

TRƯỜNG ĐẠI HỌC PHENIKAA

KHOA CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN CƠ SỞ

Nhóm: Phần mềm quản lý 3

Đề tài: Xây dựng ứng dụng quản lý bán hàng sách trực tuyến trên nền Web

Thành viên:	Trịnh Thị Ngọc Anh – 22010380 Phạm Vĩnh Khang – 22010197 Lương Trung Hiếu – 22010422 Vũ Xuân Hoan – 22010404
Giảng viên hướng dẫn:	ThS. Vũ Quang Dũng

Hà Nội, năm 2025

LỜI CẢM ƠN

-----□□□□□-----

Nhóm em muốn bày tỏ lòng biết ơn chân thành đến thầy Vũ Quang Dũng đã dành thời gian và công sức hướng dẫn tận tình, cũng như tạo điều kiện thuận lợi cho nhóm em trong quá trình thực hiện đề tài. Sự nhiệt huyết và sự hỗ trợ không ngừng từ thầy đã đóng góp quan trọng vào việc xây dựng nền tảng kiến thức của nhóm em, giúp nhóm vượt qua những thách thức trong quá trình nghiên cứu.

Trong suốt hành trình này, nhóm nhận thức được rằng không tránh khỏi những thiếu sót và khuyết điểm. Nhóm trân trọng những góp ý và sự chỉ dẫn từ quý thầy/cô, và nhóm cam kết sẽ tích cực thâu hiểu và áp dụng những phản hồi quý báu đó để đưa đề tài của mình lên một tầm cao mới. Sự hỗ trợ và tình cảm quý báu của quý thầy là động viên lớn, là động lực mạnh mẽ giúp em không ngừng phát triển trong sự nghiệp nghiên cứu. Nhóm em xin chân thành cảm ơn!

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN

Hà Nội, ngày ... tháng ... năm 2025

GIẢNG VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

LỜI CAM ĐOAN

-----□□□□□-----

Chúng em xin cam đoan rằng toàn bộ nội dung của báo cáo này là kết quả của quá trình nghiên cứu, tìm hiểu và thực hiện đề tài một cách nghiêm túc của nhóm. Các thông tin, tài liệu, dữ liệu được sử dụng trong báo cáo đều có nguồn gốc rõ ràng và được trích dẫn đầy đủ theo quy định.

Nhóm cam kết không sao chép, vi phạm bản quyền hay sử dụng bất kỳ hình thức gian lận nào trong quá trình thực hiện đề tài. Nếu có bất kỳ sai sót nào do nhóm chưa phát hiện hoặc có tranh chấp về quyền tác giả, nhóm xin hoàn toàn chịu trách nhiệm trước nhà trường và các bên liên quan.

Chúng em xin chân thành cảm ơn và cam đoan những điều trên là hoàn toàn đúng sự thật.

Hà Nội, ngày tháng năm 2025

NHÓM SINH VIÊN THỰC HIỆN

(Ký và ghi rõ họ tên)

PHÂN CÔNG NHIỆM VỤ THÀNH VIÊN

Danh sách các công việc/nhiệm vụ	Mô tả tóm tắt công việc
Công việc 1	Phân tích yêu cầu và thiết kế hệ thống
Công việc 2	Xây dựng giao diện người dùng (Frontend)
Công việc 3	Xây dựng logic nghiệp vụ và API (Backend)
Công việc 4	Tích hợp hệ thống và thử nghiệm
Công việc 5	Hoàn thành báo cáo và chuẩn bị trình bày

STT	MSSV	Họ và tên	Nhiệm vụ phân công	Nội dung đã thực hiện	Đánh giá
1	22010380	Trịnh Thị Ngọc Anh (Nhóm trưởng)	Công việc 1, Công việc 2, Công việc 5	<ul style="list-style-type: none"> - Xác định yêu cầu chức năng và phi chức năng. - Thiết kế giao diện người dùng. - Viết báo cáo hoàn chỉnh về dự án. - Chuẩn bị slide thuyết trình. - Hỗ trợ các thành viên khác 	100%
2	22010197	Phạm Vĩnh Khang	Công việc 2, Công việc 4	<ul style="list-style-type: none"> - Triển khai các chức năng - Thiết kế giao diện người dùng. - Kiểm tra tính năng, hiệu suất, và bảo mật của hệ thống. - Hỗ trợ các thành viên khác 	100%

2	22010422	Lương Trung Hiếu	Công việc 1, Công việc 3	<ul style="list-style-type: none"> - Phát triển API. - Thiết kế và quản lý cơ sở dữ liệu trên MySQL. - Nghiên cứu và thiết kế kiến trúc hệ thống. - Hỗ trợ các thành viên khác 	100%
4	22010404	Vũ Xuân Hoan	Công việc 1, Công việc 3	<ul style="list-style-type: none"> - Kết nối giữa backend và frontend. - Nghiên cứu và thiết kế kiến trúc hệ thống. - Hỗ trợ các thành viên khác 	100%

DANH MỤC CÁC KÝ HIỆU, CÁC CHỮ VIẾT TẮT

STT	Ký hiêu, viết tắt	Chú thích đầy đủ
1	UC	Use Case (kịch bản sử dụng)
2	BR	Business Requirement (Yêu cầu nghiệp vụ)

DANH MỤC CÁC BẢNG, HÌNH VẼ, ĐỒ THỊ

Hình 1 UC Tổng quát	24
Hình 2 UC Đăng nhập	28
Hình 3 UC Đăng xuất	29
Hình 4 UC Quản lý tài khoản	30
Hình 5 UC Quản lý sản phẩm	31
Hình 6 UC Quản lý đơn hàng	32
Hình 7 UC Quản lý đơn hàng cá nhân	33
Hình 8 UC Quản lý thông tin cá nhân	34
Hình 9 UC Quản lý thể loại	35
Hình 10 UC Đăng ký	36
Hình 11 UC Báo cáo	37
Hình 12 UC Thanh toán	38
Hình 13 UC Tìm kiếm sản phẩm	39
Hình 14 UC Đánh giá sản phẩm	40
Hình 15: UC Quản lý giỏ hàng	41
Hình 16 Sơ đồ kiến trúc tổng thể	42
Hình 17 MySQL	43
Hình 18 Spring boot	44
Hình 19 ReactJs	46
Hình 20 Redis	48
Hình 21 Key abstraction	50
Hình 22 Package	51
Hình 23 Phân tích xác thực	52
Hình 24 Phân tích quản lý sách	52
Hình 25 Phân tích quản lý thể loại	53
Hình 26: Phân tích quản lý giỏ hàng và thanh toán	53
Hình 27: Phân tích quản lý đơn hàng	55

Hình 28 Pha phân tích báo cáo thống kê	55
Hình 29: Sơ đồ thực thể liên kết cơ sở dữ liệu	56
Hình 30: Cấu trúc thư mục API.....	61
Hình 31: Sequence Đăng nhập	64
Hình 32 Sequence Thêm sản phẩm	65
Hình 33 Sequence Cập nhật thông tin	66
Hình 34 Sequence xem giỏ hàng	67
Hình 35 Sequence Lọc và tìm kiếm	67
Hình 36 Sequence Báo cáo.....	68
Hình 37 Sequence Cập nhật thông tin cá nhân.....	68
Hình 38 Sequence Xác nhận đơn hàng.....	69
Hình 39 Sequence Thanh toán.....	69
Hình 40 Sequence Đánh giá sản phẩm	70
Hình 41 Sequent chức năng xoá	70
Hình 42: Giao diện đăng nhập.....	75
Hình 43: Giao diện đăng ký	76
Hình 44: Giao diện tổng quan (trang quản lý).....	76
Hình 45: Giao diện quản lý sách	77
Hình 46: Giao diện thêm sửa xóa sách	77
Hình 47: Giao diện quản lý đơn hàng	78
Hình 48: Giao diện quản lý danh mục	78
Hình 49: Giao diện quản lý tài khoản khách hàng	79
Hình 50: Giao diện trang chủ	79
Hình 51: Giao diện thông tin cá nhân.....	80
Hình 52: Giao diện quản lý đơn hàng (dành cho khách hàng).....	80
Hình 53: Giao diện thông tin đơn hàng	81
Hình 54: Giao diện thanh toán	81

MỤC LỤC

LỜI CẢM ƠN	2
NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN.....	3
LỜI CAM ĐOAN	4
DANH MỤC CÁC BẢNG, HÌNH VẼ, ĐỒ THỊ.....	8
MỤC LỤC	10
LỜI MỞ ĐẦU	13
CHƯƠNG 1. PHÂN TÍCH VÀ ĐẶC TẨ PHẦN MỀM	15
1.1 Mô tả bài toán	15
1.2 Bảng mô tả thuật ngữ	15
1.3 Yêu cầu bài toán.....	17
1.3.1 Chuẩn bị dữ liệu cho ứng dụng.....	17
1.3.2 Thực hiện quy trình mua bán tại ứng dụng web	17
1.3.3 Báo cáo thống kê	18
1.4 Yêu cầu nghiệp vụ.....	19
1.4.1 Mô tả nghiệp vụ	19
1.4.2 Yêu cầu nghiệp vụ	19
1.5 Sơ đồ Use-case chức năng chính.....	24
1.5.1 Sơ đồ Use-case tổng quát	24
1.5.2 Các tác nhân trong hệ thống	25
1.5.3 Các Use-case	25

CHƯƠNG 2: THIẾT KẾ HỆ THỐNG	42
2.1 Sơ đồ kiến trúc tổng thể	42
2.1.1 Sơ đồ	42
2.1.2 Mô tả chi tiết	42
2.1.3 Công nghệ sử dụng.....	42
2.2 Mô tả chi tiết về công nghệ sử dụng.....	43
2.2.1 Hệ quản trị cơ sở dữ liệu MySQL	43
2.2.2 Spring boot (Web service)	44
2.2.3 ReactJs	46
2.2.4 Redis	48
2.3 Phân tích kiến trúc hệ thống.....	50
2.3.1 Key abstraction	50
2.3.2 Thành phần cấp cao và sự phụ thuộc.....	51
2.3.3 Biểu đồ lớp pha phân tích chức năng.....	52
2.4 Thiết kế hệ thống	56
2.4.1 Thiết kế cơ sở dữ liệu	56
2.4.2 Thiết kế API.....	61
2.4.3 Thiết kế chức năng.....	64
CHƯƠNG 3: XÂY DỰNG VÀ THỰC NGHIỆM	71
3.1 Cấu trúc thành phần đối tượng thực	71
3.1.1 Front-end	71
3.1.2 Back-end	73
3.2 Thực nghiệm sử dụng	74
3.2.1 Công cụ cần thiết.....	74

3.2.2 Lệnh chạy dự án Spring Boot	75
3.2.3 Lệnh chạy dự án ReactJS	75
3.3.1 Giao diện đăng nhập, đăng ký	75
3.3.2 Giao diện trang quản lý (Admin)	76
CHƯƠNG 4: ĐÁNH GIÁ VÀ KẾT LUẬN	82
4.1 Đánh giá kết quả đạt được và tồn tại so với yêu cầu ban đầu	82
4.1.1 Kết quả đạt được	82
4.1.2 Tồn tại và hạn chế	82
4.2. Hướng cải thiện và bảo trì (nâng cấp) sau này	83
4.2.1 Cải thiện và bổ sung tính năng	83
4.2.2 Tối ưu hiệu suất và khả năng mở rộng	83
4.3 Kết luận.....	84
TÀI LIỆU THAM KHẢO	85

LỜI MỞ ĐẦU

Trong bối cảnh phát triển mạnh mẽ của công nghệ thông tin và thương mại điện tử, hành vi mua sắm của người tiêu dùng đang dần thay đổi. Mua sắm trực tuyến không chỉ mang lại sự tiện lợi về thời gian và không gian, mà còn đáp ứng nhu cầu đa dạng của người dùng. Đặc biệt, đối với lĩnh vực mua bán sách trực tuyến, người dùng ngày càng ưa chuộng việc tìm kiếm và mua sách trực tuyến hơn là đến cửa hàng truyền thống.

Trong việc bán sách theo mô hình truyền thống, các nhà sách và cửa hàng bán lẻ phải đối mặt với nhiều khó khăn thách thức trong việc duy trì hoạt động và cạnh tranh với các hình thức mua sắm hiện đại. Một số khó khăn đáng chú ý bao gồm:

- a. Chi phí vận hành cao: Việc duy trì một cửa hàng truyền thống đòi hỏi nhiều chi phí từ việc thuê mặt bằng, chi phí quản lý, chi phí nhân công. Đặc biệt, với những cửa hàng nằm ở khu vực trung tâm có diện tích lớn thì những loại chi phí kể trên sẽ rất lớn.
- b. Giới hạn không gian trưng bày: Một cửa hàng sách truyền thống chỉ có một diện tích giới hạn để trưng bày các đầu sách. Điều này dẫn đến việc của hàng chỉ có thể lưu trữ và bán được một số lượng sách nhất định. Các khách hàng có thể khó tiếp cận đầy đủ các tựa sách mà họ mong muốn, đặc biệt những đầu sách hiếm hoặc ít phổ biến.
- c. Sự phụ thuộc vào địa lý: Việc bán sách truyền thống phụ thuộc rất nhiều vào vị trí bán của cửa hàng. Nếu cửa hàng không có vị trí thuận lợi cho việc mua bán hoặc trong những khu vực có lưu lượng người qua lại thấp, lượng khách sẽ bị hạn chế. Điều này có thể gây ảnh hưởng tiêu cực đến cửa hàng.
- d. Hạn chế về thời gian: Các cửa hàng truyền thống chỉ có thể hoạt động một cách cố định theo một thời gian hoạt động cụ thể, thường là giờ

hành chính. Điều này không đáp ứng được một số lượng khác hàng bận rộn hoặc có thời gian rảnh rỗi ngoài giờ làm việc.

Vì thế để khắc phục những hạn chế trên, ngoài việc kinh doanh một cửa hàng truyền thống, chúng tôi đã phát triển một ứng dụng web mang lại một giải pháp toàn diện hơn. Ứng dụng không chỉ cung cấp một giao diện thân thiện, dễ sử dụng, mà còn tích hợp nhiều tính năng quản lý vượt trội. Ngoài ra, hệ thống thanh toán còn được cung cấp tích hợp nhiều hình thức thanh toán, giúp người mua an tâm trong quá trình thanh toán.

Với sự phát triển của ứng dụng này, chúng tôi hướng đến việc trải nghiệm mua sách trực tuyến mượt mà, tiện lợi và đáng tin cậy cho mọi đối tượng khách hàng. Thông qua đó không chỉ giúp khách hàng dễ dàng tìm kiếm và sở hữu những cuốn sách yêu thích mà còn phát triển bên vững của ngành thương mại điện tử trong lĩnh vực sách.

Về tài liệu phân tích thiết kế sẽ chia thành các chương như sau:

- *Chương 1: Phân tích và đặc tả phần mềm*

Giới thiệu về bài toán mà hệ thống cần giải quyết và phân tích các khó khăn, thực trạng mà các bộ phận đang gặp phải, đối với việc quản lý và mua bán tại một cửa hàng nhằm đưa các giải pháp và yêu cầu nghiệp vụ cho hệ thống.

- *Chương 2: Thiết kế hệ thống*

Giới thiệu, mô tả về kiến trúc tổng thể của hệ thống và các công nghệ được sử dụng.

- *Chương 3: Xây dựng và thực nghiệm*

Mô tả triển khai hệ thống, giao diện chương trình và cách thức vận hành..

- *Chương 4: Đánh giá và kết luận*

Đánh giá mức độ đáp ứng yêu cầu, những điểm đạt được và hạn chế. Đề xuất hướng cải tiến và kế hoạch bảo trì để nâng cấp hệ thống trong tương lai.

CHƯƠNG 1. PHÂN TÍCH VÀ ĐẶC TẢ PHẦN MỀM

1.1 Mô tả bài toán

“Ứng dụng quản lý bán sách trên nền web” là kết quả của việc ứng dụng công nghệ và áp dụng tin học hoá để quản lý một cửa hàng sách truyền thống. Để ứng dụng được hoàn chỉnh, quy trình nghiệp vụ trong việc mua bán tại cửa hàng truyền thống được tìm hiểu kỹ càng, từ đó sẽ triển khai các nghiệp vụ đó thông qua việc áp dụng công nghệ.

Ứng dụng sẽ được sử dụng cho các đối tượng người dùng khác nhau:

- Đối với Người dùng, là những người không có tài khoản mua hàng trên hệ thống, họ có thể thực hiện đăng ký tại trang web và trở thành Khách hàng của hệ thống.
- Đối với Khách hàng, là người đã có tài khoản trên hệ thống, có thể thực hiện các việc tương tác với hệ thống.
- Đối với Quản trị viên, là người có quyền quản lý toàn bộ thông tin, bao gồm cả thông tin Khách hàng, thông tin về cửa hàng đó.

Ứng dụng được phát triển dựa trên nền tảng Web - nền tảng này phù hợp cho việc triển khai một trang thương mại điện tử đầy đủ chức năng, nó cũng tiện lợi cho những người sử dụng Mobile vì có thể triển khai một giao diện dành cho người dùng Mobile.

1.2 Bảng mô tả thuật ngữ

STT	Thuật ngữ	Mô tả
1	REST API	Giao diện lập trình ứng dụng dùng để kết nối và giao tiếp giữa server và client.
2	Spring Boot	Framework giúp phát triển ứng dụng Java đơn giản, hỗ trợ tạo các ứng dụng web và microservices.
3	ReactJS	Thư viện JavaScript để xây dựng giao diện người dùng (UI) động và hiệu quả.

4	MySQL	Hệ quản trị cơ sở dữ liệu quan hệ mã nguồn mở, dùng để lưu trữ và quản lý dữ liệu ứng dụng.
5	JWT (JSON Web Token)	Phương thức xác thực người dùng bằng cách mã hóa thông tin xác thực trong các token.
6	Heroku	Nền tảng đám mây hỗ trợ triển khai, quản lý và mở rộng ứng dụng web.
7	JPA (Java Persistence API)	Giao diện API trong Java dùng để quản lý và lưu trữ dữ liệu trong cơ sở dữ liệu quan hệ.
8	WebFlux	Thành phần của Spring Framework hỗ trợ lập trình phản ứng không đồng bộ, phù hợp với REST API.
9	Cloudinary	Dịch vụ lưu trữ và quản lý hình ảnh/video trên cloud, tích hợp với Spring Boot qua API.
10	JSON	Định dạng dữ liệu phổ biến dùng để trao đổi thông tin giữa client và server trong ứng dụng web.
11	Momo	Một nền tảng cung cấp dịch vụ thanh toán trực tuyến tự động.
12	Sách	Sản phẩm chính được bán trên trang web.
13	Người dùng (User)	Người sử dụng hệ thống, có thể là khách hàng hoặc quản trị viên
14	Đơn hàng (Order)	Giao dịch mà người dùng thực hiện khi mua sách, bao gồm các thông tin về sản phẩm và thanh toán
16	Quản trị viên (Admin)	Người có quyền quản lý hệ thống thêm sửa xoá bất kỳ các thông tin gì

1.3 Yêu cầu bài toán

1.3.1 Chuẩn bị dữ liệu cho ứng dụng

Để bắt đầu xây dựng hệ thống, thì việc đầu tiên cần làm là chuẩn bị đầy đủ các thông tin về mặt hàng cần bán (ở đây là những cuốn sách), dữ liệu cần thiết để duy trì hệ thống. Các dữ liệu cần chuẩn bị bao gồm:

- Danh sách thông tin về các tác giả
- Danh sách thông tin về các nhà xuất bản
- Danh sách thông tin về sách và giá tiền
- Quản lý quyền truy cập giới hạn các chức năng, phân quyền

1.3.2 Thực hiện quy trình mua bán tại ứng dụng web

Chức năng chính của phần mềm là quản lý bán sách và cung cấp các giải pháp mua bán online tiện lợi mà không cần phải đến trực tiếp cửa hàng:

- Người dùng truy cập trang web, tại đây họ có thể xem toàn bộ mặt hàng trong cửa hàng, thực hiện tìm kiếm những cuốn sách yêu thích, để có thể thực hiện việc mua, Người dùng có thể đăng ký bằng cách cung cấp đầy đủ thông tin yêu cầu và thực hiện đăng ký tại web và trở thành Khách hàng.
- Khách hàng có thể thực hiện việc thêm món hàng vào giỏ hàng, sau đó thực hiện thanh toán qua 2 hình thức chính: Thanh toán khi nhận hàng và Thanh toán online. Khách hàng có thể kiểm tra đơn hàng của mình, thay đổi thông tin cá nhân cần thiết.
- Quản trị viên, khi Khách hàng có đã thực hiện thanh toán đơn hàng, Quản trị viên nhận yêu cầu và thực hiện chuẩn bị hàng và duyệt đơn hàng. Trong trường hợp đơn không thể chuẩn bị, Quản trị viên có quyền huỷ đơn hàng đó và lý do huỷ sẽ được cập nhật tới Khách hàng. Ngoài ra quản trị viên có thể kiểm soát toàn bộ hoạt động của hệ thống như cập nhật thông tin mặt hàng, xem báo cáo thống kê,...

1.3.3 Báo cáo thống kê

Hệ thống báo cáo thống kê được thiết kế nhằm cung cấp cái nhìn toàn diện về hoạt động mua bán trên website bán sách. Cụ thể, hệ thống sẽ tự động tổng hợp và phân tích dữ liệu liên quan đến số lượng đơn hàng, các mặt hàng nổi bật, và xu hướng mua sắm của khách hàng trong ngày. Dưới đây là mô tả chi tiết về các chức năng của hệ thống báo cáo:

Số lượng đơn hàng trong ngày:

- Hệ thống sẽ thu thập và tổng hợp số lượng đơn hàng được đặt trong ngày, bao gồm tất cả các đơn hàng đã được thanh toán hoặc đang chờ xử lý.
- Dữ liệu về số lượng đơn hàng này sẽ được thể hiện dưới dạng biểu đồ hoặc bảng thống kê, giúp người quản trị dễ dàng theo dõi tình hình hoạt động kinh doanh trong thời gian thực.
- Hệ thống cũng có khả năng phân tích sự thay đổi về số lượng đơn hàng qua từng khung giờ trong ngày, giúp xác định những khoảng thời gian cao điểm mà người dùng có xu hướng mua sắm nhiều.

Những mặt hàng nổi bật (nhiều lượt mua):

- Hệ thống sẽ tự động xác định các cuốn sách được mua nhiều nhất trong ngày, từ đó xếp hạng các mặt hàng nổi bật.
- Thông tin chi tiết bao gồm: tên sách, tác giả, thể loại, giá bán, và số lượng bán ra sẽ được liệt kê cụ thể. Điều này giúp nhà quản lý nắm bắt được thị hiếu của khách hàng và có thể điều chỉnh các chiến lược bán hàng, quảng cáo sao cho hiệu quả hơn.
- Ngoài ra, hệ thống có thể cung cấp báo cáo so sánh giữa các ngày hoặc các tuần, cho thấy xu hướng tăng giảm của những sản phẩm nổi bật.

1.4 Yêu cầu nghiệp vụ

1.4.1 Mô tả nghiệp vụ

Để thực hiện công việc quản lý quy trình hoạt động của một cửa hàng sách cần thực hiện theo đầy đủ quy trình sau:

- Khi thực hiện đăng ký Người dùng có thể đăng ký trực tiếp tại trang web với thông tin yêu cầu.
- Sau khi đăng ký thành công họ có thể thực hiện công việc mua sách trên trang web và quản lý thông tin cá nhân của chính mình, quản lý đơn hàng và chi tiết đơn hàng đã mua.
- Khi thanh toán có thể thanh toán theo 2 hình thức: Thanh toán khi nhận hàng và Thanh toán online sử dụng merchant Momo.
- Khi thanh toán thành công đơn hàng khách mua có thể xem lại ở phần quản lý đơn hàng cá nhân, khi người bán chưa chuẩn bị hàng và chưa cập nhật trạng thái thì khách hàng có quyền huỷ đơn hàng đó.
- Quản trị viên:
 - o Khi có đơn hàng từ Khách hàng thì thực hiện chuẩn bị hàng và duyệt đơn, trong trường hợp quá tải đơn (đơn hàng nhiều, không đủ mặt hàng đóng hàng) thì có thể thực hiện huỷ đơn hàng với lý do cụ thể.
 - o Kiểm soát trạng thái đơn hàng, biết được khi nào đơn hàng đến tay người dùng.
 - o Quản lý thông tin sách.
 - o Quản lý thông tin khách hàng.
 - o Quản lý đơn hàng.

1.4.2 Các yêu cầu chức năng

- ❖ BR1: Nhóm chức năng chuẩn bị dữ liệu ban đầu:

Hệ thống sẽ đảm bảo khả năng quản lý (hiển thị, thêm, sửa, xoá) các dữ liệu về thông tin tài khoản người dùng, thông tin cá nhân, thông tin sách và các thông tin liên quan.

- BR1.1 Đăng nhập

Để có thể mua hàng thì hệ thống yêu cầu Người dùng phải đăng nhập để có thể thêm giỏ hàng, thực hiện thanh toán và mua hàng, với những Người dùng không đăng nhập vẫn có thể xem các mặt hàng trên trang web.

- BR1.2 Đăng xuất

Hệ thống cho phép người dùng có thể đăng xuất ra khỏi trang web hoặc khi hết phiên đăng nhập. Chức năng đảm bảo có thể thực hiện được việc đăng xuất.

- BR1.3 Quản lý tài khoản

Hệ thống cho phép Quản trị viên quản lý thông tin tài khoản của người dùng. Chức năng phải đảm bảo hiển thị đầy đủ thông tin của người dùng (ngoại trừ mật khẩu) để có thể thực hiện các thao tác với thông tin.

- BR1.4 Quản lý thông tin mặt hàng (sách)

Hệ thống cho phép Quản trị viên thực hiện việc thêm sách mới lên trang web, các thông tin liên quan.

- BR1.5 Quản lý đơn hàng

Hệ thống cho phép Quản trị viên xem các đơn hàng đã được đặt, thực hiện cập nhật trạng thái của đơn hàng của toàn bộ hệ thống

Cho phép Khách hàng quản lý xem thông tin đơn hàng mà Khách hàng đã đặt.

- BR1.6 Quản lý thông tin cá nhân

Hệ thống cho phép Khách hàng có thể thực hiện quản lý thông tin cá nhân của chính mình, cập nhật địa chỉ, thông tin liên lạc,...

- BR1.7 Quản lý thể loại

Hệ thống cho phép Quản trị viên sẽ thực hiện thêm, sửa, xoá các thể loại mới.

- BR 1.8 Đăng kí

Hệ thống cho phép Người dùng thực hiện đăng kí khi Người dùng chưa có tài khoản nằm trong hệ thống.

- BR 1.9 Báo cáo thống kê

Hệ thống cho phép Quản trị viên thực hiện xem báo cáo thống kê hiệu suất của toàn bộ hệ thống

❖ BR2: Nhóm chức năng thực hiện mua hàng

Hệ thống đảm bảo quyền lợi của Khách hàng khi thực hiện mua hàng tại hệ thống, cần được chính xác, tin cậy và bảo mật.

- BR2.1 Thực hiện thanh toán

Hệ thống cho phép Khách hàng thanh toán linh hoạt, chính xác và an toàn.

- BR2.2 Tìm kiếm sản phẩm trong trang web

Hệ thống cho phép người dùng thực hiện tìm kiếm các sản phẩm theo nhu cầu: giá cả, nhà xuất bản, tựa đề,...

- BR2.3 Đánh giá sản phẩm

Hệ thống cho phép Khách hàng khi đã mua và đã nhận sản phẩm đó có thể thực hiện đánh giá và góp ý cho sản phẩm tại trang web.

- BR2.4 Quản lý giỏ hàng

Hệ thống cho phép khách hàng thêm sản phẩm dự kiến sẽ mua lưu trữ trong giỏ hàng online của trang web, khi người dùng đăng nhập đúng tài khoản của mình sẽ xem được trong giỏ hàng cá nhân của mình

1.4.3. Các yêu cầu phi chức năng

a. Về hiệu năng

Tối ưu hoá truy vấn cơ sở dữ liệu: Ứng dụng sử dụng JPA và một số interface khác để quản lý và tương tác với cơ sở dữ liệu. Để tránh các vấn đề ảnh hưởng hiệu năng như N+1 query, các câu truy vấn gây chậm hệ thống.

Sử dụng bộ nhớ đệm: Kết hợp việc caching lưu trữ các dữ liệu tạm thời và không thay đổi thường xuyên như việc xem sách và chi tiết về cuốn sách, từ đó cải thiện hiệu năng và truy xuất, giảm tải áp lực lên cơ sở dữ liệu.

Cân bằng tải và scaling: Ứng dụng được triển khai trên nền tảng heroku có khả năng mở rộng theo chiều ngang, tức là có thể tăng số lượng dynos khi có lượng truy cập quá cáo, giúp đảm bảo hiệu năng trong mọi tình huống tải lớn, đặc biệt là những dịp có flash sale lớn.

b. Về tốc độ

Tối ưu hoá giao diện người dùng: Ứng dụng sử dụng ReactJs để xây dựng giao diện người dùng, ảnh hưởng trực tiếp đến trải nghiệm của khách hàng. Sử dụng các kĩ thuật như lazy loading tối ưu khả năng tải trang, chỉ tải những thành phần cần thiết khi người dùng trải nghiệm trang web.

Sử dụng CDN: Hình ảnh và các tài nguyên lĩnh vực khác được lưu trữ lên dịch vụ bên thứ 3 như Cloudinary và phân phối qua CDN để giảm thời gian phản hồi và tải trang cho người dùng ở các khu vực khác nhau.

c. Về bảo mật

Xác thực và phân quyền người dùng: Ứng dụng sử dụng Spring Securiy và kết hợp với JWT để thực hiện xác thực và phân quyền của người dùng, người dùng chỉ có thể truy cập các tài nguyên phù hợp với quyền của mình.

Bảo mật thanh toán: Tích hợp các dịch vụ thanh toán Momo và thẻ tín dụng với việc sử dụng HTTPS và mã hoá thông tin thanh toán, đảm bảo rằng mọi giao dịch trực tuyến trên không gian mạng diễn ra một cách an toàn và tuân thủ các tiêu chuẩn bảo mật.

Chính sách bảo mật: Mọi thông tin cá nhân và giao dịch của người dùng cần được mã hoá và lưu trữ cẩn thận trong cơ sở dữ liệu, tuân thủ quy định về bảo mật dữ liệu cá nhân.

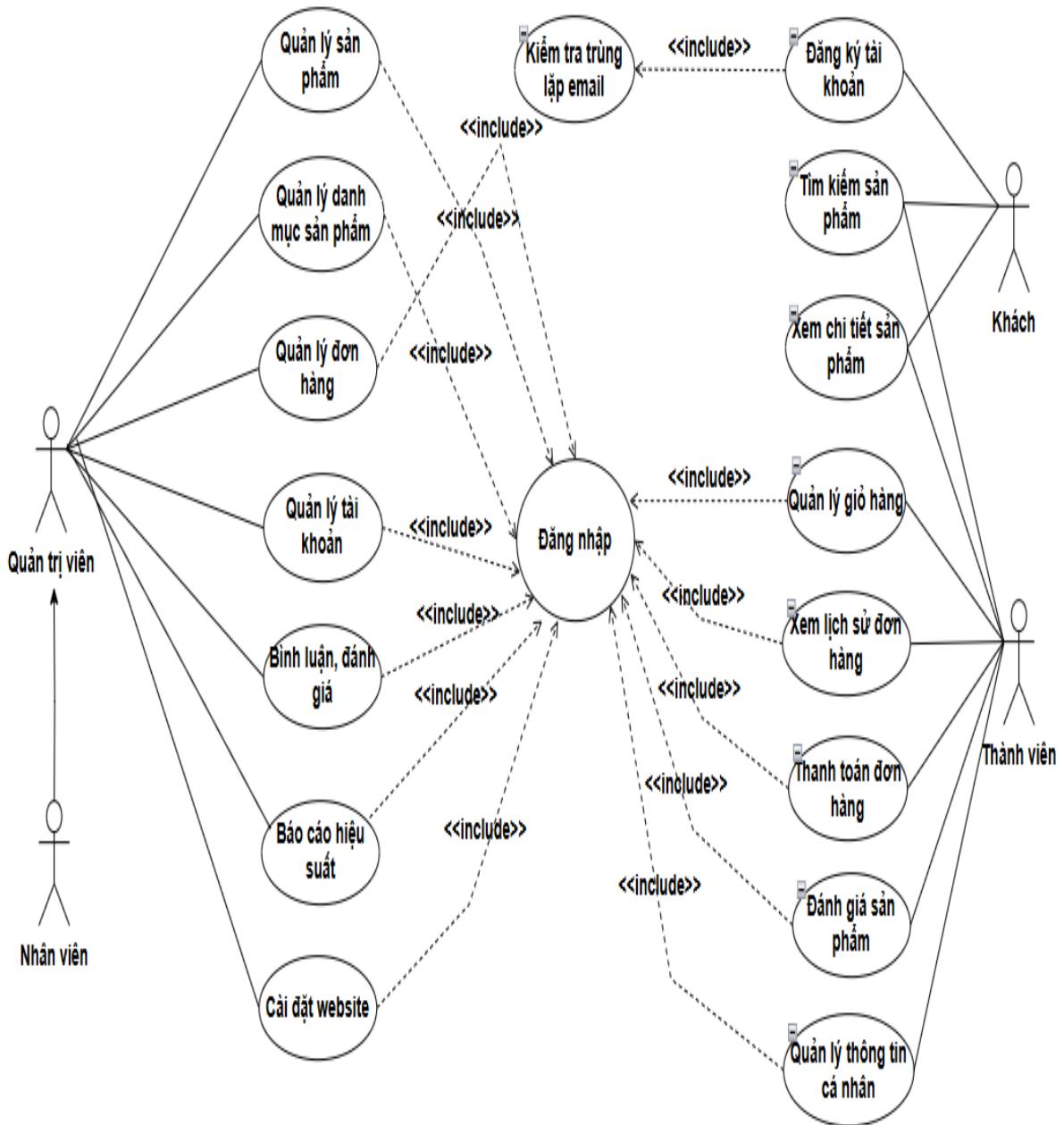
d. **Tính ổn định**

Giám sát và logging: Sử dụng các công cụ hiệu quả như Heroku Metrics và tích hợp Spring Actuator theo dõi trạng thái hệ thống. Ngoài ra, các logs được lưu trữ và phân tích để phát hiện sớm các vấn đề và đưa ra giải pháp kịp thời.

Cơ chế sao lưu và khôi phục: Dữ liệu quan trọng được sao lưu định kỳ và có khả năng khôi phục nhanh chóng khi gặp sự cố, đảm bảo tính liên tục của hệ thống.

1.5 Sơ đồ Use-case chức năng chính

1.5.1 Sơ đồ Use-case tổng quát



Hình 1 UC Tổng quát

1.5.2 Các tác nhân trong hệ thống

STT	Tên tác nhân	Mô tả
1	Người dùng	Là truy cập trang web bao gồm cả Khách hàng, Quản trị viên
2	Khách hàng	Là người thực hiện đăng nhập trên trang thương mại để thực hiện tìm kiếm và mua sản phẩm
3	Quản trị viên	Là người có vai trò lớn nhất trong hệ thống, quản lý mọi hoạt động trong hệ thống

1.5.3 Các Use-case

- ❖ Nhóm 1: Nhóm chức năng khởi tạo dữ liệu
 - UC 1.1 Đăng nhập

Chức năng này cho phép tất cả các đối tượng Người dùng có đăng nhập vào hệ thống bằng cách điền đầy đủ thông tin yêu cầu về tên đăng nhập và mật khẩu trong giao diện của hệ thống.

- UC 1.2 Đăng xuất

Chức năng này cho phép tất cả các đối tượng có thể đăng xuất khi muốn thoát ra khỏi hệ thống hoặc khi hệ thống đã biết hết phiên làm việc. Các đối tượng có thể dùng ở đây là toàn bộ những tác nhân đã đăng nhập vào hệ thống.

- UC 1.3 Quản lý tài khoản

Hệ thống cho phép Quản trị viên có thể quản lý thông tin tài khoản mà Người dùng đã đăng ký trên hệ thống.

- UC 1.4 Quản lý thông tin mặt hàng (sách)

Hệ thống cho phép Quản trị viên có thể quản lý thông tin của các mặt hàng (sách), bao gồm các công việc thêm, sửa, xoá, tạo một số báo cáo.

- UC 1.5 Quản lý đơn hàng

Hệ thống cho phép Quản trị viên quản lý đơn hàng mà Khách hàng đã đặt, thực hiện cập nhật trạng thái đơn hàng trên toàn bộ hệ thống.

Ngoài ra, Khách hàng còn có thể quản lý đơn hàng của cá nhân mình.

- UC1.6 Quản lý thông tin cá nhân

Hệ thống cho phép Khách hàng cập nhật lại thông tin cá nhân của chính mình, thông tin về địa chỉ, thông tin liên lạc,...

- UC1.7 Quản lý thẻ loại

Hệ thống cho phép Quản trị viên thêm, sửa, xoá thẻ loại, mô tả chi tiết về thẻ loại đó để thêm sách thuộc thẻ loại nào và thẻ loại nào.

- UC 1.8 Đăng ký

Hệ thống cho phép Người dùng thực hiện đăng ký tài khoản mới khi Người dùng chưa có tài khoản.

- UC 1.9 Báo cáo thống kê

Hệ thống cho phép Quản trị viên xem báo cáo thống kê hiệu suất của toàn bộ hệ thống.

- ❖ Nhóm 2: Nhóm các chức năng thực hiện mua hàng
 - UC 2.1 Thực hiện thanh toán

Khi khách hàng tiến hành thanh toán, hệ thống có 2 phương thức thanh toán linh hoạt: Thanh toán khi nhận hàng và thanh toán online thông qua merchant của Momo.

- UC 2.2 Tìm kiếm sản phẩm

Hệ thống cho phép Người dùng thực hiện tìm kiếm các sản phẩm theo nhu cầu: giá cả, nhà xuất bản, tựa đề,..

- UC 2.3 Đánh giá sản phẩm

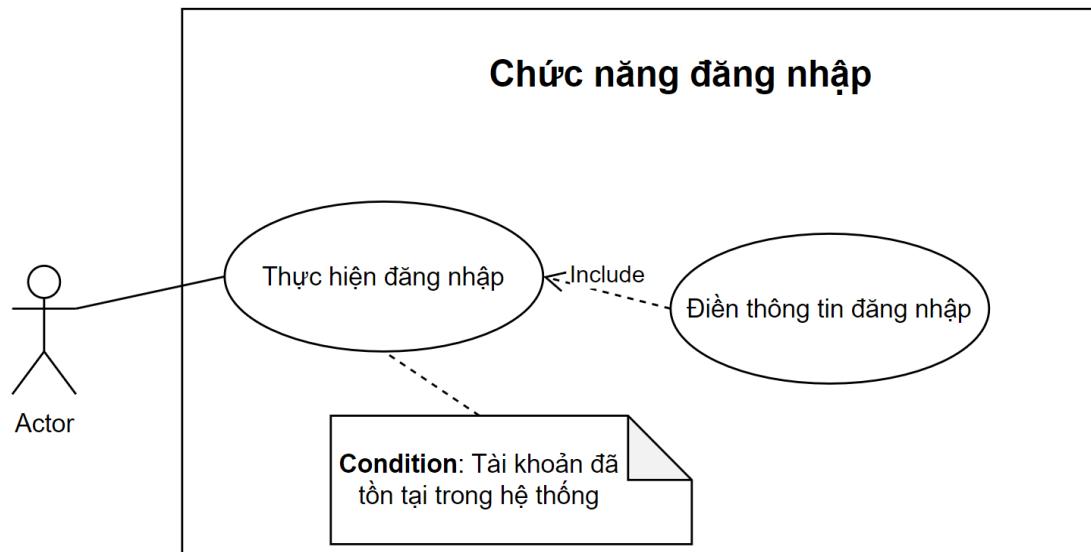
Khi Khách hàng đã thực hiện mua hàng thành công và đã nhận hàng thì có thể quay lại sản phẩm đó đã mua và thực hiện viết đánh giá sản phẩm

- UC 2.4 Quản lý giỏ hàng

Hệ thống cho phép Khách hàng thêm sản phẩm dự kiến sẽ mua lưu trữ trong giỏ hàng online của trang web, khi người dùng đăng nhập vào tài khoản của mình có thể xem lại phần giỏ hàng này.

UC	Mô tả	BR
Nhóm chức năng chuẩn bị dữ liệu ban đầu		
BR 1.1	Đăng nhập	UC 1.1
BR 1.2	Đăng xuất	UC 1.2
BR 1.3	Quản lý tài khoản	UC 1.3
BR 1.4	Quản lý thông tin mặt hàng	UC 1.4
BR 1.5	Quản lý đơn hàng	UC 1.5
BR 1.6	Quản lý thông tin cá nhân	UC 1.6
BR 1.7	Quản lý thể loại	UC 1.7
BR 1.8	Đăng ký	UC 1.8
BR 1.9	Báo cáo thống kê	UC 1.9
Nhóm chức năng thực hiện mua hàng		
BR 2.1	Thực hiện thanh toán	UC 2.1
BR 2.2	Tìm kiếm sản phẩm	UC 2.2
BR 2.3	Đánh giá sản phẩm	UC 2.3
BR 2.4	Quản lý giỏ hàng	UC 2.4

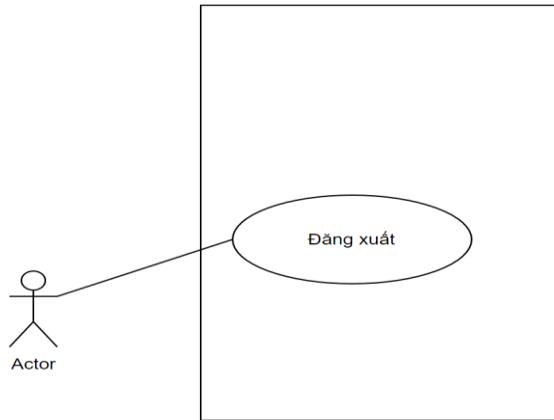
UC 1.1 – Đăng nhập



Hình 2 UC Đăng nhập

Tên use-case	Đăng nhập
Tác nhân	Người dùng
Mô tả	Tác nhân thực hiện việc đăng nhập để thực hiện một số chức năng yêu cầu đăng nhập
Sự kiện kích hoạt	Click vào “Đăng nhập”
Tiền điều kiện	Tác nhân chưa đăng nhập
Luồng sự kiện chính	- ĐIỀN THÔNG TIN YÊU CẦU VÀ THỰC HIỆN ĐĂNG NHẬP
Luồng sự kiện thay thế	- Hệ thống kiểm tra tài khoản có tồn tại trong hệ thống không, nếu không tồn tại hệ thống sẽ hiển thị thông báo lỗi tương ứng
Hậu điều kiện	- Tác nhân đã đăng nhập thành công

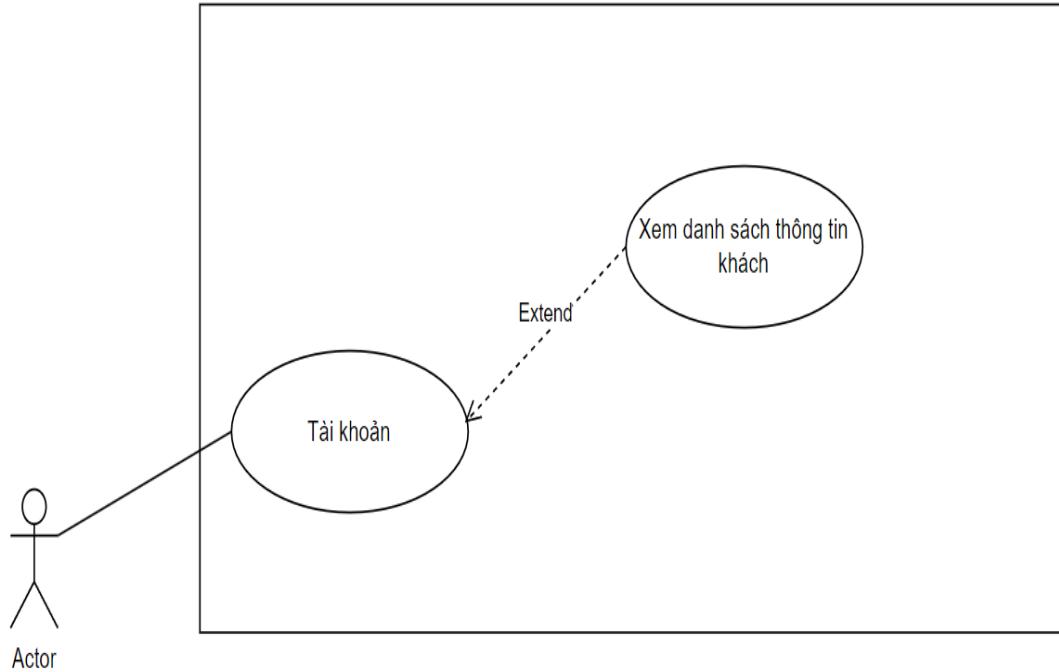
UC1.2 – Đăng xuất



Hình 3 UC Đăng xuất

Tên use-case	Đăng xuất
Tác nhân	Người dùng
Mô tả	Tác nhân thực hiện việc đăng xuất khỏi hệ thống để kết thúc phiên làm việc.
Sự kiện kích hoạt	Click vào “Đăng xuất”
Tiền kiện	Tác nhân đã đăng nhập vào hệ thống
Luồng sự kiện chính	<ol style="list-style-type: none"> Tác nhân click vào nút “Đăng xuất”. Hệ thống xác nhận hành động đăng xuất. Hệ thống tiến hành đăng xuất và xóa thông tin phiên làm việc của tác nhân.
Luồng sự kiện thay thế	- Nếu có lỗi trong quá trình đăng xuất (ví dụ: mất kết nối mạng), hệ thống sẽ hiển thị thông báo lỗi tương ứng và yêu cầu tác nhân thử lại.
Hậu kiện	Tác nhân đã đăng xuất thành công và không còn quyền truy cập vào các chức năng yêu cầu đăng nhập.

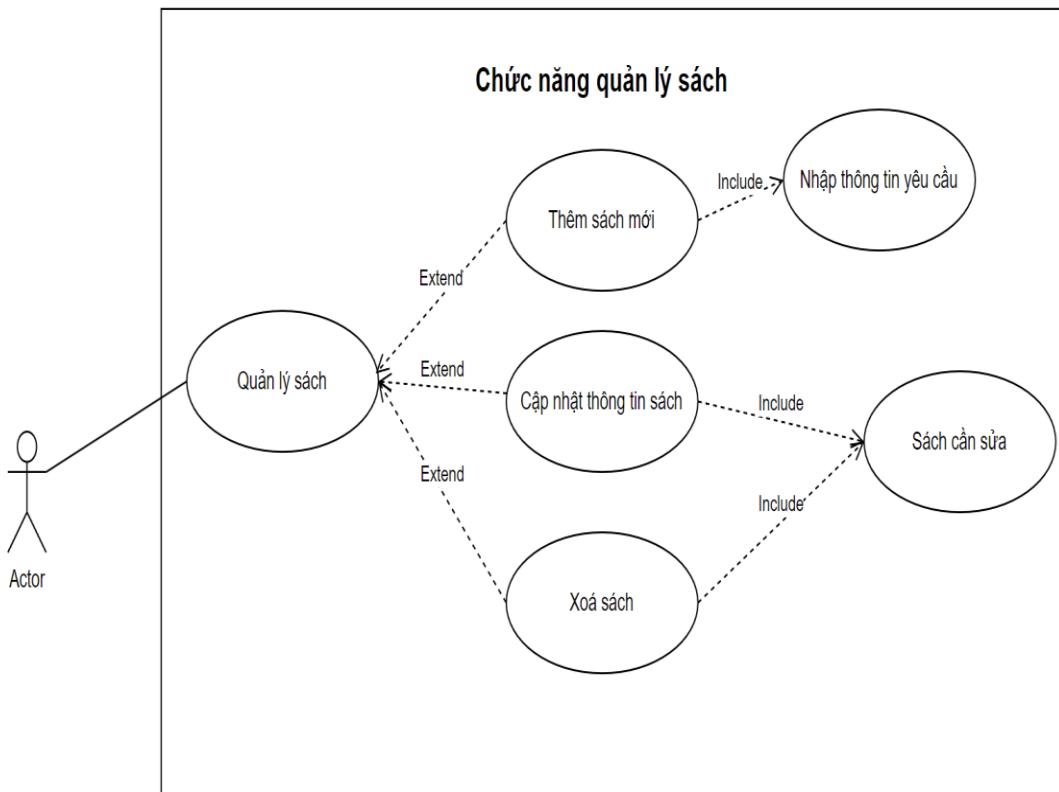
UC 1.3 – Quản lý tài khoản



Hình 4 UC Quản lý tài khoản

Tên use-case	Quản lý tài khoản
Tác nhân	Quản trị viên
Mô tả	Tác nhân thực hiện việc xem thông tin mà khách hàng đã đăng ký để mua hàng
Sự kiện kích hoạt	Click vào “Quản lý khách hàng”
Tiền điều kiện	Tác nhân đã đăng nhập vào hệ thống với vai trò Quản trị viên
Luồng sự kiện chính	<ol style="list-style-type: none"> Tác nhân click vào nút “Quản lý khách hàng”. Hệ thống hiển thị thông tin chi tiết.
Luồng sự kiện thay thế	
Hậu điều kiện	Tác nhân đã truy xuất thông tin thành công

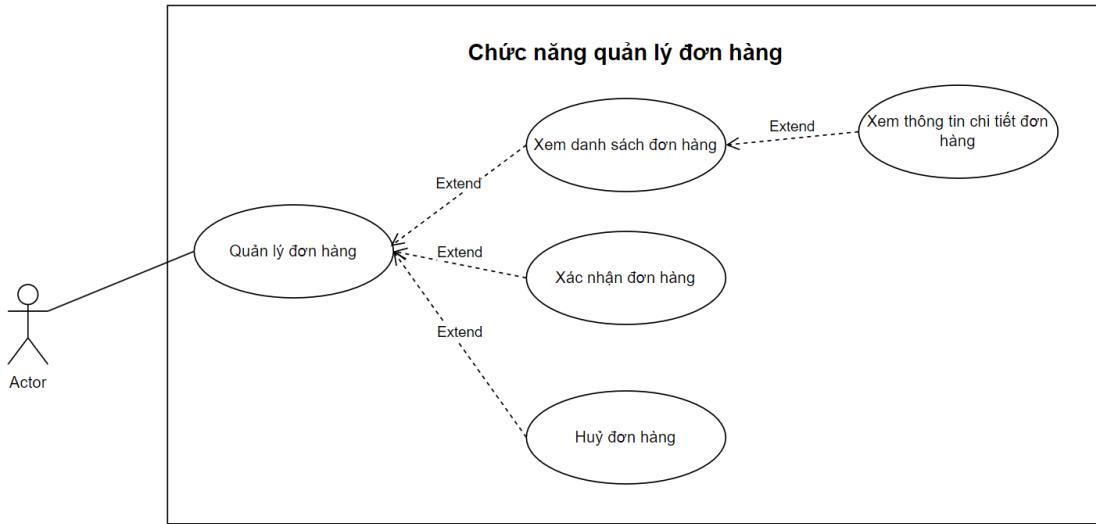
UC 1.4 – Quản lý sản phẩm (sách)



Hình 5 UC Quản lý sản phẩm

Tên use-case	Quản lý sách
Tác nhân	Quản trị viên của hệ thống
Mô tả	Tác nhân thực hiện thêm, sửa, xoá sách trong hệ thống bán hàng
Sự kiện kích hoạt	Click vào menu “Quản lý sách”
Tiền điều kiện	Tác nhân đã đăng nhập vào hệ thống với tài khoản có vài trò là quản trị viên
Luồng sự kiện thay thế	- Hệ thống kiểm tra tính hợp lệ của thông tin mà tác nhân điền vào (số, ngày tháng) nếu không đúng yêu cầu, điều kiện thì sẽ thông báo lỗi tương ứng
Hậu điều kiện	- Tác nhân đã thực hiện thêm, sửa, xoá thành công

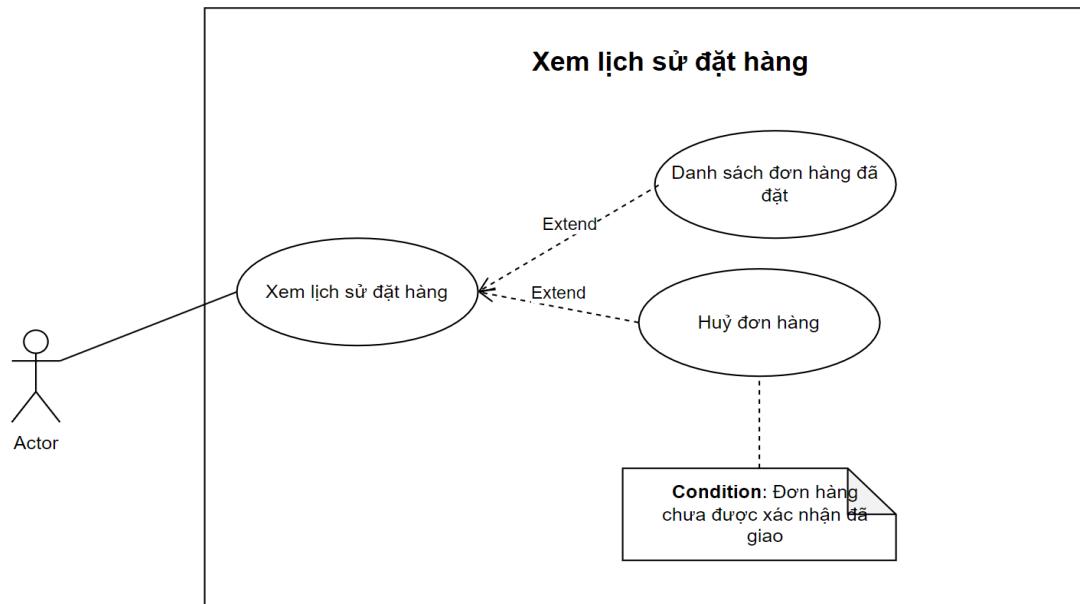
UC 1.5 – Quản lý đơn hàng



Hình 6 UC Quản lý đơn hàng

Tên use-case	Quản lý đơn hàng
Tác nhân	Quản trị viên hệ thống
Mô tả	Tác nhân thực hiện thao tác với đơn hàng: - Xem chi tiết đơn hàng của khách - Xác nhận đơn hàng - Huỷ đơn hàng
Sự kiện kích hoạt	Click vào “Quản lý đơn hàng”
Tiền điều kiện	Tác nhân đã đăng nhập vào hệ thống với tài khoản có vai trò là quản trị viên
Luồng sự kiện chính	- Xem chi tiết đơn hàng mà khách hàng đã đặt - Xác nhận đơn hàng - Huỷ đơn hàng
Hậu điều kiện	- Tác nhân đã thao tác với yêu cầu mua hàng của khách hàng thành công

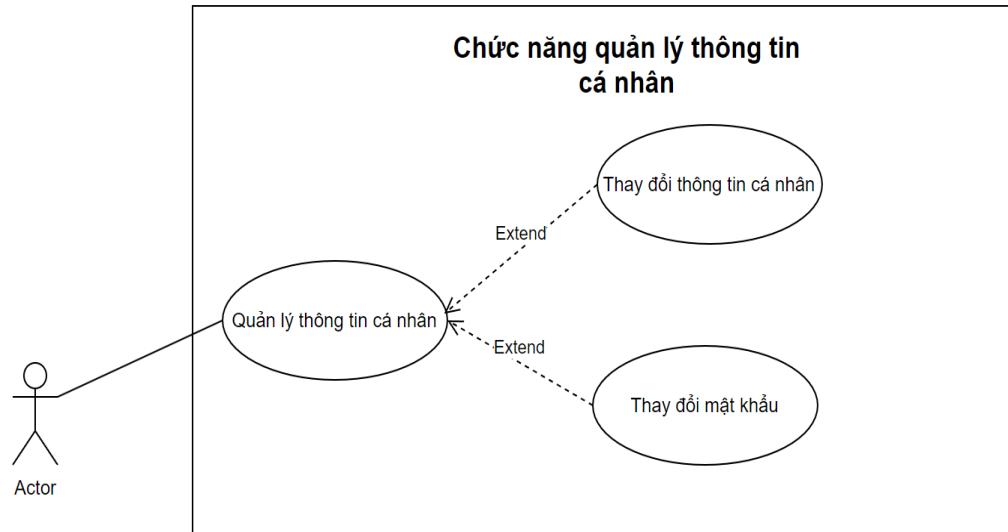
Quản lý đơn hàng cá nhân



Hình 7 UC Quản lý đơn hàng cá nhân

Tên use-case	Quản lý đơn hàng
Tác nhân	Khách hàng
Mô tả	Tác nhân thực hiện thao tác với đơn hàng: - Xem chi tiết đơn hàng của mình đã mua - Xác nhận đã nhận hàng - Huỷ đơn hàng
Sự kiện kích hoạt	Click vào “Đơn hàng của tôi”
Tiền điều kiện	Tác nhân đã đăng nhập vào hệ thống với tài khoản cá nhân của mình
Luồng sự kiện chính	- Xem chi tiết đơn hàng mà mình đã mua - Xác nhận đã nhận hàng - Huỷ đơn hàng
Hậu điều kiện	- Tác nhân đã thao tác với đơn hàng mình mua

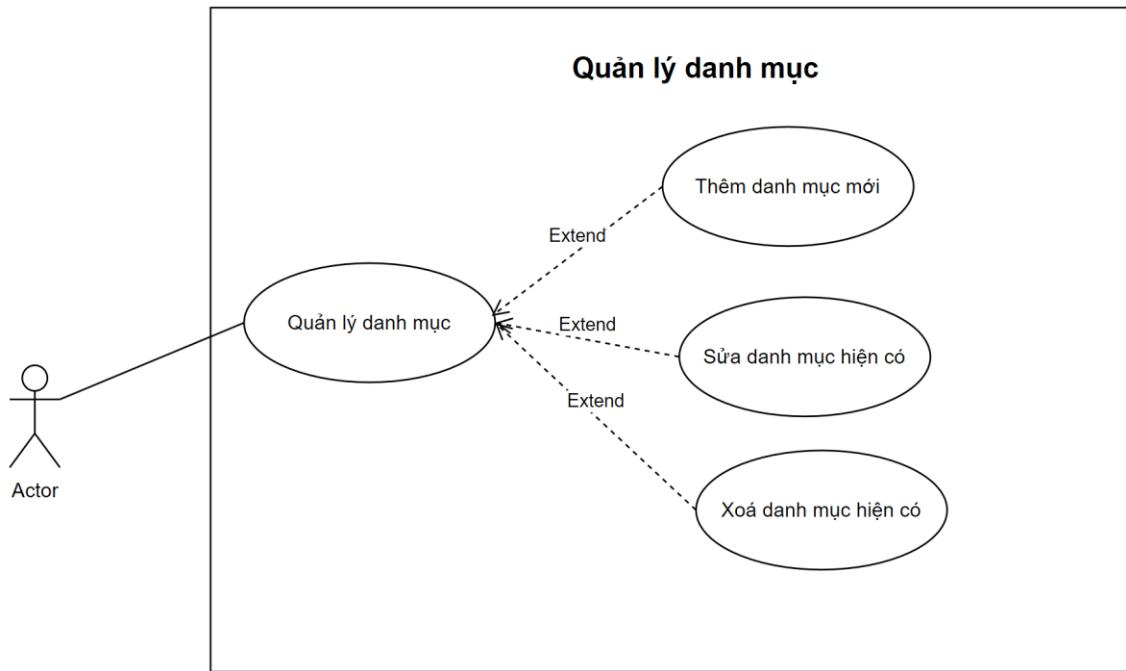
UC 1.6 – Quản lý thông tin cá nhân



Hình 8 UC Quản lý thông tin cá nhân

Tên use-case	Quản lý thông tin cá nhân
Tác nhân	Khách hàng
Mô tả	Tác nhân thực hiện việc cập nhật lại thông tin cần thiết
Sự kiện kích hoạt	Click avatar
Tiền điều kiện	Tác nhân đã đăng nhập với vai trò là khách hàng
Luồng sự kiện chính	- Điền thông tin mới cần sửa
Luồng sự kiện thay thế	- Hệ thống kiểm tra tính hợp lệ của dữ liệu (email, số điện thoại, ngày tháng) nếu không đúng định dạng sẽ hiển thị thông báo tương ứng
Hậu điều kiện	- Tác nhân đã thực hiện cập nhật thông tin thành công

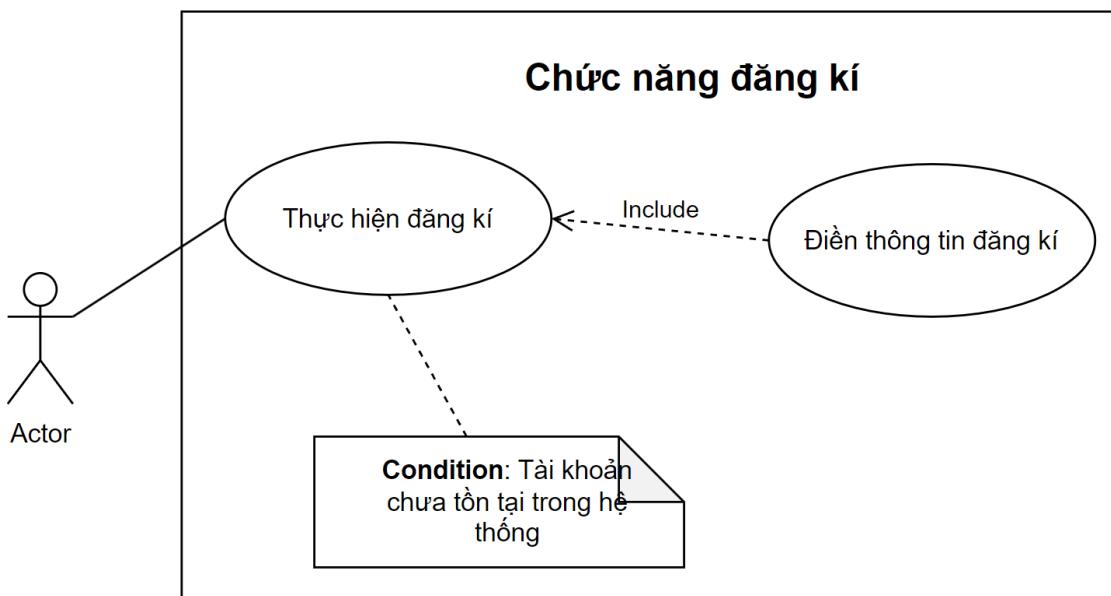
UC 1.7 – Quản lý thể loại



Hình 9 UC Quản lý thể loại

Tên use-case	Quản lý danh mục
Tác nhân	Quản trị viên của hệ thống
Mô tả	Tác nhân thực hiện thêm sửa xoá danh mục trong hệ thống
Sự kiện kích hoạt	Click vào menu “Quản danh mục”
Tiền điều kiện	Tác nhân đã đăng nhập vào hệ thống với tài khoản có vài trò là quản trị viên
Luồng sự kiện chính	<ul style="list-style-type: none"> - Đèn thông tin thể danh mục - Chọn danh mục hiện tại để cập nhật thông tin - Chọn danh mục hiện tại để xoá thông tin
Luồng sự kiện thay thế	<ul style="list-style-type: none"> - Hệ thống kiểm tra xem danh mục này đã tồn tại hay chưa, nếu đã tồn tại hiển thị thông báo tương ứng
Hậu điều kiện	<ul style="list-style-type: none"> - Tác nhân đã thực hiện thêm, sửa, xoá thành công

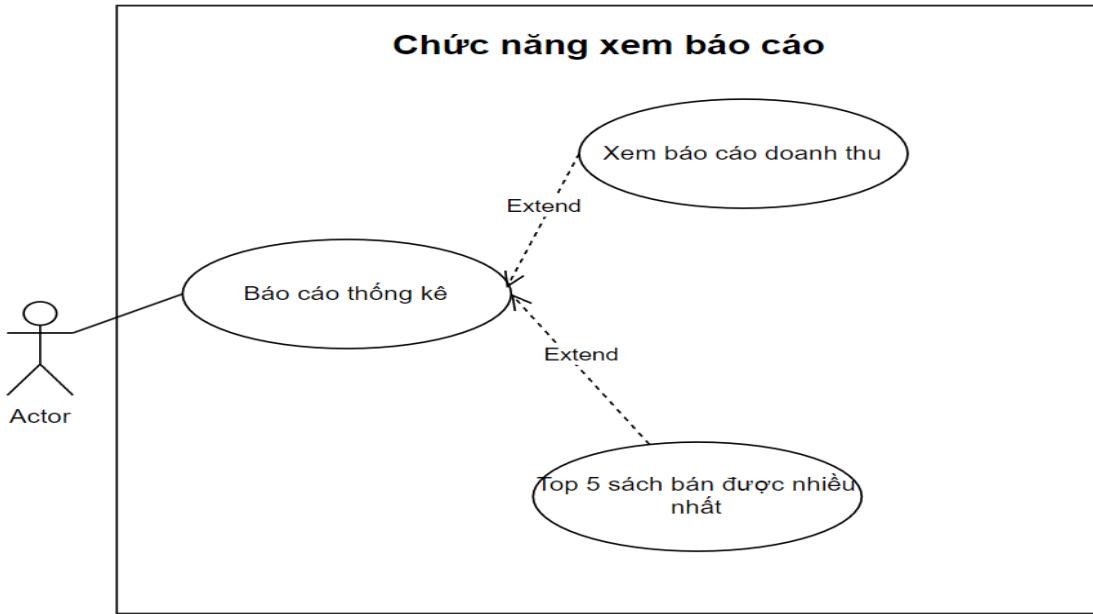
UC 1.8 – Đăng kí



Hình 10 UC Đăng kí

Tên use-case	Đăng kí
Tác nhân	Người dùng
Mô tả	Tác nhân thực hiện đăng kí tài khoản mới để sử dụng dịch vụ của hệ thống.
Sự kiện kích hoạt	Click vào “Đăng kí”
Tiền điều kiện	Tác nhân chưa có tài khoản và không đăng nhập
Luồng sự kiện chính	- Đèn thông tin yêu cầu và thực hiện đăng kí tài khoản mới
Luồng sự kiện thay thế	<ul style="list-style-type: none"> - Hệ thống kiểm tra xem tài khoản đã tồn tại hay chưa, nếu đã tồn tại hiển thị thông báo tương ứng - Kiểm tra định dạng dữ liệu (vd: email, số điện thoại) nếu không đúng định dạng sẽ báo lỗi tương ứng
Hậu điều kiện	- Tác nhân đã đăng nhập thành công

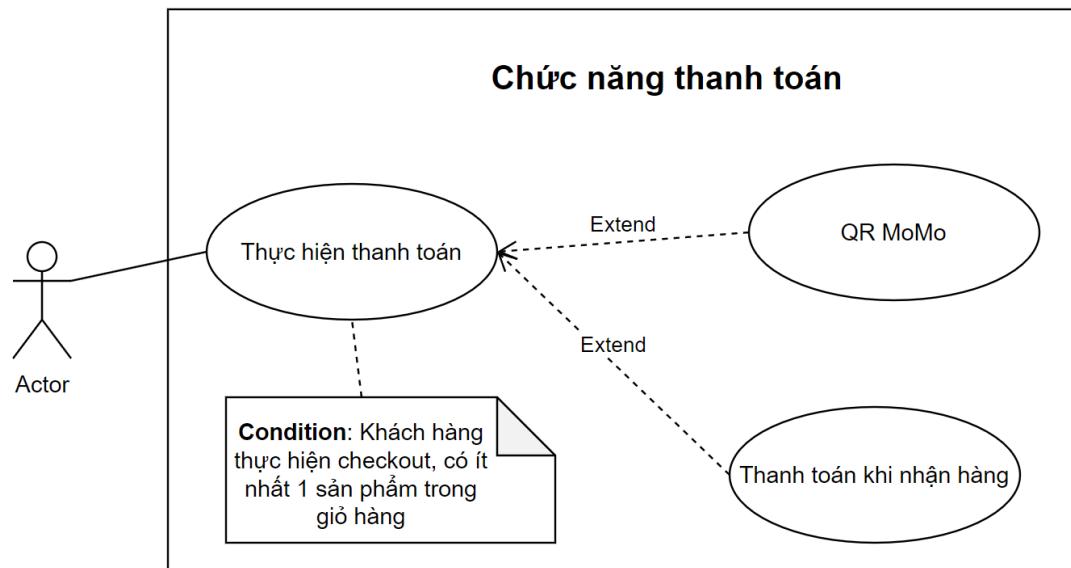
UC 1.9 – Báo cáo thống kê



Hình 11 UC Báo cáo

Tên use-case	Xem báo cáo hệ thống
Tác nhân	Quản trị viên hệ thống
Mô tả	Tác nhân xem báo cáo hiệu suất của cửa hàng: - Xem doanh thu - Xem số lượng đơn hàng trong ngày - Xem top những sách bán chạy
Sự kiện kích hoạt	Chọn các báo cáo cần xem
Tiền điều kiện	Tác nhân đã đăng nhập vào hệ thống với tài khoản có vài trò là quản trị viên
Luồng sự kiện chính	- Xem được doanh thu - Xem được top sách bán nhiều nhất - Xem được số lượng đơn hàng trong ngày
Hậu điều kiện	- Tác nhân theo dõi được báo cáo hiệu suất của cửa hàng

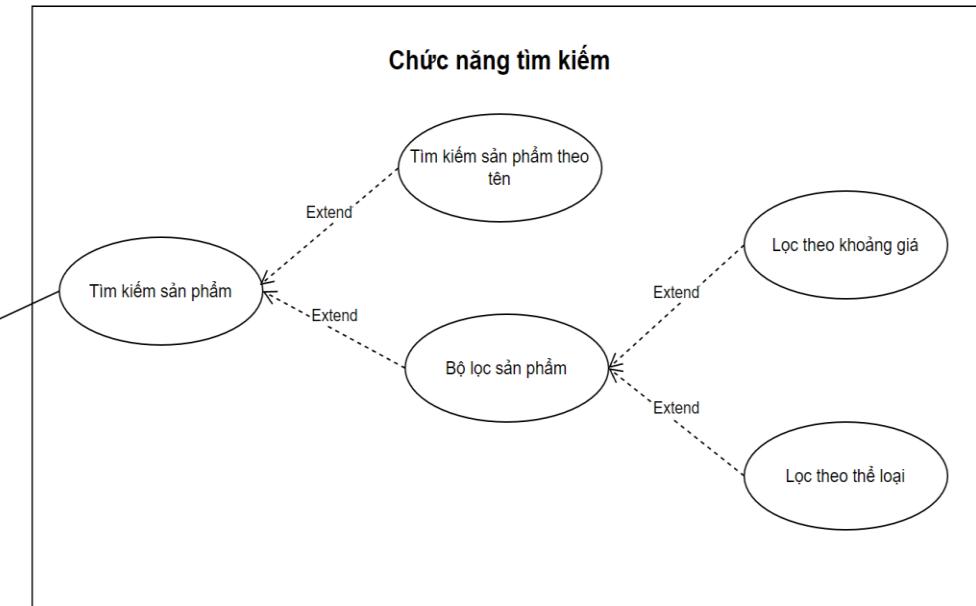
UC 2.1 – Mua hàng và thanh toán



Hình 12 UC Thanh toán

Tên use-case	Thực hiện thanh toán
Tác nhân	Khách hàng
Mô tả	Tác nhân thực hiện thanh toán cho đơn hàng của mình
Sự kiện kích hoạt	Click vào giỏ hàng và chọn thanh toán
Tiền điều kiện	Tác nhân đã đăng nhập với vai trò là khách hàng
Luồng sự kiện chính	- Chọn phương thức thanh toán phù hợp nhu cầu
Luồng sự kiện thay thế	- Hoàn tiền khi thanh toán thất bại khi phương thức thanh toán là Momo
Hậu điều kiện	- Tác nhân đã thanh toán thành công

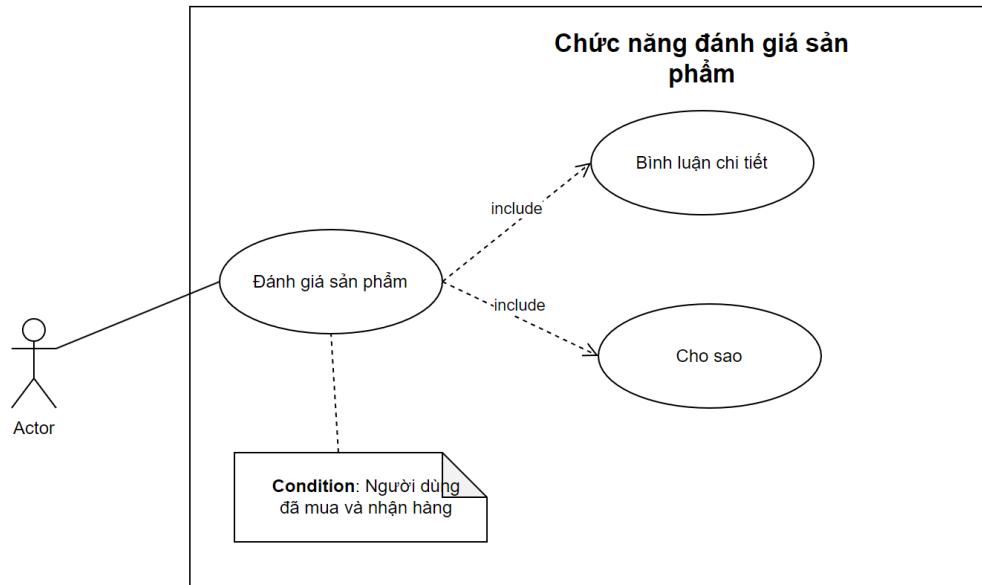
UC 2.2 – Tìm kiếm sản phẩm



Hình 13 UC Tìm kiếm sản phẩm

Tên use-case	Chức năng tìm kiếm
Tác nhân	Người dùng
Mô tả	Tác nhân thực hiện việc tìm kiếm sản phẩm theo nhu cầu cá nhân
Sự kiện kích hoạt	Click vào và điền điều kiện tìm kiếm trên thanh tìm kiếm của trang web
Tiền điều kiện	
Luồng sự kiện chính	- Điền thông tin cần tìm kiếm và lọc
Hậu điều kiện	- Tác nhân đã tìm kiếm và thấy được sản phẩm theo điều kiện đã điền

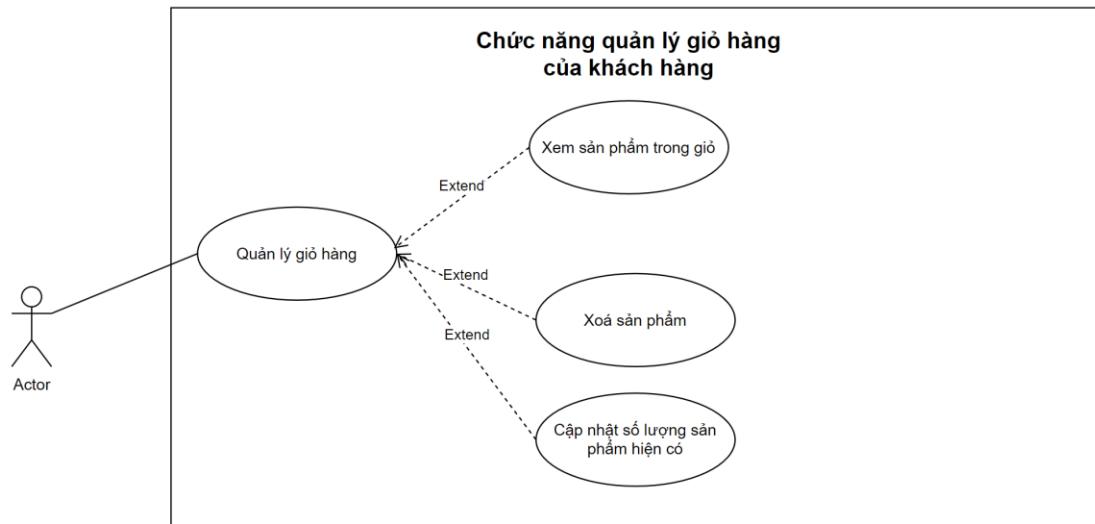
UC 2.3 – Đánh giá sản phẩm



Hình 14 UC Đánh giá sản phẩm

Tên use-case	Đánh giá sản phẩm
Tác nhân	Khách hàng
Mô tả	Tác nhân thực hiện đánh giá sản phẩm
Sự kiện kích hoạt	Click vào sản phẩm mình đã mua
Tiền điều kiện	Tác nhân đã đăng nhập vào hệ thống và đã nhận đơn hàng
Luồng sự kiện chính	<ul style="list-style-type: none"> - Thực hiện đánh giá sản phẩm - Cho sao
Luồng sự kiện thay thế	
Hậu điều kiện	<ul style="list-style-type: none"> - Tác nhân đã đánh giá sản phẩm thành công

UC 2.4 Quản lý giỏ hàng



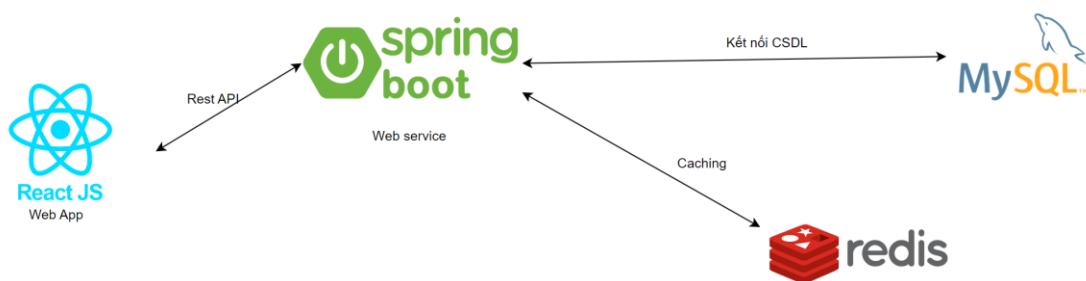
Hình 15: UC Quản lý giỏ hàng

Tên use-case	Quản lý giỏ hàng
Tác nhân	Khách hàng
Mô tả	Tác nhân thực hiện việc thao tác với giỏ hàng của mình
Sự kiện kích hoạt	Click vào biểu tượng của giỏ hàng
Tiền điều kiện	Tác nhân đã đăng nhập với vai trò là khách hàng
Luồng sự kiện chính	<ul style="list-style-type: none"> - Xem sản phẩm có trong giỏ hàng - Cập nhật số lượng của sản phẩm hiện có - Xoá sản phẩm ra khỏi giỏ hàng
Luồng sự kiện thay thế	<ul style="list-style-type: none"> - Hệ thống kiểm tra số lượng tăng thêm có > số lượng sản phẩm hiện có hay không, hệ thống sẽ ra thông báo tương ứng
Hậu điều kiện	<ul style="list-style-type: none"> - Tác nhân đã thao tác với giỏ hàng của mình thành công

CHƯƠNG 2: THIẾT KẾ HỆ THỐNG

2.1 Sơ đồ kiến trúc tổng thể

2.1.1 Sơ đồ



Hình 16 Sơ đồ kiến trúc tổng thể

2.1.2 Mô tả chi tiết

Sơ đồ kiến trúc tổng thể của hệ thống trong Hình 2 mô tả quy trình, công nghệ sử dụng để triển khai bài toán. Người dùng sẽ sử dụng chức năng của ứng dụng web, những chức năng đó sẽ gửi các request thông qua các rest api đến web service xử lý và trả lại kết quả tương ứng. Cuối cùng sẽ hiển thị kết quả đó cho người dùng thông qua giao diện trực quan.

2.1.3 Công nghệ sử dụng

- Spring boot là service trung tâm nơi thực hiện các yêu cầu từ phía client thông qua các RESTful API
- Cơ sở dữ liệu: MySQL lưu trữ toàn bộ dữ liệu của hệ thống
- Web client UI: Sử dụng ReactJS để xây dựng và phát triển giao diện người dùng.

2.2 Mô tả chi tiết về công nghệ sử dụng

2.2.1 Hệ quản trị cơ sở dữ liệu MySQL



Hình 17 MySQL

MySQL là một hệ quản trị cơ sở dữ liệu quan hệ (RDBMS – Relational Databases Management System) mã nguồn mở, sử dụng ngôn ngữ truy vấn SQL (Structured Query Language). Nó được phát triển bởi MySQL AB, sau đó đc Oracle mua lại vào năm 2010. Với khả năng xử lý dữ liệu mạnh mẽ, tốc độ nhanh và tính ổn định cao, MySQL đã trở thành một trong những hệ quản trị cơ sở dữ liệu phổ biến nhất trên thế giới. Nó được sử dụng rộng rãi trong các ứng dụng web, đặc biệt là trong các ứng dụng dựa trên kiến trúc LAMP (Linux, Apache, MySQL, PHP/Perl/Python).

Mã nguồn mở: MySQL là mã nguồn mở theo giấy phép GPL, người dùng có thể tùy chỉnh và sử dụng miễn phí. Ngoài ra, MySQL cũng có các phiên bản thương mại với các tính năng cao cấp hơn.

Khả năng mở rộng và hiệu suất cao: MySQL có thể xử lý một lượng lớn dữ liệu và người dùng đồng thời nhờ vào cơ chế tối ưu hóa bộ nhớ và công cụ lưu trữ. Nó cũng hỗ trợ các tính năng như nén bảng, phân vùng dữ liệu và chỉ mục.

Đa người dùng, đa phiên làm việc: MySQL cho phép nhiều người dùng kết nối và thực hiện các truy vấn đồng thời, với khả năng kiểm soát xung đột và đồng bộ hóa dữ liệu.

Bảo mật: Hệ quản trị này cung cấp các tính năng bảo mật mạnh mẽ như quản lý người dùng và quyền truy cập theo từng cấp độ (từ cơ sở dữ liệu, bảng, cột cho đến cấp độ hàng), mã hóa dữ liệu và hỗ trợ xác thực qua SSL.

Sao lưu và phục hồi: MySQL hỗ trợ nhiều phương pháp sao lưu (hot backup, cold backup) và phục hồi dữ liệu, giúp bảo vệ dữ liệu trong trường hợp xảy ra sự cố.

Đa nền tảng: MySQL chạy trên nhiều hệ điều hành khác nhau, bao gồm Linux, Windows, macOS và nhiều hệ điều hành khác.

MySQL hỗ trợ nhiều giao thức và công nghệ kết nối, bao gồm:

- ODBC (Open Database Connectivity): Chuẩn kết nối cho phép MySQL có thể được tích hợp với nhiều ngôn ngữ lập trình khác nhau như C++, Java, Python, và .NET.
- JDBC (Java Database Connectivity): Giao thức kết nối đặc biệt cho các ứng dụng Java.
- API đa ngôn ngữ: MySQL cung cấp nhiều API hỗ trợ các ngôn ngữ lập trình phổ biến như C, C++, Java, PHP, Python, Ruby, và Perl, giúp dễ dàng phát triển các ứng dụng web và desktop.

2.2.2 Spring boot (Web service)



Hình 18 Spring boot

Spring Boot là một phần mở rộng của Spring Framework, một nền tảng phát triển ứng dụng Java phổ biến. Được phát triển bởi Pivotal Software, Spring Boot ra đời nhằm mục tiêu đơn giản hóa quá trình xây dựng ứng dụng Java, đặc biệt là các ứng dụng web và web service. Nó giúp loại bỏ các cấu hình phức tạp và nặng nề của Spring Framework thông thường, đồng thời cung cấp các tính năng mạnh mẽ để phát triển nhanh chóng các ứng dụng với khả năng mở rộng cao.

Spring Boot được xây dựng trên nền tảng của Spring Framework, với các đặc điểm nổi bật như:

- Tự động cấu hình (Auto-Configuration): Spring Boot tự động cấu hình các thành phần cần thiết dựa trên các dependency có trong pom.xml (với Maven) hoặc build.gradle (với Gradle). Điều này giúp giảm thiểu việc lập trình viên phải cấu hình thủ công, tiết kiệm thời gian và tránh lỗi.
- Cấu trúc dự án đơn giản: Với Spring Boot, các ứng dụng có thể được cấu trúc theo mô hình tiêu chuẩn của Java, giúp quản lý mã nguồn dễ dàng hơn.
- Embedded Server: Spring Boot tích hợp sẵn các máy chủ như Tomcat, Jetty, hoặc Undertow, giúp lập trình viên có thể chạy ứng dụng như một ứng dụng Java độc lập mà không cần cài đặt thêm server bên ngoài.
- Hỗ trợ tạo RESTful Web Services: Spring Boot cung cấp các công cụ và annotation mạnh mẽ để phát triển các API RESTful một cách dễ dàng và nhanh chóng.

Một số tính năng nổi bật:

- Tự động cấu hình (Auto-Configuration): Spring Boot tự động phát hiện và cấu hình các thành phần như cơ sở dữ liệu, security, web server, mà không cần cấu hình tỉ mỉ. Nó giúp giảm bớt khối lượng cấu hình XML hoặc Java-based.
- Starter POMs: Các starter trong Spring Boot là các dependency đã được đóng gói sẵn, giúp bạn dễ dàng tích hợp các thư viện cần thiết mà không cần phải cài đặt từng thư viện riêng lẻ. Ví dụ: spring-boot-starter-web cho các ứng dụng web, spring-boot-starter-data-jpa cho cơ sở dữ liệu, v.v.
- Tích hợp dễ dàng với Spring Ecosystem: Spring Boot tận dụng toàn bộ các tính năng của hệ sinh thái Spring như Spring Data (truy cập cơ sở dữ liệu), Spring Security (bảo mật), Spring Batch (xử lý lô), Spring Cloud (microservices), v.v.

- Embedded Servers: Spring Boot tích hợp sẵn Tomcat, Jetty, và Undertow, cho phép ứng dụng được khởi chạy mà không cần triển khai trên máy chủ ứng dụng riêng biệt. Điều này hỗ trợ tốt cho việc phát triển và thử nghiệm nhanh chóng.
- Quản lý cấu hình tập trung: Spring Boot sử dụng tệp application.properties hoặc application.yml để quản lý cấu hình, giúp dễ dàng tùy chỉnh ứng dụng khi chuyển đổi giữa các môi trường (phát triển, kiểm thử, sản xuất).

2.2.3 ReactJs



Hình 19 ReactJs

ReactJS là một thư viện JavaScript mã nguồn mở dùng để xây dựng giao diện người dùng (UI), đặc biệt là trong các ứng dụng web đơn trang (Single Page Applications - SPA). Được phát triển bởi Facebook và ra mắt vào năm 2013, React đã nhanh chóng trở thành một trong những công nghệ phổ biến nhất trong lĩnh vực phát triển front-end, nhờ vào cách tiếp cận độc đáo với DOM (Document Object Model) và khả năng tái sử dụng các thành phần (component-based architecture).

Điểm nổi bật của React là nó cho phép phát triển giao diện người dùng một cách linh hoạt và hiệu quả, nhờ vào tính năng Virtual DOM và JSX (JavaScript XML). React không phải là một framework toàn diện như Angular hay Vue, mà nó tập trung vào việc xử lý giao diện người dùng, cho phép dễ dàng tích hợp với các thư viện và công nghệ khác.

ReactJS hoạt động dựa trên mô hình Component-Based Architecture, trong đó giao diện được chia thành các thành phần nhỏ và độc lập. Mỗi thành phần

(component) có thể tái sử dụng nhiều lần, giúp mã nguồn trở nên dễ bảo trì, mở rộng và tối ưu hóa.

Component (Thành phần)

Component là yếu tố cốt lõi trong React. Mỗi component đại diện cho một phần của giao diện người dùng (UI). Một component có thể là một nút bấm, một form nhập liệu, hoặc thậm chí toàn bộ trang web. Các component có thể được kết hợp với nhau để tạo thành giao diện phức tạp.

Mỗi component có thể chứa cả logic (trong JavaScript) và giao diện (dưới dạng JSX), giúp tách biệt rõ ràng giữa giao diện và chức năng.

JSX (JavaScript XML)

JSX là một cú pháp mở rộng của JavaScript, cho phép viết các phần tử UI dưới dạng mã XML trực tiếp trong JavaScript. JSX giúp mã React trở nên dễ đọc hơn, gần giống với HTML nhưng lại có sự linh hoạt và sức mạnh của JavaScript.

Virtual DOM (DOM ảo)

Virtual DOM là một bản sao nhẹ của DOM thật, được lưu trong bộ nhớ. Mỗi khi có sự thay đổi trong giao diện, React cập nhật Virtual DOM trước, so sánh với phiên bản trước đó, và chỉ áp dụng những thay đổi cần thiết lên DOM thật. Điều này giúp tăng hiệu suất và giảm thiểu số lần truy cập và cập nhật DOM, vốn là một tác vụ tốn kém tài nguyên.

State và Props

State: Mỗi component có thể lưu trữ trạng thái (state) của nó. State là dữ liệu động, có thể thay đổi trong quá trình hoạt động của ứng dụng. Khi state thay đổi, React sẽ tự động cập nhật và render lại giao diện.

Props: Props là viết tắt của "properties", được dùng để truyền dữ liệu từ component cha đến component con. Khác với state, props là không thay đổi (immutable) và chỉ có thể đọc (read-only).

2.2.4 Redis



Hình 20 Redis

Redis (viết tắt của Remote Dictionary Server) là một hệ thống lưu trữ dữ liệu dưới dạng key-value (bộ giá trị khóa) mã nguồn mở, hoạt động chủ yếu trong bộ nhớ (in-memory), cung cấp hiệu suất rất cao. Redis được phát triển bởi Salvatore Sanfilippo vào năm 2009 và đã trở thành một trong những hệ thống lưu trữ dữ liệu phổ biến nhất cho các ứng dụng yêu cầu xử lý dữ liệu nhanh chóng và thời gian truy cập thấp.

Redis không chỉ là một cơ sở dữ liệu key-value đơn giản, mà còn hỗ trợ các cấu trúc dữ liệu phức tạp như chuỗi (strings), danh sách (lists), tập hợp (sets), bảng băm (hashes), và cây có thứ tự (sorted sets), cùng với nhiều tính năng khác.

Redis hoạt động chủ yếu trong bộ nhớ RAM, điều này cho phép nó có thể cung cấp thời gian phản hồi rất nhanh (thường là trong khoảng microsecond). Dữ liệu có thể được ghi đĩa (persistent storage) để đảm bảo an toàn khi gặp sự cố mất điện hoặc hệ thống bị khởi động lại, nhưng phần lớn các thao tác đọc và ghi đều được thực hiện trong bộ nhớ.

Key-Value Store

Redis lưu trữ dữ liệu dưới dạng các cặp khóa-giá trị (key-value). Mỗi key là duy nhất trong cơ sở dữ liệu và có thể được ánh xạ tới một value là bất kỳ loại dữ liệu nào mà Redis hỗ trợ. Các loại dữ liệu được Redis hỗ trợ bao gồm:

- String: Dữ liệu chuỗi đơn giản, là loại phổ biến nhất.
- List: Danh sách các giá trị được sắp xếp.
- Set: Tập hợp các giá trị không trùng lặp, không được sắp xếp.
- Sorted Set: Tập hợp các giá trị có thứ tự, được sắp xếp theo điểm số (score).

- Hash: Một bảng băm, ánh xạ từ khóa con đến các giá trị khác nhau.
- Bitmap: Dữ liệu nhị phân cho các thao tác bitwise.

In-memory storage

Redis hoạt động hoàn toàn trong bộ nhớ (RAM), giúp cho việc đọc và ghi dữ liệu diễn ra rất nhanh chóng. Điều này làm cho Redis trở thành một lựa chọn lý tưởng cho các ứng dụng yêu cầu hiệu suất cao, chẳng hạn như cache, hệ thống quản lý phiên (session management), và xếp hàng công việc (task queues).

Persistence (Lưu trữ bền vững)

Mặc dù Redis hoạt động trong bộ nhớ, nó vẫn cung cấp nhiều cơ chế lưu trữ bền vững (persistence) để đảm bảo dữ liệu không bị mất khi hệ thống gặp sự cố:

- Snapshot (RDB - Redis Database Backup): Redis tạo ra các bản snapshot (ảnh chụp) của dữ liệu tại một thời điểm nhất định và lưu chúng vào đĩa. Đây là cách lưu trữ ít tốn tài nguyên, nhưng có thể không lưu lại toàn bộ dữ liệu gần nhất nếu hệ thống gặp sự cố.
- Append-only file (AOF): Redis ghi lại mọi thao tác ghi dữ liệu vào tệp tin riêng, đảm bảo rằng tất cả các thay đổi đều được lưu trữ và có thể khôi phục lại sau sự cố.

Replication (Nhân bản)

Redis hỗ trợ cơ chế nhân bản master-slave, trong đó dữ liệu từ một Redis server (master) có thể được sao chép sang một hoặc nhiều Redis server khác (slave). Điều này cung cấp khả năng chịu lỗi cao (high availability) và phân phối tải (load balancing).

Pub/Sub (Publish/Subscribe)

Redis cung cấp tính năng publish/subscribe, cho phép xây dựng các hệ thống nhắn tin theo thời gian thực. Trong mô hình này, các client có thể đăng ký (subscribe) vào các kênh nhất định và nhận các thông báo (message) khi có bất kỳ client nào khác phát (publish) thông tin trên cùng kênh.

2.3 Phân tích kiến trúc hệ thống

2.3.1 Key abstraction



Hình 21 Key abstraction

Định nghĩa Key abstraction:

Book: Là bản ghi chứa thông tin về sách

Cơ chế phân tích: Persistency

Category: Là bản ghi chứa thông tin về thể loại

Cơ chế phân tích: Persistency

Comment: Là bản ghi chứa thông tin về đánh giá sản phẩm

Cơ chế phân tích: Persistency

Customer: Là bản ghi chứa thông tin về khách hàng

Cơ chế phân tích: Persistency

Order: Là bản ghi chứa thông tin của đơn hàng

Cơ chế phân tích: Persistency

OrderDetail: Là bản ghi chứa thông tin đơn hàng chi tiết

Cơ chế phân tích: Persistency

Rating: Là bản ghi chứa thông tin của cho đánh giá theo sao

Cơ chế phân tích: Persistency

Role: Là bản ghi chứa thông tin về quyền hạn

Cơ chế phân tích: Persistency

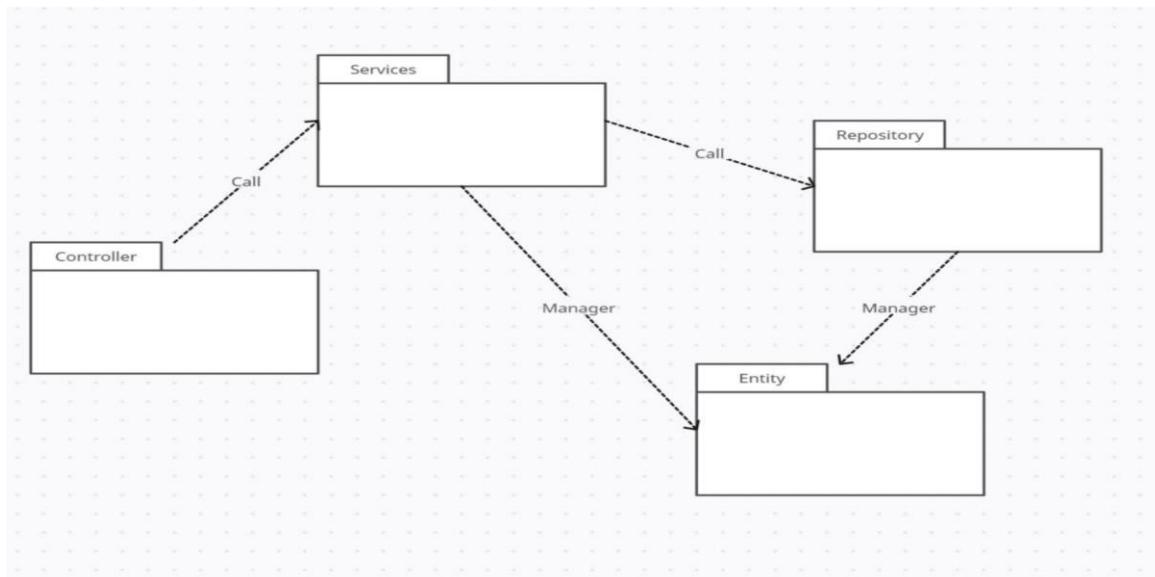
ShoppingCart: Là bản ghi chứa thông tin về giỏ hàng của khách hàng

Cơ chế phân tích: Persistency

User: Là bản ghi chứa thông tin tài khoản

Cơ chế phân tích: Persistency

2.3.2 Thành phần cấp cao và sự phụ thuộc



Hình 22 Package

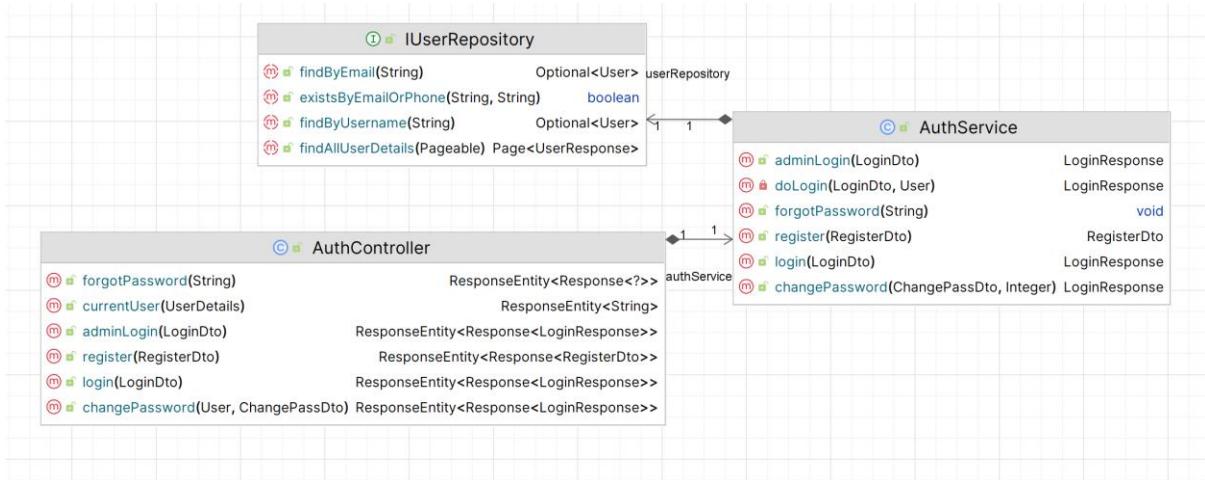
Mô tả:

- Package Service: Chứa các thành phần liên quan đến business logic của ứng dụng. Trong gói này chịu trách nhiệm triển khai dịch vụ (services) để xử lý các yêu cầu từ Controller và thực hiện business logic. Các dịch vụ này có thể bao gồm phương thức để thực hiện các chức năng cụ thể của ứng dụng.
- Package Entity: Chứa các thành phần liên quan để biểu diễn dữ liệu của ứng dụng. Trong gói này định nghĩa các lớp biểu diễn các thực thể trong cơ sở dữ liệu.

- Package Repository: Chứa các thành phần liên quan đến truy cập và thao tác với cơ sở dữ liệu. Trong gói này thực hiện các thao tác CRUD trên dữ liệu trong cơ sở dữ liệu.
- Package Controller: Chứa các Controller, nơi nhận các yêu cầu http đến từ client.

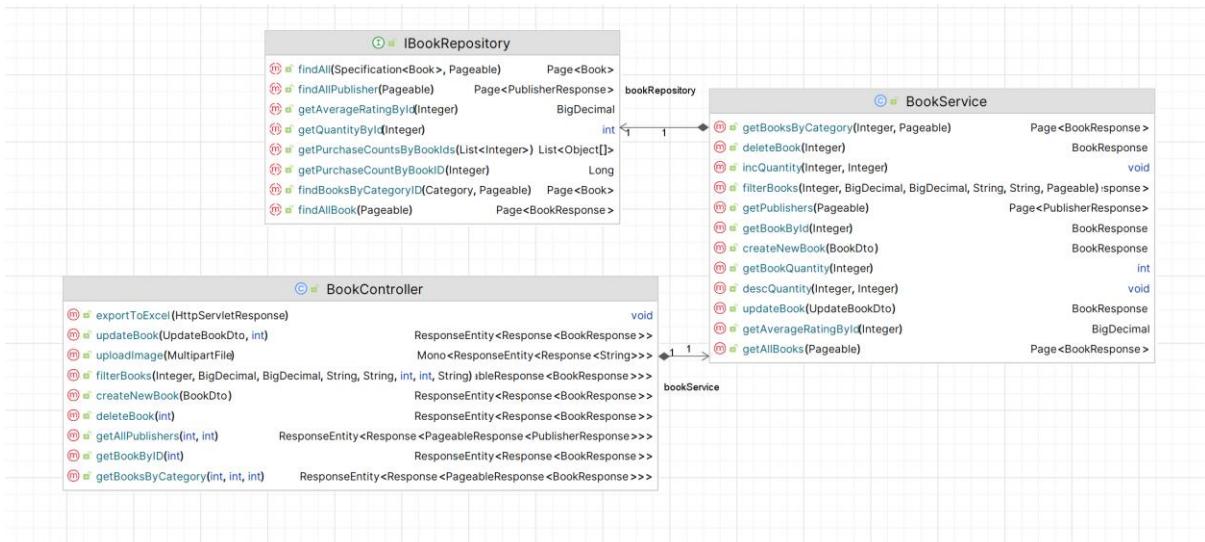
2.3.3 Biểu đồ lớp phân tích chức năng

Sơ đồ lớp phân tích chức năng xác thực tài khoản, quên mật khẩu



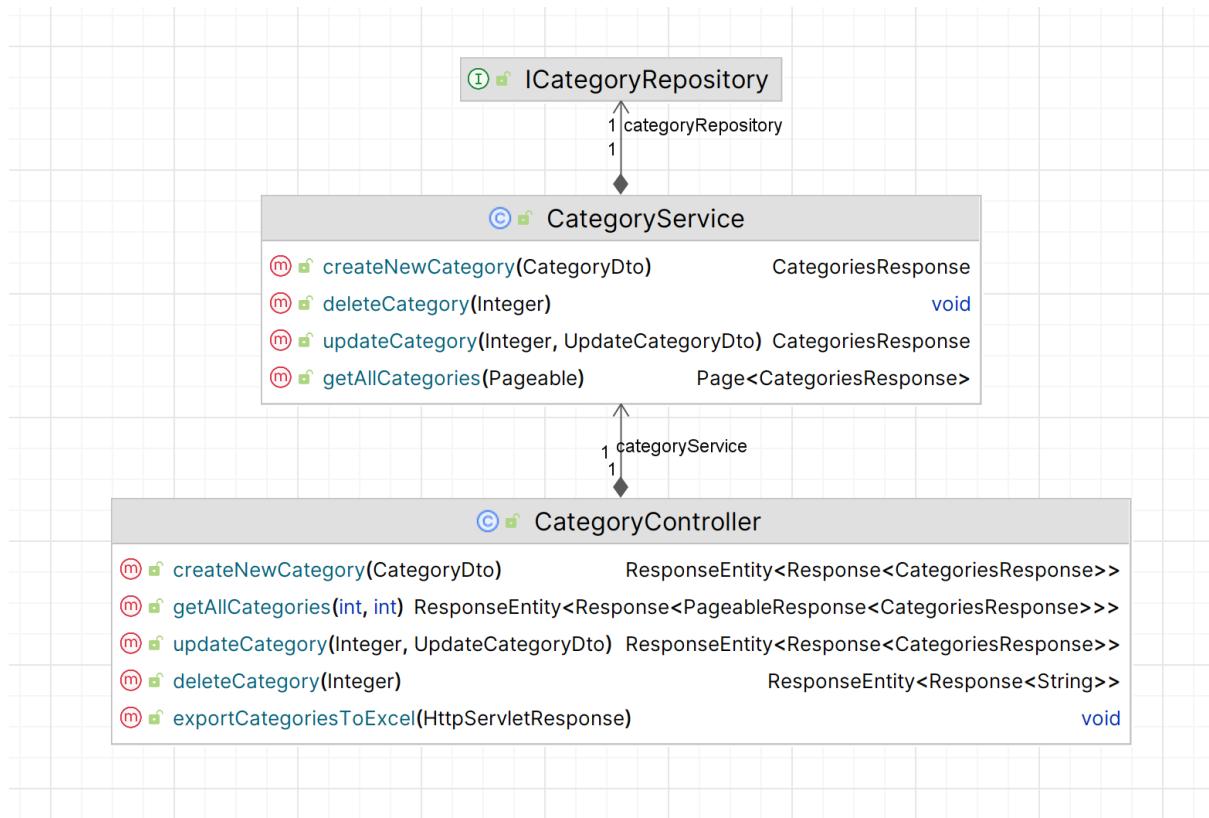
Hình 23 Phân tích xác thực

Sơ đồ lớp phân tích chức năng quản lý sách



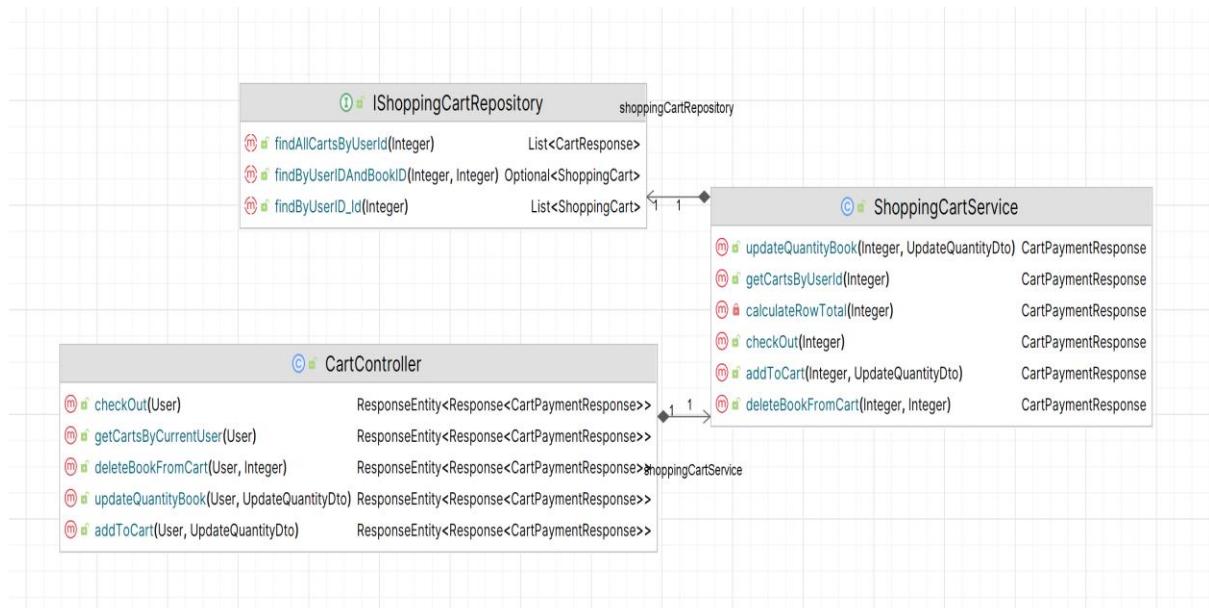
Hình 24 Phân tích quản lý sách

Sơ đồ lớp phân tích chức năng quản lý thẻ loại



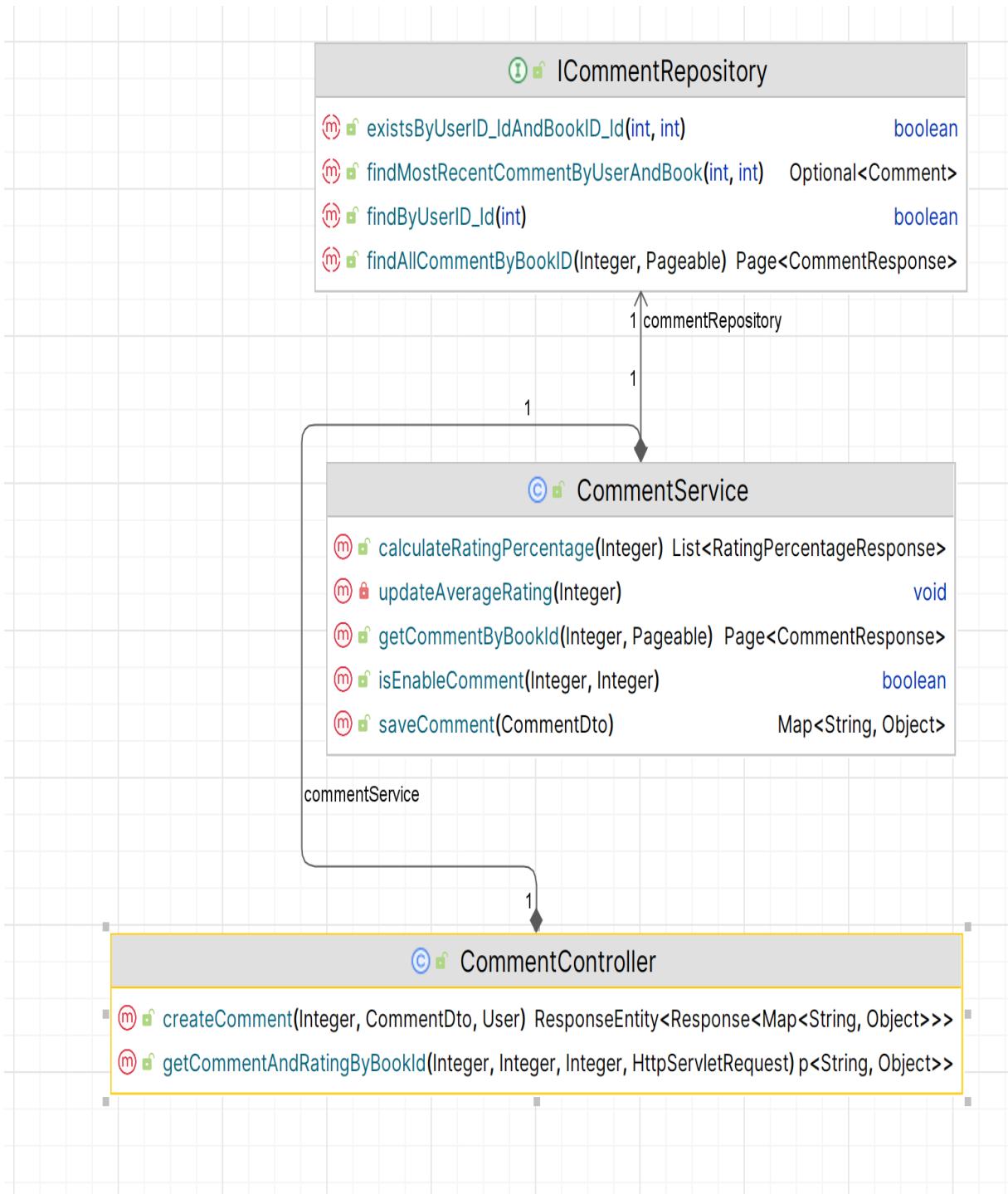
Hình 25 Phân tích quản lý thẻ loại

Sơ đồ lớp cho chức năng quản lý giỏ hàng và thanh toán



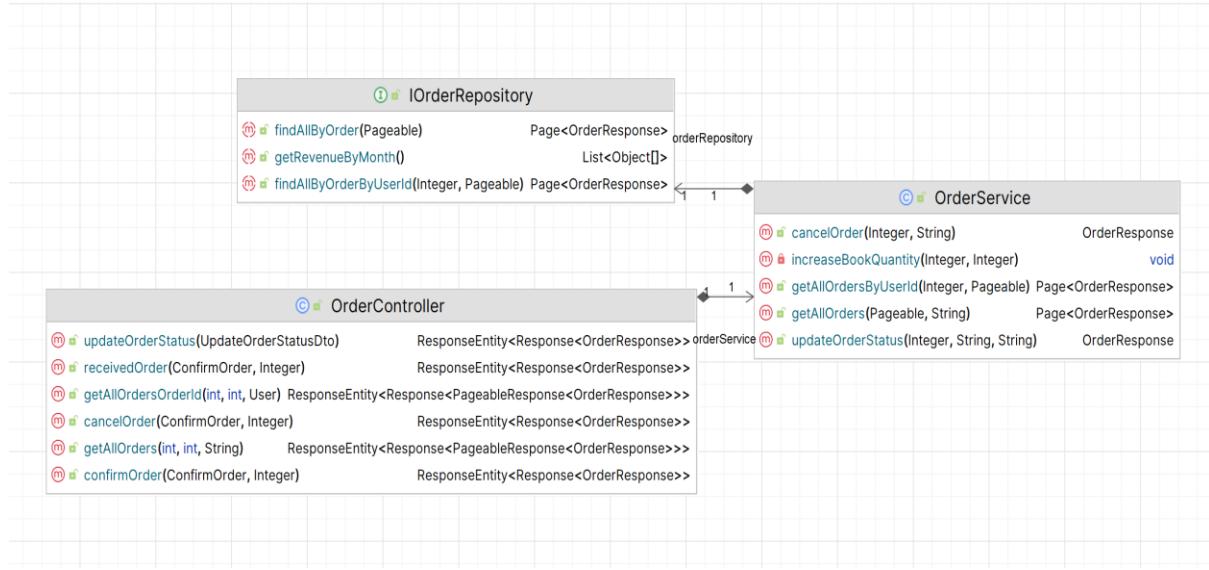
Hình 26: Phân tích quản lý giỏ hàng và thanh toán

Sơ đồ lớp phân tích cho chức năng đánh giá sản phẩm



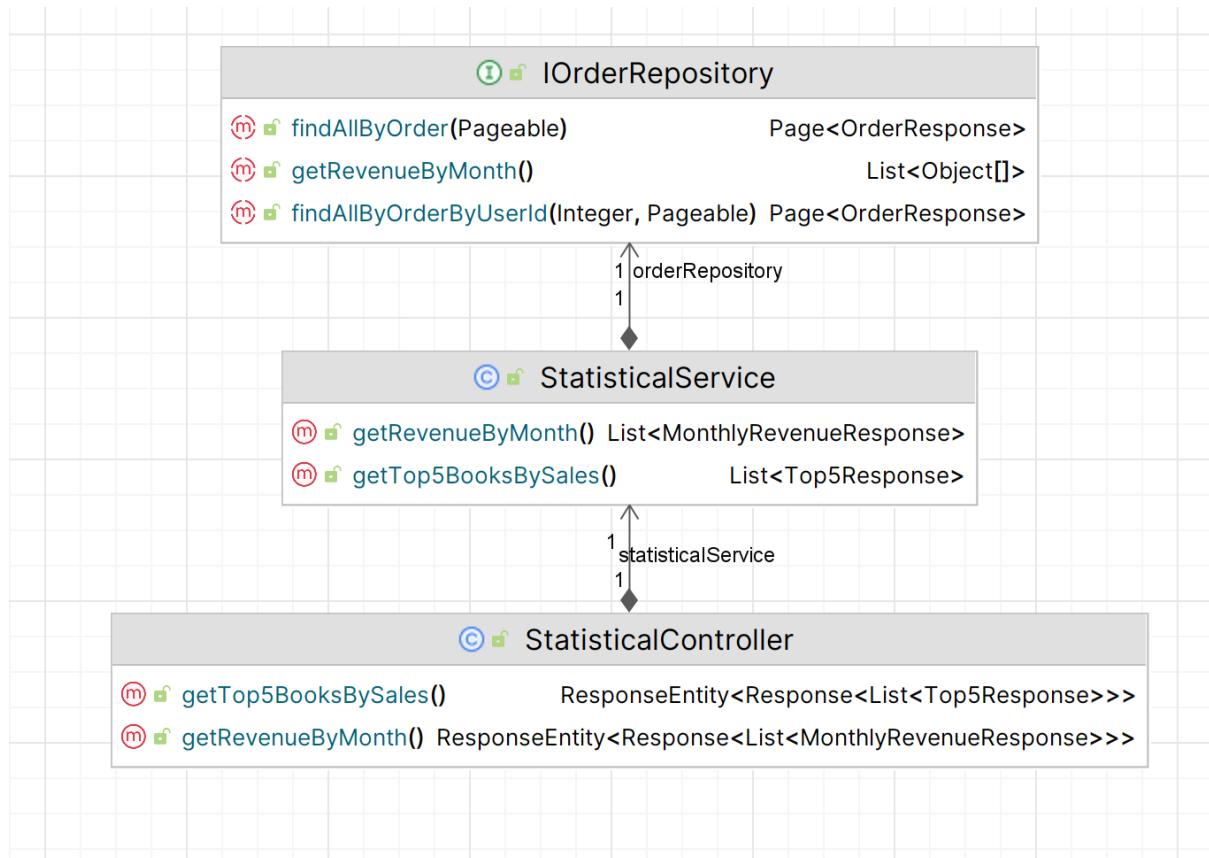
Hình 27 Phân tích đánh giá sản phẩm

Sơ đồ lớp phân tích cho chức năng quản lý đơn hàng



Hình 27: Phân tích quản lý đơn hàng

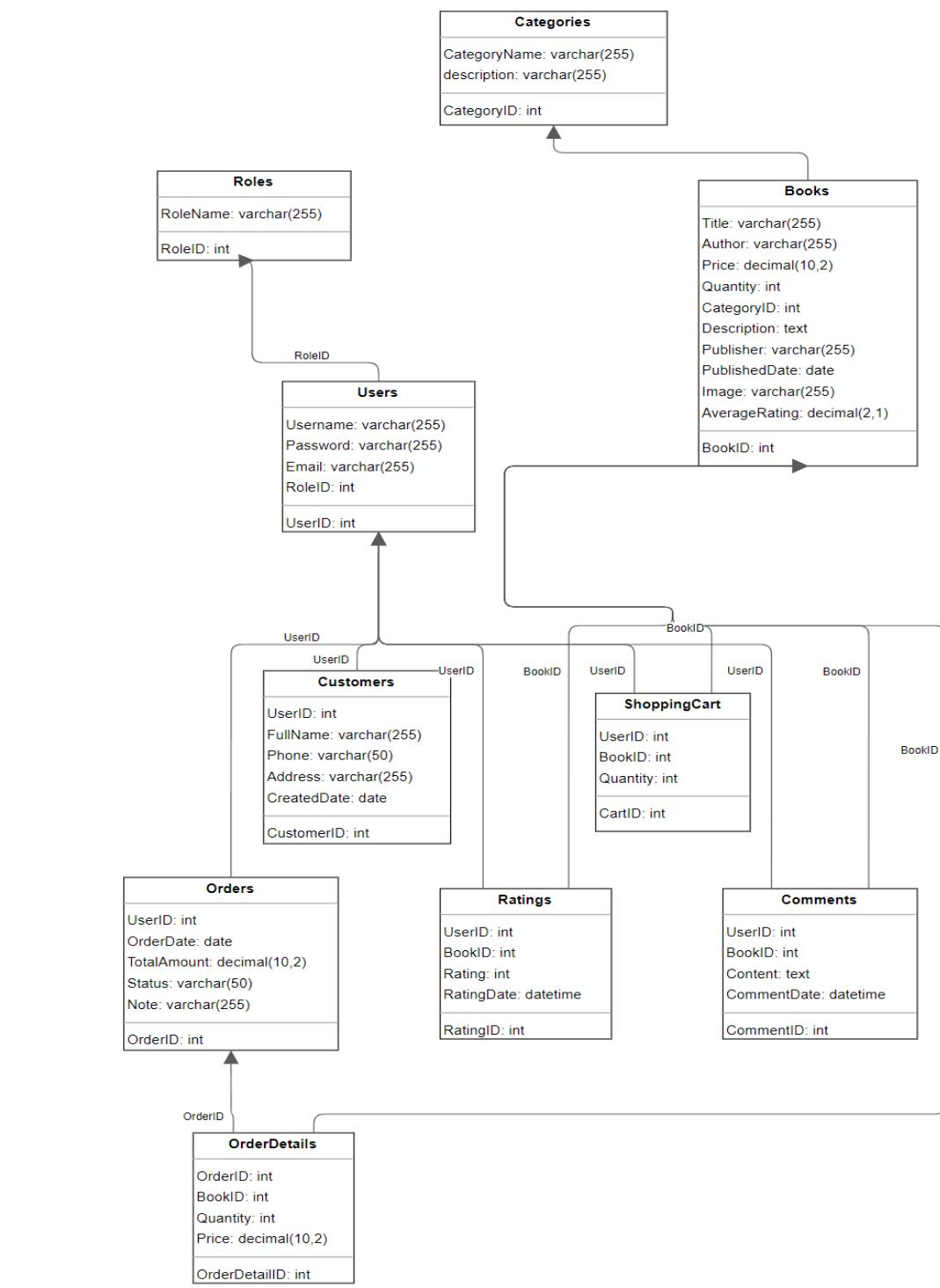
Sơ đồ lớp phân tích cho báo cáo thống kê



Hình 28 Phân tích báo cáo thống kê

2.4 Thiết kế hệ thống

2.4.1 Thiết kế cơ sở dữ liệu



Hình 29: Sơ đồ thực thể liên kết cơ sở dữ liệu

Books

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
BookID	int		YES	NO	
Title	varchar(255)			NO	
Author	varchar(255)			YES	
Price	decimal(10,2)			NO	
Quantity	int			NO	
CategoryID	int			YES	
Description	text			YES	
Publisher	varchar(255)			YES	
PublishedDate	date			YES	
Image	varchar(255)			YES	
AverageRating	decimal(2,1)			YES	

Categories

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
CategoryID	int		YES	NO	
CategoryName	varchar(255)			NO	
description	varchar(255)			YES	

Comments

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
CommentID	int		YES	NO	
UserID	int			NO	
BookID	int			NO	
Content	text			NO	
CommentDate	datetime			NO	

Customers

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
CustomerID	int		YES	NO	
UserID	int			NO	
FullName	varchar(255)			YES	
Phone	varchar(50)			YES	
Address	varchar(255)			YES	
CreatedDate	date			YES	

OrderDetails

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định

OrderDetailID	int		YES	NO	
OrderID	int			NO	
BookID	int			NO	
Quantity	int			NO	
Price	decimal(10,2)			NO	

Orders

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
OrderID	int		YES	NO	
UserID	int			NO	
OrderDate	date			NO	
TotalAmount	decimal(10,2)			NO	
Status	varchar(50)			NO	
Note	varchar(255)			YES	

Ratings

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
RatingID	int		YES	NO	
UserID	int			NO	

BookID	int			NO	
Rating	int			NO	
RatingDate	datetime			NO	

Roles

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
RoleID	int		YES	NO	
RoleName	varchar(255)			NO	

ShoppingCart

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
CartID	int		YES	NO	
UserID	int			NO	
BookID	int			NO	
Quantity	int			NO	

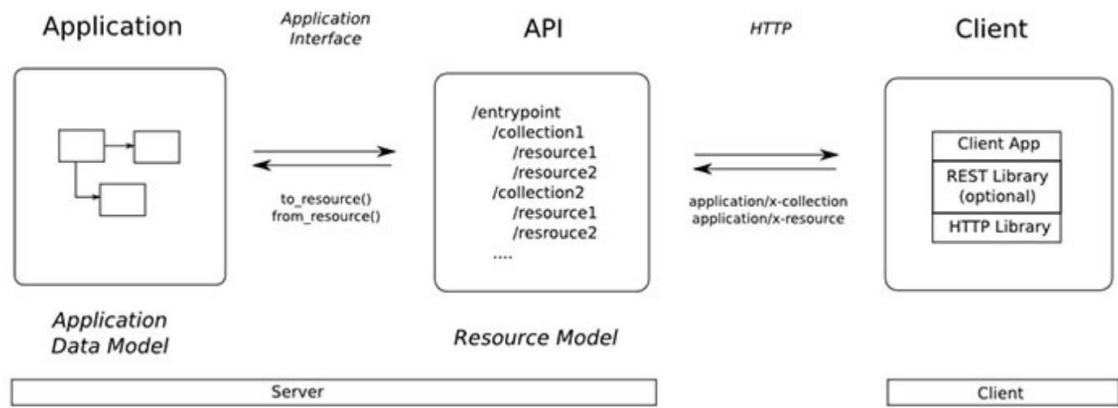
Users

Tên cột	Kiểu dữ liệu	Ghi chú	Khoá chính	Cho phép rỗng	Giá trị mặc định
UserID	int		YES	NO	

Username	varchar(255)			NO	
Password	varchar(255)			NO	
Email	varchar(255)			YES	
RoleID	int			NO	

2.4.2 Thiết kế API

2.4.2.1 Cấu trúc thư mục của API



Hình 30: Cấu trúc thư mục API

RESTful API (Representational State Transfer Application Programming Interface) là một kiểu kiến trúc cho phép các ứng dụng giao tiếp với nhau thông qua HTTP. Nó thường được sử dụng để xây dựng các dịch vụ web, cho phép các client (như trình duyệt hoặc ứng dụng di động) truy cập và tương tác với dữ liệu trên server một cách dễ dàng và hiệu quả.

Các đặc điểm chính của RESTful API:

- Stateless: Mỗi yêu cầu từ client đến server phải chứa tất cả thông tin cần thiết để server xử lý yêu cầu đó. Server không lưu trạng thái của client giữa các yêu cầu.
- Resource-Based: RESTful API hoạt động dựa trên các tài nguyên (resources) mà có thể được truy cập thông qua các URL.

Các HTTP Methods phổ biến trong RESTful API:

- GET: Dùng để lấy thông tin từ server. Ví dụ, GET /users sẽ trả về danh sách người dùng.
- POST: Dùng để gửi dữ liệu mới đến server, thường được sử dụng để tạo một tài nguyên mới. Ví dụ, POST /users với dữ liệu người dùng mới trong body sẽ tạo một người dùng mới.
- PUT: Dùng để cập nhật toàn bộ thông tin của một tài nguyên hiện có. Ví dụ, PUT /users/1 sẽ cập nhật thông tin của người dùng có ID 1.
- PATCH: Dùng để cập nhật một phần thông tin của một tài nguyên hiện có. Ví dụ, PATCH /users/1 với dữ liệu thay đổi chỉ bao gồm các trường cần thiết sẽ cập nhật phần đó của người dùng có ID 1.
- DELETE: Dùng để xóa một tài nguyên hiện có. Ví dụ, DELETE /users/1 sẽ xóa người dùng có ID 1.

2.4.2.1 Danh sách API

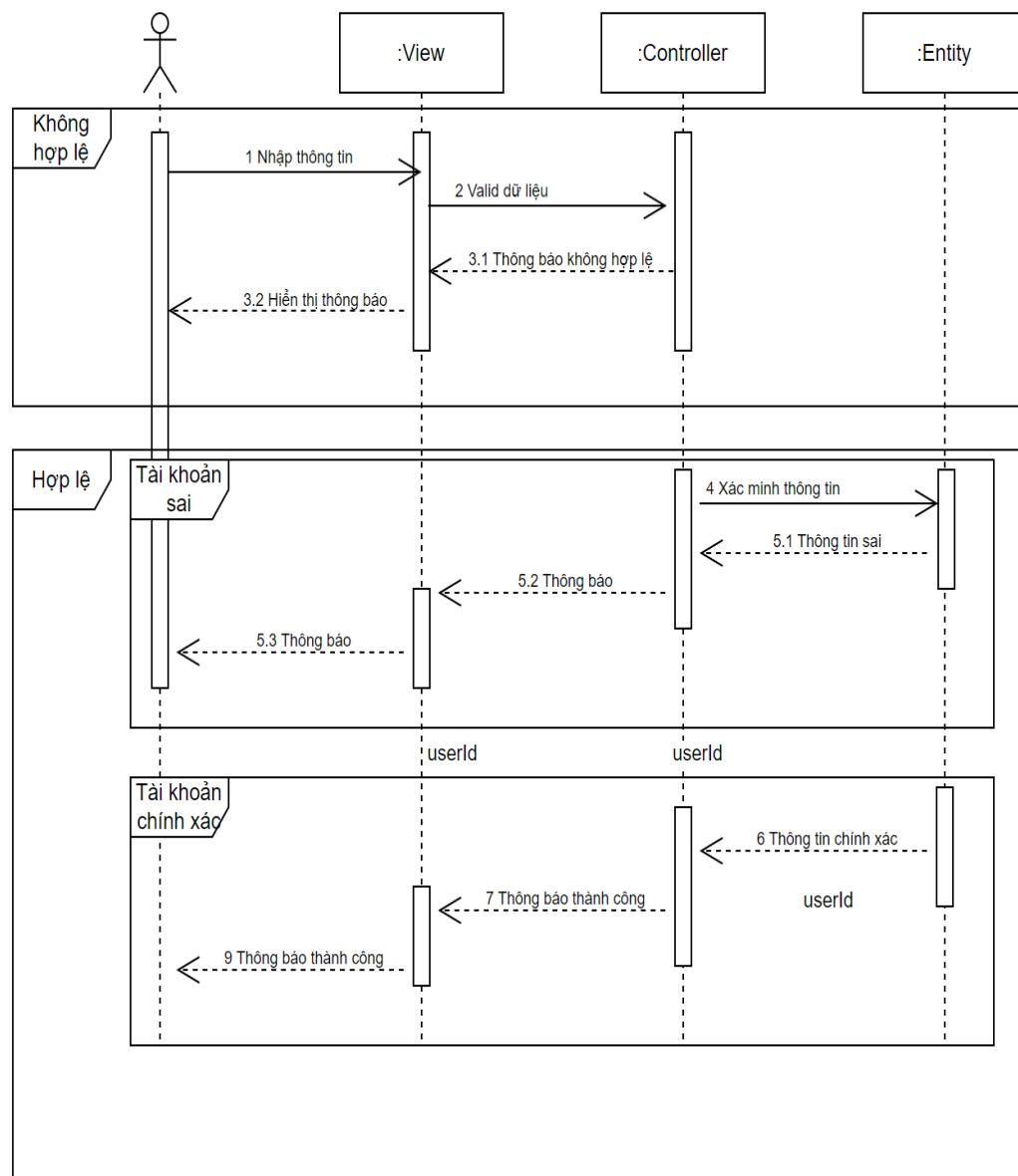
STT	URL	Phương thức
Auth		
1	/api/v1/auth/regisger	Post
2	/api/v1/auth/login	Post
3	/api/v1/auth/forgot-password	Post
4	/api/v1/auth/change-password	Post
5	/api/v1/auth/admin	Post
User		
6	/api/v1/users/update-information	Put
7	/api/v1/users/information	Get
8	/api/v1/users/address	Get
9	/api/v1/users	Get
Order		
10	/api/v1/orders/user/recevived/{orderId}	Put

11	/api/v1/orders/cancel/{orderId}	Put
12	/api/v1/orders/admin/confirm/{orderId}	Put
13	/api/v1/orders	Get
14	/api/v1/orders/user	Get
Category		
15	/api/v1/categories/update-category/{id}	Put
16	/api/v1/categories	Get
17	/api/v1/categories	Post
18	/api/v1/categories/{id}	Delete
Cart		
19	/api/v1/carts/update-quantity	Put
20	/api/v1/carts	Get
21	/api/v1/carts	Post
22	/api/v1/carts/checkout	Post
23	/api/v1/carts/delete	Delete
Payment		
23	/api/v1/payments	Post
Comment		
24	/api/v1/comments/{bookId}	Get
25	/api/v1/comments/{bookId}	Post
Statiscal		
26	/api/v1/statiscal/top5-books	Get
27	/api/v1/statiscal/revenue-by-month	Get
Order Detail		
28	/api/v1/order-details/user/{orderId}	Get
29	/api/v1/order-details/admin/{orderId}	Get

2.4.3 Thiết kế chức năng

Chức năng đăng nhập

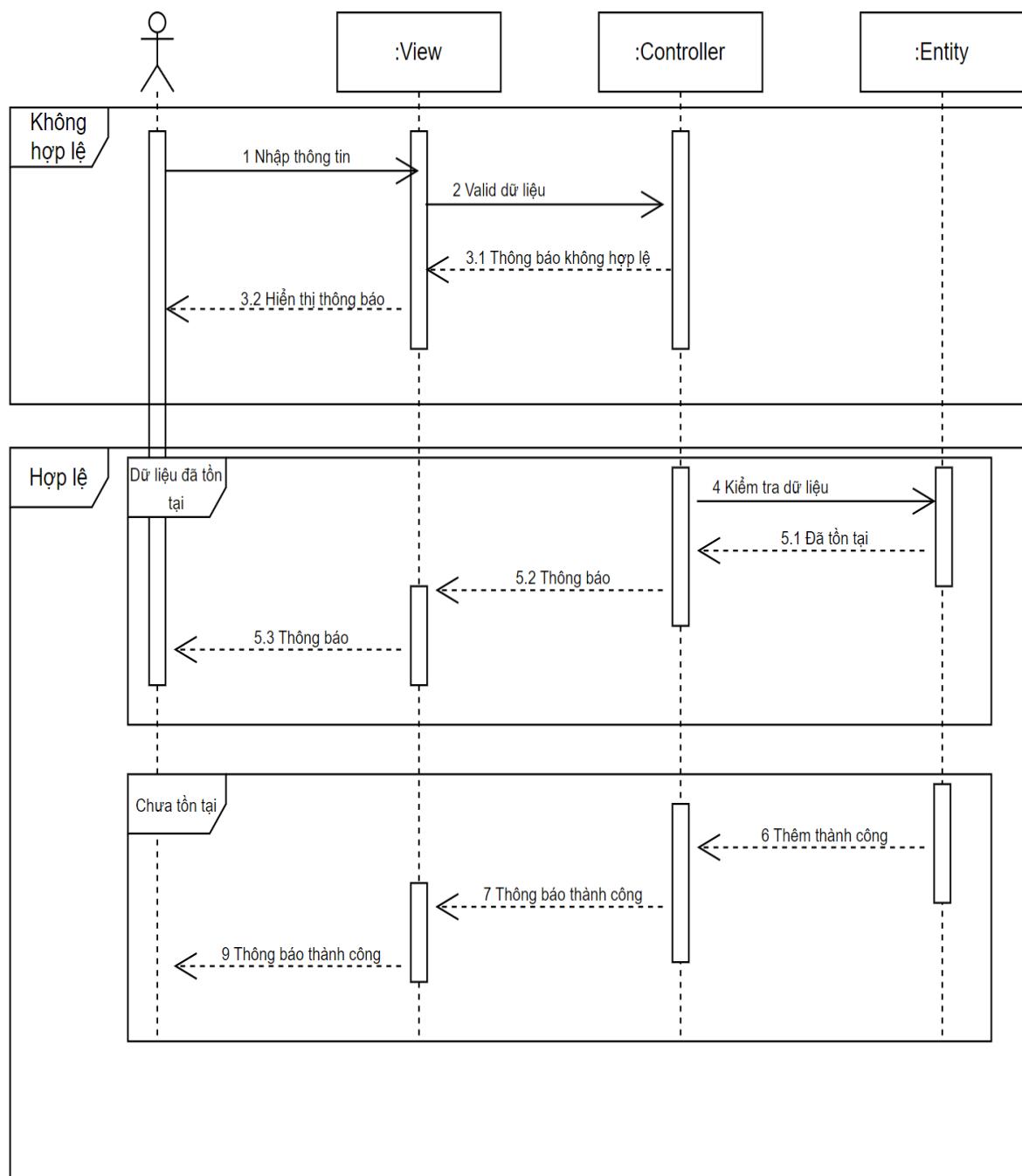
Biểu đồ tuần tự chức năng đăng nhập



Hình 31: Sequence Đăng nhập

Chức năng thêm

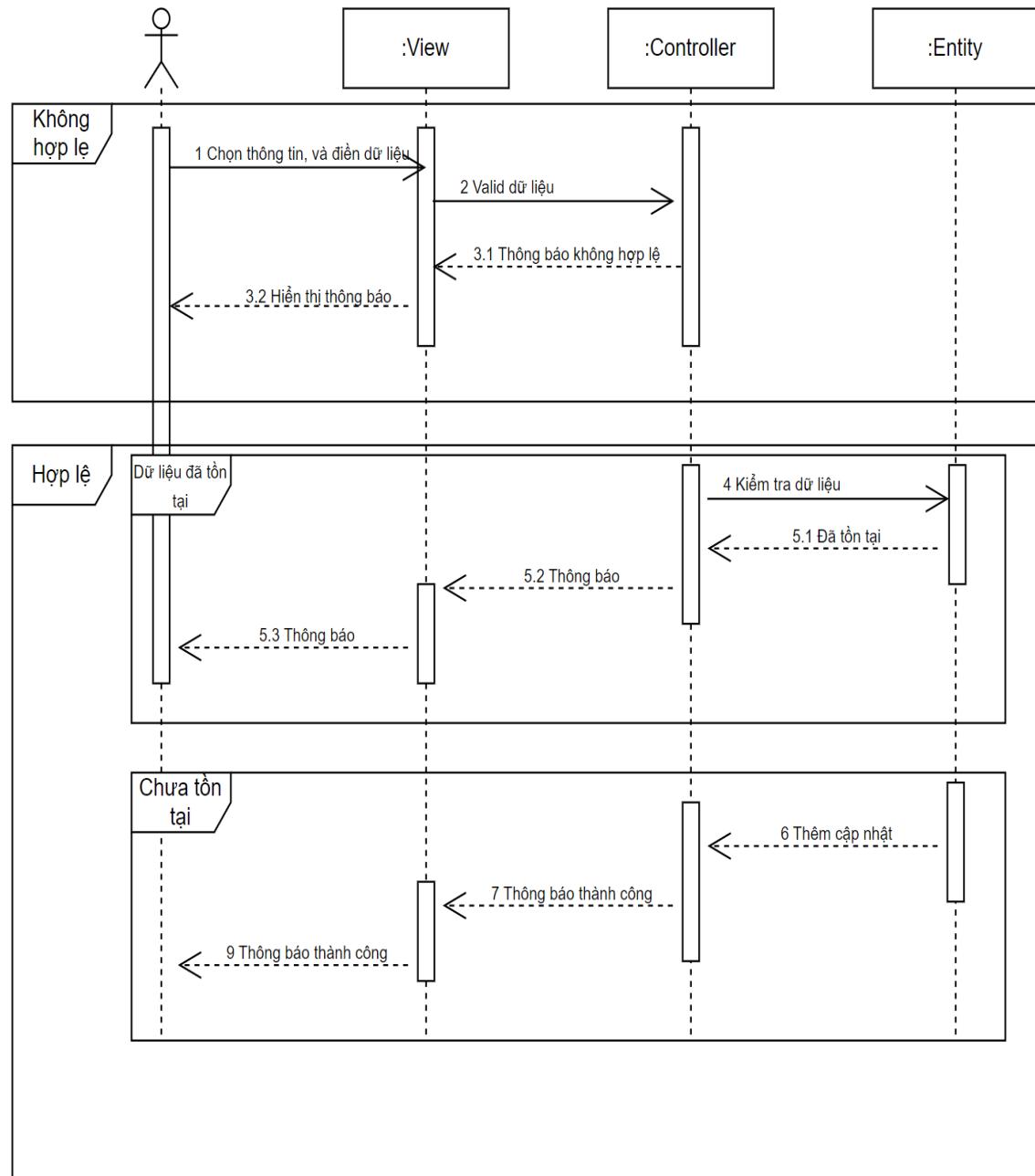
Biểu đồ tuần tự chức năng thêm



Hình 32 Sequence Thêm sản phẩm

Chức năng cập nhật

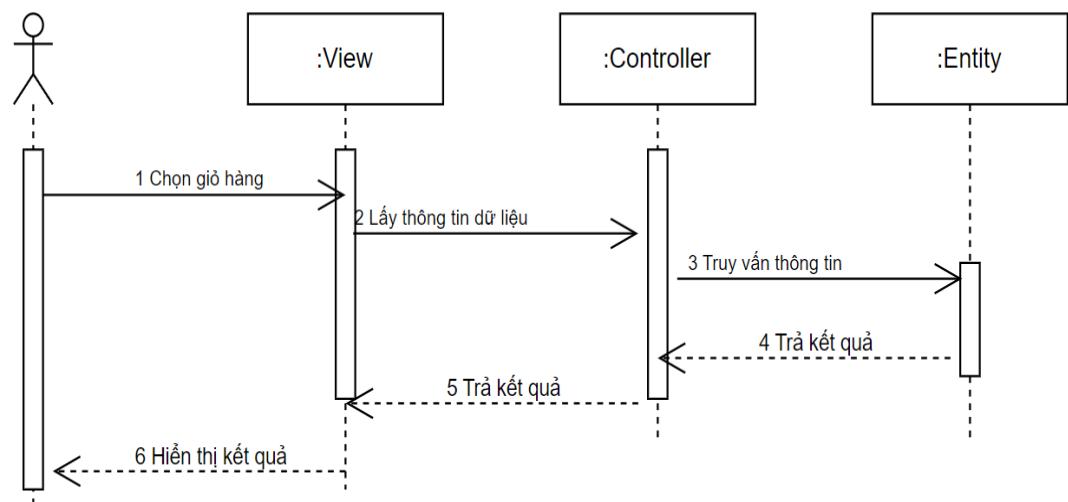
Biểu đồ tuần tự chức năng cập nhật



Hình 33 Sequence Cập nhật thông tin

Chức năng cho giờ hàng

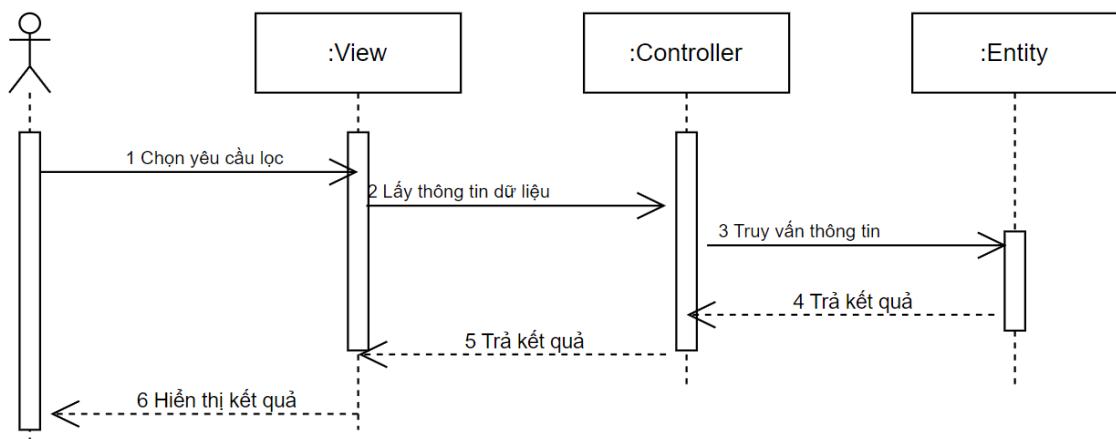
Chức năng giờ hàng



Hình 34 Sequence xem giờ hàng

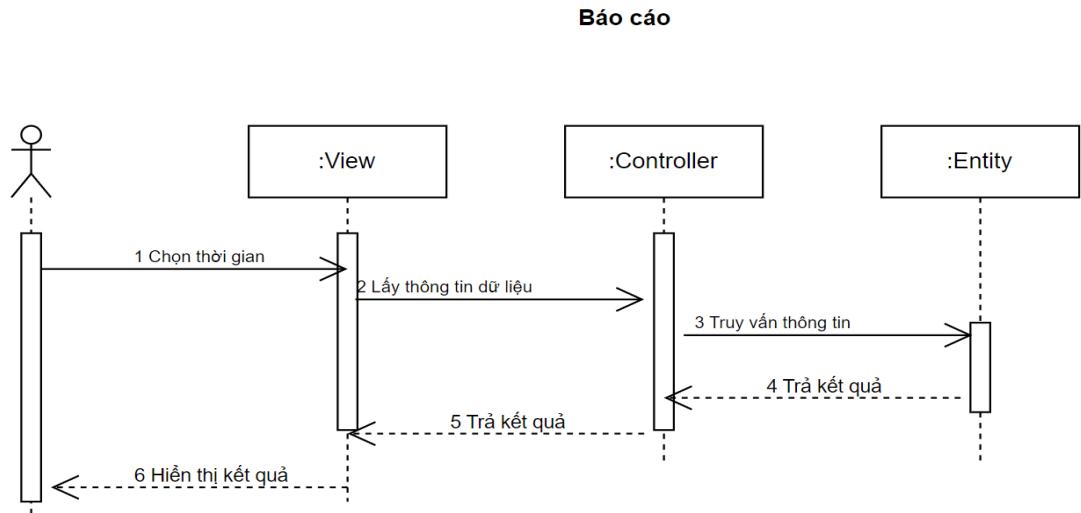
Chức năng lọc và tìm kiếm

Lọc và tìm kiếm



Hình 35 Sequence Lọc và tìm kiếm

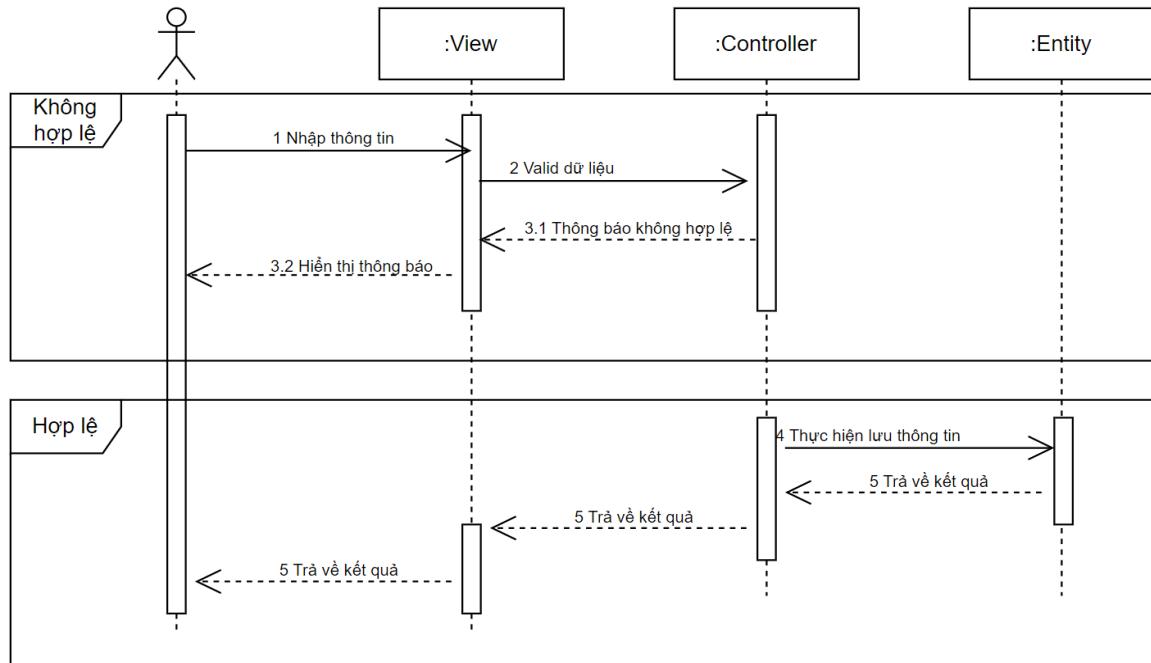
Chức năng báo cáo



Hình 36 Sequence Báo cáo

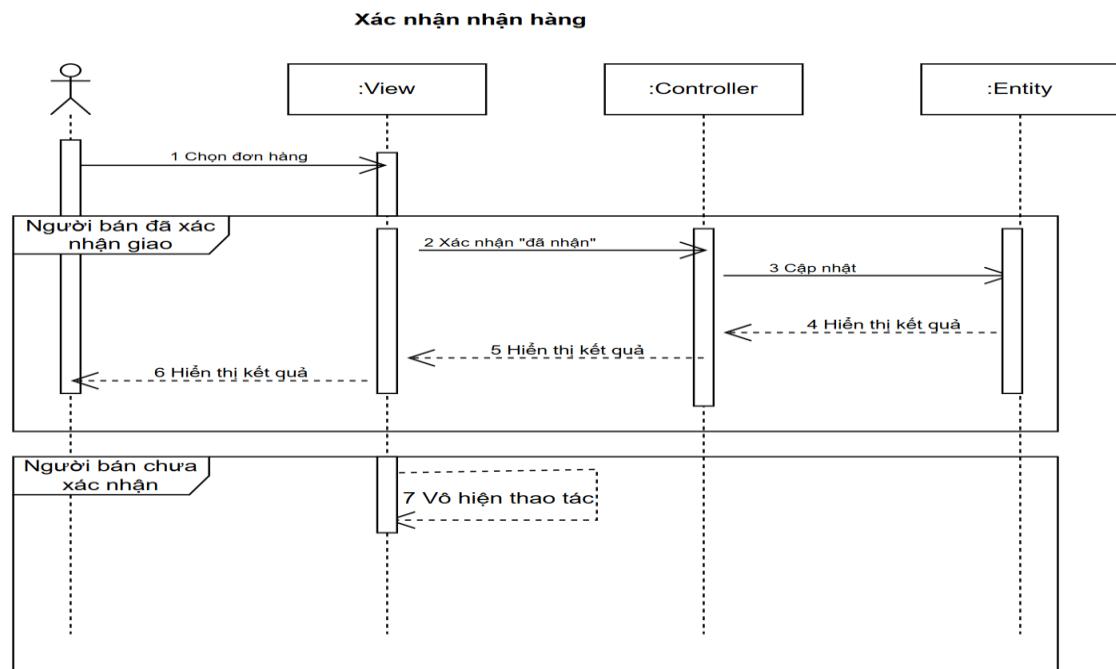
Chức năng cập nhật thông tin cá nhân

Cập nhật thông tin cá nhân và mật khẩu



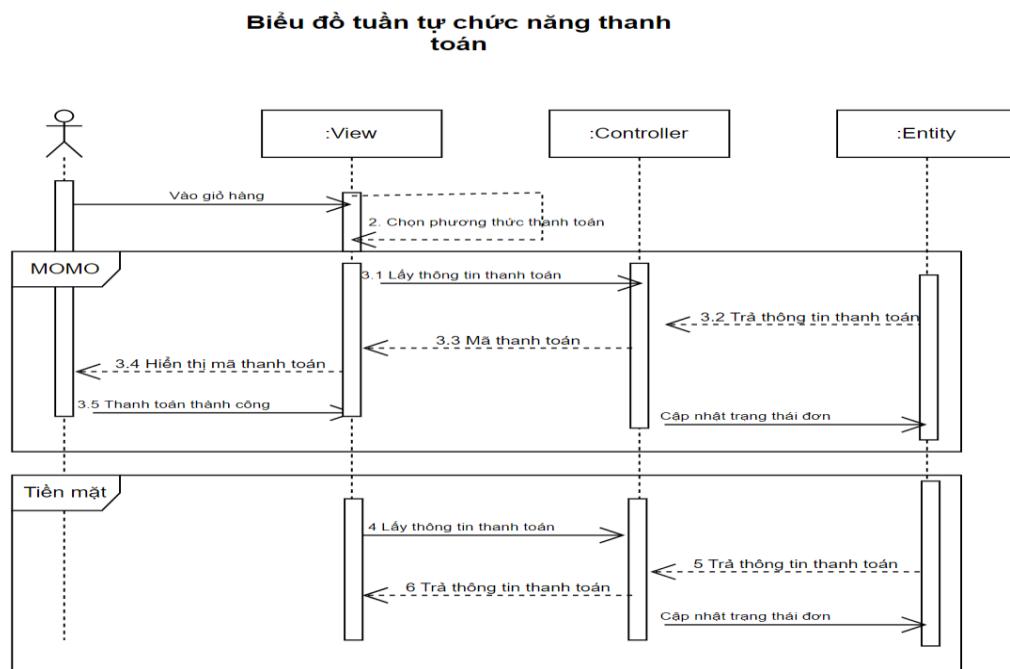
Hình 37 Sequence Cập nhật thông tin cá nhân

Xác nhận hàng



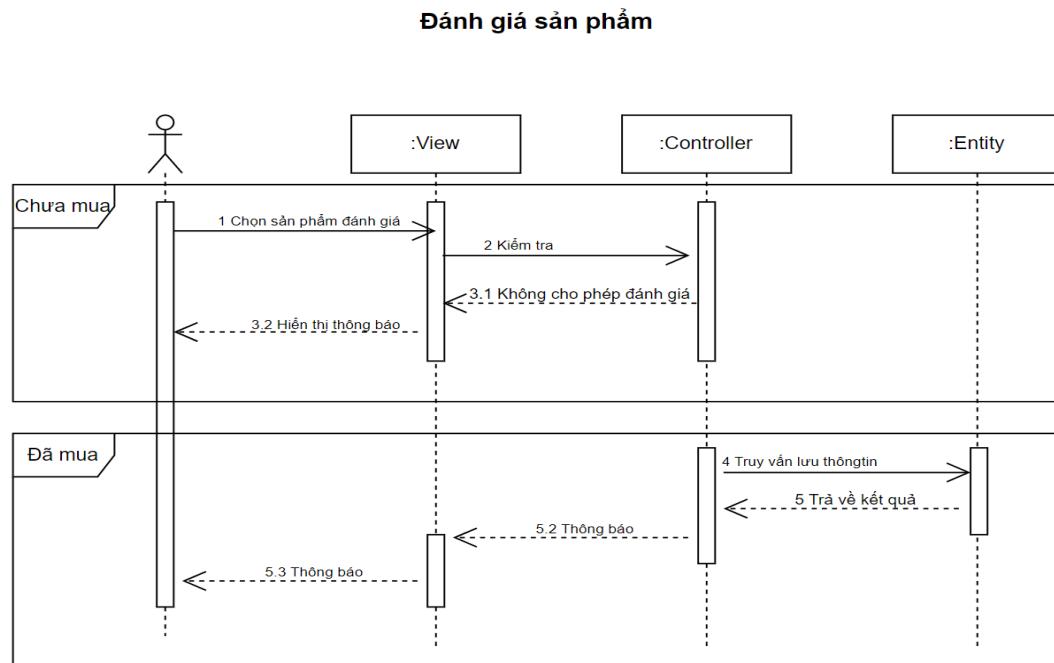
Hình 38 Sequence Xác nhận đơn hàng

Chức năng thanh toán



Hình 39 Sequence Thanh toán

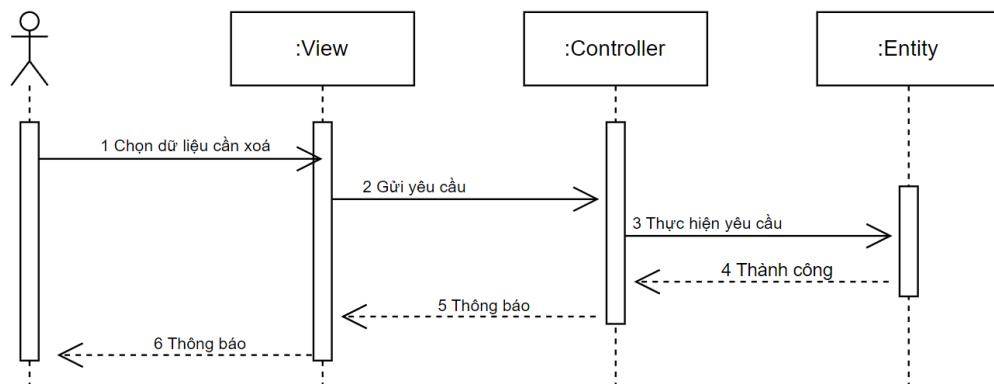
Chức năng đánh giá sản phẩm



Hình 40 Sequence Đánh giá sản phẩm

Chức năng xoá

Biểu đồ tuần tự chức năng xoá



Hình 41 Sequen chức năng xoá

CHƯƠNG 3: XÂY DỰNG VÀ THỰC NGHIỆM

3.1 Cấu trúc thành phần đối tượng thực

3.1.1 Front-end

STT	Thành phần	Mô tả
1	api	Chứa các module giao tiếp với API bên ngoài hoặc dịch vụ backend của ứng dụng.
2	assets	Chứa tài nguyên tĩnh như hình ảnh, font, hoặc các tệp đa phương tiện khác.
3	components	Chứa các thành phần giao diện tái sử dụng trong ứng dụng, như nút, biểu mẫu, hoặc các thành phần UI khác.
4	constants	Chứa các hằng số dùng chung trong ứng dụng, ví dụ như URL API, tên action trong redux, hoặc các giá trị không đổi khác.
5	helpers	Chứa các hàm tiện ích hỗ trợ xử lý logic không liên quan đến UI, như các hàm xử lý dữ liệu, định dạng ngày giờ, hoặc kiểm tra dữ liệu.
6	pages	Chứa các trang chính của ứng dụng, mỗi trang có thể là một màn hình hoàn chỉnh với nhiều thành phần.
7	redux	Chứa cấu trúc redux của ứng dụng, bao gồm các slice, reducer và action để quản lý trạng thái toàn cục của ứng dụng.
8	routes	Chứa cấu hình cho các route trong ứng dụng, để điều hướng giữa các trang.
9	services	Chứa các dịch vụ để làm việc với dữ liệu, ví dụ như xử lý các yêu cầu API, logic dữ liệu, hoặc các dịch vụ liên quan đến backend.

10	styles	Chứa các tệp CSS hoặc SCSS để tạo kiểu dáng cho giao diện ứng dụng.
11	types	Chứa các tệp khai báo kiểu TypeScript dùng trong toàn ứng dụng để hỗ trợ quản lý kiểu dữ liệu và tăng độ ổn định cho code.
12	utils	Chứa các hàm tiện ích chung không liên quan đến domain cụ thể nào, có thể tái sử dụng trong nhiều phần của ứng dụng.
13	App.tsx	Thành phần gốc của ứng dụng, nơi cấu hình và kết hợp các thành phần chính của ứng dụng.
14	main.tsx	Điểm vào chính của ứng dụng, nơi ứng dụng được khởi chạy và render vào HTML.
15	vite-env.d.ts	Chứa các khai báo môi trường cho Vite, giúp TypeScript nhận biết các biến môi trường khi biên dịch.
16	.env	Tệp cấu hình môi trường chứa các biến môi trường nhạy cảm như API keys, các cấu hình backend, chỉ được truy cập qua server-side.
17	.eslintrc.cjs	Tệp cấu hình cho ESLint, dùng để kiểm tra và định dạng lại mã nguồn theo quy tắc nhất định, giúp mã sạch và chuẩn hóa.
18	.gitignore	Tệp cấu hình Git để bỏ qua các tệp và thư mục không cần thiết khi đẩy code lên repository.
19	.prettierrc	Tệp cấu hình cho Prettier, giúp định dạng mã nguồn tự động theo quy tắc đã định nghĩa sẵn.
20	index.html	Tệp HTML chính của ứng dụng, nơi gắn kết điểm vào chính của ứng dụng và khởi động giao diện người dùng.

21	package.json	Tệp khai báo cấu hình cho npm/yarn, chứa các thông tin như các package cần thiết, các script để chạy, và các thuộc tính khác của dự án.
22	README.md	Tệp hướng dẫn và mô tả tổng quan về dự án, giúp người khác hiểu về cách cài đặt, chạy, và phát triển thêm cho dự án.
23	tsconfig.app.json	Tệp cấu hình TypeScript đặc thù cho ứng dụng.
24	tsconfig.json	Tệp cấu hình TypeScript chung của dự án, quy định các cài đặt như kiểm tra kiểu, các đường dẫn, và hỗ trợ biên dịch TypeScript.
25	tsconfig.node.json	Tệp cấu hình TypeScript đặc thù cho node modules.
26	vite.config.ts	Tệp cấu hình cho Vite, một công cụ bundler để biên dịch và chạy ứng dụng, cho phép cài đặt các plugin và tùy chỉnh khác.
27	yarn.lock	Tệp khóa cho Yarn, chứa các phiên bản chính xác của các package và các dependencies đã cài đặt để đảm bảo tính đồng nhất khi cài đặt lại.

3.1.2 Back-end

STT	Thành phần	Mô tả
1	Component	Thư mục chứa các thành phần con của hệ thống
2	Config	Thư mục chứa các cài đặt, setting cho hệ thống
3	Controllers	Thư mục chịu trách nhiệm nhận các yêu cầu từ client và trả dữ liệu cho client
4	Dto	Thư mục chứa cái kiểu client thực hiện yêu cầu đến hệ thống

5	Entities	Thư mục chứa các đối tượng đại diện cho các bảng trong cơ sở dữ liệu
6	Exceptions	Thư mục chứa các ngoại lệ, xử lý ngoại lệ của hệ thống
7	Filter	Thư mục chứa các bộ lọc xác thực của hệ thống
8	Repositories	Thư mục chịu trách nhiệm truy vấn dữ liệu với cơ sở dữ liệu
9	Responses	Thư mục chứa các kiểu dữ liệu trả về cho client
10	Services	Thư mục chứa các logic của hệ thống
11	Specifications	Thư mục chứa các truy vấn đặc trưng, nâng cao của hệ thống
12	Utils	Thư mục chứa các tính năng, tiện ích chung của toàn hệ thống
13	Resource/db.migration	Thư mục chứa các phiên bản của cơ sở dữ liệu
14	Application.properties	File chứa toàn bộ biến môi trường của hệ thống
15	Pom.xml	File khai báo thư viện, các phụ thuộc, build của hệ thống

3.2 Thực nghiệm sử dụng

3.2.1 Công cụ cần thiết

- Java Development Kit (JDK): Phiên bản 11 trở lên.
- Maven: Để quản lý dự án Java và các phụ thuộc.
- Node.js và NPM: Để chạy và quản lý dự án ReactJS.

3.2.2 Lệnh chạy dự án Spring Boot

- Bước 1: Cài đặt phụ thuộc bằng Maven

```
mvn install
```

- Bước 2: Chạy ứng dụng Spring Boot

```
mvn spring-boot:run
```

- Bước 3: Hoặc chạy trực tiếp từ tệp JAR sau khi build:

```
java -jar target/your-app.jar
```

3.2.3 Lệnh chạy dự án ReactJS

- Bước 1: Cài đặt các phụ thuộc cho React

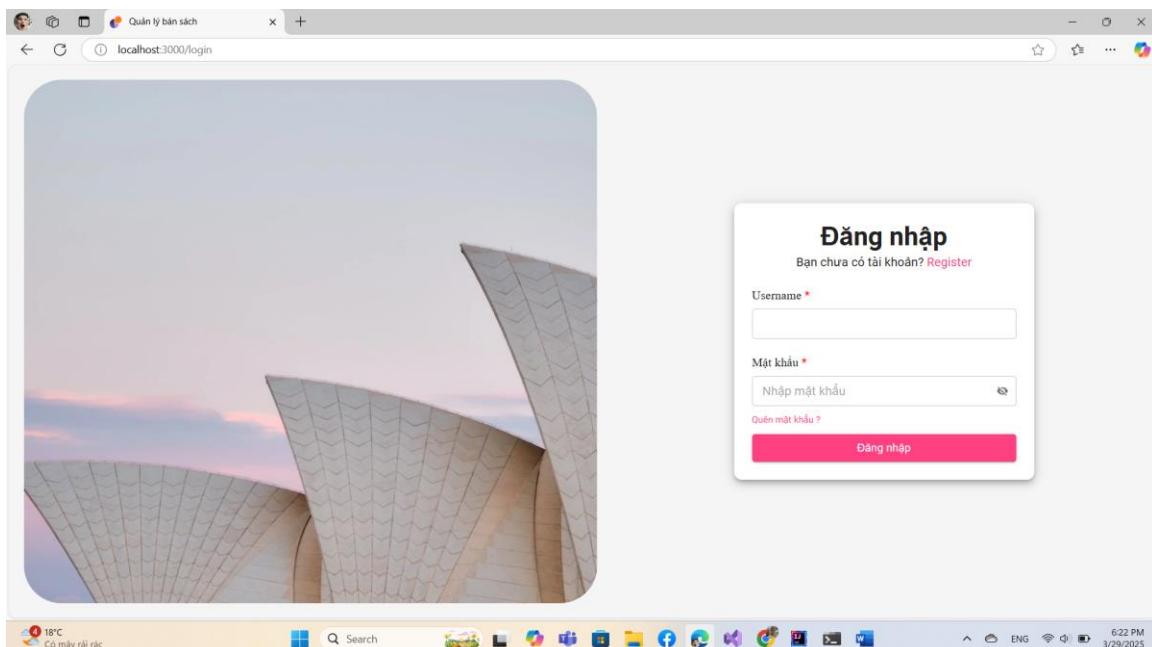
```
npm install
```

- Bước 2: Chạy ứng dụng React

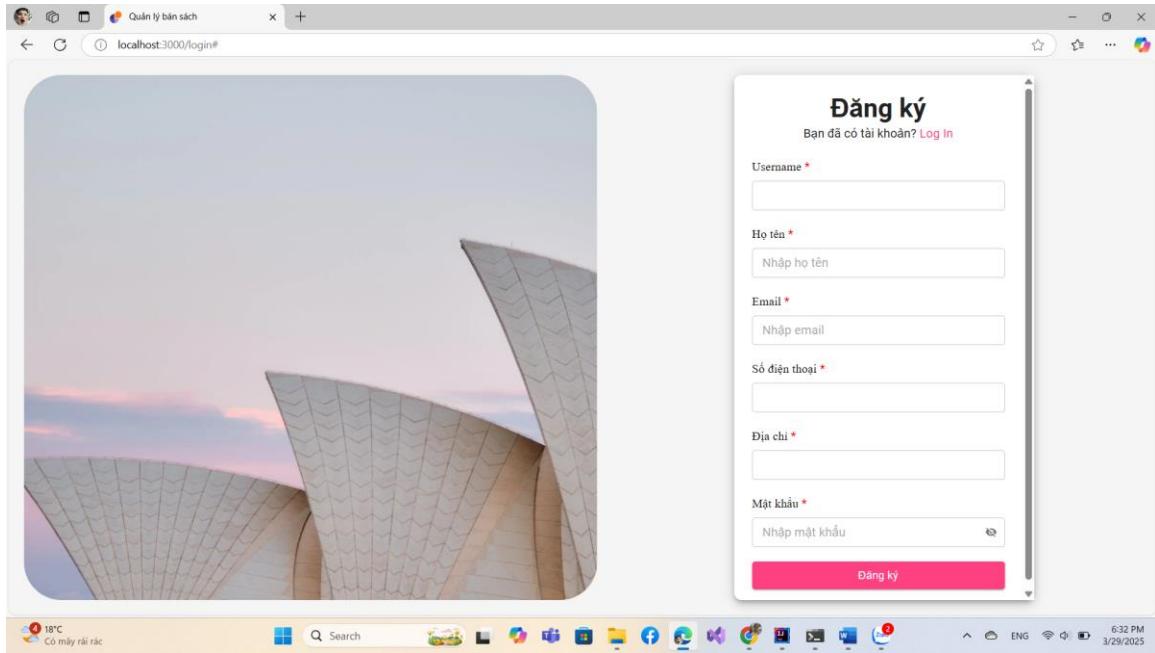
```
npm start
```

3.3. Giao diện chương trình

3.3.1 Giao diện đăng nhập, đăng ký



Hình 42: Giao diện đăng nhập



Hình 43: Giao diện đăng ký

3.3.2 Giao diện trang quản lý (Admin)

Hình 44: Giao diện tổng quan (trang quản lý)

The screenshot shows a web-based application titled "Book Heaven". On the left, there is a sidebar with a logo and four menu items: "Tổng quan", "Quản lý sách" (selected), "Quản lý đơn hàng", and "Quản lý danh mục". The main content area has a title "Book Heaven" and a subtitle "Quản lý sách". At the top right, there is a user profile icon and a button "+ Thêm sách". Below this is a table listing books with columns: STT, Ánh, Tiêu đề, Tác giả, Thể loại, Nhà xuất bản, Năm xuất bản, Giá, Số lượng, Mô tả, and Thao tác. The table contains 4 rows of book data. At the bottom of the table, there are pagination controls: "Rows per page: 10", "1–10 of 10", and navigation arrows.

Hình 45: Giao diện quản lý sách

This screenshot shows the same application interface as Figure 45, but with a modal dialog box in the center titled "Thêm Sách". The dialog contains fields for "Tiêu đề" (Title), "Tác giả" (Author), "Giá" (Price), "Thể loại" (Category), "Số lượng" (Quantity), "Ngày xuất bản" (Publication Date), and "Nhà xuất bản" (Publisher). There is also a "Mô tả" (Description) field and two buttons at the bottom: "Hủy" (Cancel) and "Thêm" (Add). The background table and sidebar are visible but dimmed.

Hình 46: Giao diện thêm sửa xóa sách

STT	Tên khách hàng	Email	Ngày đặt hàng	Tổng tiền	Lý do hủy	Trạng thái	Thao tác
1	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	18750.00		Delivering	
2	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	56250.00		Đang chờ	
3	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	47973.00		Đang chờ	
4	PVKhang	aaaaaaaaaaaa@gmail.com	27-12-2024	38970.00		Xác nhận Từ chối	
5	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	45000.00		Completed	

Hình 47: Giao diện quản lý đơn hàng

STT	Tên danh mục	Mô tả	Thao tác
1	Fiction		
2	Science		
3	Biography		
4	Children		
5	Mystery		

Hình 48: Giao diện quản lý danh mục

STT	Username	Email	Họ tên	Số điện thoại	Địa chỉ	Ngày tạo
1	customer1	customer1@example.com	John Doe	1234567890	123 Elm Street	2024-01-01
2	khang	22010197@st.phenikaa-uni.edu.vn	PVKharg	0975766528	Hà Nội	2024-11-03
3	khang2	jgejoekf@gmail.com	PVKdz	0988888888	wegawrg agreg	2024-11-11
4	khangdz	eijfkjdk@gmail.com	PVKdz	0999999999	etaweg aergw	2024-11-11
5	thinhtran3803	thinhtran383.au@gmail.com	Thinh Tran Duc	0947445420	Ha Nam	2024-11-03
6	khang3	khaekgekr@gmail.com	PVKdz	0987888888	egwaeg	2024-11-11
7	Thinhtran382	thinhtran382.cina@gmail.com	Thinh	0916.66.16.16	Ha Nam	2024-11-11

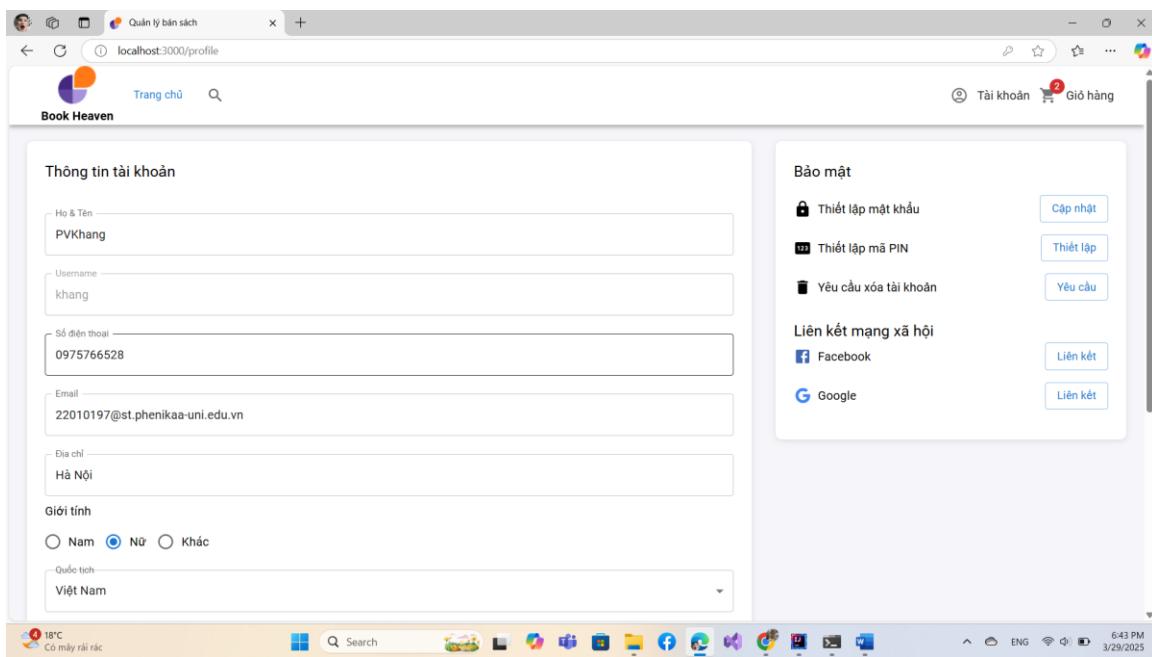
Rows per page: 10 < 1–7 of 7 >

Hình 49: Giao diện quản lý tài khoản khách hàng

3.3.3 Giao diện chính của shop (trang khách hang)

Category	Books
Biography	Ngôi nhà của Người Hà Lan
Children	Cuốn theo chiều gió
Mystery	Stich and Lilo
Ngôn tình	Thiên đường của bạn?
Tiểu thuyết	Fuki Conan

Hình 50: Giao diện trang chủ



Hình 51: Giao diện thông tin cá nhân

The screenshot shows a web browser window for 'Quản lý bán sách' at 'localhost:3000/orders/history'. The title is 'Đơn hàng của tôi' (My Orders). The table lists eight orders with columns: Mã đơn hàng (Order ID), Khách hàng (Customer), Email, Ngày đặt (Date Placed), Tổng tiền (Total Price), Lý do hủy (Reason Canceled), Trạng thái (Status), and Thao tác (Action). The status column includes dropdowns for 'Delivering', 'Đang chờ', and 'Completed'. The action column includes edit icons. The table has a vertical scroll bar on the right.

Mã đơn hàng	Khách hàng	Email	Ngày đặt	Tổng tiền	Lý do hủy	Trạng thái	Thao tác
Đơn hàng #136829	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	18.750 đ		Delivering	
Đơn hàng #340080	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	56.250 đ		Đang chờ	
Đơn hàng #951816	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	47.973 đ		Đang chờ	
Đơn hàng #69202	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	45.000 đ		Completed	
Đơn hàng #358811	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	10.000 đ		Completed	
Đơn hàng #476280	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	12.990 đ		Completed	
Đơn hàng #556702	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	278.193 đ		Completed	

Hình 52: Giao diện quản lý đơn hàng (dành cho khách hàng)

Mã đơn hàng	Khách hàng	Email	Ngày đặt	Tổng tiền	Lý do hủy	Trạng thái	Thao tác
Đơn hàng #136829	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	18.750 đ		Delivering	
Đơn hàng #340080	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	56.250 đ		Đang chờ	
Đơn hàng #951816	PVKhang	22010197@st.phenikaa-uni.edu.vn	10-03-2025	47.973 đ		Đang chờ	
Đơn hàng #69202	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	45.000 đ		Completed	
Đơn hàng #358811	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	10.000 đ		Completed	
Đơn hàng #476280	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	12.990 đ		Completed	
Đơn hàng #556702	PVKhang	22010197@st.phenikaa-uni.edu.vn	11-11-2024	278.193 đ		Completed	

Hình 53: Giao diện thông tin đơn hàng

Sản phẩm	Giá	Số lượng	Tổng tiền
Ngôi Nhà Của Người Hà Lan	199,200 đ	1	199,200 đ
Cuốn theo chiều gió	196,700 đ	1	196,700 đ

KHUYẾN MÃI

MÃ FREESHIP 20K CHO ĐƠN HÀNG TỪ 250K
Áp dụng duy nhất ngày 10.10
Có thể áp dụng đồng thời nhiều mã

Nhận quà

Tổng Số Tiền (gồm VAT) **395,900 đ**

THANH TOÁN

Đăng ký nhận ưu đãi mới của chúng tôi!

Hình 54: Giao diện thanh toán

CHƯƠNG 4: ĐÁNH GIÁ VÀ KẾT LUẬN

4.1 Đánh giá kết quả đạt được và tồn tại với yêu cầu ban đầu

Sau quá trình phát triển và triển khai hệ thống web bán sách trực tuyến, nhóm đã đạt được nhiều kết quả tích cực nhưng cũng còn một số tồn tại cần cải thiện.

4.1.1 Kết quả đạt được

Hoàn thành đầy đủ các chức năng cốt lõi: Ứng dụng hỗ trợ tìm kiếm sách, đặt hàng, thanh toán và quản lý đơn hàng theo đúng quy trình đã phân tích ban đầu.

Thiết kế giao diện thân thiện, dễ sử dụng: Nhóm đã tập trung vào trải nghiệm người dùng (UI/UX), giúp khách hàng dễ dàng thao tác trên cả nền tảng web và thiết bị di động.

Xây dựng cơ sở dữ liệu và API có tính mở rộng: Hệ thống quản lý sách, người dùng, đơn hàng và kho hàng được thiết kế theo mô hình quan hệ, đảm bảo dễ dàng nâng cấp trong tương lai.

Triển khai các chức năng bảo mật cơ bản: Ứng dụng đã áp dụng cơ chế xác thực và bảo mật khi đăng nhập, cũng như đảm bảo an toàn dữ liệu trong quá trình thanh toán trực tuyến.

4.1.2 Tồn tại và hạn chế

Chưa tối ưu quản lý kho hàng: Hệ thống chưa có tính năng kiểm soát số lượng tồn kho chi tiết, dẫn đến khó khăn trong việc theo dõi hàng hóa và cảnh báo khi hàng sắp hết.

Thiếu tính năng cá nhân hóa nội dung: Hiện tại, hệ thống chưa hỗ trợ gợi ý sách dựa trên sở thích và lịch sử mua hàng của người dùng. Điều này làm giảm trải nghiệm khách hàng và khả năng gia tăng doanh số.

Hệ thống thanh toán và giao hàng còn hạn chế: Ứng dụng chưa hỗ trợ nhiều phương thức thanh toán trực tuyến (ví điện tử, cỗng thanh toán đa dạng) và chưa tích hợp đầy đủ với các đơn vị vận chuyển lớn.

Hiệu suất và khả năng mở rộng: Khi số lượng người dùng tăng cao, hệ thống có thể gặp tình trạng quá tải do chưa áp dụng các kỹ thuật cân bằng tải và tối ưu hiệu suất truy vấn.

4.2. Hướng cải thiện và bảo trì (nâng cấp) sau này

Để khắc phục các hạn chế và nâng cao chất lượng hệ thống, nhóm đề xuất các giải pháp phát triển và bảo trì trong tương lai:

4.2.1 Cải thiện và bổ sung tính năng

Hoàn thiện chức năng quản lý kho hàng: Xây dựng hệ thống kiểm soát tồn kho, tự động cập nhật số lượng khi có đơn hàng mới và gửi cảnh báo khi sách sắp hết hàng.

Phát triển hệ thống gợi ý sách thông minh: Sử dụng thuật toán phân tích hành vi người dùng để đề xuất sách dựa trên lịch sử tìm kiếm và mua hàng.

Mở rộng phương thức thanh toán: Tích hợp thêm các cỗng thanh toán điện tử như MoMo, VNPay, ZaloPay để tăng sự tiện lợi cho người dùng.

Nâng cao trải nghiệm vận chuyển: Liên kết với các đối tác giao hàng lớn như GHN, GHTK để tối ưu quy trình vận chuyển và theo dõi đơn hàng.

4.2.2 Tối ưu hiệu suất và khả năng mở rộng

Sử dụng hệ thống phân tán và cân bằng tải: Áp dụng các giải pháp như load balancing, caching và microservices để đảm bảo hệ thống hoạt động ổn định ngay cả khi lượng truy cập lớn.

Tối ưu cơ sở dữ liệu: Cải tiến hệ thống truy vấn dữ liệu, sử dụng indexing và caching để giảm thời gian xử lý đơn hàng và tìm kiếm.

Cải thiện bảo mật: Triển khai các cơ chế mã hóa dữ liệu nhạy cảm, bảo vệ thông tin người dùng và giao dịch trước các cuộc tấn công an ninh mạng.

4.3 Kết luận

Dự án hệ thống bán sách trực tuyến đã đạt được những mục tiêu quan trọng, đáp ứng phần lớn yêu cầu đề ra. Tuy nhiên, vẫn còn một số hạn chế cần cải thiện trong tương lai. Với các kế hoạch nâng cấp và bảo trì hợp lý, hệ thống có thể phát triển mạnh mẽ hơn, mang lại trải nghiệm tốt hơn cho người dùng và đảm bảo khả năng mở rộng lâu dài.

TÀI LIỆU THAM KHẢO

1. <https://spring.io/why-spring>
2. <https://react.dev/>
3. <https://redis.io/>
4. Custom Software Requirements Specification Document Example (International Standard) - Software Development Company
5. IT4490: Software Design and Construction - Nguyen Thi Thu Trang, trangntt@soict.hust.edu.vn.
6. IEEE Recommended Practice for Software Requirements Specifications," in IEEE Std 830-1998, vol., no., pp.1-40, 20 Oct. 1998