

TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG HÀ NỘI
KHOA CÔNG NGHỆ THÔNG TIN



KỲ THI KẾT THÚC HỌC PHẦN
HỌC KỲ II NĂM HỌC 2020 – 2021

Đề tài bài tập lớn: Xây dựng website hỗ trợ công tác quản lý sách của thư viện

Họ và tên học sinh/sinh viên: Hoàng Phúc
Mã học sinh/sinh viên: 1811061739
Lớp: ĐH8C6
Tên học phần: Công nghệ Java
Giảng viên hướng dẫn: Nguyễn Văn Hách

Hà Nội, ngày 10 tháng 11 năm 2021

MỤC LỤC

MỞ ĐẦU.....	1
1.1 Giới thiệu đề tài	1
1.2 Giới thiệu công nghệ sử dụng làm đề tài	1
CHƯƠNG 1: PHÂN TÍCH HỆ THỐNG.....	3
1.1 Giới thiệu	3
1.1.1 Lí do chọn đề tài	3
1.1.2 Tầm quan trọng của việc ứng dụng máy tính cho hệ thống	3
1.1.3 Nghiên cứu sơ bộ về hệ thống hiện hành	3
1.2 Xác định các yêu cầu của hệ thống.....	3
1.3 Xây dựng biểu đồ ca sử dụng	4
1.3.1 Xác định các tác nhân của hệ thống	4
1.3.2 Xác định các ca sử dụng của hệ thống	4
1.3.3 Đặc tả các ca sử dụng	5
1.4 Biểu đồ hoạt động.....	7
1.4.1 Biểu đồ hoạt động đăng nhập.....	7
1.4.2 Biểu đồ hoạt động tìm kiếm thông tin.....	8
1.4.3 Biểu đồ hoạt động quản lí sách.	8
1.4.4 Biểu đồ hoạt động quản lí loại sách.	9
1.4.5 Biểu đồ hoạt động quản lí mượn trả sách.....	9
1.5 Biểu đồ trình tự	10
1.5.1 Biểu đồ trình tự chức năng đăng nhập	10
1.5.2 Biểu đồ trình tự chức năng tìm kiếm thông tin sách và loại sách	10
1.5.3 Biểu đồ trình tự chức năng thêm sách.....	10
1.5.4 Biểu đồ trình tự chức năng mượn sách	10
1.6 Xây dựng biểu đồ lớp lĩnh vực	11
1.6.1 Sơ đồ lớp của bài toán	11
1.6.2 Mô hình dữ liệu quan hệ.....	11
CHƯƠNG 2: THIẾT KẾ GIAO DIỆN.....	12
2.1 Giao diện đăng nhập	12
2.2 Giao diện cho thủ thư.....	12
2.3 Giao diện cho độc giả	15
KẾT LUẬN.....	17
PHỤ LỤC.....	18

TÀI LIỆU THAM KHẢO.....	27
TÀI LIỆU CÓ NGUỒN TỪ INTERNET	27

MỞ ĐẦU

1.1 Giới thiệu đề tài

Thư viện là nơi lưu trữ một khối lượng kiến thức đồ sộ và hết sức quý giá đối với việc học tập, nghiên cứu của sinh viên và giảng viên. Nhưng hiện nay, việc quản lý thư viện theo cách truyền thống ít nhiều đã gây khó khăn cho độc giả lẫn người quản lý. Về phía độc giả, họ không thể nắm được danh mục sách tại thư viện đó, cũng như không có gì đảm bảo cho việc họ có thể mượn được quyển sách ưng ý khi đến thư viện. Còn về phía người quản lý thư viện, công việc quản lý mượn sách một cách thủ công chiếm của họ khá nhiều thời gian, sức lực và vật chất. Chính những lý do trên đã khiến cho việc tin học hóa các công tác văn phòng, thủ tục hành chính trở thành một xu thế tất yếu. Những thiết bị lưu trữ, hệ thống thông tin đang dần thay thế những tủ hồ sơ khổng lồ. Trước những lợi ích mà công nghệ thông tin mang lại, việc thay đổi cách thức quản lý thư viện đã trở thành một nhu cầu thực tiễn và hoàn toàn nằm trong tầm tay.

1.2 Giới thiệu công nghệ sử dụng làm đề tài

1.2.1 Java

Java là một ngôn ngữ lập trình và là một công nghệ được sử dụng để làm cơ sở nền tảng cho việc phát triển các ứng dụng. Java là một ngôn ngữ lập trình cấp cao, mạnh mẽ, hướng đối tượng và an toàn. Java được phát triển bởi Sun Microsystems (hiện là công ty con của Oracle) vào năm 1995. James Gosling được biết đến như là cha đẻ của Java. Trước Java, tên của nó là Oak. Vì Oak đã là một công ty đã đăng ký, vì vậy James Gosling và nhóm của ông đã đổi tên từ Oak thành Java. Java đang có 4 nền tảng hoặc phiên bản: Java SE (Java Standard Edition), Java EE (Java Enterprise Edition), Java ME (Java Micro Edition), JavaFX.

Hiện nay Java đang được rất nhiều thiết bị sử dụng như máy tính để bàn, các ứng dụng web, các thiết bị di động, games, robotics, smart card, v.v. Ước tính có khoảng 3 tỷ thiết bị đang chạy Java.

Java chủ yếu có 4 loại ứng dụng có thể được tạo bằng lập trình Java:

- Standalone Application

Các ứng dụng chạy độc lập hay còn được gọi là ứng dụng máy tính để bàn hoặc các ứng dụng trên windows. Đây là những phần mềm truyền thống mà chúng ta cần cài đặt trên mọi máy. Ví dụ: Media player, antivirus, v.v. AWT và Swing được sử dụng trong Java để tạo các ứng dụng độc lập.

- **Web Application**

Là một ứng dụng chạy ở phía máy chủ và tạo ra một trang web động được gọi là ứng dụng web. Hiện tại có các công nghệ như Servlet, JSP, Struts, Spring, Hibernate, JSF, v.v. được sử dụng để tạo các ứng dụng web bằng Java.

- **Enterprise Application**

Là một ứng dụng được phân phối trong thị trường ngày nay, chẳng hạn như các ứng dụng ngân hàng, v.v. được gọi là ứng dụng doanh nghiệp. Nó có những ưu điểm như bảo mật cấp cao, cân bằng tải và phân cụm. Trong Java, EJB được sử dụng để tạo các ứng dụng doanh nghiệp.

- **Mobile Application**

Ứng dụng được tạo cho thiết bị di động được gọi là ứng dụng di động. Hiện tại như là Android và Java ME được sử dụng để tạo các ứng dụng di động.

1.2.2 Servlet

Công nghệ Servlet được sử dụng để tạo một ứng dụng web (nằm ở phía máy chủ và tạo ra một trang web động). Công nghệ Servlet mạnh mẽ và có thể mở rộng nhờ ngôn ngữ java. Trước Servlet, ngôn ngữ kịch bản CGI (Common Gateway Interface) phổ biến như một ngôn ngữ lập trình phía máy chủ. Tuy nhiên, có nhiều nhược điểm đối với công nghệ này. Chúng tôi đã thảo luận về những nhược điểm này dưới đây. Có nhiều giao diện và lớp trong API Servlet như Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse, v.v.

1.2.3 JSP

Công nghệ JSP được sử dụng để tạo ứng dụng web giống như công nghệ Servlet. Nó có thể được coi là một phần mở rộng cho Servlet vì nó cung cấp nhiều chức năng hơn servlet như ngôn ngữ biểu thức, JSTL, v.v. Một trang JSP bao gồm các thẻ HTML và các thẻ JSP. Các trang JSP dễ bảo trì hơn Servlet vì chúng ta có thể tách biệt việc thiết kế và phát triển. Nó cung cấp một số tính năng bổ sung như Expression Language, Custom Tags, v.v.

CHƯƠNG 1: PHÂN TÍCH HỆ THỐNG

1.1 Giới thiệu

1.1.1 Lí do chọn đề tài

Thư viện trong một trường Đại Học là nơi các sinh viên dành nhiều thời gian để thực hiện việc nghiên cứu, học tập, cũng như tìm thấy những quyển sách hay để trau dồi kiến thức của mình. Do đó việc phát triển và quản lý tốt thư viện là hết sức cần thiết cho việc tự học của sinh viên.

Nắm bắt được tình hình hiện tại và nhu cầu của người dùng cùng với sự phát triển của công nghệ hiện nay em đã thực hiện đề tài: “Xây dựng website quản lý thư viện trường Đại học”. Dựa trên ngôn ngữ Java cùng với hệ quản trị cơ sở dữ liệu MySQL.

1.1.2 Tầm quan trọng của việc ứng dụng máy tính cho hệ thống

Từ lâu, việc ứng dụng công nghệ thông tin trong quản lý đã trở thành xu hướng bởi nó không chỉ mang lại hiệu quả vượt trội mà còn tiết kiệm nhân lực, tiền bạc và thời gian hơn hẳn. Nhưng đối với một thư viện điện tử, nếu chỉ dừng lại ở quản lý sách trong thư viện thì vẫn còn chưa đủ. Trong thời đại internet bùng nổ như hiện nay, việc tích hợp tính năng tra cứu và đặt mượn sách trực tuyến hứa hẹn khả năng phục vụ sinh viên mọi lúc, mọi nơi, và cũng tối ưu hóa vai trò của một thư viện điện tử.

1.1.3 Nghiên cứu sơ bộ về hệ thống hiện hành

Hệ thống thư viện gồm: quản lý sinh viên, quản lý mượn trả sách, quản lý sách.

Các loại sách mà thư viện có: Sách Chính trị – pháp luật, Sách Khoa học công nghệ – Kinh tế, Sách Văn học nghệ thuật, ...

1.2 Xác định các yêu cầu của hệ thống

Để đáp ứng yêu cầu người dùng, hệ thống quản lý sách thư viện bao gồm hai phân hệ chính: phân hệ cho người quản trị để quản trị nội dung và quản trị hoạt động quản lý online, phân hệ cho người dùng để cho người dùng có thể xem thông tin về các cuốn sách. Các yêu cầu chi tiết của hai phân hệ như sau:

- Cho phép ghi sổ các loại nghiệp vụ liên quan đến thư viện như: mượn trả sách.
- Hỗ trợ công cụ tìm kiếm trong các màn hình danh mục.
- Cho phép quản lý nhiều loại sách trên cùng một máy tính.

❖ **Yêu cầu quản trị:**

- Cho phép quản trị viên sẽ quản lý thông tin về sách, loại sách, mượn trả sách. Khi các thông tin này có sai sót hệ thống cho phép quản trị viên có thể sửa hoặc xóa thông tin liên quan.

❖ **Yêu cầu người dùng:**

- Khi người dùng sử dụng ứng dụng, ứng dụng sẽ hiển thị các thông tin sách hiện có trong thư viện.
- Khi người dùng xem thông tin của sách có thể thực hiện chức năng tìm kiếm sách theo thông tin sách cần tìm.

❖ **Các chức năng của quản trị**

Bảng 1.1: Các chức năng của hệ quản trị

STT	Chức năng	Mô tả
1	Nhập sách	Nhập thông tin sách vào web.
2	Phân loại sách	Phân loại các cuốn sách trên web.
3	Mượn trả sách	Quản lý thông tin mượn sách, trả sách của độc giả.
4	Chỉnh sửa thông tin sách	Người quản trị quản lý thông tin sách bao gồm các chức năng: sửa, xóa, tìm kiếm.

1.3 Xây dựng biểu đồ ca sử dụng

1.3.1 Xác định các tác nhân của hệ thống

Bảng 1.2: Xác định các tác nhân của hệ thống

Tác nhân	Mô tả
Độc giả	Người thực hiện các tác vụ tìm kiếm thông tin sách, loại sách
Thủ thư	Người quản lý thông tin sách, loại sách, mượn trả sách

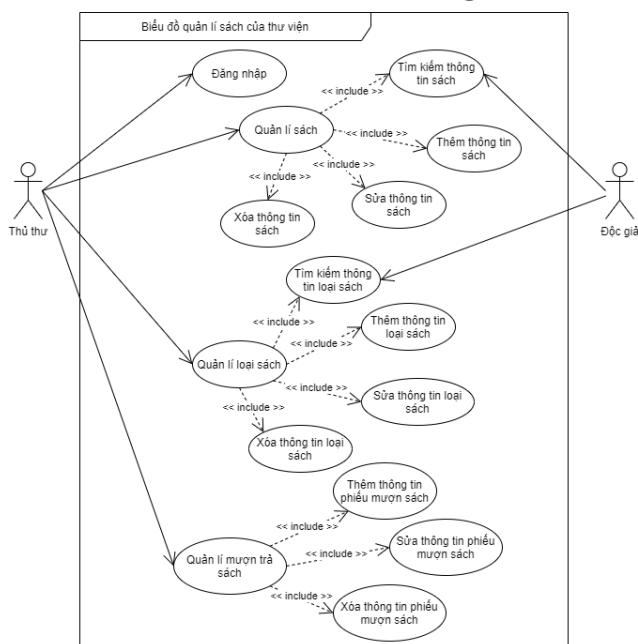
1.3.2 Xác định các ca sử dụng của hệ thống

Bảng 1.3: Xác định các ca sử dụng của hệ thống

Ca sử dụng	Tác nhân	Thực hiện
Tìm kiếm thông tin sách	Độc giả	Tìm kiếm thông tin liên quan về sách
Tìm kiếm thông tin loại sách		Tìm kiếm thông tin liên quan về loại sách
Đăng nhập	Thủ thư	Đăng nhập vào hệ thống để thực hiện các tác vụ quản trị

Quản lí sách		Thêm, sửa, xoá, tìm kiếm thông tin sách
Quản lí loại sách		Thêm, sửa, xoá, tìm kiếm thông tin loại sách
Mượn trả sách		Xem, tìm kiếm, thêm, sửa thông tin mượn trả sách

1.3.3 Đặc tả các ca sử dụng



Hình 1.1: Biểu đồ usecase chính

❖ Mô tả ca sử dụng đăng nhập hệ thống:

- ✓ Mục đích: Đảm bảo xác thực người sử dụng và các yêu cầu về bảo mật của hệ thống.
- ✓ Tác nhân: Thủ thư.
- ✓ Các luồng sự kiện chính:
 - Các tác nhân chọn chức năng đăng nhập, hệ thống hiển thị đăng nhập.
 - Tác nhân nhập thông tin đăng nhập sau đó gửi thông tin đến hệ thống, hệ thống kiểm tra thông tin đăng nhập, nếu đúng thì cho phép truy cập hệ thống, sai thì thông báo lỗi.
- ✓ Luồng sự kiện chính: Nếu dùng tên truy cập và mật khẩu không đúng thì hệ thống sẽ yêu cầu nhập lại nhưng chỉ nhập một số lần xác định.
- ✓ Điều kiện trước: Tên truy cập và mật khẩu.
- ✓ Điều kiện sau: Những thông tin mà người sử dụng yêu cầu.

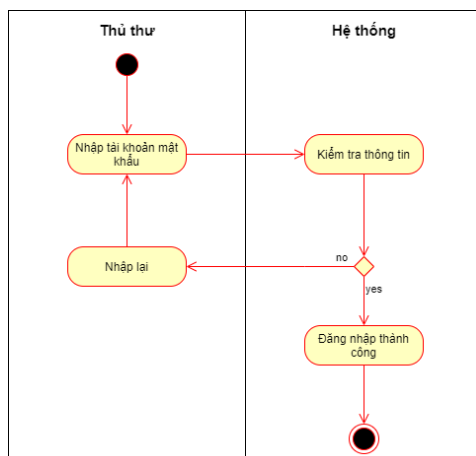
❖ Mô tả ca sử dụng tìm kiếm thông tin sách:

- ✓ Mục đích: Độc giả tìm kiếm, xem thông tin sách.
- ✓ Tác nhân: Độc giả.
- ✓ Luồng sự kiện chính:
 - Tác nhân nhập thông tin sách.
 - Tác nhân gửi thông tin cho hệ thống, sau đó hệ thống cập nhập lại cơ sở dữ liệu, cập nhập kết quả và đưa ra thông báo.
- ✓ Điều kiện trước: Thông tin về sách.
- ✓ Điều kiện sau: Độc giả xem, tìm kiếm thông tin sách.
- ❖ Mô tả ca sử dụng tìm kiếm thông tin loại sách:
 - ✓ Mục đích: Độc giả tìm kiếm, xem thông tin loại sách.
 - ✓ Tác nhân: Độc giả.
 - ✓ Luồng sự kiện chính:
 - Tác nhân nhập thông tin loại sách.
 - Tác nhân gửi thông tin cho hệ thống, sau đó hệ thống cập nhập lại cơ sở dữ liệu, cập nhập kết quả và đưa ra thông báo.
 - ✓ Điều kiện trước: Thông tin về loại sách.
 - ✓ Điều kiện sau: Độc giả xem, tìm kiếm về loại sách.
- ❖ Mô tả ca sử dụng quản lý loại sách:
 - ✓ Mục đích: Dùng để quản lý thông tin chi tiết loại sách.
 - ✓ Tác nhân: Thủ thư.
 - ✓ Các luồng sự kiện chính:
 - Chọn chức năng “Quản lý loại sách”: hệ thống sẽ hiển thị các chức năng con trong mục quản lý loại sách.
 - Hiển thị danh sách sắp xếp loại sách trong thư viện theo thứ tự.
 - Thoát: người thủ thư có thể thoát ra tại bất cứ thời gian nào của ca sử dụng, người thủ thư cũng có thể ghi lại thông tin trước khi thoát.
 - Nếu thông tin đã có thì cho phép sửa đổi thông tin.
 - ✓ Điều kiện trước: Thủ thư phải đăng nhập trước sau đó có thể thực thêm - sửa - xoá.
 - ✓ Điều kiện sau: Đưa ra được danh sách loại hàng.
- ❖ Mô tả ca sử dụng quản lý mượn trả sách:
 - ✓ Mục đích: Dùng để quản lý thông tin mượn trả sách.
 - ✓ Tác nhân: Thủ thư.

- ✓ Các luồng sự kiện chính:
 - Chọn chức năng “Quản lý mượn trả sách”: hệ thống sẽ hiển thị các chức năng con trong mục quản lý mượn trả sách.
 - Hiển thị danh sách mượn trả sách sắp xếp trong thư viện theo thứ tự.
 - ✓ Các luồng sự kiện rẽ nhánh:
 - Thoát: người thủ thư có thể thoát ra tại bất cứ thời gian nào của ca sử dụng, người thủ thư cũng có thể ghi lại thông tin trước khi thoát.
 - Nếu thông tin đã có thì cho phép sửa đổi thông tin.
 - ✓ Điều kiện trước: Thủ thư phải đăng nhập trước sau đó có thể thực thêm - sửa - xoá.
 - ✓ Điều kiện sau: Đưa ra được danh sách mượn trả sách.
- ❖ Mô tả ca sử dụng quản lý sách:
- ✓ Mục đích: giúp thủ thư nắm rõ tình trạng sách có thể thêm - sửa - xoá thông tin sách.
 - ✓ Tác nhân: Thủ thư.
 - ✓ Các luồng sự kiện chính:
 - Cập nhật thông tin sách.
 - Thêm, xóa, sửa thông tin sách.
 - ✓ Các luồng sự kiện rẽ nhánh:
 - Người dùng có thể thoát tại bất cứ thời điểm nào.
 - ✓ Điều kiện trước: Thủ thư phải đăng nhập hệ thống.
 - ✓ Điều kiện sau: Hiển thị thông tin sách theo yêu cầu.

1.4 Biểu đồ hoạt động

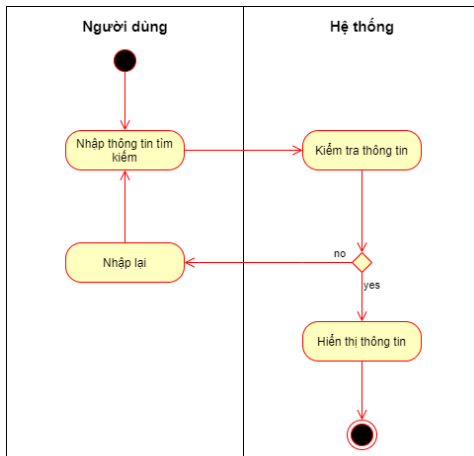
1.4.1 Biểu đồ hoạt động đăng nhập



- Thủ thư nhập tài khoản và mật khẩu để đăng nhập vào hệ thống. Nếu hệ thống kiểm tra thông tin là đúng thì cho phép thủ thư vào hệ thống còn nếu đăng nhập sai hệ thống sẽ yêu cầu nhập lại.

Hình 1.2: Biểu đồ hoạt động đăng nhập hệ thống

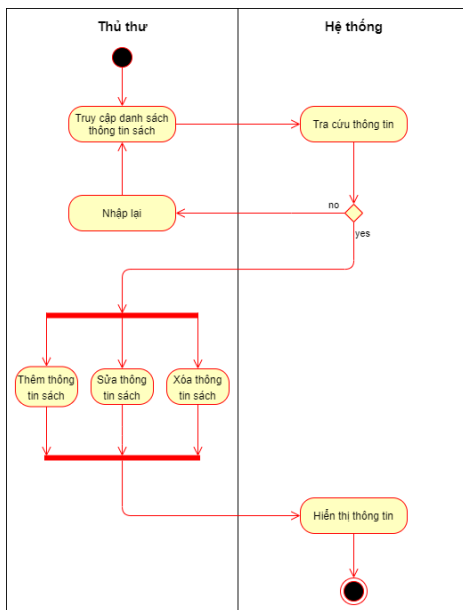
1.4.2 Biểu đồ hoạt động tìm kiếm thông tin



- Người dùng nhập thông tin cần tìm kiếm vào ô tìm kiếm (search). Nếu thông tin tồn tại hệ thống sẽ hiển thị những thông tin liên quan nếu thông tin nhập không tồn tại thì hệ thống sẽ không hiển thị thông tin.

Hình 1.3: Biểu đồ hoạt động tìm kiếm thông tin

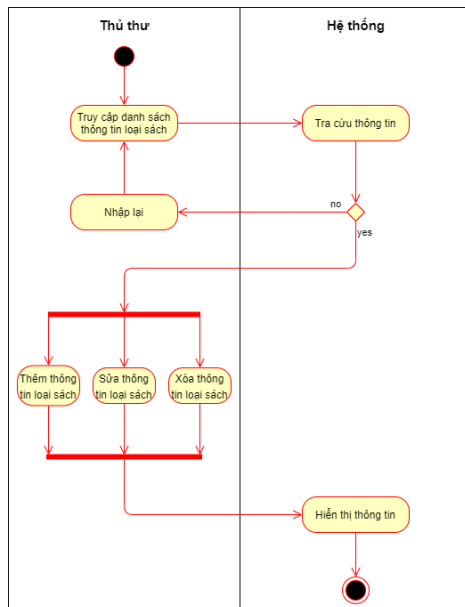
1.4.3 Biểu đồ hoạt động quản lý sách.



- Thủ thư truy cập vào danh sách chức năng thông tin sách. Thủ thư có thể thực hiện các thao tác như tìm kiếm, thêm, sửa, xóa thông tin sách.

Hình 1.4: Biểu đồ hoạt động quản lý sách

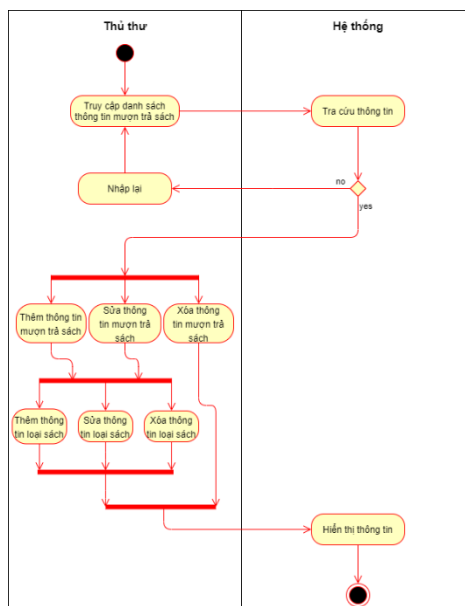
1.4.4 Biểu đồ hoạt động quản lý loại sách.



- Thủ thư truy cập vào danh sách chức năng thông tin loại sách. Thủ thư có thể thực hiện các thao tác như tìm kiếm, thêm, sửa, xóa thông tin loại sách.

Hình 1.5: Biểu đồ hoạt động quản lý loại sách

1.4.5 Biểu đồ hoạt động quản lý mượn trả sách.

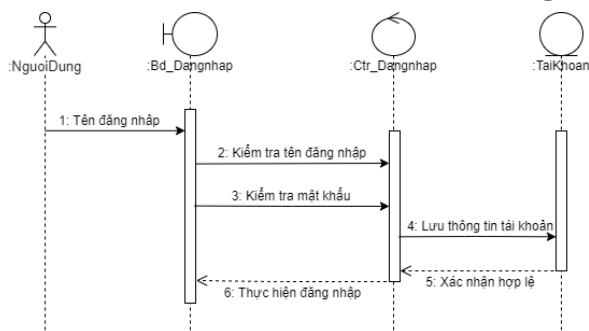


- Thủ thư truy cập vào danh sách chức năng thông tin mượn trả sách. Thủ thư có thể thực hiện các thao tác như tìm kiếm, thêm, sửa, xóa thông tin mượn trả sách cũng như thêm, sửa, xóa thông tin sách trong phiếu mượn.

Hình 1.6: Biểu đồ hoạt động quản lý mượn trả sách

1.5 Biểu đồ trình tự

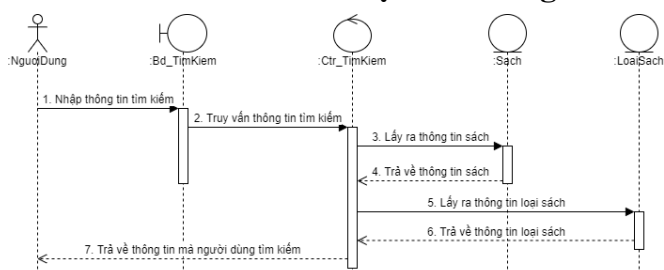
1.5.1 Biểu đồ trình tự chức năng đăng nhập



- Thủ thư nhập tài khoản và mật khẩu để đăng nhập vào hệ thống. Nếu hệ thống kiểm tra thông tin là đúng thì cho phép thủ thư vào hệ thống còn nếu đăng nhập sai hệ thống sẽ yêu cầu nhập lại.

Hình 1.7: Biểu đồ tuần tự chức năng đăng nhập

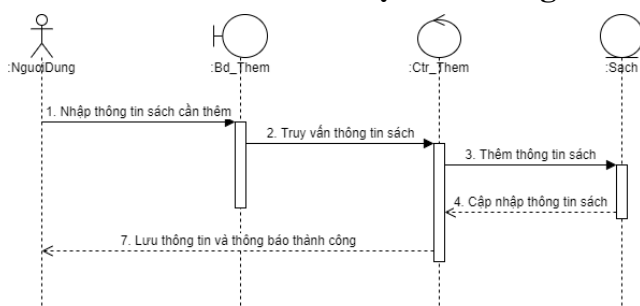
1.5.2 Biểu đồ trình tự chức năng tìm kiếm thông tin sách và loại sách



- Người dùng nhập thông tin cần tìm kiếm vào ô tìm kiếm (search). Nếu thông tin tồn tại hệ thống sẽ hiển thị những thông tin liên quan.

Hình 1.8: Biểu đồ tuần tự chức năng tìm kiếm thông tin sách và loại sách

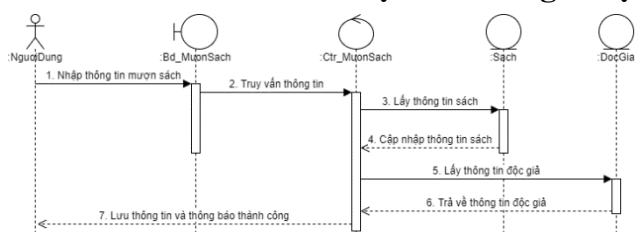
1.5.3 Biểu đồ trình tự chức năng thêm sách



- Người dùng nhập thông tin cần thêm vào biểu mẫu thêm mới sách. Nếu thêm thành công thì cập nhập lên danh sách.

Hình 1.9: Biểu đồ tuần tự chức năng thêm sách

1.5.4 Biểu đồ trình tự chức năng mượn sách

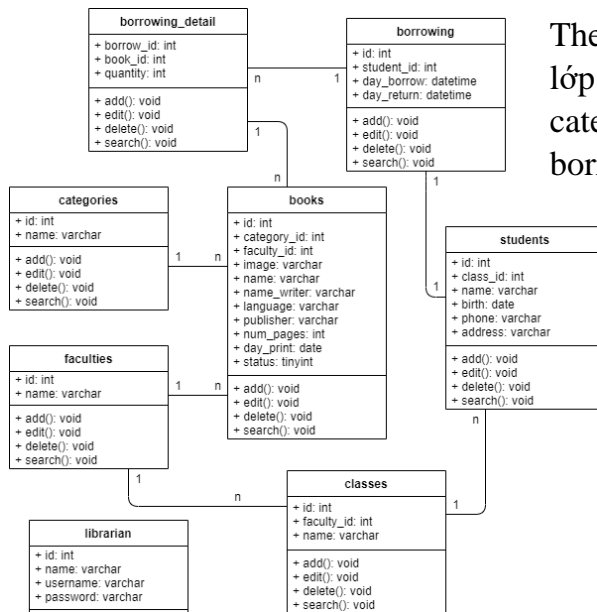


- Thủ thư nhập thông tin cần cho mượn sách vào biểu mẫu thêm thông tin mượn sách. Nếu thêm thành công thì cập nhập lên danh sách.

Hình 1.10: Biểu đồ tuần tự chức năng mượn sách

1.6 Xây dựng biểu đồ lớp lĩnh vực

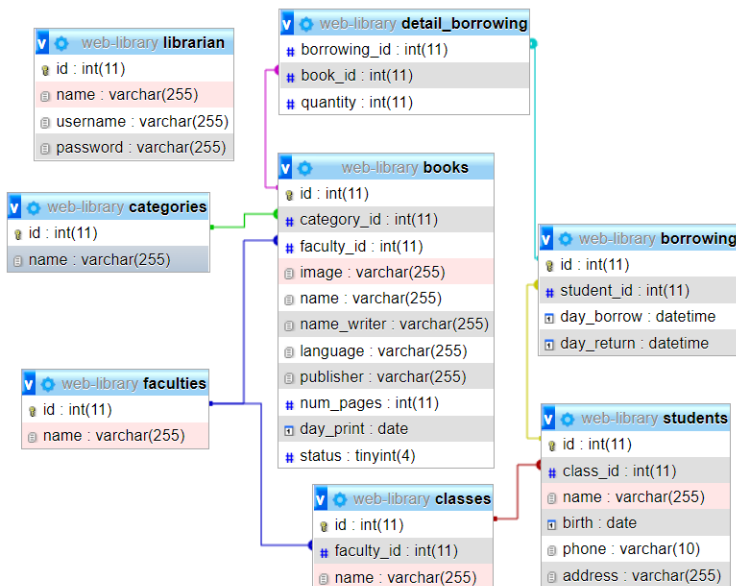
1.6.1 Sơ đồ lớp của bài toán



Theo Hình 1.11 là phân trình bày biểu đồ lớp của website. Biểu đồ lớp gồm: books, categories, faculties, students, borrowing, borrowing_detail, classes.

Hình 1.11: Sơ đồ lớp

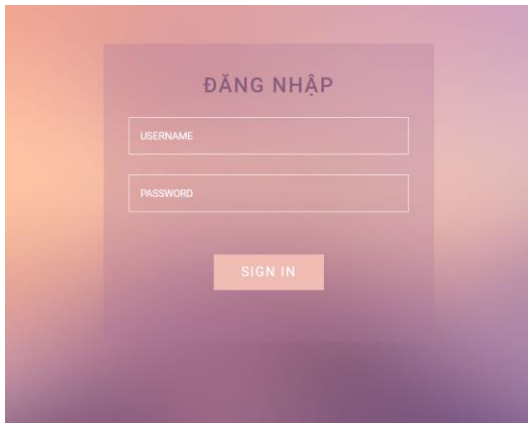
1.6.2 Mô hình dữ liệu quan hệ



Hình 1.12: Mô hình dữ liệu quan hệ

CHƯƠNG 2: THIẾT KẾ GIAO DIỆN

2.1 Giao diện đăng nhập

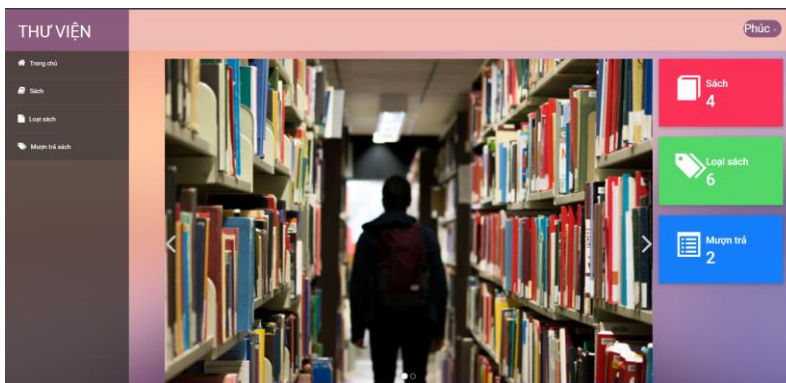


❖ Giao diện đăng nhập

- Giao diện đăng nhập để người dùng đăng nhập vào hệ thống và phân quyền sử dụng.

Hình 2.1: Giao diện đăng nhập

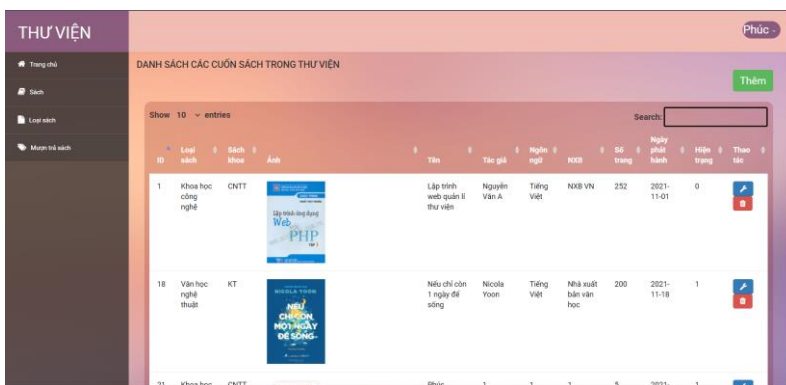
2.2 Giao diện cho thủ thư



❖ Giao diện trang chủ

- Thủ thư: xem thông tin tổng số sách, loại sách, mượn trả sách.

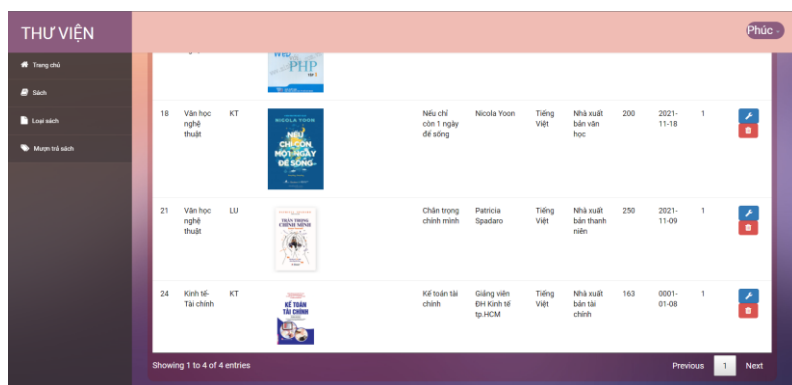
Hình 2.2: Giao diện cho thủ thư mục trang chủ



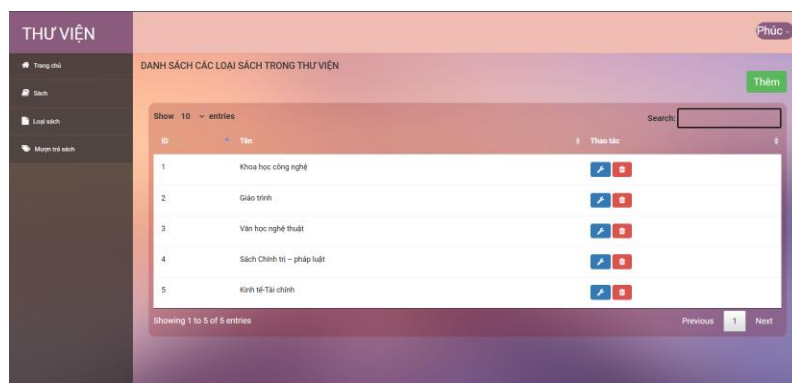
❖ Giao diện quản lí sách

- Thủ thư: xem, tìm kiếm thông tin sách.

Hình 2.3: Giao diện cho thủ thư mục sách phần trên



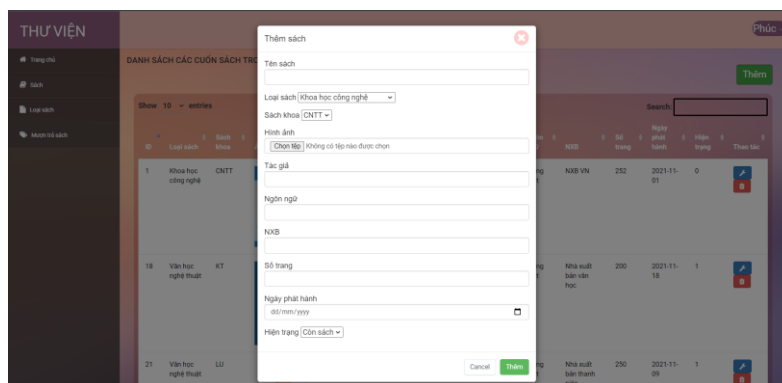
Hình 2.4: Giao diện cho thủ thư mục sách phần dưới



Hình 2.5: Giao diện cho thủ thư mục loại sách

❖ Giao diện quản lí loại sách

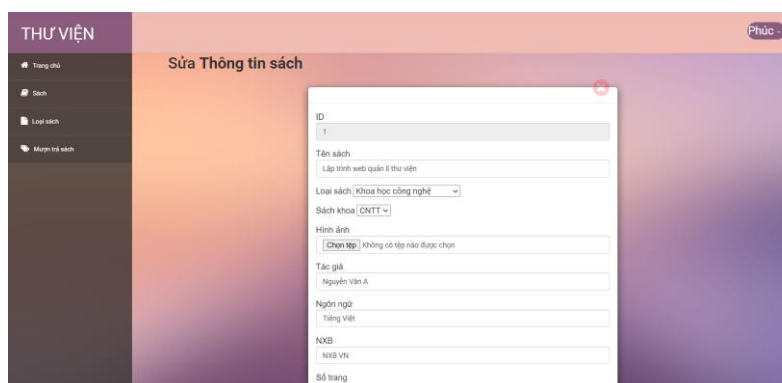
- Thủ thư: xem, tìm kiếm thông tin loại sách.



Hình 2.6: Giao diện cho thủ thư mục thêm sách mới

❖ Giao diện thêm sách mới

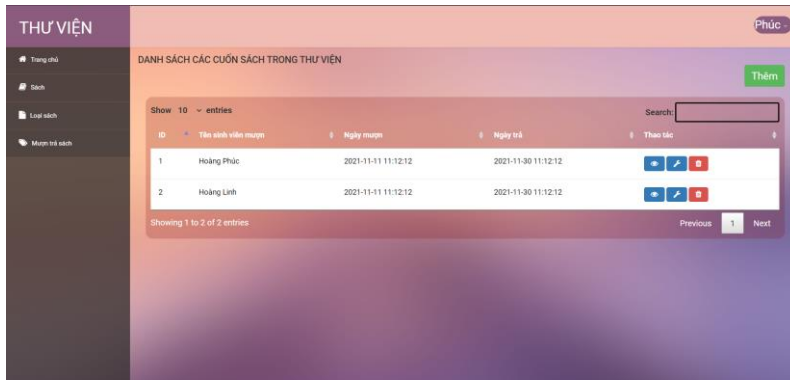
- Thủ thư có thể thêm thông tin sách mới vào thư viện



❖ Giao diện sửa xóa sách

- Thủ thư có thể sửa, xóa thông tin sách.

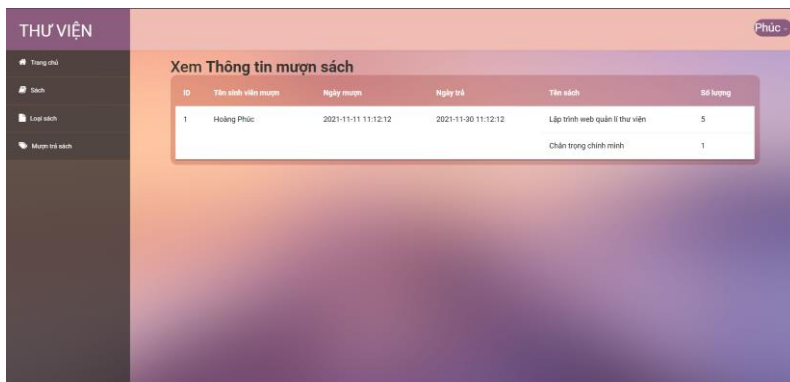
Hình 2.7: Giao diện cho thủ thư mục sửa, xóa sách mới



❖ **Giao diện quản lý mượn trả sách**

- Thủ thư có thể tìm kiếm thông tin mượn trả sách.

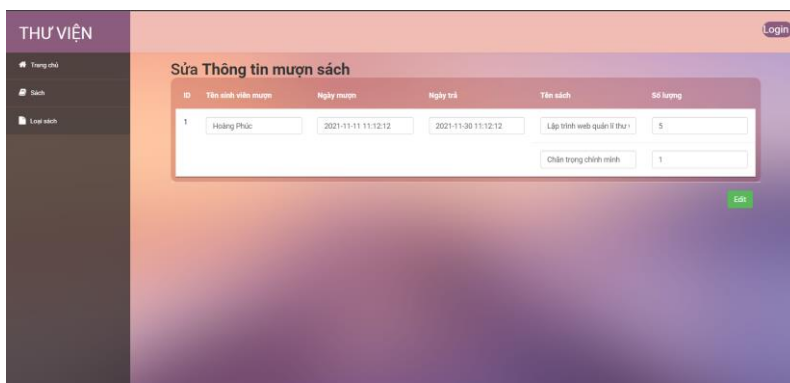
Hình 2.8: Giao diện cho thủ thư mục quản lý phiếu mượn trả sách



❖ **Giao diện xem chi tiết**

- Thủ thư có thể xem thông tin chi tiết mượn trả sách.

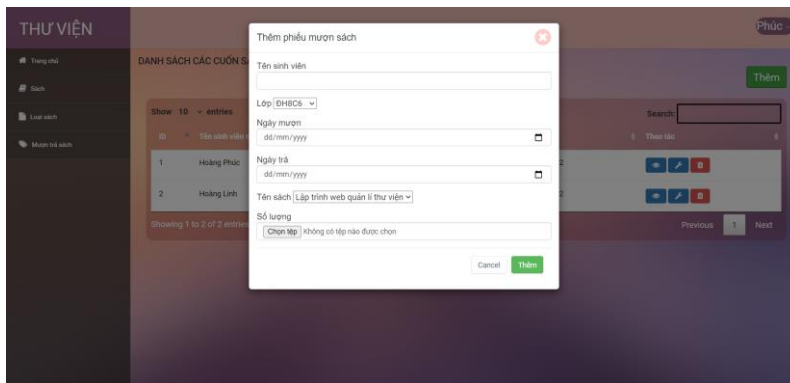
Hình 2.9: Giao diện cho thủ thư xem chi tiết phiếu mượn



❖ **Giao diện sửa thông tin mượn trả sách**

- Thủ thư có thể sửa thông tin chi tiết mượn trả sách.

Hình 2.10: Giao diện cho thủ thư sửa phiếu mượn

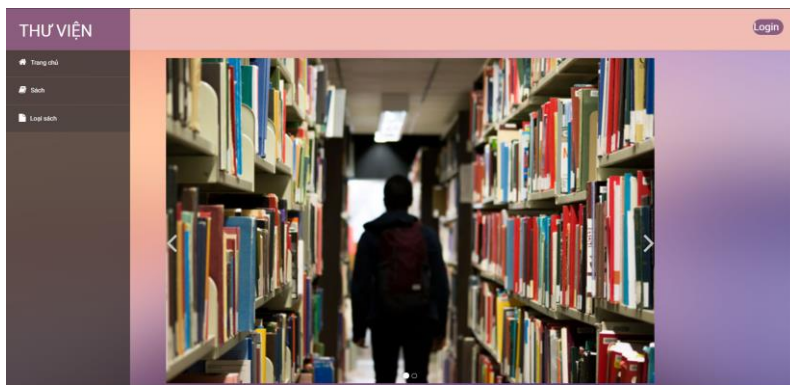


❖ **Giao diện thêm thông tin mượn trả sách**

- Thủ thư có thể thêm thông tin chi tiết mượn trả sách.

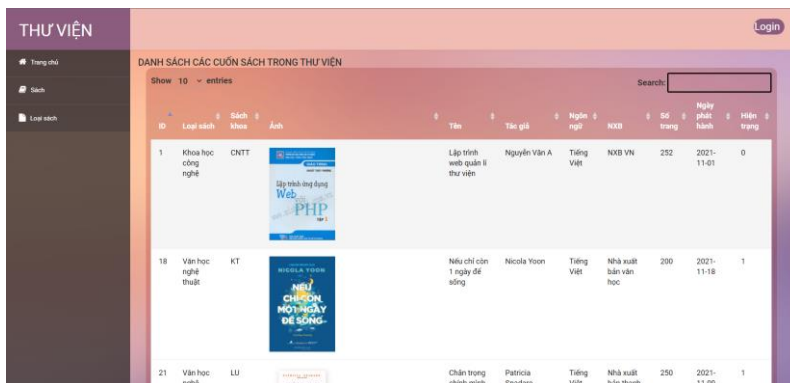
Hình 2.11: Giao diện cho thủ thư thêm phiếu mượn sách

2.3 Giao diện cho độc giả



❖ **Giao diện trang chủ của độc giả**

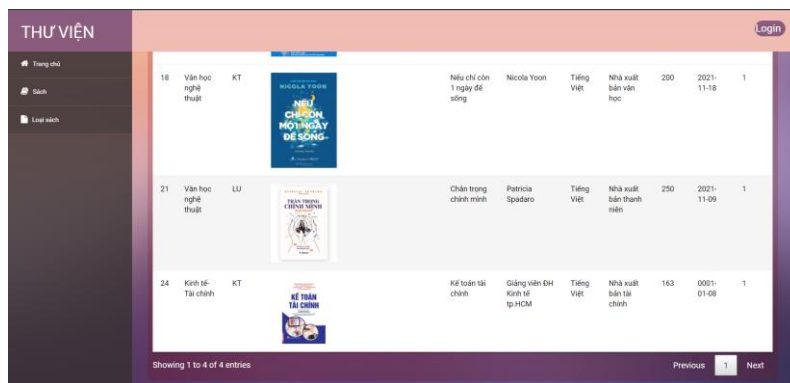
Hình 2.12: Giao diện trang chủ của độc giả



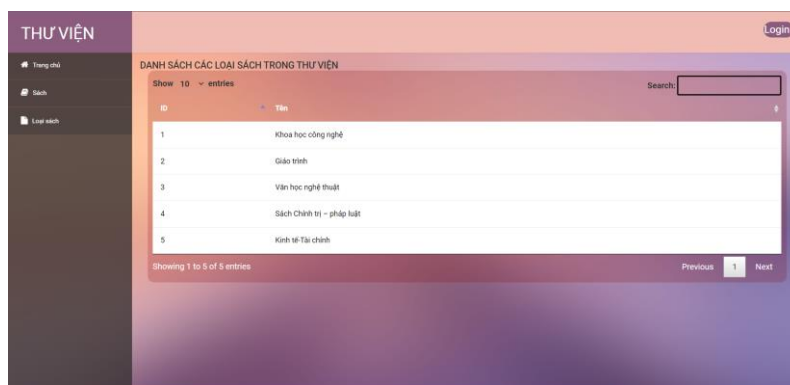
❖ **Giao diện mục sách của độc giả**

- Cho phép độc giả: xem, tìm kiếm thông tin sách.

Hình 2.13: Giao diện cho độc giả tìm kiếm thông tin sách phần trên



Hình 2.14: Giao diện cho độc giả tìm kiếm thông tin sách phần dưới



- ❖ **Giao diện mục loại sách của độc giả**
- Cho phép độc giả: xem, tìm kiếm thông tin loại sách.

Hình 2.15: Giao diện cho độc giả tìm kiếm thông tin loại sách

KẾT LUẬN

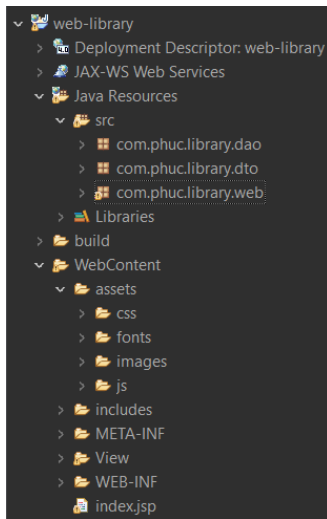
Trong bối cảnh thị trường đầy cạnh tranh như hiện nay thì việc nâng cao chất lượng dịch vụ là vô cùng quan trọng. Việc quản lí sách thư viện hiệu quả sẽ rất cần thiết cho người thủ thư. Vì thế việc có một ứng dụng quản lí sẽ rất hữu ích, vừa tiết kiệm thời gian, vừa ít tốn nhân lực và việc đưa sản phẩm đến người tiêu dùng.

Em hi vọng thông qua bài báo cáo này sẽ góp một phần nhỏ bé vào công việc quản lí sách thư viện. Việc thực hiện đề tài đã giúp chúng em trau dồi lại những kiến thức đã học, đồng thời tìm hiểu nghiên cứu những kiến thức mới để ứng dụng vào việc phát triển một hệ thống thông tin.

Do hiểu biết về lí thuyết cũng như thực tế còn hạn chế, bài báo cáo sẽ không tránh khỏi những thiếu sót. Em kính mong thầy cô giáo nhận xét, góp ý để bài báo cáo của em được hoàn thiện hơn.

PHỤ LỤC

❖ Trang web được xây dựng theo mô hình 3 lớp:



❖ Một số hàm truy vấn cơ sở dữ liệu cho mục quản lí sách (com.phuc.library.dao):

- Hàm truy vấn lấy ra danh sách thông tin sách của thư viện

```
public List<BookDTO> getListBook() {  
    List<BookDTO> list = new ArrayList<>();  
    try {  
        conn = getConnection();  
        sql = "SELECT books.id, books.image, books.name, books.name_writer, "  
            + "books.language, books.publisher, books.num_pages, books.day_print, "  
            + "books.status, categories.name AS 'category_id', faculties.name AS 'faculty_id' "  
            + "FROM books, faculties, categories WHERE books.faculty_id = faculties.id "  
            + "AND books.category_id = categories.id";  
        ps = conn.prepareStatement(sql);  
        rs = ps.executeQuery();  
        while (rs.next()) {  
            BookDTO bookDTO = new BookDTO(rs.getInt("id"), rs.getInt("num_pages"), rs.getInt("status"),  
                rs.getString("image"), rs.getString("name"), rs.getString("name_writer"),  
                rs.getString("language"), rs.getString("publisher"), rs.getString("day_print"),  
                rs.getString("category_id"), rs.getString("faculty_id"));  
            list.add(bookDTO);  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
    return list;  
}
```

- Hàm truy vấn lấy ra thông tin sách theo “id” cho trước

```
public BookDTO getListBookBy(String id) {  
    try {  
        conn = getConnection();  
        sql = "SELECT books.id, books.image, books.name, books.name_writer, books.language, "  
            + "books.publisher, books.num_pages, books.day_print, books.status, "  
            + "categories.name AS 'category_id', faculties.name AS 'faculty_id' "  
            + "FROM books, faculties, categories WHERE books.faculty_id = faculties.id "  
            + "AND books.category_id = categories.id AND books.id = ?";  
        ps = conn.prepareStatement(sql);  
        ps.setString(1, id);  
        rs = ps.executeQuery();  
        while (rs.next()) {  
            return new BookDTO(rs.getInt("id"), rs.getInt("num_pages"), rs.getInt("status"), rs.getString("image"),  
                rs.getString("name"), rs.getString("name_writer"), rs.getString("language"),  
                rs.getString("publisher"), rs.getString("day_print"), rs.getString("category_id"),  
                rs.getString("faculty_id"));  
        }  
    } catch (Exception e) {  
        e.printStackTrace();  
    }  
    return null;  
}
```

- Hàm truy vấn thêm sách mới vào thư viện

```
public void insertBook(int category_id, int faculty_id, String image, String name, String name_writer,
String language, String publisher, int num_pages, String day_print, int status) {
    try {
        conn = getConnection();
        sql = "INSERT INTO books (category_id, faculty_id, image, name, name_writer, language, publisher,
            + "num_pages, day_print, status) VALUES (?, ?, ?, ?, ?, ?, ?, ?)";
        ps = conn.prepareStatement(sql);
        ps.setInt(1, category_id);
        ps.setInt(2, faculty_id);
        ps.setString(3, image);
        ps.setString(4, name);
        ps.setString(5, name_writer);
        ps.setString(6, language);
        ps.setString(7, publisher);
        ps.setInt(8, num_pages);
        ps.setString(9, day_print);
        ps.setInt(10, status);
        ps.executeUpdate();
    } catch (SQLException e) {
        e.printStackTrace();
    }
}
```

- Hàm truy vấn sửa thông tin sách

```
public void updateBook(int category_id, int faculty_id, String name, String name_writer, String language,
String publisher, int num_pages, String day_print, int status, String image, int id) {
    try {
        conn = getConnection();
        sql = "UPDATE books SET category_id = ?, faculty_id = ?, name = ?, name_writer = ?, language = ?, publisher = ?,
            + "num_pages = ?, day_print = ?, status = ?, image = ? WHERE id = ?";
        ps = conn.prepareStatement(sql);
        ps.setInt(1, category_id);
        ps.setInt(2, faculty_id);
        ps.setString(3, name);
        ps.setString(4, name_writer);
        ps.setString(5, language);
        ps.setString(6, publisher);
        ps.setInt(7, num_pages);
        ps.setString(8, day_print);
        ps.setInt(9, status);
        ps.setString(10, image);
        ps.setInt(11, id);
        ps.executeUpdate();
    } catch (SQLException e) {
        e.printStackTrace();
    }
}
```

- Hàm truy vấn xóa thông tin sách

```
public void deleteBook(int id) {
    try {
        conn = getConnection();
        sql = "delete from books where id = ?";
        ps = conn.prepareStatement(sql);
        ps.setInt(1, id);
        ps.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

- ❖ Lớp BookDTO giúp định nghĩa cho các thuộc tính có trong mục quản lý sách (com.phuc.library.dto):

```
package com.phuc.library.dto;

public class BookDTO {
    int id, num_pages, status;
    String image, name, name_writer, language,
    publisher, day_print, category_id, faculty_id;

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public int getNum_pages() {
        return num_pages;
    }

    public void setNum_pages(int num_pages) {
        this.num_pages = num_pages;
    }

    public int getStatus() {
        return status;
    }

    public void setStatus(int status) {
        this.status = status;
    }

    public String getImage() {
        return image;
    }

    public void setImage(String image) {
        this.image = image;
    }

    public String getName() {
        return name;
    }

    public void setName(String name) {
        this.name = name;
    }

    public String getName_writer() {
        return name_writer;
    }

    public void setName_writer(String name_writer) {
        this.name_writer = name_writer;
    }

    public String getLanguage() {
        return language;
    }

    public void setLanguage(String language) {
        this.language = language;
    }

    public String getPublisher() {
        return publisher;
    }

    public void setPublisher(String publisher) {
        this.publisher = publisher;
    }

    public String getDay_print() {
        return day_print;
    }

    public void setDay_print(String day_print) {
        this.day_print = day_print;
    }

    public String getCategory_id() {
        return category_id;
    }

    public void setCategory_id(String category_id) {
        this.category_id = category_id;
    }

    public String getFaculty_id() {
        return faculty_id;
    }

    public void setFaculty_id(String faculty_id) {
        this.faculty_id = faculty_id;
    }
}

public BookDTO(int id, int num_pages, int status, String image, String name, String name_writer, String language,
    String publisher, String day_print, String category_id, String faculty_id) {
    super();
    this.id = id;
    this.num_pages = num_pages;
    this.status = status;
    this.image = image;
    this.name = name;
    this.name_writer = name_writer;
    this.language = language;
    this.publisher = publisher;
    this.day_print = day_print;
    this.category_id = category_id;
    this.faculty_id = faculty_id;
}

@Override
public String toString() {
    return "BookDTO [id=" + id + ", num_pages=" + num_pages + ", status=" + status + ", image=" + image + ", name="
        + name + ", name_writer=" + name_writer + ", language=" + language + ", publisher=" + publisher
        + ", day_print=" + day_print + ", category_id=" + category_id + ", faculty_id=" + faculty_id + "]";
}
```

❖ Code file servlet trong mục quản lí sách (com.phuc.library.web):

- Hàm khai báo mảng chứa thông tin sách, loại sách,.. trong file DAO và chuyển các mảng ra file book.jsp để hiển thị danh sách

```
private void load(HttpServletRequest request, HttpServletResponse response)
    throws SQLException, IOException, ServletException {
    // GET data from DAO
    DAO dao = new DAO();
    List<BookDTO> list = dao.getListBook();
    List<CategoryDTO> list1 = dao.getListCategory();
    List<FacultyDTO> list2 = dao.getListFaculty();
    // SET data to JSP
    request.setAttribute("lists", list);
    request.setAttribute("list1", list1);
    request.setAttribute("list2", list2);
    request.getRequestDispatcher("/View/book.jsp").forward(request, response);
}
```

- Hàm tạo mới thông tin sách lấy dữ liệu nhận được từ trình duyệt khi khách hàng truy cập gửi

```
private void add(HttpServletRequest request, HttpServletResponse response)
    throws SQLException, IOException, ServletException {
    request.setCharacterEncoding("UTF-8");
    response.setCharacterEncoding("UTF-8");

    File file = new File(request.getParameter("image")); // Khai báo file textfile.txt
    String absolutePath = file.getAbsolutePath();
    String filePath = absolutePath.substring(20);

    int category_id = Integer.parseInt(request.getParameter("category_id"));
    int faculty_id = Integer.parseInt(request.getParameter("faculty_id"));
    String image = "assets/images/books/" + filePath;
    String name = request.getParameter("name");
    String name_writer = request.getParameter("name_writer");
    String language = request.getParameter("language");
    String publisher = request.getParameter("publisher");
    int num_pages = Integer.parseInt(request.getParameter("num_pages"));
    String day_print = request.getParameter("day_print");
    int status = Integer.parseInt(request.getParameter("status"));
    DAO dao = new DAO();
    dao.insertBook(category_id, faculty_id, image, name, name_writer, language, publisher, num_pages, day_print,
        status);
    response.sendRedirect("book");
}
```

- Hàm lấy id từ web khi khách hàng bấm xóa rồi thực hiện xóa thông tin chứa id đó theo hàm xóa trong file DAO

```
private void deleteBook(HttpServletRequest request, HttpServletResponse response)
    throws SQLException, IOException, ServletException {
    // GET id from JSP
    int id = Integer.parseInt(request.getParameter("id"));
    // PASS id to DAO
    DAO dao = new DAO();
    dao.deleteBook(id);
    response.sendRedirect("book");
}
```

- Hàm lấy id từ web khi khách hàng bấm sửa rồi thực hiện hiển thị thông tin sản phẩm chứa id đó theo hàm hiển thị mảng trong file DAO

```
private void showupdate(HttpServletRequest request, HttpServletResponse response)
    throws SQLException, IOException, ServletException {
    // GET id from JSP
    request.setCharacterEncoding("UTF-8");
    response.setCharacterEncoding("UTF-8");
    String id = request.getParameter("id");
    // PASS id to DAO
    DAO dao = new DAO();
    BookDTO dto = dao.getListBookBy(id);
    List<CategoryDTO> list1 = dao.getListCategory();
    List<FacultyDTO> list2 = dao.getListFaculty();
    request.setAttribute("detail", dto);
    request.setAttribute("list1", list1);
    request.setAttribute("list2", list2);
    request.getRequestDispatcher("/View/book_op.jsp").forward(request, response);
}
```


- Hàm thực hiện lấy các giá trị mà người dùng cập nhật thông tin sách từ web rồi đưa vào hàm cập nhật trong file DAO

```
private void update(HttpServletRequest request, HttpServletResponse response)
    throws SQLException, IOException, ServletException {
    // GET id from JSP
    request.setCharacterEncoding("UTF-8");
    response.setCharacterEncoding("UTF-8");

    File file = new File(request.getParameter("image")); // Khai báo file textfile.txt
    String absolutePath = file.getAbsolutePath();
    String filePath = absolutePath.substring(20);

    int category_id = Integer.parseInt(request.getParameter("category_id"));
    int faculty_id = Integer.parseInt(request.getParameter("faculty_id"));
    String image = "assets/images/books/" + filePath;
    String name = request.getParameter("name");
    String name_writer = request.getParameter("name_writer");
    String language = request.getParameter("language");
    String publisher = request.getParameter("publisher");
    int num_pages = Integer.parseInt(request.getParameter("num_pages"));
    String day_print = request.getParameter("day_print");
    int status = Integer.parseInt(request.getParameter("status"));
    int id = Integer.parseInt(request.getParameter("id"));
    // PASS id to DAO
    DAO dao = new DAO();
    dao.updateBook(category_id, faculty_id, name, name_writer, language, publisher, num_pages, day_print, status,
        image, id);
    response.sendRedirect("book");
}
```

❖ Code đăng nhập đăng xuất trong servlet:

- Code đăng nhập

```
protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    // TODO Auto-generated method stub
    String username = request.getParameter("username");
    String password = request.getParameter("password");
    DAO dao = new DAO();
    LibrarianDTO librarianDTO = dao.login(username, password);
    if (librarianDTO == null) {
        request.setAttribute("message", "Sai tài khoản mật khẩu!");
        request.getRequestDispatcher("/View/login.jsp").forward(request, response);
    } else {
        HttpSession httpSession = request.getSession();
        httpSession.setAttribute("librarianDTO", librarianDTO);
        response.sendRedirect("index");
    }
}
```

- Code đăng xuất

```
protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
    // TODO Auto-generated method stub
    HttpSession httpSession = request.getSession();
    httpSession.removeAttribute("librarianDTO");
    response.sendRedirect("index");
}
```

❖ Code đăng nhập đăng xuất trong jsp:

- Code đăng nhập

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<!DOCTYPE html>
<jsp:include page="/includes/head.jsp" />
<body>
    <div class="log-w3">
        <div class="w3layouts-main">
            <h2>Đăng nhập</h2>
            <form action="login" method="post">
                <input type="text" class="ggg" name="username"
                    placeholder="USERNAME" required=""> <input type="password"
                    class="ggg" name="password" placeholder="PASSWORD" required="">
                <div class="clearfix"></div>
                <c:if test="${message != null}">
                    <div class="alert alert-danger" role="alert">${message}</div>
                </c:if>
                <input type="submit" value="Sign In" name="login">
            </form>
        </div>
    </div>
    <jsp:include page="/includes/script.jsp" />
</body>
</html>
```

- Code nút đăng nhập - đăng xuất - hiển thị tên

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core"%>
<header class="header fixed-top clearfix">
    <!-- logo start -->
    <div class="brand">
        <a href="index" class="Logo"> Thư viện </a>
    </div>
    <!-- user info end -->
    <div class="top-nav clearfix">
        <!-- user info start -->
        <ul class="nav pull-right top-menu">
            <!-- user login dropdown start -->
            <li class="dropdown" data-toggle="dropdown">
                <a href="#" style="font-size: 22px; class="username">${sessionScope.librarianDTO.name}</span> <b>
                    class="caret"</b>
                </a>
                <ul class="dropdown-menu extended Logout">
                    <li><a href="Logout"><i class="fa fa-key"></i> Log Out</a></li>
                </ul></li>
            <!-- user login dropdown end -->
            <!-- user info end -->
        </ul>
    </div>
</header>
```

❖ Code tìm kiếm cho tất cả các bảng

- Thêm dòng css

```
<link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.11.3/css/jquery.dataTables.css">
```

- Thêm dòng js

```
<script type="text/javascript" charset="utf8"
    src="https://cdn.datatables.net/1.11.3/js/jquery.dataTables.js"></script>
<script src="assets/js/dataTables.js"></script>
```

- Trong file dataTable.js

```
$(document).ready(function() {
    $("#listTable").DataTable();
});
```

Với #listTable là id của table

❖ Code trang quản lí sách trong jsp:

- Form hiển thị sách

```
<div class="card-body">
  <table id="listTable"
    class="table table-hover">
    <thead>
      <tr>
        <th scope="col">ID</th>
        <th scope="col">Loại sách</th>
        <th scope="col">Sách khoa</th>
        <th scope="col">Ảnh</th>
        <th scope="col">Tên</th>
        <th scope="col">Tác giả</th>
        <th scope="col">Ngôn ngữ</th>
        <th scope="col">NXB</th>
        <th scope="col">Số trang</th>
        <th scope="col">Ngày phát hành</th>
        <th scope="col">Hiện trạng</th>
        <c:if test="${sessionScope.librarianDTO != null}">
          <th scope="col">Thao tác</th>
        </c:if>
      </tr>
    </thead>
    <tbody>
      <c:forEach items="${lists}" var="x">
        <tr>
          <td>${x.id}</td>
          <td>${x.category_id}</td>
          <td>${x.faculty_id}</td>
          <td></td>
          <td>${x.name}</td>
          <td>${x.name_writer}</td>
          <td>${x.language}</td>
          <td>${x.publisher}</td>
          <td>${x.num_pages}</td>
          <td>${x.day_print}</td>
          <td>${x.status}</td>
          <c:if test="${sessionScope.librarianDTO != null}">
            <td><a href="edit?id=${x.id}"
              class="btn btn-primary fa fa-wrench"></a> <a
              href="delete?id=${x.id}"
              onclick="return confirm('Bạn có thực sự muốn xóa?');">
              class="btn btn-danger fa fa-trash"> </a></td>
          </c:if>
        </tr>
      </c:forEach>
    </tbody>
  </table>
</div>
```

- Form thêm sách mới

```

<form action="add" method="post">
  <div class="modal-header">
    <h4 class="modal-title">Thêm sách</h4>
    <button type="button" class="close fa fa-times-circle"
      data-dismiss="modal" aria-hidden="true"></button>
  </div>
  <div class="modal-body">
    <div class="form-group">
      <label>Tên sách</label> <input name="name" type="text"
        class="form-control" required>
    </div>
    <div class="form-group">
      <label>Loại sách</label> <select name="category_id"
        class="form-select" aria-label="Default select example">
        <c:forEach items="${listS1}" var="x">
          <option value="${x.id}">${x.name}</option>
        </c:forEach>
      </select>
    </div>
    <div class="form-group">
      <label>Sách khoa</label> <select name="faculty_id"
        class="form-select" aria-label="Default select example">
        <c:forEach items="${listS2}" var="x">
          <option value="${x.id}">${x.name}</option>
        </c:forEach>
      </select>
    </div>
    <div class="form-group">
      <label>Hình ảnh</label> <input name="image" type="file"
        class="form-control" required accept="image/*">
    </div>
    <div class="form-group">
      <label>Tác giả</label> <input name="name_writer" type="text"
        class="form-control" required>
    </div>
    <div class="form-group">
      <label>Ngôn ngữ</label> <input name="Language" type="text"
        class="form-control" required>
    </div>
    <div class="form-group">
      <label>NXB</label> <input name="publisher" type="text"
        class="form-control" required>
    </div>
    <div class="form-group">
      <label>Số trang</label> <input name="num_pages" type="number"
        class="form-control" required>
    </div>
    <div class="form-group">
      <label>Ngày phát hành</label> <input name="day_print" type="date"
        class="form-control" required>
    </div>
    <div class="form-group">
      <label>Hiện trạng</label> <select name="status"
        class="form-select" aria-label="Default select example">
        <option value="1">Còn sách</option>
        <option value="2">Hết sách</option>
      </select>
    </div>
  </div>
  <div class="modal-footer">
    <input type="button" class="btn btn-default" data-dismiss="modal"
      value="Cancel"> <input type="submit"
      class="btn btn-success" value="Thêm">
  </div>
</form>

```

- Form sửa thông tin sách

```
<form action="edit" method="post">
  <div class="modal-header">
    <h4 class="modal-title"></h4>
    <a href="book" type="button" class="close fa fa-times-circle"
      data-dismiss="modal" aria-hidden="true"></a>
  </div>
  <div class="modal-body">
    <div class="form-group">
      <label>ID</label> <input value="{detail.id}" name="id"
        type="text" class="form-control" readonly required>
    </div>
    <div class="form-group">
      <label>Tên sách</label> <input value="{detail.name}"
        name="name" type="text" class="form-control" required>
    </div>
    <div class="form-group">
      <label>Loại sách</label> <select name="category_id"
        class="form-select" aria-label="Default select example">
        <:forEach items="{listS1}" var="o">
          <option value="{o.id}" {o.name == detail.category_id ? 'selected="selected"' : ''}>{o.name}</option>
        </c:forEach>
      </select>
    </div>
    <div class="form-group">
      <label>Sách khoa</label> <select name="faculty_id"
        class="form-select" aria-label="Default select example">
        <:forEach items="{listS2}" var="o">
          <option value="{o.id}" {o.name == detail.faculty_id ? 'selected="selected"' : ''}>{o.name}</option>
        </c:forEach>
      </select>
    </div>
    <div class="form-group">
      <label>Hình ảnh</label> <input value="{detail.image}"
        name="image" type="file" class="form-control" required accept="image/*">
    </div>
    <div class="form-group">
      <label>Tác giả</label> <input value="{detail.name_writer}"
        name="name_writer" type="text" class="form-control" required>
    </div>
    <div class="form-group">
      <label>Ngôn ngữ</label> <input value="{detail.language}"
        name="language" type="text" class="form-control" required>
    </div>
    <div class="form-group">
      <label>NXB</label> <input value="{detail.publisher}"
        name="publisher" type="text" class="form-control" required>
    </div>
    <div class="form-group">
      <label>Số trang</label> <input value="{detail.num_pages}"
        name="num_pages" type="number" class="form-control" required>
    </div>
    <div class="form-group">
      <label>Ngày phát hành</label> <input
        value="{detail.day_print}" name="day_print" type="date"
        class="form-control" required>
    </div>
    <div class="form-group">
      <label>Hiện trạng</label> <select name="status"
        class="form-select" aria-label="Default select example">
        <:if test="{detail.status == 1}">
          <option value="1">Còn sách</option>
          <option value="0">Hết sách</option>
        </c:if>
        <:if test="{detail.status == 0}">
          <option value="0">Hết sách</option>
          <option value="1">Còn sách</option>
        </c:if>
      </select>
    </div>
  </div>
  <div class="modal-footer">
    <input type="submit" class="btn btn-success" value="Edit">
  </div>
</form>
```

TÀI LIỆU THAM KHẢO

- [1] Bryan Basham, Kathy Sierra, and Bert Bates (2008), Head First Servlets and JSP Second Edition, O'Reilly Media, United States of America.
- [2] Nguyễn Văn Vy (2007), Phân tích và thiết kế hệ thống thông tin quản lý, NXB Khoa học tự nhiên và công nghệ, Hà Nội.

TÀI LIỆU CÓ NGUỒN TỪ INTERNET

- [1] JSP Tutorial, javatpoint, <https://www.javatpoint.com/jsp-tutorial>.
- [2] Servlets | Servlet Tutorial, javatpoint, <https://www.javatpoint.com/servlet-tutorial>.