

**BỘ KHOA HỌC VÀ CÔNG NGHỆ
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



BÁO CÁO ĐỒ ÁN MÔN HỌC

ĐỀ TÀI: QUẢN LÝ THƯ VIỆN

Môn học: LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG & CƠ SỞ DỮ LIỆU

Giảng viên hướng dẫn: NGUYỄN THỊ TUYẾT HẢI

Thực hiện bởi nhóm sinh viên, bao gồm:

- | | | |
|---------------------|------------|---------------|
| 1. LÊ NGỌC UYÊN | N23DCCN135 | <Trưởng nhóm> |
| 2. TRẦN QUANG KHOAN | N23DCCN099 | <Thành viên> |
| 3. PHAN THIỆN VỸ | N23DCCN137 | <Thành viên> |

TP.HCM, tháng 11/2025

MỤC LỤC

MỤC LỤC	2
DANH SÁCH HÌNH, BẢNG.....	4
TÓM TẮT.....	5
CHƯƠNG I. TỔNG QUAN.....	6
1. Giới thiệu đề tài.	6
2. Cơ sở lý thuyết.....	6
CHƯƠNG II. THIẾT KẾ HỆ THỐNG.....	8
1. Phân tích hệ thống	8
1.1. Mục tiêu hệ thống.....	8
1.2. Phạm vi	8
1.3. Tác nhân	8
1.4. Yêu cầu chức năng	8
1.4.1. Sách	8
1.4.2. Quản lý độc giả.....	8
1.4.3. Quản lý nhân viên.....	8
1.4.4. Quản lý tài khoản.....	8
1.4.5. Mượn trả	9
1.4.6. Danh mục phụ.....	9
2. Sơ đồ phân cấp chức năng	9
3. Yêu cầu chức năng	9
4. Thiết kế cơ sở dữ liệu	10
4.1. Thiết kế cơ sở dữ liệu mức khái niệm	10
4.2 Thiết kế cơ sở dữ liệu mức logic	10
4.3 thiết kế cơ sở dữ liệu mức vật lý.....	13
CHƯƠNG III. TRIỂN KHAI HỆ THỐNG.....	20
1. Giới thiệu các thư viện sử dụng.....	20
1.1.JPA Validation (Sử dụng Hibernate Validator Implementation)	20
1.2.Jakarta Mail	20
1.3. Spring AOP.....	20

1.4. MySQL	21
1.5. FlatLaf UI Manager	21
1.6. Một số thư viện khác	21
2. Phân Tích Chức Năng Theo Các Tầng Kiến Trúc	22
2.1. Data Access Layer (Tầng Truy Cập Dữ Liệu)	22
2.2. Business Logic Layer (Tầng Nghiệp Vụ).....	23
2.3. Presentation Layer (Tầng Trình Bày/Giao Diện)	23
3. Giao diện màn hình.....	25
3.1. Trang đăng nhập	25
3.2. Trang chủ.....	26
3.3. Sách	26
3.4. Độc giả.....	27
3.5. Danh mục.....	27
3.6. Tác giả	28
3.7. Nhà xuất bản.....	28
3.8. Kệ sách	29
3.9. Phiếu mượn.....	29
3.10. Nhân viên.....	30
3.11. Thùng rác	30
4. Xử lý truy vấn dữ liệu tìm kiếm	31
4.1. Tìm kiếm sách với nhiều điều kiện	31
4.2. Tìm kiếm độc giả, tác giả	31
4.3. Tìm kiếm nhân viên với tài khoản.....	32
4.4. Tìm kiếm phiếu mượn	32
4.5. Tìm sách đang mượn	32
4.6. Xóa tài khoản và chi tiết sách.....	32
CHƯƠNG IV. KẾT LUẬN.....	33
TÀI LIỆU THAM KHẢO	34

DANH SÁCH HÌNH

HÌNH 1. SƠ ĐỒ PHÂN CẤP CHỨC NĂNG.....	9
HÌNH 2. EER DIAGRAM.....	10
HÌNH 3. LƯỚI ĐỒ CƠ SỞ DỮ LIỆU MỨC LOGIC	12
HÌNH 4. DIAGRAM CƠ SỞ DỮ LIỆU MỨC VẬT LÍ.....	13
HÌNH 5. VIEW ĐĂNG NHẬP.....	25
HÌNH 6. VIEW TRANG CHỦ.....	26
HÌNH 7. VIEW SÁCH.....	26
HÌNH 8. VIEW ĐỘC GIẢ.....	27
HÌNH 9. VIEW DANH MỤC.....	27
HÌNH 10. VIEW TÁC GIẢ.....	28
HÌNH 11. VIEW NHÀ XUẤT BẢN.....	28
HÌNH 12. VIEW KỆ SÁCH	29
HÌNH 13. VIEW PHIẾU MƯỢN.....	29
HÌNH 14. VIEW NHÂN VIÊN.....	30
HÌNH 15. VIEW THÙNG RÁC.....	30
HÌNH 16. SQL TÌM KIẾM SÁCH	31
HÌNH 17. SQL TÌM KIẾM ĐỘC GIẢ, TÁC GIẢ	31
HÌNH 18. SQL TÌM KIẾM NHÂN VIÊN.....	32
HÌNH 19. SQL TÌM KIẾM PHIẾU MƯỢN	32
HÌNH 20. SQL TÌM SÁCH ĐANG MƯỢN.....	32
HÌNH 21. XÓA TÀI KHOẢN VÀ XÓA CHI TIẾT SÁCH.....	32

DANH SÁCH BẢNG

BẢNG 1. CƠ SỞ DỮ LIỆU MỨC LOGIC.....	11
BẢNG 2. ACCOUNT.....	14
BẢNG 3. AUTHOR.....	14
BẢNG 4. BOOK.....	15
BẢNG 5. BOOK_AUTHOR	15
BẢNG 6. BOOK_BORROW_CARD	15
BẢNG 7. BORROW_CARD	16
BẢNG 8. CATEGORY.....	16
BẢNG 9. CATEGORY_BOOK.....	16
BẢNG 10. PUBLISHER.....	17
BẢNG 11. READER.....	17
BẢNG 12. ROLE.....	18
BẢNG 13. SHELF	18
BẢNG 14. STAFF	19
BẢNG 15. THƯ VIỆN JPA VALIDATION.....	20
BẢNG 16. THƯ VIỆN JAKARTA MALL	20
BẢNG 17. THƯ VIỆN SPRING AOP	21
BẢNG 18. MYSQL.....	21
BẢNG 19. THƯ VIỆN FLATLAF UI MANAGER.....	21
BẢNG 20. DESIGN PATTERN.....	22

TÓM TẮT

Đồ án **Phát triển Hệ thống Quản lý Thư viện (Library Management)** nhằm mục đích xây dựng một hệ thống phần mềm hoàn chỉnh, hỗ trợ số hóa và tự động hóa các nghiệp vụ cốt lõi của một thư viện hiện đại. Hệ thống được phát triển bằng ngôn ngữ lập trình **Java 17**, sử dụng công cụ quản lý dependency **Maven**, và lưu trữ dữ liệu trên cơ sở dữ liệu **MySQL**.

Hệ thống cung cấp một cơ sở dữ liệu có cấu trúc, cho phép quản lý chi tiết các thông tin về sách, tác giả, nhà xuất bản, thể loại và vị trí kệ sách. Các chức năng nghiệp vụ chính bao gồm quản lý thông tin độc giả, nhân viên, quản lý tài khoản người dùng với cơ chế phân quyền rõ ràng, cùng với việc xử lý các giao dịch mượn/trả sách thông qua BorrowCard. Hệ thống có khả năng theo dõi trạng thái mượn sách (Đang mượn, Đã trả, Quá hạn) và áp dụng các hình thức phạt. Các công cụ hỗ trợ như thư viện jbcrypt cho mã hóa mật khẩu và cơ chế Transactional đã được tích hợp để đảm bảo tính bảo mật và toàn vẹn dữ liệu cho các thao tác phức tạp (ví dụ: tạo/xóa tài khoản, xóa vĩnh viễn sách).

CHƯƠNG I. TỔNG QUAN

1. Giới thiệu đề tài.

Cùng với sự phát triển của nền kinh tế thì giáo dục cũng được mở rộng về quy mô và chất lượng để có thể cung cấp nhân lực có trình độ, chuyên môn, cơ sở vật chất ... phục vụ cho nền kinh tế ngày càng vững mạnh. Bộ phận thư viện là bộ phận không thể thiếu trong các trường đại học cũng như trong các trường phổ thông, trung học ... để phục vụ cho việc nghiên cứu, tìm kiếm các tài liệu học tập một cách tốt nhất cho các học sinh, sinh viên. Chính vì thế nhu cầu tin học hóa các chương trình quản lý trong thư viện cũng đang rất cần thiết để có thể đổi mới với quy trình quản lý, đạt hiệu quả cao.

Ứng dụng tin học trong công tác quản lý đã phát triển mạnh mẽ giúp cho công tác quản lý ngày càng trở nên hiệu quả hơn như nâng cao hiệu suất trong công việc, đưa ra các báo cáo, các số liệu thống kê một cách nhanh chóng chính xác và kịp thời. Đồng thời nhờ có việc ứng dụng tin học đã tiết kiệm được rất nhiều thời gian, công sức của con người, nó làm giảm nhẹ bộ máy quản lý vốn rất cồng kềnh từ trước tới nay. Trong quá trình học tập chúng em thấy hệ thống thư viện ngày càng gần gũi với học sinh sinh viên trong trường từ việc mượn trả sách đến việc đọc sách trong thư viện cũng là cách học hết sức bổ ích. Chính vì lý do đó, em đã chọn đề tài “quản lý thư viện” để cho môn học của mình.

Hệ thống quản lý thư viện ra đời nhằm giải quyết các thách thức trong việc quản lý số lượng lớn sách, hồ sơ độc giả, và các giao dịch mượn/trả thủ công. Đề tài tập trung xây dựng một ứng dụng quản lý toàn diện, từ nhập/xuất sách, theo dõi tồn kho, đến quản lý quy trình mượn trả và xử lý các trường hợp quá hạn.

Mục tiêu chính:

- Xây dựng mô hình cơ sở dữ liệu tối ưu cho nghiệp vụ thư viện.
- Phát triển các Service Layer xử lý logic nghiệp vụ, đảm bảo tính toàn vẹn và hiệu suất.
- Cung cấp các chức năng quản lý người dùng (Độc giả, Nhân viên) và sách.
- Triển khai các cơ chế bảo mật cơ bản (mã hóa mật khẩu) và quản lý giao dịch (transactional).

2. Cơ sở lý thuyết.

Ngôn ngữ lập trình: Java 17.

Cơ sở dữ liệu: MySQL (Sử dụng mysql-connector-j).

Kiến trúc: Sử dụng kiến trúc đa tầng (Multi-tier Architecture) với:

- **Data Access Layer (DAO):** Chịu trách nhiệm giao tiếp trực tiếp với cơ sở dữ liệu.
- **Service Layer (Business Logic):** Chứa logic nghiệp vụ cốt lõi (ví dụ: BookService, ReaderService, v.v.).
- **Data Transfer Objects (DTO):** Dùng để truyền dữ liệu giữa các tầng.

Thư viện chính: jpa validation-hibernate implement, jarkata mail, spring aop, mysql, flatlaf ui manager

Design pattern: proxy, singleton, factory method, dao pattern, dto pattern

Bảo mật: Sử dụng thư viện jbcrypt để mã hóa mật khẩu một chiều cho tài khoản người dùng.

Quản lý giao dịch: Áp dụng annotation @Transactional để đảm bảo các thao tác đa bước (ví dụ: insertBook, deletePermanently, CreateStaffAccount) được thực hiện một cách nguyên tử (Atomic).

CHƯƠNG II. THIẾT KẾ HỆ THỐNG

1. Phân tích hệ thống

1.1. Mục tiêu hệ thống

Quản lý vòng đời sách và hoạt động mượn trả. Kiểm soát tác giả, thể loại, nhà xuất bản, kệ sách, độc giả, nhân sự, tài khoản, phân quyền. Giám thao tác thủ công, loại bỏ sai lệch dữ liệu, theo dõi trạng thái sách và lịch sử giao dịch.

1.2. Phạm vi

Quản lý sách. Quản lý độc giả. Quản lý nhân viên. Quản lý tài khoản và phân quyền. Quản lý mượn trả. Quản lý danh mục (kệ, thể loại, tác giả, NXB). Không bao gồm thanh toán, thống kê nâng cao, tích hợp hệ thống ngoài.

1.3. Tác nhân

- Độc giả: đăng ký, xem sách, mượn, trả.
- Nhân viên: xử lý mượn trả, cập nhật sách, cập nhật danh mục, xử lý vi phạm.
- Admin: quản lý toàn bộ dữ liệu, tài khoản, phân quyền.

1.4. Yêu cầu chức năng

1.4.1. Sách

- Thêm, sửa, xóa mềm.
- Gán tác giả, NXB, thể loại, kệ sách.
- Quản lý số lượng, trạng thái.

1.4.2. Quản lý độc giả

- Lưu thông tin cá nhân, liên hệ, ngày sinh.
- Theo dõi lịch sử mượn trả.

1.4.3. Quản lý nhân viên

- Lưu thông tin cá nhân, lương, tài khoản liên kết.

1.4.4. Quản lý tài khoản

- Đăng nhập theo username/password.
- Gán role.

- Phân quyền theo role.

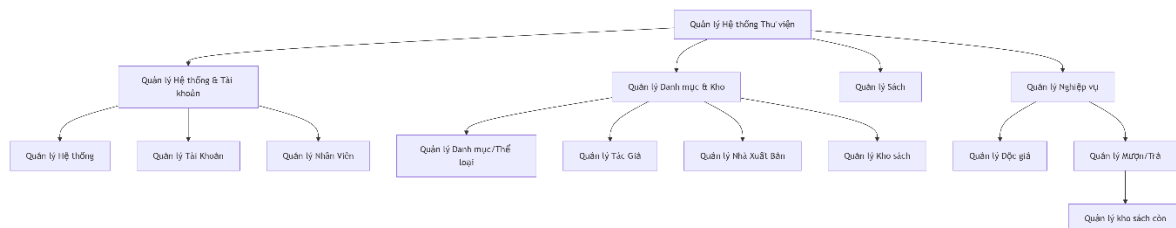
1.4.5. Mượn trả

- Tạo phiếu mượn.
- Ghi thời gian mượn, hạn trả, trạng thái.
- Ghi phạt nếu quá hạn.
- Ghi chi tiết từng cuốn trong phiếu mượn.

1.4.6. Danh mục phụ

- Thể loại.
- Tác giả.
- Kệ.
- Nhà xuất bản.

2. Sơ đồ phân cấp chức năng



Hình 1. Sơ đồ phân cấp chức năng

3. Yêu cầu chức năng

- Chức năng lưu trữ:

+ Thông tin thành viên: họ tên, mật khẩu, loại độc giả (người lớn/trẻ em), ngày sinh, địa chỉ, email, ngày lập thẻ.

+ Thông tin sách: tên sách, thể loại, tác giả, năm xuất bản, nhà xuất bản, ngày nhập sách, trạng thái (có thể/không thể cho mượn).

+ Thông tin giao dịch, mượn sách: họ tên thành viên mượn sách, ngày mượn sách, sách mượn.

- Chức năng truy vấn:

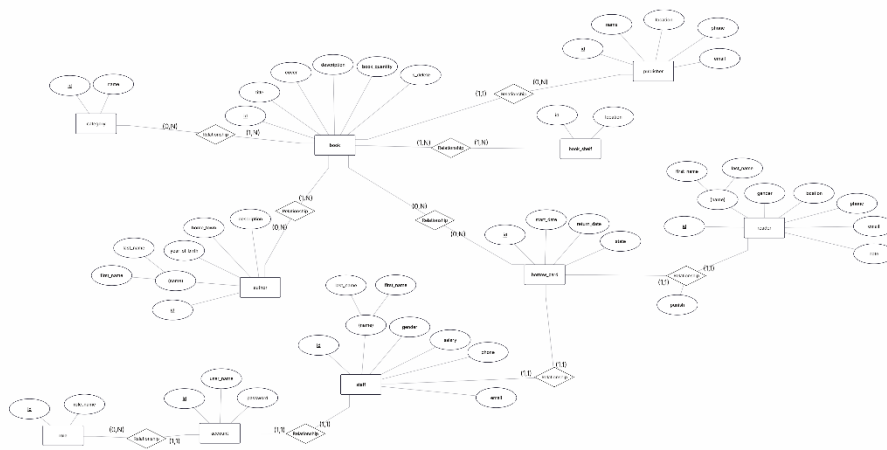
+ Tra cứu thông tin thành viên.

+ Tra cứu thông tin sách.

- + Tra cứu đầu sách, cuốn sách.
- + Tra cứu sách đã mượn.
- + Tra cứu sách trong wishlist.
- + Chức năng báo cáo thống kê.
 - Số phiếu mượn sách trong năm.
 - Thành viên mượn nhiều nhất.
 - Sách được mượn nhiều nhất.

4. Thiết kế cơ sở dữ liệu

4.1. Thiết kế cơ sở dữ liệu mức khái niệm



Hình 2. EER Diagram

4.2 Thiết kế cơ sở dữ liệu mức logic

book (id, title, cover, description, book_quantity, is_delete)

category (id, name)

author (id, first_name, last_name, year_of_birth, home_town, description)

publisher (id, name, location, phone, email)

shelf (id, location)

reader (id, first_name, last_name, gender, location, phone, email, note)

borrow_card (id, start_date, return_date, state)

staff (id, first_name, last_name, gender, location, salary, phone, email)

Báo cáo Đồ án LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG & CƠ SỞ DỮ LIỆU

account (id, user_name, password)

role (id, role_name)

		Phụ thuộc hàm chính
BOOK		id → title, cover, description, book_quantity, is_delete
CATEGORY		id → name
AUTHOR	id	id → first_name, last_name, year_of_birth, home_town, description
PUBLISHER	id	id → name, location, phone, email
BOOK_SHELF	id	id → location
READER	id	id → first_name, last_name, gender, location, phone, email, note
BORROW_CARD	id	id → start_date, return_date, state
STAFF	id	id → first_name, last_name, gender, location, salary, phone, email
ACCOUNT	id	id → user_name, password
ROLE	id	id → role_name

Bảng 1. Cơ sở dữ liệu mức logic

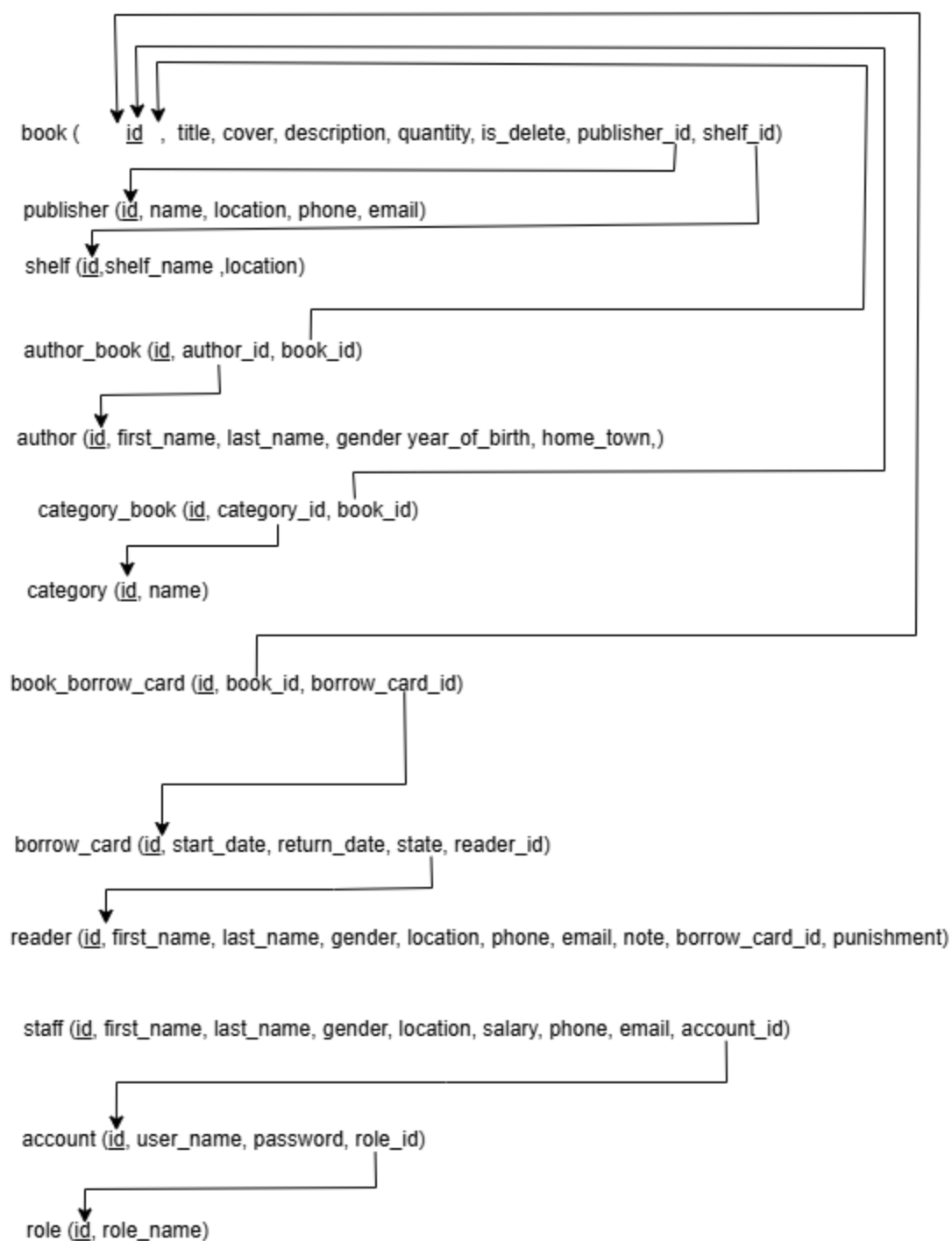
Chuẩn hóa các quan hệ với các phụ thuộc hàm đã xác định ở trên

1 NF: Tất cả các quan hệ trên đều thỏa mãn

2 NF: Tất cả các quan hệ trên đều thỏa mãn

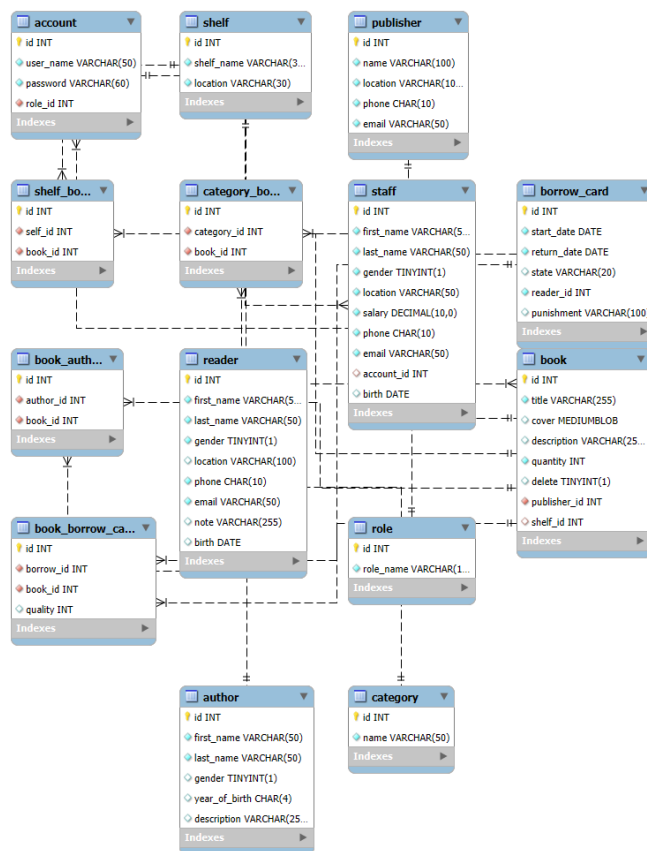
3 NF: Tất cả các quan hệ trên đều thỏa mãn

BCNF: Tất cả các quan hệ trên đều thỏa mãn



Hình 3. Lược đồ cơ sở dữ liệu mức logic

4.3 Thiết kế cơ sở dữ liệu mức vật lý



Hình 4. Diagram cơ sở dữ liệu mức vật lý

Bảng account (Tài khoản)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của tài khoản.
user_name	VARCHAR(50) (NOT NULL, UNIQUE)	Tên đăng nhập của tài khoản (duy nhất).
password	VARCHAR(60) (NOT NULL)	Mật khẩu đã được mã hóa.
role_id	INT (NOT NULL, Foreign Key)	Mã vai trò của tài khoản (liên kết với bảng role).

Bảng 2. Account

Bảng author (tác giả)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của tác giả.
first_name	VARCHAR(50) (NOT NULL)	Tên gọi của tác giả.
last_name	VARCHAR(50) (NOT NULL)	Họ của tác giả.
gender	TINYINT(1)	Giới tính (0 hoặc 1).
year_of_birth	CHAR(4)	Năm sinh của tác giả.
description	VARCHAR(255)	Mô tả hoặc tiểu sử ngắn về tác giả.

Bảng 3. Author

Bảng book (sách)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của sách.
title	VARCHAR(255) (NOT NULL)	Tiêu đề hoặc tên sách.
cover	MEDIUMBLOB	Dữ liệu hình ảnh bìa sách (có thể là NULL).
description	VARCHAR(255)	Tóm tắt nội dung sách.

quantity	INT (NOT NULL)	Số lượng sách hiện có trong kho.
delete	TINYINT(1)	Cờ trạng thái xóa mềm (0: chưa xóa, 1: đã xóa).
publisher_id	INT (NOT NULL, Foreign Key)	Mã nhà xuất bản (liên kết với bảng publisher).
shelf_id	INT (Foreign Key)	Mã kệ sách chứa sách (liên kết với bảng shelf).

Bảng 4. Book

Bảng book_author (sách-tác giả)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của mỗi quan hệ.
author_id	INT (NOT NULL, Foreign Key)	Mã tác giả (liên kết với bảng author).
book_id	INT (NOT NULL, Foreign Key)	Mã sách (liên kết với bảng book).

Bảng 5. Book_Author

Bảng book_borrow_card (chi tiết phiếu mượn)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của chi tiết mượn sách.
borrow_id	INT (NOT NULL, Foreign Key)	Mã phiếu mượn (liên kết với bảng borrow_card).
book_id	INT (NOT NULL, Foreign Key)	Mã sách được mượn (liên kết với bảng book).
quality	INT	Số lượng cuốn sách được mượn (mặc định là 1).

Bảng 6. Book_borrow_card

Bảng borrow_card (phiếu mượn)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của phiếu mượn.
start_date	DATE (NOT NULL)	Ngày bắt đầu mượn sách.
return_date	DATE (NOT NULL)	Ngày dự kiến trả sách.
state	VARCHAR(20)	Trạng thái của phiếu mượn (Đang mượn, Đã trả, Quá hạn...).
reader_id	INT (NOT NULL)	Mã độc giả (liên kết với bảng reader).
punishment	VARCHAR(100)	Thông tin phạt nếu có (quá hạn, hư hỏng).

Bảng 7. Borrow_card

Bảng category (danh mục)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của danh mục.
name	VARCHAR(50) (NOT NULL, UNIQUE)	Tên của danh mục (thể loại) sách (duy nhất).

Bảng 8. Category

Bảng category_book (danh mục-sách)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của mối quan hệ.
category_id	INT (NOT NULL, Foreign Key)	Mã danh mục (liên kết với bảng category).
book_id	INT (NOT NULL, Foreign Key)	Mã sách (liên kết với bảng book).

Bảng 9. Category_book

Bảng publisher (nhà xuất bản)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của nhà xuất bản.
name	VARCHAR(100) (NOT NULL)	Tên nhà xuất bản.
location	VARCHAR(100) (NOT NULL)	Địa chỉ của nhà xuất bản.
phone	CHAR(10) (NOT NULL, UNIQUE)	Số điện thoại của nhà xuất bản (duy nhất).
email	VARCHAR(50) (NOT NULL, UNIQUE)	Địa chỉ email của nhà xuất bản (duy nhất).

Bảng 10. Publisher

Bảng reader (độc giả)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã độc giả (tương đương Ma_Khach_hang).
first_name	VARCHAR(50) (NOT NULL)	Tên gọi của độc giả.
last_name	VARCHAR(50) (NOT NULL)	Họ của độc giả.
gender	TINYINT(1) (NOT NULL)	Giới tính (0 hoặc 1).
location	VARCHAR(100)	Địa chỉ của độc giả.
phone	CHAR(10) (NOT NULL)	Số điện thoại (tương đương Phone).
email	VARCHAR(50) (NOT NULL)	Địa chỉ email (tương đương Email).
note	VARCHAR(255)	Ghi chú hoặc thông tin bổ sung.
birth	DATE	Ngày sinh của độc giả.

Bảng 11. Reader

Bảng role(vai trò)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của vai trò.
role_name	VARCHAR(10) (NOT NULL, UNIQUE)	Tên vai trò (Ví dụ: Admin, Staff).

Bảng 12. Role

Bảng shelf (kệ sách)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của kệ sách.
shelf_name	VARCHAR(30) (NOT NULL, UNIQUE)	Tên kệ sách (Ví dụ: Kệ A1, Kệ B2).
location	VARCHAR(30) (NOT NULL)	Vị trí cụ thể của kệ sách (Ví dụ: Tầng 1 - Khu A).

Bảng 13. Shelf

Bảng staff (nhân viên)

Tên trường	Kiểu dữ liệu	Mô tả
id	INT (Auto Increment, Primary Key)	Mã định danh duy nhất của nhân viên.
first_name	VARCHAR(50) (NOT NULL)	Tên gọi của nhân viên.
last_name	VARCHAR(50) (NOT NULL)	Họ của nhân viên.
gender	TINYINT(1) (NOT NULL)	Giới tính (0 hoặc 1).
location	VARCHAR(50) (NOT NULL)	Địa chỉ của nhân viên.

salary	DECIMAL(10,0) (NOT NULL)	Mức lương của nhân viên.
phone	CHAR(10) (NOT NULL)	Số điện thoại của nhân viên.
email	VARCHAR(50) (NOT NULL)	Địa chỉ email của nhân viên.
account_id	INT (Foreign Key)	Mã tài khoản liên kết (liên kết với bảng account, có thể NULL).
birth	DATE	Ngày sinh của nhân viên.

Bảng 14. Staff

CHƯƠNG III. TRIỂN KHAI HỆ THỐNG

1. Giới thiệu các thư viện sử dụng

1.1.JPA Validation (Sử dụng Hibernate Validator Implementation)

Thư Viện	Mô tả Chức năng
JPA Validation	Là API chuẩn Java EE/Jakarta EE (thường là Bean Validation) được sử dụng để định nghĩa và áp dụng các ràng buộc (constraints) trên các trường dữ liệu (fields) của đối tượng mô hình (DTO/Entity).
Hibernate Validator	Là triển khai (implementation) phổ biến nhất của chuẩn JPA Validation. Nó cung cấp các annotation như <code>@NotNull</code> , <code>@Size</code> , <code>@Email</code> , <code>@Pattern</code> ,... để khai báo quy tắc kiểm tra dữ liệu đầu vào, đảm bảo tính hợp lệ của dữ liệu trước khi chúng được xử lý bởi tầng nghiệp vụ hoặc lưu vào cơ sở dữ liệu.
Vai trò	Kiểm tra dữ liệu đầu vào (Input Validation) ngay tại tầng ứng dụng, giúp tăng tính bảo mật và độ tin cậy của hệ thống.

Bảng 15. Thư viện JPA Validation

1.2.Jakarta Mail

Thư Viện	Mô tả Chức năng
Jakarta Mail	Là một framework chuẩn Java (trước đây là JavaMail) cung cấp các API để gửi (send) và nhận (receive) email.
Vai trò	Cho phép hệ thống thực hiện các chức năng liên quan đến email như: Gửi thông báo tự động (ví dụ: xác nhận mượn sách, thông báo quá hạn), thiết lập lại mật khẩu , hoặc các thông báo khẩn cấp cho nhân viên/độc giả.

Bảng 16. Thư viện Jakarta Mail

1.3. Spring AOP

Thư Viện	Mô tả Chức năng
Spring AOP (Aspect-Oriented Programming)	Là một module của Spring Framework cho phép lập trình hướng khía cạnh. Nó được dùng để tách các Concern Xuyên Suốt (Cross-cutting Concerns) khỏi logic nghiệp vụ chính.

Vai trò	Được sử dụng để triển khai các chức năng mà không cần sửa đổi code nghiệp vụ: Ghi nhật ký (Logging) các thao tác quan trọng (@Before, @After), Quản lý giao dịch (Transaction Management) , Xác thực và Ủy quyền (Security/Authorization) , hoặc đo lường hiệu năng của các phương thức.
Ví dụ	Bạn có thể dùng AOP để ghi lại ai đã tạo/xóa sách nào mà không cần nhúng code ghi log vào mọi phương thức BookService.

Bảng 17. Thư viện Spring AOP

1.4. MySQL

Công Nghệ	Mô tả Chức năng
MySQL	Là hệ quản trị cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở rất phổ biến.
Vai trò	Đóng vai trò là Data Store (nơi lưu trữ dữ liệu) chính cho toàn bộ hệ thống. Nó lưu trữ thông tin về Sách, Độc giả, Phiếu mượn, Tài khoản,... và đảm bảo các mối quan hệ dữ liệu được duy trì.

Bảng 18. MySQL

1.5. FlatLaf UI Manager

Thư Viện	Mô tả Chức năng
FlatLaf UI Manager	Là một thư viện Look and Feel (L&F) hiện đại cho các ứng dụng Desktop Java (sử dụng Swing/JavaFX).
Vai trò	Thiết lập giao diện người dùng (UI) cho ứng dụng Desktop Thư viện. Nó cung cấp một giao diện phẳng (flat design) , hiện đại, và có thể tùy chỉnh dễ dàng (ví dụ: chế độ sáng/tối - light/dark mode), làm cho ứng dụng trở nên chuyên nghiệp và thân thiện hơn so với giao diện Java Swing mặc định.

Bảng 19. Thư viện FlatLaf UI Manager

1.6. Một số thư viện khác

Tên Pattern	Loại Pattern	Mục Đích Chính	Vai Trò trong Ứng Dụng

Proxy	Cấu trúc (Structural)	Cung cấp một đối tượng đại diện (Surrogate) cho một đối tượng khác, nhằm kiểm soát việc truy cập hoặc thêm chức năng.	Dùng để kiểm soát truy cập (access control) , tải chậm (lazy loading) , hoặc ghi log trước khi gọi phương thức thực.
Singleton	Khởi tạo (Creational)	Đảm bảo một lớp chỉ có DUY NHẤT một thể hiện (instance) trong toàn bộ ứng dụng và cung cấp một điểm truy cập toàn cục tới thể hiện đó.	Thường được dùng cho các tài nguyên dùng chung như Connection Pool , Logger , hoặc lớp Cấu hình (Configuration Manager) .
Factory Method	Khởi tạo (Creational)	Định nghĩa một giao diện (interface) chung để tạo đối tượng, nhưng để các lớp con (subclasses) quyết định lớp nào sẽ được khởi tạo.	Tách biệt logic tạo đối tượng khỏi code sử dụng nó. Giúp code linh hoạt và dễ dàng mở rộng để tạo ra các loại đối tượng khác nhau.
DAO Pattern	Kiến trúc/J2EE	Tách biệt logic truy cập dữ liệu ra khỏi logic nghiệp vụ của ứng dụng.	Chuẩn hóa các thao tác CRUD (Create, Read, Update, Delete) với cơ sở dữ liệu. Bảo vệ tầng nghiệp vụ khỏi sự thay đổi của công nghệ lưu trữ.
DTO Pattern	Kiến trúc/J2EE	Dùng để truyền tải dữ liệu giữa các tầng (layers) hoặc qua mạng.	Đóng gói dữ liệu cần thiết cho một thao tác, giúp giảm tải dữ liệu thừa và tăng cường bảo mật bằng cách chỉ lộ ra các trường cần thiết.

Bảng 20. Design Pattern

2. Phân Tích Chức Năng Theo Các Tầng Kiến Trúc

2.1. Data Access Layer (Tầng Truy Cập Dữ Liệu)

Tầng này chịu trách nhiệm duy nhất là tương tác trực tiếp với **Cơ sở Dữ liệu (CSDL)**.

- **Các thành phần chính:** Các tệp **DAO** (Data Access Object) như AccountDAO, BookDAO, ReaderDAO, v.v.
- **Chức năng:**
 - Thực thi các câu lệnh **SQL** (SELECT, INSERT, UPDATE, DELETE).
 - Sử dụng **JDBC** để thiết lập kết nối CSDL và xử lý các thao tác I/O.
 - Chuyển đổi dữ liệu từ dạng **Record/ResultSet** của CSDL thành các đối tượng **DTO** hoặc **Entity** để chuyển lên tầng trên.
 - Đảm bảo tính toàn vẹn (Integrity) của dữ liệu trong CSDL.
- **Mục tiêu:** Cô lập logic truy cập dữ liệu khỏi các logic khác, giúp dễ dàng thay đổi loại CSDL mà không ảnh hưởng đến tầng nghiệp vụ.

2.2. Business Logic Layer (Tầng Nghiệp Vụ)

Tầng này chứa logic kinh doanh và các quy tắc nghiệp vụ của hệ thống. Đây là nơi các quyết định được đưa ra.

- **Các thành phần chính:** Các tệp **Service** (ví dụ: BookService, BorrowCardService, ReaderService).
- **Chức năng:**
 - **Xử lý Logic Nghiệp vụ:** Kiểm tra tính hợp lệ của dữ liệu (Validation), xử lý tính toán, và áp dụng các quy tắc kinh doanh (ví dụ: kiểm tra số lượng sách tồn kho trước khi cho mượn, kiểm tra độc giả có đang nợ phạt không).
 - **Điều phối (Coordination):** Gọi các phương thức từ một hoặc nhiều **DAO** để thực hiện một tác vụ phức tạp (ví dụ: khi mượn sách, BorrowCardService phải gọi BorrowCardDAO để tạo phiếu và BookDAO để giảm số lượng sách).
 - Sử dụng các đối tượng **DTO** để nhận dữ liệu từ tầng Presentation và gửi kết quả đã xử lý đi.
- **Mục tiêu:** Đảm bảo tính nhất quán và đúng đắn của các quy trình nghiệp vụ.

2.3. Presentation Layer (Tầng Trình Bày/Giao Diện)

Tầng này chịu trách nhiệm hiển thị thông tin cho người dùng và tiếp nhận yêu cầu từ họ.

- **Các thành phần chính:** Các tệp **Controller** (nếu là ứng dụng Web) hoặc các lớp **View** (nếu là ứng dụng Desktop), và các đối tượng **DTO** (Data Transfer Object).
- **Chức năng:**
 - **Hiển thị:** Trình bày dữ liệu đã được xử lý từ tầng Business Logic dưới dạng giao diện thân thiện.
 - **Tiếp nhận yêu cầu:** Thu thập dữ liệu đầu vào từ người dùng (ví dụ: thông tin độc giả mới, ID sách mượn).
 - **Giao tiếp:** Gọi các phương thức của tầng Business Logic (Service) và sử dụng các đối tượng **DTO** để truyền và nhận dữ liệu.
- **Mục tiêu:** Đảm bảo trải nghiệm người dùng (UX) tốt và thu thập dữ liệu đầu vào chính xác.

3. Giao diện màn hình

3.1. Trang đăng nhập

Đăng nhập

—

×

ĐĂNG NHẬP VÀO HỆ THỐNG

Tên đăng nhập

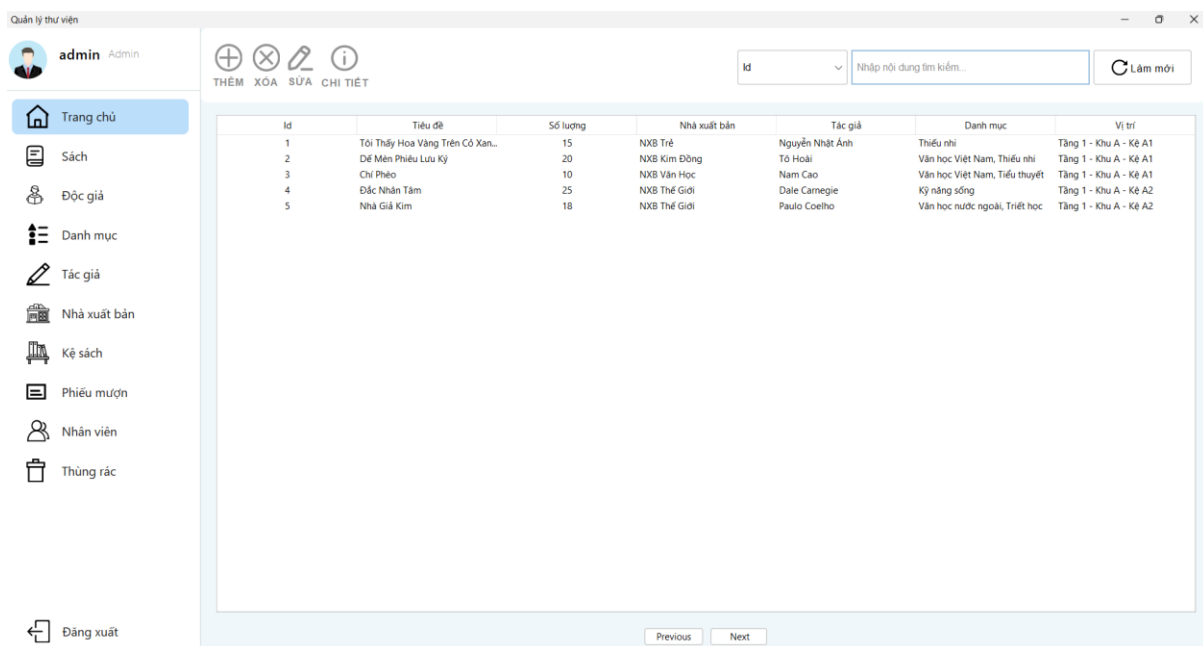
Mật khẩu

[Quên mật khẩu](#)

ĐĂNG NHẬP

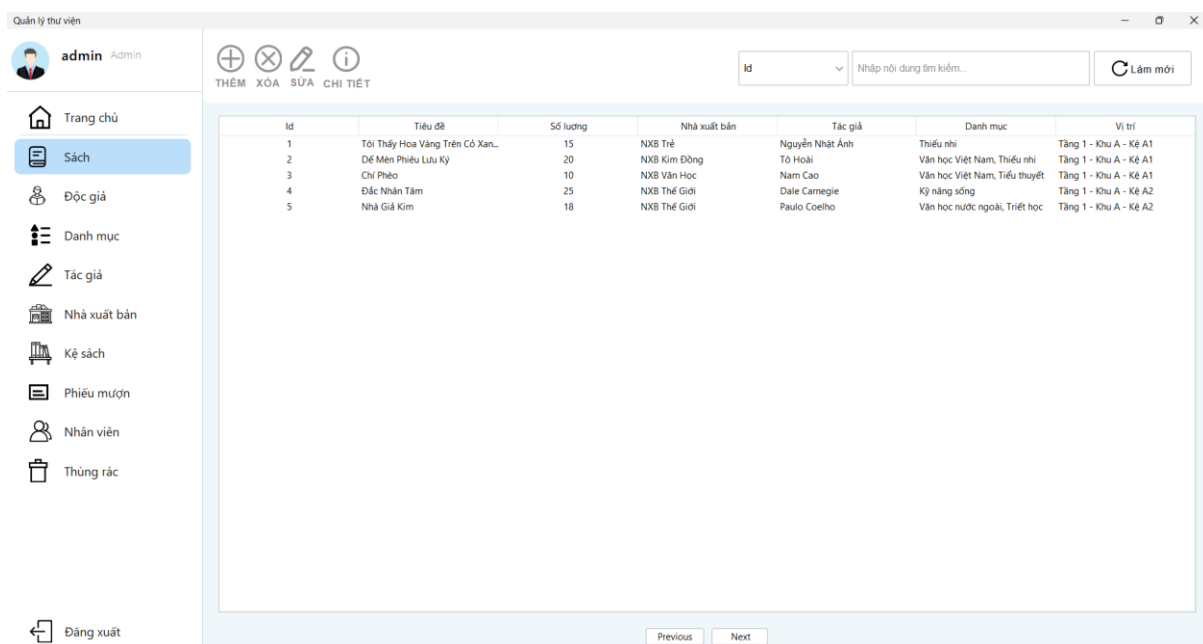
Hình 5. View Đăng nhập

3.2. Trang chủ



Hình 6. View Trang chủ

3.3. Sách



Hình 7. View Sách

3.4. Đọc giả

Quản lý thư viện

admin Admin

THÊM XÓA SỬA

Id Nhập nội dung tìm kiếm... Làm mới

Id	Họ và tên	Giới tính	Địa chỉ	Ngày sinh	Số điện thoại	Email
1	Trần Minh Anh	Nữ	Quận 1, TP.HCM		0909123401	tranminhanh@email.c...
2	Nguyễn Văn Bình	Nam	Quận 3, TP.HCM		0909123402	nguyenvanbinh@email...
3	Lê Thị Cúc	Nữ	Quận 5, TP.HCM		0909123403	lethicuc@email.com
4	Phạm Hoàng Dũng	Nam	Quận 7, TP.HCM		0909123404	phamhoangdung@em...
5	Võ Thị Em	Nữ	Quận 10, TP.HCM		0909123405	vothiemail.com
6	Đặng Văn Phong	Nam	Quận Bình Thạnh, TP.H...		0909123406	dangvanphong@email...
7	Hoàng Thị Giang	Nữ	Quận Tân Bình, TP.HC...		0909123407	hoangthi Giang@email...
8	Bùi Văn Hải	Nam	Quận Phú Nhuận, TP.H...		0909123408	buivanhai@email.com
9	Đinh Thị Lan	Nữ	Quận 2, TP.HCM		0909123409	dinhthilam@email.com
10	Trịnh Văn Khoa	Nam	Quận 4, TP.HCM		0909123410	trinhvankhoa@email.c...
11	Nam Nguyễn Văn	Nữ	Quận 1, TP.HCM		0909123401	tranminhanh@email.c...
12	Nguyễn Văn Long	Nam	Quận 6, TP.HCM	1995-05-15	0909123411	nguyenvanlong@emai...
13	Trần Thị Mai	Nữ	Quận 8, TP.HCM	2005-08-20	0909123412	tranthimai@email.com
14	Lê Văn Nam	Nam	Quận 9, TP.HCM	1990-03-10	0909123413	levanam@email.com
15	Phạm Thị Oanh	Nữ	Quận 11, TP.HCM	1988-12-05	0909123414	phamthioanh@email.c...
16	Võ Văn Phúc	Nam	Quận 12, TP.HCM	1992-07-25	0909123415	vovanphuc@email.co...
17	Đặng Thị Quỳnh	Nữ	Quận Gò Vấp, TP.HCM	2002-11-30	0909123416	dangthiquynh@email...
18	Hoàng Văn Rộng	Nam	Quận Thủ Đức, TP.HCM	1985-04-18	0909123417	hoangvanrong@email...
19	Bùi Thị Sơn	Nữ	Quận Bình Tân, TP.HC...	1998-09-12	0909123418	buthison@email.com
20	Đinh Văn Tài	Nam	Huyện Bình Chánh, TP...	1987-02-28	0909123419	dinhvantai@email.com
21	Trịnh Thị Uyên	Nữ	Huyện Hóc Môn, TP.H...	1983-06-14	0909123420	trinhthiuyen@email.co...
22	Phan Văn Vũ	Nam	Huyện Củ Chi, TP.HCM	1996-10-07	0909123421	phanvannu@email.co...
23	Mai Thị Xuân	Nữ	Huyện Nhà Bè, TP.HCM	2000-01-22	0909123422	maithixuan@email.co...
24	Lý Văn Yên	Nam	Huyện Cần Giuộc, TP.HC...	1991-05-19	0909123423	lyvanyen@email.com
25	Dương Thị Ánh	Nữ	Quận 1, TP.HCM	1994-08-03	0909123424	duongthianh@email.c...
26	Cao Văn Bảo	Nam	Quận 2, TP.HCM	1986-11-28	0909123425	caovanbao@email.co...
27	Huỳnh Thị Chi	Nữ	Quận 3, TP.HCM	1999-03-15	0909123426	huynhthichi@email.co...
28	Tô Văn Đức	Nam	Quận 4, TP.HCM	1989-07-09	0909123427	tovanduc@email.com
29	Vương Thị Hoa	Nữ	Quận 5, TP.HCM	1993-12-21	0909123428	vuongthihoa@email.c...
30	Đỗ Văn Khang	Nam	Quận 6, TP.HCM	1997-04-06	0909123429	dovankhang@email.c...
31	Chu Thị Linh	Nữ	Quận 7, TP.HCM	1990-09-17	0909123430	chuthilinh@email.com
32	Tà Văn Minh	Nam	Quận 8, TP.HCM	1995-02-11	0909123431	tavanminh@email.com

Đăng xuất

Hình 8. View Đọc giả

3.5. Danh mục

Quản lý thư viện

admin Admin

THÊM XÓA SỬA

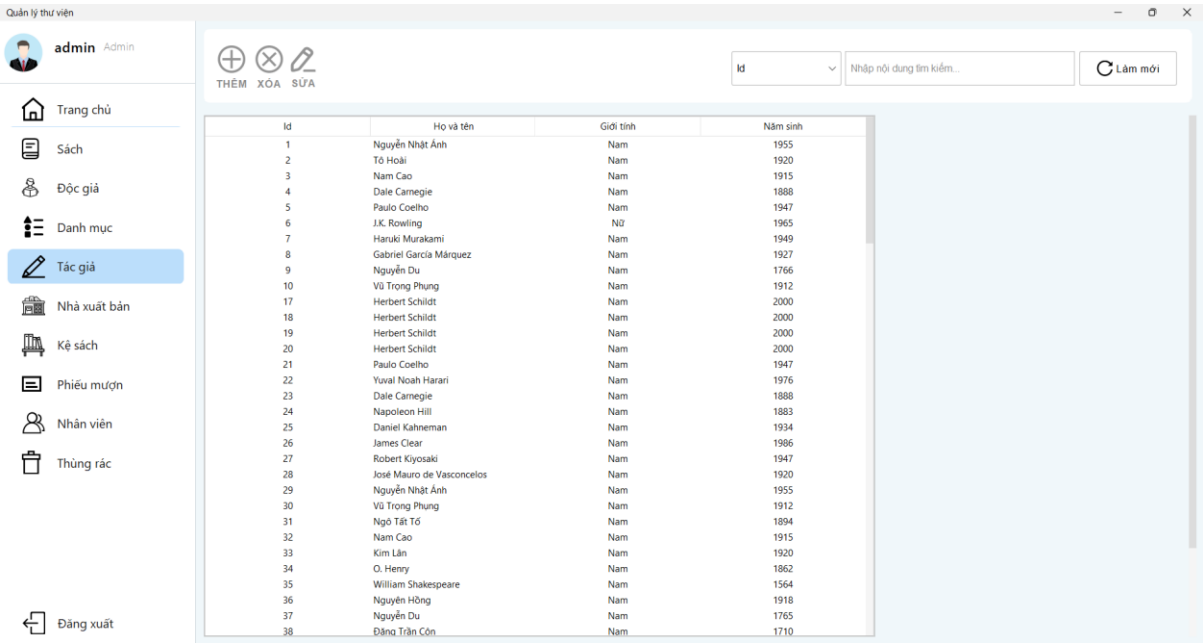
Id Nhập nội dung tìm kiếm... Làm mới

Id	Danh mục
96	Agile
35	Algorithms
73	Apache Spark
56	API Development
64	Automation
60	AWS
61	Azure
49	Bách khoa
53	Backend Development
72	Big Data
36	Blockchain
76	Business Intelligence
65	CI/CD
93	Clean Architecture
34	Clean code
37	Cloud computing
14	Cơ sở dữ liệu
78	Computer Vision
10	Công nghệ
54	Container Technology
71	Cryptography
40	Cybersecurity
75	Data Visualization
92	DDD
39	Deep Learning
33	Design patterns
41	DevOps
106	Địa lý
87	Django
69	Ethical Hacking
89	FastAPI
88	Flask

Đăng xuất

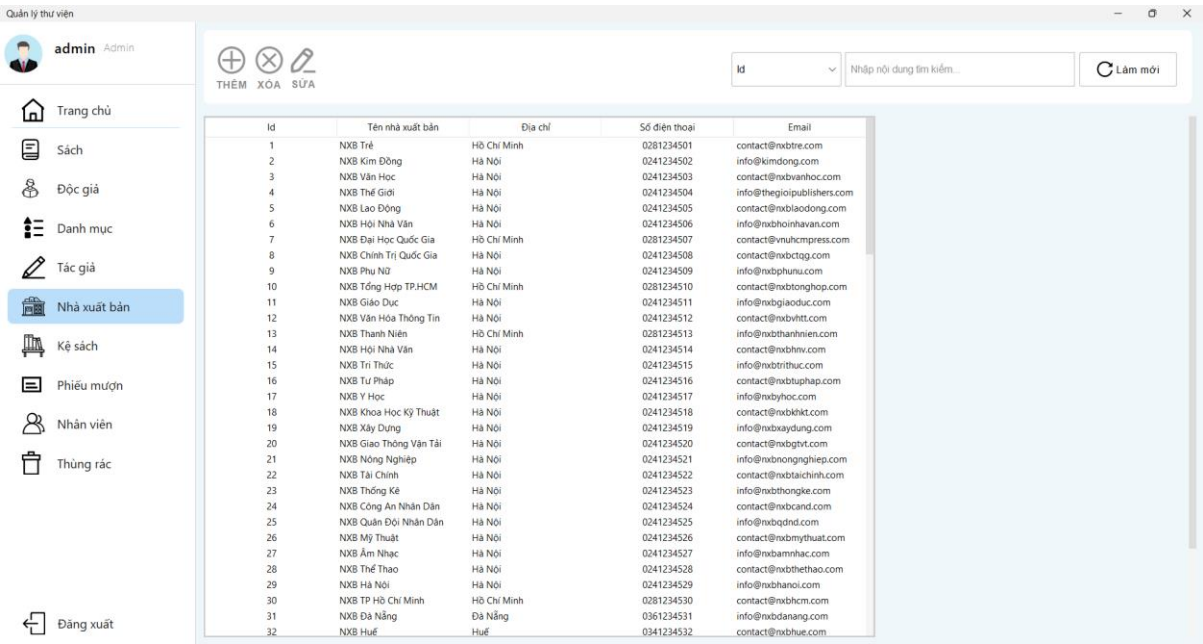
Hình 9. View Danh mục

3.6. Tác giả



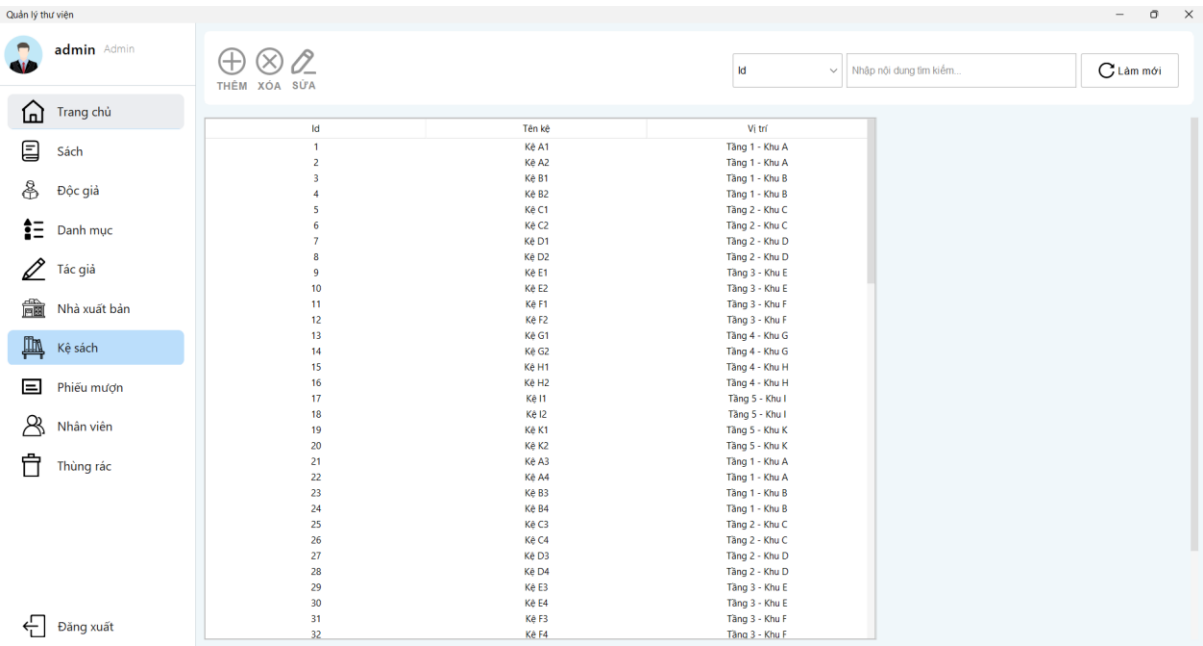
Hình 10. View Tác giả

3.7. Nhà xuất bản



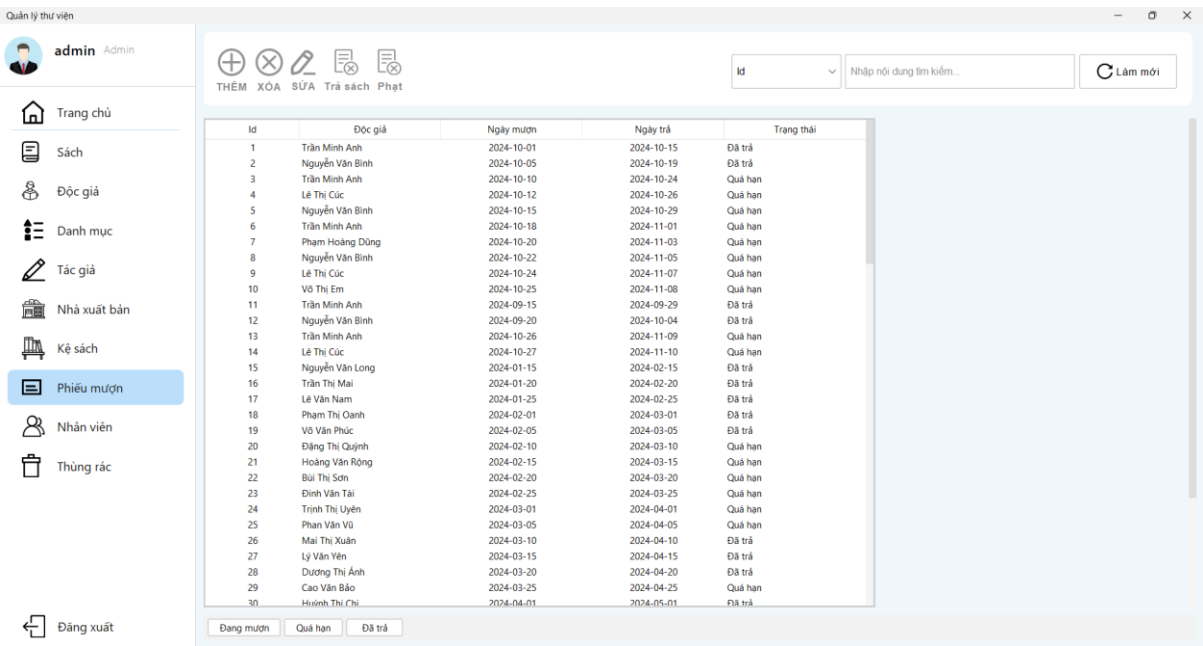
Hình 11. View Nhà xuất bản

3.8. Kệ sách



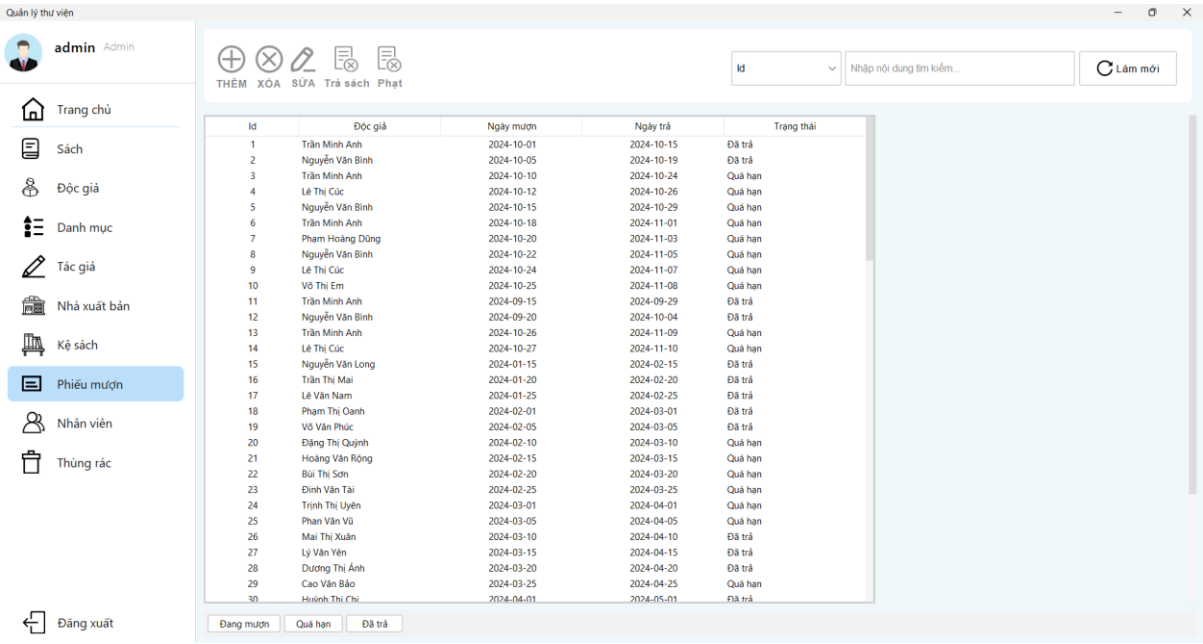
Hình 12. View kệ sách

3.9. Phiếu mượn



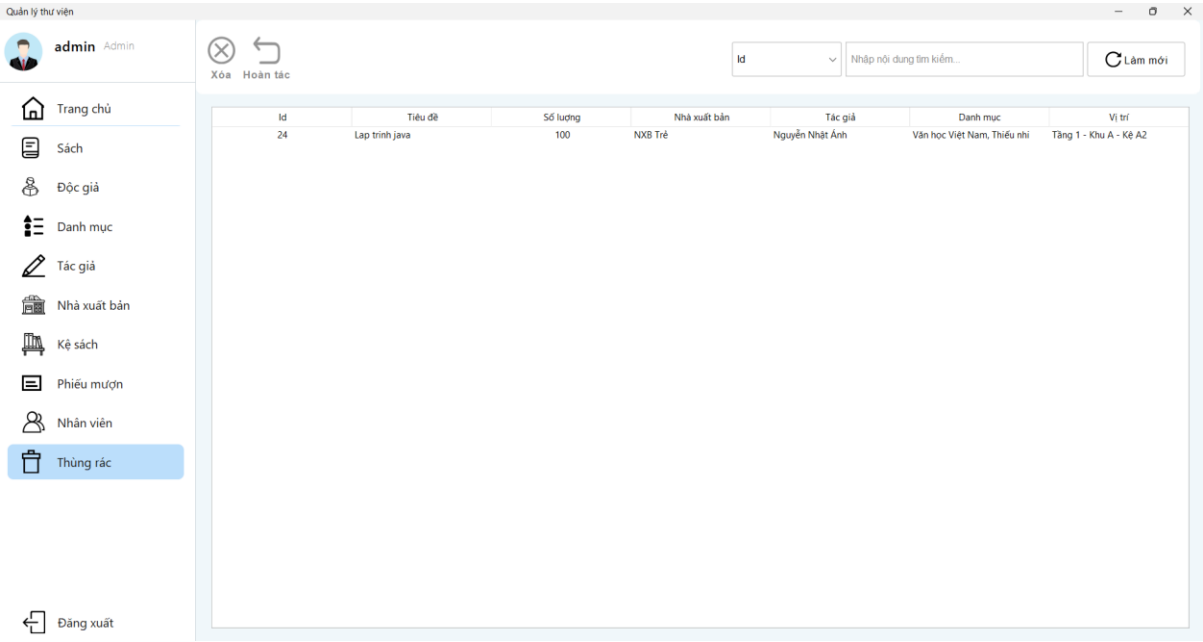
Hình 13. View Phiếu mượn

3.10. Nhân viên



Hình 14. View Nhân viên

3.11. Thùng rác



Hình 15. View Thùng rác

4. Xử lý truy vấn dữ liệu tìm kiếm

4.1. Tìm kiếm sách với nhiều điều kiện

```
SELECT * FROM book
JOIN publisher ON book.publisher_id = publisher.id
JOIN shelf ON book.shelf_id = shelf.id
WHERE (book.id = ? OR ? IS NULL)
      AND (book.title LIKE CONCAT('%',?, '%') OR ? IS NULL)
      AND book.`delete` = ?
LIMIT ? OFFSET ?;
```

Hình 16. SQL Tìm kiếm sách

- Tìm sách kết hợp với nhà xuất bản và kệ sách
- Điều kiện linh hoạt: có thể tìm theo ID, tiêu đề, hoặc cả hai
- Sử dụng ? IS NULL để xử lý tham số tùy chọn
- Phân trang với LIMIT và OFFSET
- Lọc sách chưa bị xóa (delete = false)

4.2. Tìm kiếm độc giả, tác giả

```
SELECT * FROM reader
WHERE (id = ? OR ? IS NULL)
      AND (CONCAT(first_name, ' ', last_name) LIKE CONCAT('%', ?, '%') OR ? IS NULL);
SELECT * FROM author
WHERE (id = ? OR ? IS NULL)
      AND (CONCAT(first_name, ' ', last_name) LIKE CONCAT('%', ?, '%') OR ? IS NULL);
```

Hình 17. SQL Tìm kiếm độc giả, tác giả

- Tìm theo ID hoặc tên (hoặc cả hai)
- Ghép họ và tên để tìm kiếm toàn diện
- Điều kiện linh hoạt với tham số NULL

4.3. Tìm kiếm nhân viên với tài khoản

```
SELECT * FROM staff
JOIN account ON staff.account_id = account.id
WHERE (staff.id = ? OR ? IS NULL)
AND (CONCAT(staff.first_name, ' ', staff.last_name) LIKE CONCAT('%',?, '%') OR ? IS NULL)
AND (account.user_name LIKE CONCAT('%',?, '%') OR ? IS NULL);
```

Hình 18. SQL Tìm kiếm nhân viên

- Kết hợp thông tin nhân viên và tài khoản
- Tìm theo ID nhân viên, tên nhân viên, username
- Điều kiện tìm kiếm đa chiều

4.4. Tìm kiếm phiếu mượn

```
SELECT * FROM borrow_card
JOIN reader ON borrow_card.reader_id = reader.id
WHERE (borrow_card.id = ? OR ? IS NULL)
AND (CONCAT(reader.first_name, ' ', reader.last_name) LIKE CONCAT('%', ?, '%') OR ? IS NULL);
```

Hình 19. SQL Tìm kiếm phiếu mượn

- Kết hợp phiếu mượn với thông tin độc giả
- Tìm theo ID phiếu hoặc tên độc giả

4.5. Tìm sách đang mượn

```
SELECT * FROM borrow_card
JOIN reader ON borrow_card.reader_id = reader.id
JOIN book_borrow_card ON borrow_card.id = book_borrow_card.borrow_id
JOIN book ON book_borrow_card.book_id = book.id
WHERE borrow_card.state != 'Đã trả' AND reader.id = ?;
```

Hình 20. SQL Tìm sách đang mượn

4.6. Xóa tài khoản và chi tiết sách

```
DELETE FROM account WHERE id = ?;
DELETE FROM book_borrow_card WHERE borrow_id = ?;
```

Hình 21. Xóa tài khoản và xóa chi tiết sách

CHƯƠNG IV. KẾT LUẬN

Đồ án "Quản lý Thư viện" đã được triển khai đáp ứng đầy đủ các yêu cầu đề ra ban đầu. Hệ thống không chỉ giải quyết bài toán quản lý thư viện truyền thống mà còn ứng dụng các công nghệ hiện đại để nâng cao hiệu quả vận hành. Về mặt chức năng, hệ thống đã xây dựng thành công các module cốt lõi: quản lý kho sách với đầy đủ thông tin tác giả, nhà xuất bản, thể loại; quản lý độc giả và nhân viên; xử lý nghiệp vụ mượn/trả sách linh hoạt; cùng hệ thống phân quyền bảo mật. Về kỹ thuật, đồ án áp dụng thành công kiến trúc đa tầng, sử dụng Java 17 và MySQL, kết hợp các design pattern phù hợp giúp code dễ bảo trì và mở rộng.

Trong quá trình thực hiện, nhóm đã vượt qua nhiều thách thức từ việc thiết kế cơ sở dữ liệu phức tạp đến xử lý các nghiệp vụ đa dạng. Những khó khăn này đem lại bài học quý giá về phân tích hệ thống, quản lý dự án và làm việc nhóm. Thành phẩm cuối cùng là một sản phẩm tương đối hoàn chỉnh với giao diện thân thiện, tính năng đầy đủ, có tính ứng dụng thực tế cao. Đồ án không chỉ khép lại môn học mà còn là bước đệm quan trọng, trang bị cho cả nhóm kinh nghiệm thực tế để sẵn sàng cho các dự án chuyên nghiệp trong tương lai.

TÀI LIỆU THAM KHẢO

- [1] Cay S. Horstmann (2022). *Core Java Volume I: Fundamentals*. Pearson Education.
- [2] Martin Fowler (2019). *Patterns of Enterprise Application Architecture*. Addison-Wesley.
- [3] Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides (1994). *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley.
- [4] Craig Walls (2022). *Spring in Action*. Manning Publications.
→ Hướng dẫn sử dụng Spring Framework, AOP và transactional management

TÀI LIỆU TRỰC TUYẾN VÀ WEBSITE

- [5] Oracle Corporation. *Java Platform, Standard Edition Documentation*.
<https://docs.oracle.com/en/java/javase/17/>
- [6] MySQL Documentation. *MySQL 8.0 Reference Manual*.
<https://dev.mysql.com/doc/refman/8.0/en/>
- [7] Hibernate Validator. *Reference Guide*.
https://docs.jboss.org/hibernate/validator/7.0/reference/en-US/html_single/
- [8] FlatLaf Documentation.
<https://www.formdev.com/flatlaf/>
- [9] Jakarta Mail Specification.
<https://eclipse-ee4j.github.io/mail/>
- [10] Refactoring Guru
<https://refactoring.guru/>