT

Lê Công Thành

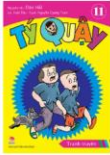





Quản lý sách

Danh sách

Tất cả các cuốn sách trong thư viện

+

Thêm sách mới

#	Tiêu đề	Ảnh	Giá bìa	Số lượng	Hiện có	Số trang	Tác giả	Danh mục	Ngân sách	Nhà xuất bản	Năm XB	Thời gian tạo	Tùy chọn
1.	Tý Quậy - Tập 11		40.000 đ	10	9	100	Đào Hải	Sách, Chuyên khảo, Tuyển tập	Truyện tranh	NXB Kim Đồng	2015	00:00 - 15/01/2024	 
2.	Luật an ninh mạng		25.000 đ	3	3	64	Quốc hội nước CHXHCNVN	Ấn phẩm định kỳ	Pháp luật 1	NXB Công An Nhân Dân	2019	00:00 - 15/01/2024	 

Hình 2.34: Kết quả giao diện màn hình Quản lý sách

Hệ thống

Tổng quát

Quản lý mượn/trả

Quản lý phiếu mượn

Quỹ cộc sách

Sách

Quản lý sách

Quản lý tủ/ngăn sách

Quản lý danh mục

Tài khoản

Quản lý nhân sự

Quản lý thành viên

Quản lý thư viện

Trường ĐHCN Hà Nội

Khoa CNTT

Lê Công Thành

Quản lý tủ/ngăn sách

Quản lý tủ

Danh sách tất cả tủ trong thư viện

Thêm tủ sách

#	Tên tủ	Vị trí	Chứa ngăn sách	Thời gian tạo	Tùy chọn
1.	Sách Chính trị - Xã hội	1-1	Chính trị - Xã hội 2 Chính trị - Xã hội 1	00:00 - 17/05/2023	
2.	Sách khác	3-2	Sách khác 1 Sách khác 2	00:00 - 17/05/2023	
3.	Sách Ngoại ngữ	2-1	Tiếng Trung Tiếng Anh	00:00 - 17/05/2023	
4.	Sách Pháp luật	2-3	Pháp luật 1	00:00 - 05/01/2024	
5.	Truyện	3-1	Truyện tranh Tiểu thuyết	00:00 - 06/01/2024	
6.	Văn học - Nghệ thuật	1-3	Văn học - Nghệ thuật 1	00:00 - 17/05/2023	
7.	SGK - Giáo trình	2-2	SGK Giáo trình	00:00 - 20/08/2023	
8.	Sách Khoa học - Công nghệ	1-2	Khoa học - Công nghệ 2 Khoa học - Công nghệ 1	00:00 - 17/05/2023	

Hình 2.35: Kết quả giao diện màn hình Quản lý tủ/ngăn sách

Hệ thống

Tổng quát

Quản lý mượn/trả

Quản lý phiếu mượn

Quỹ cộc sách

Sách

Quản lý sách

Quản lý tủ/ngăn sách

Quản lý danh mục

Tài khoản

Quản lý nhân sự

Quản lý thành viên

Quản lý thư viện

Trường ĐHCN Hà Nội

Khoa CNTT

Lê Công Thành

Quản lý danh mục

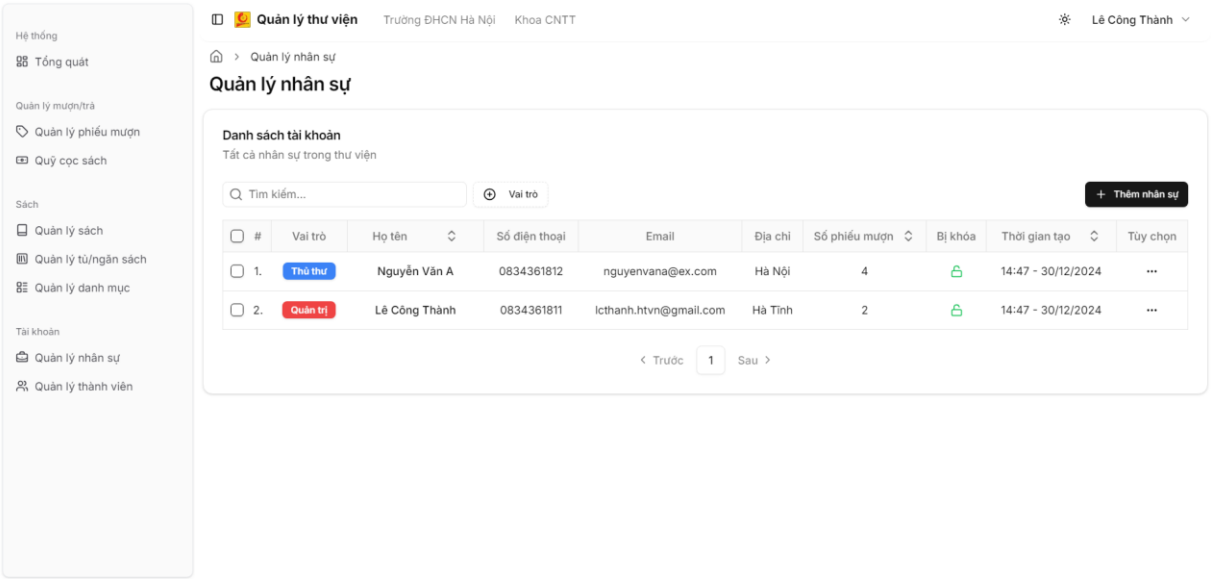
Danh sách

Danh mục các cuốn sách trong thư viện

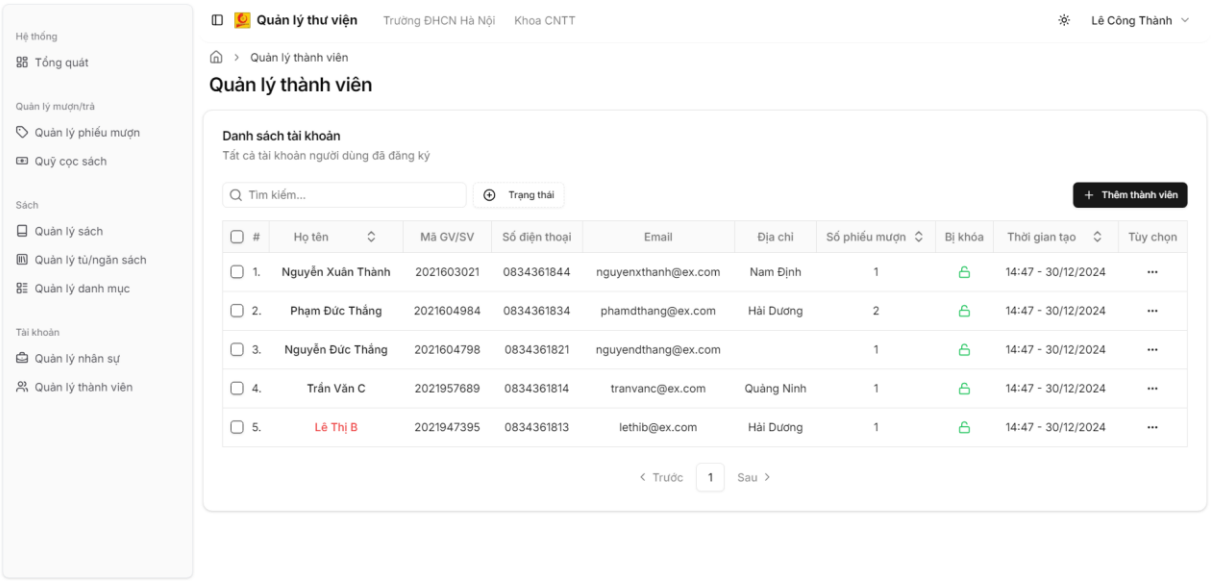
Thêm danh mục

#	Tên danh mục	Số lượng sách	Mô tả	Thời gian tạo	Tùy chọn
1.	Bài trích	1	Các bài viết trích từ các tài liệu khác nhau, thường là từ các ấn phẩm định kỳ hoặc tuyển tập.	00:00 - 17/05/2023	
2.	Sách, Chuyên khảo, Tuyển tập	10	Bao gồm các loại sách giáo trình, sách tham khảo, chuyên khảo và các tuyển tập bài viết khoa học.	00:00 - 17/05/2023	
3.	Ấn phẩm định kỳ	3	Các ấn phẩm được xuất bản định kỳ như tạp chí, báo và các bản tin.	00:00 - 17/05/2023	

Hình 2.36: Kết quả giao diện màn hình Quản lý danh mục



Hình 2.37: Kết quả giao diện màn hình Quản lý nhân sự



Hình 2.38: Kết quả giao diện màn hình Quản lý thành viên

2.9. Kiểm thử ứng dụng

Bảng 2.10: Bảng kế hoạch kiểm thử

Người kiểm thử	Nội dung	Thời gian	Cách thức
----------------	----------	-----------	-----------

Lê Công Thành	Kiểm thử chức năng Quản lý phiếu mượn, Quản lý sách, Phân quyền và Xác thực người dùng.	09/12/2024	Kiểm thử thủ công
Nguyễn Đức Thắng	Kiểm thử chức năng Quản lý danh mục, Báo cáo thống kê.	09/12/2024	Kiểm thử thủ công
Phạm Đức Thắng	Kiểm thử chức năng Quản lý nhân sự, Quản lý thành viên.	09/12/2024	Kiểm thử thủ công
Nguyễn Xuân Thành	Kiểm thử chức năng Quản lý tủ, ngăn sách.	09/12/2024	Kiểm thử thủ công

## 2.10. Cài đặt và triển khai ứng dụng

### **Bước 1:** Cài đặt SQL Server:

- Truy cập đường link: <https://www.microsoft.com/en-us/sql-server/sql-server-downloads>
- Chọn 1 trong 2 phiên bản Developer hoặc Express.

Or, download a free specialized edition



Developer

SQL Server 2022 Developer is a full-featured free edition, licensed for use as a development and test database in a non-production environment.

[Download now](#)



Express

SQL Server 2022 Express is a free edition of SQL Server, ideal for development and production for desktop, web, and small server applications.

[Download now](#)

- Nhấn **Download** và tiến hành cài đặt.
- Truy cập <https://learn.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16> để tải SSMS (SQL Server Management Studio).
- Tiến hành cài đặt SSMS như bình thường.

### **Bước 2:** Mở SMSS và tạo CSDL:

- Mở ứng dụng SSMS vừa cài đặt.
- Đăng nhập bằng tài khoản SQL Server (chọn Authentication: SQL Server Authentication hoặc Windows Authentication).
- Trong **Object Explorer**, nhấp chuột phải vào **Databases** > **New Database**.
- Nhập tên cơ sở dữ liệu, ví dụ: "LibraryDB".

Script ? Help

Database name: LibraryDB

Owner: <default> ...

☒ Use full-text indexing

Database files:

Logical Name	File Type	Filegroup	Initial Size (MB)	Autogrowth / Maxsize	Path
LibraryDB	ROWS ...	PRIMARY	8	By 64 MB, Unlimited	C:\Program
LibraryDB_log	LOG	Not Applicable	8	By 64 MB, Unlimited	C:\Program

- Nhấn **OK** để tạo cơ sở dữ liệu.

### **Bước 3:** Cài đặt Microsoft Visual Studio và .NET SDK:

- Truy cập: <https://dotnet.microsoft.com/en-us/download/dotnet/6.0>
- Tải bản **.NET SDK 6.0** (phiên bản yêu cầu cho dự án).

Build apps - SDK ⓘ

## SDK 6.0.428

OS	Installers	Binaries
Linux	<a href="#">Package manager instructions</a>	<a href="#">Arm32</a>   <a href="#">Arm32 Alpine</a>   <a href="#">Arm64</a>   <a href="#">Arm64 Alpine</a>   <a href="#">x64</a>   <a href="#">x64 Alpine</a>
macOS	<a href="#">Arm64</a>   <a href="#">x64</a>	<a href="#">Arm64</a>   <a href="#">x64</a>
Windows	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>   <a href="#">winget instructions</a>	<a href="#">x64</a>   <a href="#">x86</a>   <a href="#">Arm64</a>
All	<a href="#">dotnet-install scripts</a>	

**Visual Studio support**  
Visual Studio 2022 (v17.3)

**Included runtimes**  
.NET Runtime 6.0.36  
ASP.NET Core Runtime 6.0.36  
.NET Desktop Runtime 6.0.36

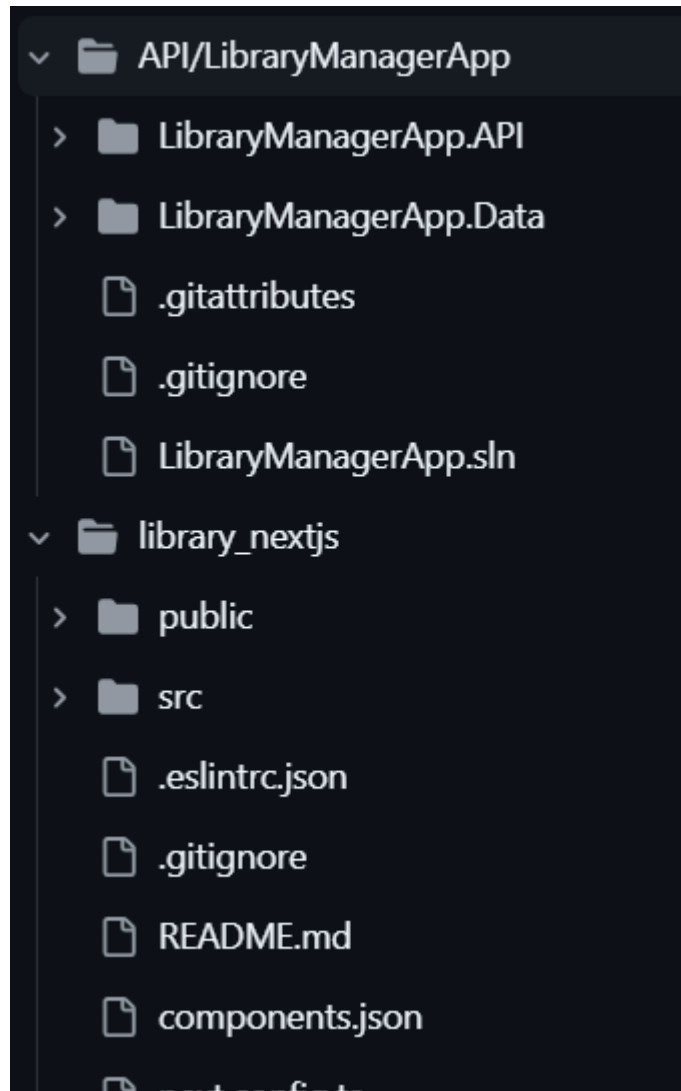
**Language support**  
C# 10.0  
F# 6.0  
Visual Basic 16.9

- Chọn phiên bản cài phù hợp cho máy tính, tải về và tiến hành cài đặt.
- Truy cập <https://visualstudio.microsoft.com/> để tải Visual Studio 2022 nếu chưa cài đặt.

**Bước 4:** Tải mã nguồn của dự án:

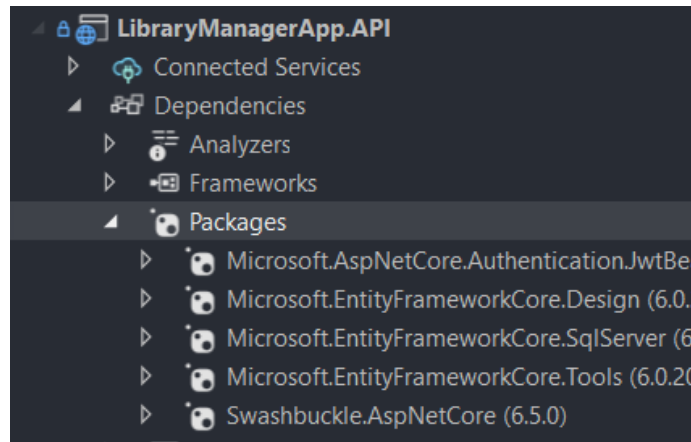
- Truy cập đường link Github của dự án (nhánh Main): [https://github.com/lc-thanh/DACN\\_Nhom21/tree/main](https://github.com/lc-thanh/DACN_Nhom21/tree/main)
- Có thể tải trực tiếp mã nguồn về hoặc dùng lệnh “**git clone**” sau đó “**git checkout**” sang nhánh **Main**.
- Cấu trúc mã nguồn của dự án:



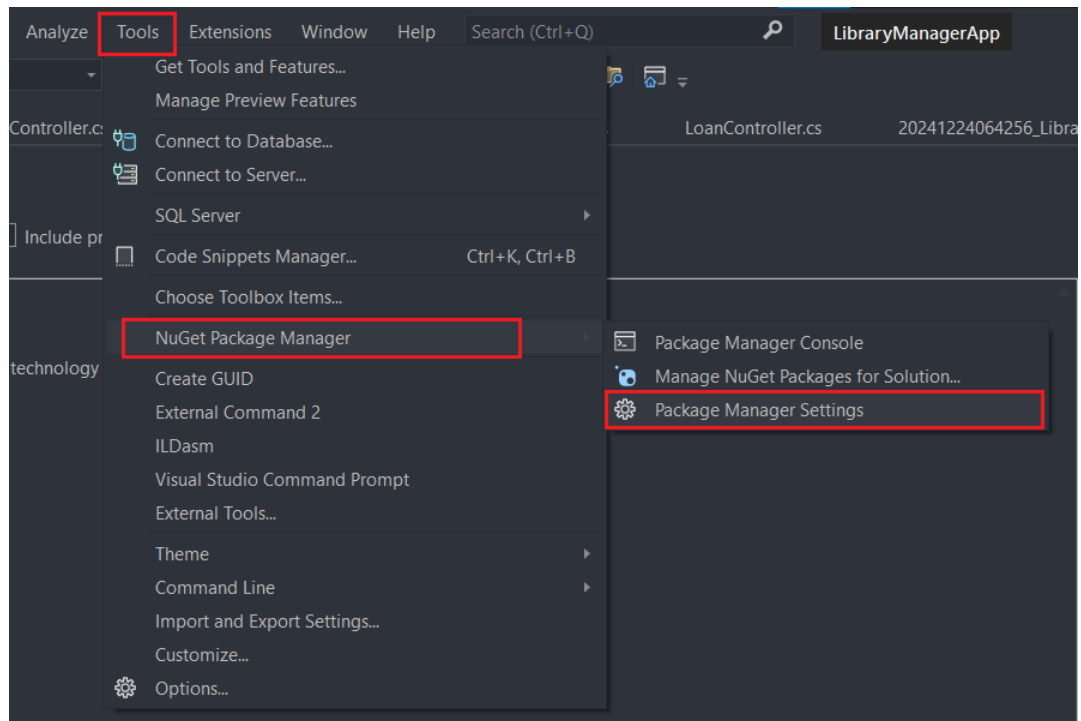


**Bước 5:** Triển khai Backend:

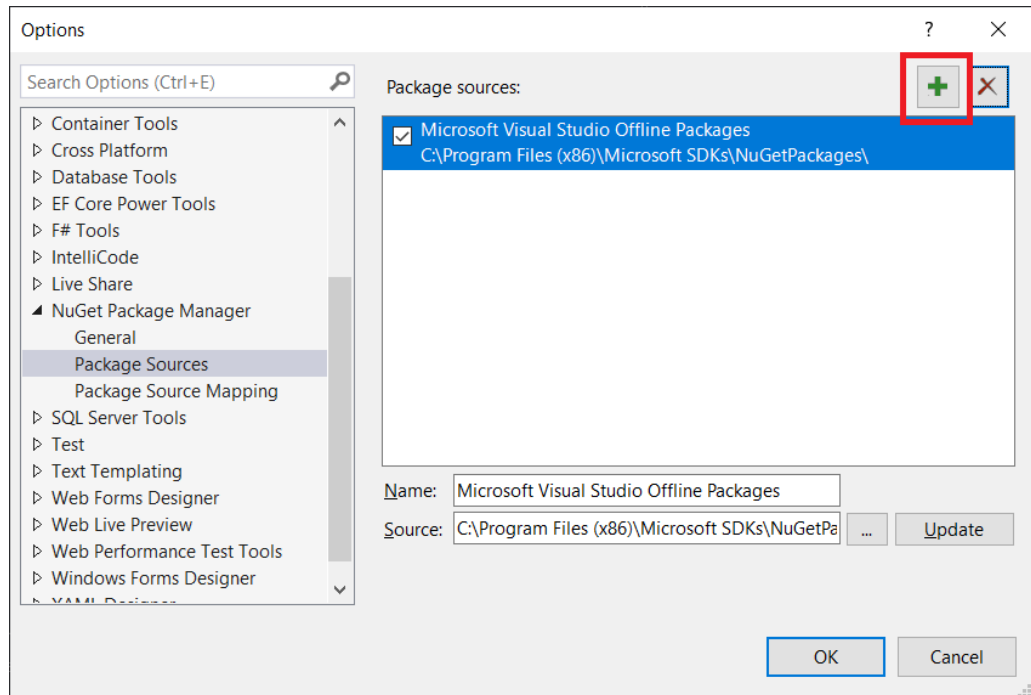
- Mở Visual Studio Solution **LibraryManagerApp.sln** nằm trong thư mục /API/LibraryManagerApp/.
- Nếu các Packages của 2 Project trong Solution báo lỗi (nếu không bị lỗi thì có thể bỏ qua các bước này):



1) Vào **Tools > NuGet Package Manager > Package Manager Settings**.

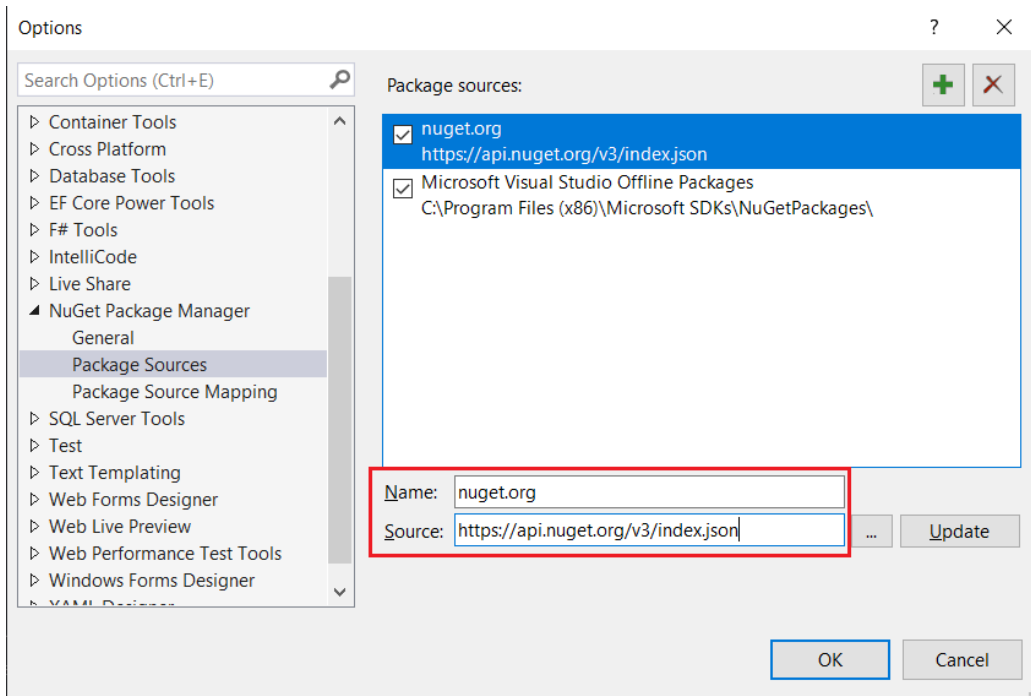


2) Vào **Package Sources**, nhấp vào biểu tượng + để thêm nguồn mới.



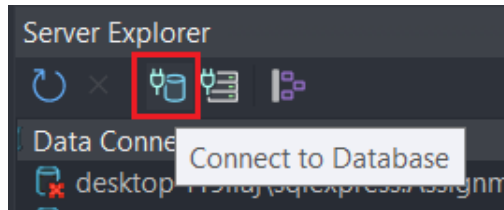
### 3) Điền thông tin:

- + **Name:** nuget.org và
- + **Source:** <https://api.nuget.org/v3/index.json>

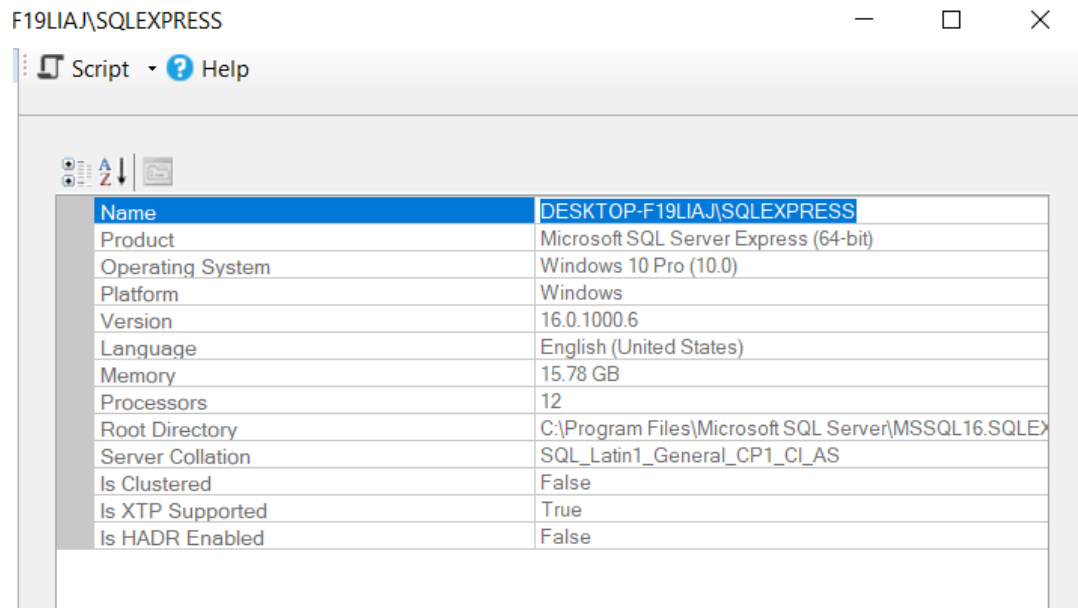


### 4) Nhấn **Update** và **OK** để lưu thay đổi.

- Vào **View > Other Windows > Server Explorer** (hoặc nhấn Ctrl + Alt + S) để mở cửa sổ Server Explorer.
- Chọn biểu tượng Connect to Database.



- Trở lại SSMS, chuột phải vào Server của mình chọn **Properties**, rồi sao chép Server Name.



- Dán Server Name vào ô Textbox rồi chọn Database đã tạo ở bước 2.

Add Connection ? X

Enter information to connect to the selected data source or click "Change" to choose a different data source and/or provider.

Data source: Microsoft SQL Server (SqlClient) Change...

Server name: **DESKTOP-F19LIAJ\SQLEXPRESS** Refresh

Log on to the server

Authentication: Windows Authentication

User name:

Password:

☐ Save my password

Connect to a database

☒ Select or enter a database name:

**LibraryManagerAppDB**

☐ Attach a database file:

Browse...

Logical name:

Advanced...

Test Connection OK Cancel

- Vào **Advanced...** sửa **Trust Server Certificate** thành **True**.

Advanced Properties ? X

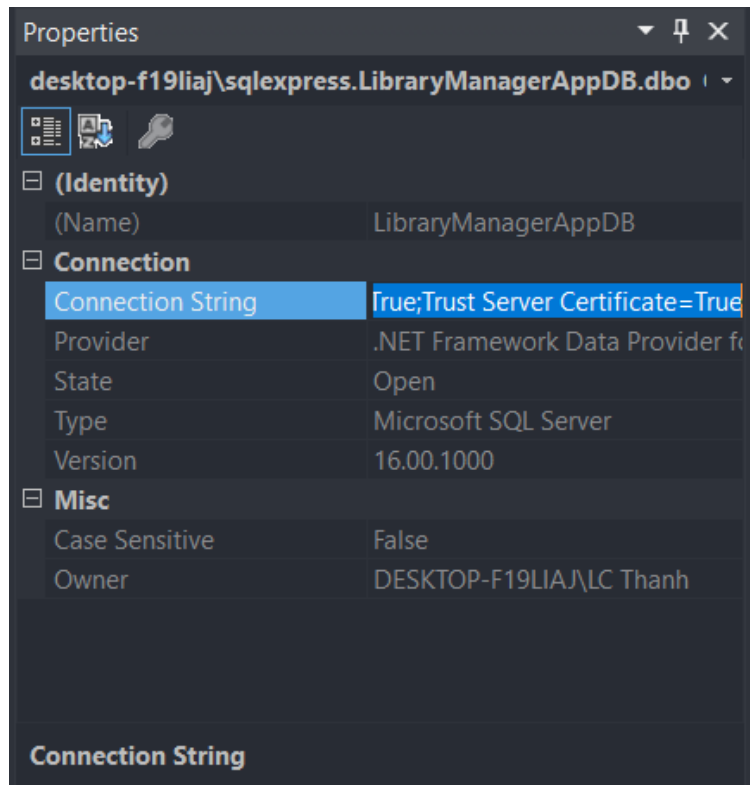
Encrypt	False
Integrated Security	<b>True</b>
IP Address Preference	IPv4First
Password	
Persist Security Info	False
<b>Trust Server Certificate</b>	<b>True</b>
User ID	
<b>Source</b>	
AttachDbFilename	
Context Connection	False
Data Source	<b>DESKTOP-F19LIAJ\SQLEXPRESS</b>
Failover Partner	

**Trust Server Certificate**  
When true (and encrypt=true), SQL Server uses SSL encryption for all data sent between the client and...

Data Source=DESKTOP-F19LIAJ\SQLEXPRESS;Initial C;

OK Cancel

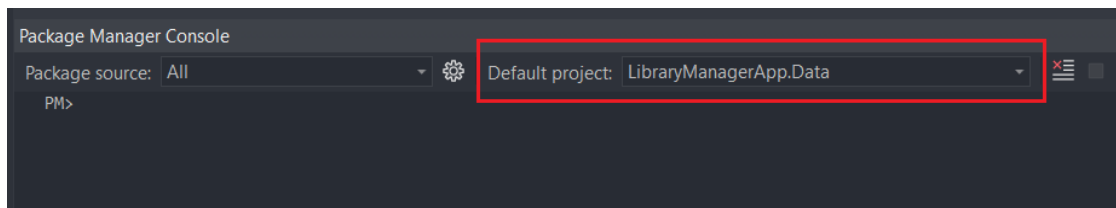
- Sau khi kết nối thành công, nhấn phải chuột vào Database vừa kết nối và chọn **Properties**, rồi sao chép Connection String.



- Trở lại Solution Explorer, mở file **appsettings.json** và thay đổi giá trị của “DefaultConnection” bằng Connection String vừa sao chép ở trên.

```
"ConnectionStrings": {
  "DefaultConnection": "Data Source=DESKTOP-F19LIAJ\\SQLEXPRESS;Initial Catalog=LibraryManagerAppDB;Integrated Security=True;Trust Server Certificate=True"
}
```

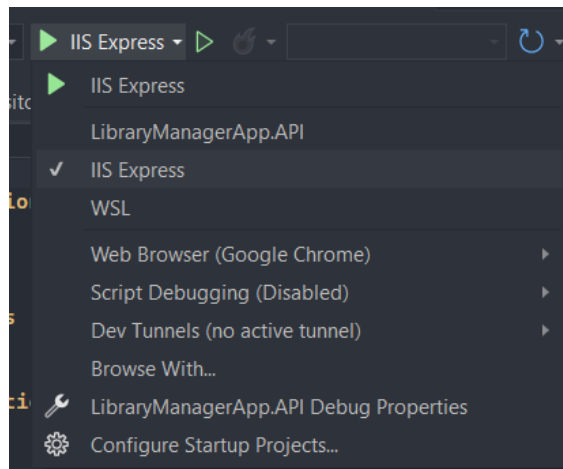
- Vào **Tools > NuGet Package Manager > Package Manager Console**, chọn Default Project là **LibraryManagerApp.Data**.



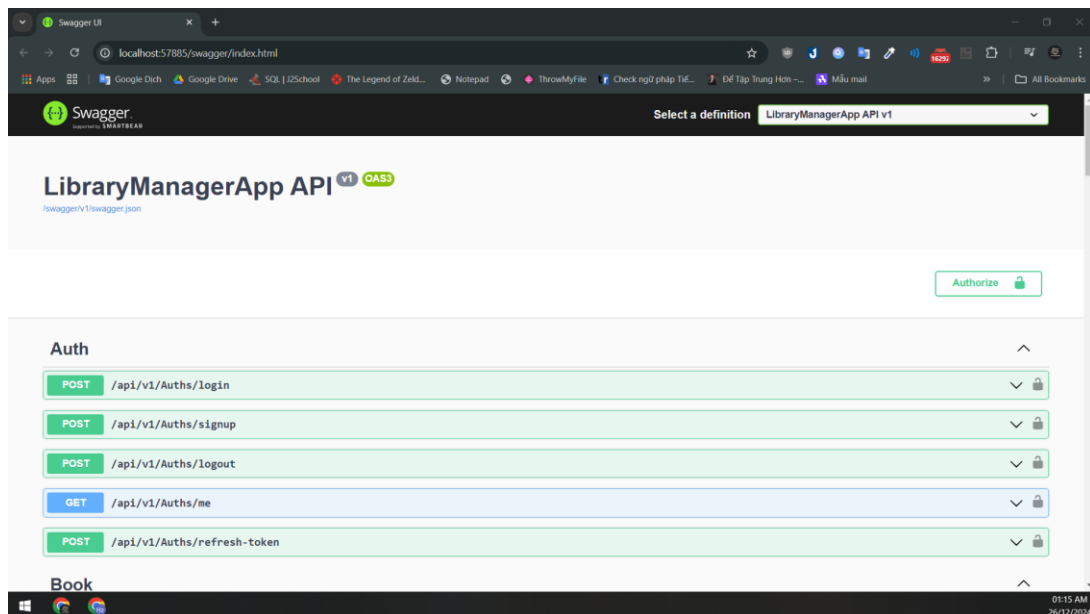
- Nhập lệnh “Add-Migration LibraryDb”, và Enter. Nếu nhận được kết quả như sau là thành công:

```
PM> Add-Migration LibraryDb
Build started...
Build succeeded.
Microsoft.EntityFrameworkCore.Infrastructure[10403]
    Entity Framework Core 6.0.20 initialized 'LibraryManagerAppDbContext' using provider 'Microsoft.EntityFrameworkCore.SqlServer:6.0.20' with options: None
To undo this action, use Remove-Migration.
PM>
```

- Tiếp theo chạy lệnh “Update-Database”, nếu thành công sẽ nhận được kết quả “Done.” ở dòng cuối cùng.
- Tiến hành chạy Project LibraryManagerApp.API bằng **IIS Express**.



- Nếu màn hình hiển thị SwaggerUI, là đã triển khai Backend thành công.



**Bước 6:** Triển khai Frontend:

- Truy cập <https://nodejs.org/> và cài đặt **Node.js** nếu như chưa cài.
- Chạy lệnh “npm install -g pnpm” bằng Window PowerShell để cài đặt **pnpm**.
- Nhập lệnh “cd” để di chuyển đến thư mục chứa mã nguồn của Frontend là **library\_nextjs**.
- Chạy lệnh “pnpm install” để cài đặt các package phụ thuộc.
- Sửa file có tên “.env.example” thành “.env” (chỉnh sửa API ENPOINT nếu như Backend có đường dẫn khác với đường dẫn có sẵn trong file)
- Tiến hành chạy lệnh “pnpm dev” để khởi chạy Website, nếu chương trình đang “Ready”, thì đã có thể truy cập được ứng dụng.

```
> next dev --turbo
```

```
▲ Next.js 15.0.3 (Turbopack)
- Local:      http://localhost:3000
- Environments: .env

✓ Starting...
✓ Compiled in 599ms
✓ Ready in 2.1s
```

- Truy cập <http://localhost:3000/> để sử dụng ứng dụng.



## CHƯƠNG 3: KẾT LUẬN VÀ BÀI HỌC KINH NGHIỆM

### 3.1. Kiến thức đã học được

Trong quá trình thực hiện đề tài “*Xây dựng website quản lý mượn trả sách cho thư viện Trường ĐHCNH*”, cá nhân từng thành viên trong nhóm đều đã tích lũy được nhiều kinh nghiệm, kiến thức chuyên ngành, bao gồm:

- Kiến thức về công nghệ: Sử dụng hiệu quả các công nghệ như ASP.NET Core Web API để xây dựng backend, Next.js để xây dựng frontend, và SQL Server để quản lý cơ sở dữ liệu.
- Quản lý người dùng và phân quyền: Hiểu rõ cách thiết kế và triển khai hệ thống phân quyền, đảm bảo chỉ các vai trò cụ thể (Quản trị, Thủ thư, Thành viên) có quyền truy cập các tính năng nhất định.
- Quản lý dữ liệu phức tạp: Xây dựng cơ sở dữ liệu và các API để quản lý dữ liệu liên quan đến sách, tủ sách, phiếu mượn, tài khoản người dùng, và danh mục một cách tối ưu.
- Tối ưu giao diện người dùng: Ứng dụng thư viện giao diện ShadcnUI trong việc xây dựng giao diện trực quan và thân thiện.

### 3.2. Bài học kinh nghiệm

- Lập kế hoạch rõ ràng: Phân chia công việc và thời gian hợp lý giữa các thành viên trong nhóm.
- Khả năng giải quyết vấn đề: Trong quá trình phát triển, nhóm đã phải đối mặt với nhiều thách thức kỹ thuật như xử lý lỗi khi gọi API, tối ưu hóa hiệu suất API, và đảm bảo tính bảo mật cho hệ thống.
- Tầm quan trọng của kiểm thử: Để đảm bảo hệ thống hoạt động ổn định, nhóm đã học cách kiểm thử API, giao diện người dùng, và các kịch bản sử dụng khác nhau.

- Giao tiếp trong nhóm: Học cách phối hợp và giao tiếp hiệu quả giữa các thành viên thông qua các công cụ như Git và báo cáo tiến độ hàng tuần.

### **3.3. Tính khả thi của đề tài**

- Tính ứng dụng thực tế cao: Đề tài đáp ứng được nhu cầu thực tiễn trong quản lý thư viện, đặc biệt là trong việc tự động hóa quy trình thủ công.
- Khả năng mở rộng: Hệ thống được thiết kế với kiến trúc module, cho phép dễ dàng thêm các tính năng mới như thông báo mượn sách, tích hợp thanh toán trực tuyến, hoặc báo cáo nâng cao.
- Tính bảo mật: Sử dụng JWT để xác thực người dùng và các tiêu chuẩn mã hóa đảm bảo an toàn dữ liệu.

### **3.4. Những thuận lợi và khó khăn**

- Thuận lợi:
  - + Có sẵn các tài liệu tham khảo phong phú từ các nguồn học thuật và trực tuyến.
  - + Sự hỗ trợ từ các giảng viên và các thành viên trong nhóm.
  - + Công nghệ hiện đại như ASP.NET Core và Next.js giúp rút ngắn thời gian phát triển.
- Khó khăn:
  - + Hạn chế về thời gian do song song thực hiện các môn học khác.
  - + Một số lỗi phức tạp liên quan đến cơ sở dữ liệu và tích hợp frontend-backend mất nhiều thời gian để xử lý.
  - + Thiếu kinh nghiệm thực tế trong xử lý tình huống khi hệ thống có lượng dữ liệu lớn.

## TÀI LIỆU THAM KHẢO

### Tiếng Việt

- [1] Nguyễn Thị Thanh Huyền, Ngô Thị Bích Thúy (2011), *Giáo trình cơ sở dữ liệu*, Trường ĐH Công nghiệp Hà Nội.
- [2] Vũ Thị Dương, Phùng Đức Hòa, Nguyễn Thị Hương Lan (2015), *Giáo trình phân tích thiết kế hướng đối tượng*, Trường ĐH Công nghiệp Hà Nội.

### Tiếng Anh

- [3] Microsoft Docs. (n.d.). *ASP.NET Core Documentation*.
- [4] Next.js Documentation (n.d.). <https://nextjs.org/docs>
- [5] Galloway, J., Wilson, P., Allen, B., & Matson, K. (2018). *ASP.NET Core in Action. Manning Publications*.
- [6] SQL Server Documentation. (n.d.). *SQL Server Documentation*.