

BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG



BÁO CÁO THỰC TẬP
TỐT NGHIỆP ĐẠI HỌC

Đề tài: “ XÂY DỰNG ỨNG DỤNG WEB TÌM KIẾM
ĐẶT MUA SÁCH TRỰC TUYẾN”

Người hướng dẫn : TH.S NGUYỄN ANH HÀO
Sinh viên thực hiện : PHAN XUÂN HUYNH
Mã số sinh viên : N20DCCN023
Lớp : D20CQCNPM01-N
Khoa : 2020-2025
Ngành : CÔNG NGHỆ THÔNG TIN
Hệ : ĐẠI HỌC CHÍNH QUY

TP.HCM, tháng 07 /2024

**BỘ THÔNG TIN VÀ TRUYỀN THÔNG
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**BÁO CÁO THỰC TẬP
TỐT NGHIỆP ĐẠI HỌC**

**Đề tài: “ XÂY DỰNG ỨNG DỤNG WEB TÌM KIẾM
ĐẶT MUA SÁCH TRỰC TUYẾN”**

Người hướng dẫn : **TH.S NGUYỄN ANH HÀO**
Sinh viên thực hiện : **PHAN XUÂN HUYNH**
Mã số sinh viên : **N20DCCN023**
Lớp : **D20CQCNPM01-N**
Khoá : **2020-2025**
Ngành : **CÔNG NGHỆ THÔNG TIN**
Hệ : **ĐẠI HỌC CHÍNH QUY**

TP.HCM, tháng 07 /2024

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập- Tự do- Hạnh phúc

TP. Hồ Chí Minh, ngày tháng năm 2024

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN
THỰC TẬP TỐT NGHIỆP ĐẠI HỌC

- Tên đề tài: Xây dựng ứng dụng web tìm kiếm đặt mua sách trực tuyến
- Sinh viên: Phan Xuân Huynh Lớp: D20CQCNPM01-N
- Giáo viên hướng dẫn: Th.S Nguyễn Anh Hào
- Nơi công tác: khoa Công nghệ thông tin 2

NỘI DUNG NHẬN XÉT

1. Đánh giá chung:

.....
.....
.....
.....

2. Đánh giá chi tiết:

.....
.....
.....
.....
.....
.....
.....
.....

3. Nhận xét về tinh thần, thái độ làm việc:

.....
.....
.....

4. Kết luận:

.....
.....
.....

5. Điểm hướng dẫn:

GIẢNG VIÊN HƯỚNG DẪN
(Ký, ghi rõ họ tên)

LỜI CẢM ƠN

Đề tài “Xây dựng ứng dụng đặt mua sách trực tuyến” là nội dung nghiên cứu và làm luận văn thực tập tốt nghiệp sau bốn năm theo học chương trình đại học chuyên ngành Công nghệ thông tin tại trường Học viện Công Nghệ Bưu Chính Viễn Thông cơ sở TP. HCM của em.

Để hoàn thành quá trình nghiên cứu và hoàn thiện luận văn này, lời đầu tiên em xin chân thành cảm ơn sâu sắc đến thầy Nguyễn Anh Hào. Thầy đã trực tiếp chỉ bảo và hướng dẫn em trong suốt quá trình nghiên cứu để em hoàn thiện luận văn này. Ngoài ra em xin chân thành cảm ơn các thầy, cô trong Khoa Công nghệ thông tin cùng các bạn bè đã đóng góp những ý kiến quý báu cho luận văn.

Thành phố Hồ Chí Minh, tháng 7 năm 2024
Sinh viên thực hiện

Phan Xuân Huynh

MỤC LỤC

LỜI CẢM ƠN	i
MỞ ĐẦU	vii
CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI	1
1.1. Giới thiệu ngắn gọn về đề tài	1
1.2. Mục đích của đề tài	1
1.3. Mục tiêu của đề tài	1
1.4. Phương pháp tiến hành	1
CHƯƠNG 2. CƠ SỞ KHOA HỌC	3
2.1. Mô hình Boolean trong tìm kiếm	3
2.1.1. Khái niệm về mô hình Boolean	3
2.1.2. Nguyên lý hoạt động của mô hình Boolean	3
2.1.3. Áp dụng mô hình Boolean trong tìm kiếm sách	3
2.2. Công nghệ sử dụng trong phát triển PM	4
2.2.1. Spring boot	4
2.2.2. Reactjs	5
2.2.3. MySQL	5
2.2.4. RESTful API	6
2.3. Ứng dụng công nghệ trong đề tài	7
CHƯƠNG 3. PHẦN TÍCH HỆ THỐNG	8
3.1. Ngữ cảnh của hệ thống	8
3.1.1. Bối cảnh và mục tiêu của đề tài	8
3.1.2. Hiện trạng trước khi sử dụng PM	8
3.1.3. Đề xuất giải pháp của đề tài	9
3.2. Mô hình usecase	10
3.2.1. Mô tả các use case:	10
3.2.2. Biểu đồ tuần tự tương ứng với usecase	15
3.2.3. Biểu đồ hoạt động	17
3.3. Các yêu cầu đối với PM:	18
3.3.1. Từ môi trường nghiệp vụ	18
3.3.2. Từ môi trường vận hành của PM	19
3.3.3. Từ môi trường phát triển	19
CHƯƠNG 4. THIẾT KẾ PHẦN MỀM	20
4.1. Phân quyền và chức năng	20
4.2. Lược đồ usecase cho thiết kế phần mềm	20
4.2.1. Các Usecase trong hệ thống	20
4.2.2. Biểu đồ tuần tự	38
4.2.3. Biểu đồ hoạt động	42
4.3. Thiết kế cơ sở dữ liệu và ràng buộc	45
4.3.1. Xác định thực thể	45
4.3.2. Lược đồ ERD	46
4.3.3. Mô hình cơ sở dữ liệu quan hệ	46
4.3.4. Diagram	52
4.4. Tạo API cho chương trình	53
4.4.1. Trang đăng nhập, đăng ký	53
4.4.2. Trang chủ	53
4.4.3. Trang chi tiết sản phẩm	54
4.4.4. Trang tìm kiếm theo Boolean Search	54
4.4.5. Trang giỏ hàng của người dùng	55
4.4.6. Trang thông tin cá nhân	56

4.4.7. Trang đơn hàng của người dùng	57
4.4.8. Trang đặt hàng	58
4.4.9. Trang chi tiết đơn hàng	60
4.4.10. Trang thống kê	60
4.4.11. Trang quản lý đơn hàng	61
4.4.12. Trang quản lý sản phẩm	61
4.4.13. Trang quản lý thể loại	63
4.4.14. Trang quản lý nhà xuất bản	63
4.4.15. Trang quản lý tác giả	65
4.4.16. Trang quản lý người dùng	66
4.5. Thiết kế giao diện	67
4.5.1. Trang chủ	67
4.5.2. Đăng ký	67
4.5.3. Đăng nhập	68
4.5.4. Trang shop	69
4.5.5. Chi tiết sản phẩm	70
4.5.6. Giao diện form chọn tiêu chí tìm kiếm sách bằng Boolean Search	71
4.5.7. Giao diện form tìm kiếm sách theo Boolean Search	72
4.5.8. Giỏ hàng	73
4.5.9. Trang đặt hàng	74
4.5.10. Thông tin cá nhân	75
4.5.11. Danh lịch sử đơn hàng	76
4.5.12. Giao diện trang Dashboard	77
4.5.13. Giao diện quản lý sản phẩm	78
4.5.14. Giao diện quản lý thể loại	81
4.5.15. Giao diện quản lý nhà xuất bản	81
4.5.16. Giao diện quản lý tag	82
4.5.17. Giao diện quản lý đơn hàng	82
CHƯƠNG 5. KẾT LUẬN VÀ KIẾN NGHỊ	85
5.1. Kết luận	85
5.2. Kiến nghị phát triển	85
DANH MỤC TÀI LIỆU THAM KHẢO	87

DANH MỤC CÁC BẢNG, SƠ ĐỒ, HÌNH

Bảng 3.1.Yêu cầu đối với phần mềm từ môi trường nghiệp vụ	19
Bảng 3.2. Yêu cầu đối với phần mềm từ môi trường vận hàng của PM	19
Bảng 3.3. Yêu cầu đối với phần mềm từ môi trường phát triển	19
Bảng 4.1.Bảng authors	47
Bảng 4.2. Bảng authors_products	47
Bảng 4.3.Bảng categories	47
Bảng 4.4.Bảng orderdetails	48
Bảng 4.5. Bảng orders	48
Bảng 4.6. Bảng products	49
Bảng 4.7.Bảng products_tags	49
Bảng 4.8. Bảng publishers	50
Bảng 4.9. Bảng reviews	50
Bảng 4.10. Bảng tags	50
Bảng 4.11. Bảng users	51
Sơ đồ 3.1.Usecase tìm sản phẩm	10
Sơ đồ 3.2.Usecase xem chi tiết sản phẩm	11
Sơ đồ 3.3.Usecase đặt mua sách	12
Sơ đồ 3.4. Usecase quản lý sản phẩm	13
Sơ đồ 3.5. Usecase xác nhận giao hàng thành công	14
Sơ đồ 3.6. Biểu đồ tuần tự tìm kiếm sách	15
Sơ đồ 3.7. Biểu đồ tuần tự đặt mua sách	16
Sơ đồ 3.8. Biểu đồ hoạt động thêm sản phẩm vào giỏ hàng	17
Sơ đồ 3.9. Biểu đồ hoạt động đặt mua sách	18
Sơ đồ 4.1.Biểu đồ usecase tổng quát	20
Sơ đồ 4.2.Biểu đồ usecase đăng nhập	21
Sơ đồ 4.3Biểu đồ usecase đổi mật khẩu	22
Sơ đồ 4.4.Biểu đồ usecase đăng xuất	23
Sơ đồ 4.5.Biểu đồ usecase đăng ký	24
Sơ đồ 4.6.Biểu đồ usecase quản lý thông tin cá nhân	25
Sơ đồ 4.7.Biểu đồ usecase Tìm kiếm sản phẩm	26
Sơ đồ 4.8. Biểu đồ usecase thêm sản phẩm vào giỏ hàng	28
Sơ đồ 4.9.Biểu đồ usecase sửa số lượng trong giỏ hàng	29
Sơ đồ 4.10.Biểu đồ usecase xóa sản phẩm trong giỏ hàng	30
Sơ đồ 4.11. Biểu đồ usecase đặt hàng	31
Sơ đồ 4.12. Biểu đồ usecase xem lịch sử đơn hàng	33
Sơ đồ 4.13. Biểu đồ usecase duyệt đơn hàng	34
Sơ đồ 4.14. Biểu đồ usecase thêm sản phẩm	35
Sơ đồ 4.15. Biểu đồ usecase sửa sản phẩm	36
Sơ đồ 4.16. Biểu đồ usecase xóa sản phẩm	37

Sơ đồ 4.17. Biểu đồ tuần tự đăng nhập	38
Sơ đồ 4.18. Biểu đồ tuần tự đăng ký	39
Sơ đồ 4.19. Biểu đồ tuần tự xem giỏ hàng	39
Sơ đồ 4.20. Biểu đồ tuần tự tiêm kiểm sản phẩm	40
Sơ đồ 4.21. Biểu đồ tuần tự đặt hàng	40
Sơ đồ 4.22. Biểu đồ tuần tự duyệt đơn hàng	41
Sơ đồ 4.23. Biểu đồ tuần tự thêm sản phẩm	41
Sơ đồ 4.24. Biểu đồ hoạt động đăng nhập	42
Sơ đồ 4.25. Biểu đồ hoạt động đổi mật khẩu	43
Sơ đồ 4.26. Biểu đồ hoạt động thêm sản phẩm vào giỏ	44
Sơ đồ 4.27. Biểu đồ hoạt động đặt hàng	45
Sơ đồ 4.28. Lược đồ ERD	46

Hình 2.1. Giới thiệu Spring boot	4
Hình 2.2. Giới thiệu MySQL	5
Hình 2.3. Giới thiệu RESTful API	6
Hình 4.1. Diagram trong hệ thống	52
Hình 4.2. Giao diện trang chủ	67
Hình 4.3. Giao diện đăng kí	68
Hình 4.4. Giao diện đăng nhập	68
Hình 4.5. Giao diện trang shop	69
Hình 4.6. Giao diện trang chi tiết sản phẩm	70
Hình 4.7. Giao diện chọn tiêu chí tìm kiếm Boolean Search	71
Hình 4.8. Giao diện tìm kiếm sách Boolean Search	72
Hình 4.9. Giao diện giỏ hàng người dùng	73
Hình 4.10. Trang đặt hàng của người dùng	74
Hình 4.11. Giao diện thông tin cá nhân	75
Hình 4.12. Giao diện danh sách đơn hàng của người dùng	76
Hình 4.13. Giao diện trang Dashboard	77
Hình 4.14. Giao diện trang quản lý sản phẩm	78
Hình 4.15. Form thêm mới sản phẩm	79
Hình 4.16. Giao diện quản lý thể loại	81
Hình 4.17. Giao diện quản lý nhà xuất bản	81
Hình 4.18. Giao diện quản lý tag sách	82
Hình 4.19. Giao diện quản lý đơn hàng đã đặt của khách	82
Hình 4.20. Giao diện quản lý đơn hàng đang giao của khách	83
Hình 4.21. Giao diện quản lý đơn hàng đã hủy của khách	83
Hình 4.22. Giao diện quản lý đơn hàng đã giao đến cho khách	83
Hình 4.23. Giao diện chi tiết đơn hàng khi admin xem thông tin đơn hàng	84

KÍ HIỆU CÁC CỤM TỪ VIẾT TẮT

API	Application Program Interface
SQL	Structured Query Language
PM	Phần mềm
SDT	Số điện thoại
ID	Identification
ERD	Entity Relationship Diagram
JS	Javascript
REST	Representational State Transfer
GHN	Dịch vụ Giao hàng nhanh

MỞ ĐẦU

Ngày nay, đời sống con người được nâng cao, kéo theo nhu cầu giải trí và học hỏi cũng phát triển không ngừng. Đọc sách đã trở thành một phần quan trọng trong đời sống tinh thần của nhiều người. Các nền tảng mua bán sách trực tuyến đã ra đời để đáp ứng nhu cầu này. Tuy nhiên, việc tìm kiếm và mua sách trực tuyến vẫn còn nhiều hạn chế, đặc biệt là khi người dùng muốn tìm kiếm sách theo nhiều tiêu chí khác nhau và so sánh giá cả từ nhiều nguồn. Để đáp ứng tốt hơn nhu cầu của người dùng, cần có những công cụ tìm kiếm và đặt mua sách trực tuyến hiệu quả, tiện lợi.

Đề tài “Website tìm kiếm và đặt mua sách trực tuyến” được chọn làm đề tài thực tập tốt nghiệp của em, mong có thể góp phần nâng cao chất lượng và hiệu quả của công tác tìm kiếm và mua sắm sách trực tuyến.

Đề tài được nghiên cứu trong phạm vi hoạt động mua bán sách trực tuyến, dựa vào những tài liệu và thông tin thu thập được từ nhiều nguồn và tham khảo các tài liệu về công nghệ và mô hình tìm kiếm để phát triển một hệ thống trợ giúp tìm kiếm và đặt mua sách. Công cụ và môi trường để phát triển hệ thống bao gồm hệ quản trị cơ sở dữ liệu MySQL Server, ngôn ngữ lập trình Java với Spring Boot, và ReactJS cho phần giao diện người dùng. Chương trình được phát triển sẽ hỗ trợ rất tốt việc tìm kiếm và mua sách trực tuyến, đem đến sự tiện lợi, tiết kiệm công sức

CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI

1.1. Giới thiệu ngắn gọn về đề tài

Đề tài “Website tìm kiếm và đặt mua sách trực tuyến” nhằm phát triển một ứng dụng web giúp người dùng dễ dàng tìm kiếm và mua sách trực tuyến. Ứng dụng này sử dụng mô hình Boolean để cải thiện hiệu quả tìm kiếm dựa trên nhiều tiêu chí khác nhau. Hệ thống cũng hỗ trợ quy trình đặt mua sách với các chính sách ưu đãi và tính phí giao hàng, mang lại trải nghiệm tốt hơn cho người dùng và hỗ trợ hiệu quả cho các nhà sách và nhà xuất bản.

1.2. Mục đích của đề tài

Phần mềm của đề tài được thiết kế để cung cấp một công cụ tìm kiếm và đặt mua sách trực tuyến hiệu quả. Ứng dụng này giúp người dùng tìm kiếm sách theo nhiều tiêu chí và trọng số khác nhau, đặt mua sách một cách tiện lợi với các chính sách ưu đãi và phí giao hàng hợp lý. Lợi ích của PM này bao gồm:

- Nâng cao trải nghiệm mua sắm sách trực tuyến cho người dùng.
- Giúp người dùng dễ dàng tìm kiếm và so sánh sách từ nhiều nguồn khác nhau.
- Tối ưu hóa quy trình đặt mua sách, tiết kiệm thời gian và công sức cho người dùng.

1.3. Mục tiêu của đề tài

Phần mềm giải quyết các vấn đề chính sau:

- Tìm kiếm sách hiệu quả: Sử dụng mô hình Boolean để tìm kiếm sách theo nhiều tiêu chí như tên sách, tác giả, thể loại, nhà xuất bản, năm xuất bản, từ khóa tùy chỉnh, số lượng mua và đánh giá sao. Người dùng có thể gán trọng số cho từng tiêu chí để có kết quả tìm kiếm ưu tiên.
- Đặt mua sách tiện lợi: Hỗ trợ quy trình đặt mua sách với các chính sách ưu đãi dựa trên phân loại khách hàng (thân thiết, VIP, v.v.) và tính phí giao hàng dựa trên trọng lượng sách và khu vực giao hàng.

Đối tượng chính của PM là người dùng cá nhân muốn tìm kiếm và mua sách trực tuyến, cũng như các nhà sách và nhà xuất bản cần một công cụ hỗ trợ quản lý và kinh doanh sách.

1.4. Phương pháp tiến hành

Phương pháp tiến hành được áp dụng trong đề tài là phân tích và thiết kế hệ thống theo hướng đối tượng. Các bước cụ thể bao gồm:

- Phân tích yêu cầu: Xác định các yêu cầu chức năng và phi chức năng của hệ thống.
- Thiết kế hệ thống: Thiết kế kiến trúc hệ thống, bao gồm giao diện người dùng, cơ sở dữ liệu, và các dịch vụ API.
- Phát triển và triển khai: Sử dụng các công nghệ như ReactJS cho frontend, Spring Boot cho backend và MySQL cho cơ sở dữ liệu để xây dựng và triển khai hệ thống.
- Kiểm thử và đánh giá: Thực hiện kiểm thử hệ thống để đảm bảo các chức năng hoạt động đúng và hiệu quả, sau đó đánh giá và cải tiến hệ thống dựa trên phản hồi từ người dùng.

Phương pháp này giúp đảm bảo hệ thống được phát triển một cách có cấu trúc, dễ bảo trì và mở rộng, đồng thời đáp ứng tốt các yêu cầu của người dùng và nhà sách

CHƯƠNG 2. CƠ SỞ KHOA HỌC

2.1. Mô hình Boolean trong tìm kiếm

2.1.1. Khái niệm về mô hình Boolean

Mô hình Boolean là một phương pháp tìm kiếm thông tin dựa trên đại số Boolean, nơi các tài liệu được biểu diễn dưới dạng các từ khóa (keywords) và các từ khóa này được kết hợp với nhau bằng các toán tử logic: AND, OR và NOT. Mô hình này giúp xác định tài liệu nào phù hợp với truy vấn của người dùng dựa trên các điều kiện logic mà người dùng đưa ra.

2.1.2. Nguyên lý hoạt động của mô hình Boolean

- AND: Kết hợp hai hoặc nhiều tiêu chí tìm kiếm, chỉ trả về các tài liệu chứa tất cả các từ khóa được liên kết.
- OR: Trả về các tài liệu chứa ít nhất một trong các từ khóa.
- NOT: Loại bỏ các tài liệu chứa từ khóa được liên kết.

Ví dụ: Truy vấn “(author: 'J.K. Rowling' AND title: 'Harry Potter') OR (category: 'Fantasy' AND NOT publisher: 'ABC Press')” sẽ trả về tất cả các sách của J.K. Rowling với tiêu đề "Harry Potter" hoặc tất cả các sách thuộc thể loại Fantasy nhưng không được xuất bản bởi ABC Press.

2.1.3. Áp dụng mô hình Boolean trong tìm kiếm sách

Trong hệ thống tìm kiếm sách của đề tài, mô hình Boolean sẽ được sử dụng để tìm kiếm sách theo nhiều tiêu chí khác nhau như tên sách, tác giả, thể loại, nhà xuất bản, năm xuất bản, từ khóa tùy chỉnh (tag), số lượng mua và đánh giá sao. Mỗi tiêu chí sẽ được người dùng gán trọng số khác nhau để ưu tiên kết quả tìm kiếm.

a. Cấu trúc tổng quát

Công thức tổng quát cho việc tính toán điểm ưu tiên của một sản phẩm dựa trên các tiêu chí tìm kiếm và trọng số tương ứng được định nghĩa như sau:

$$\text{Score} = \sum_{i=1}^n (\text{Match}_i \times \text{Weight}_i)$$

Trong đó:

- Score là điểm ưu tiên của sản phẩm do khách hàng chọn.
- n là số lượng tiêu chí tìm kiếm.
- Match_i là giá trị boolean (0 hoặc 1) cho biết tiêu chí thứ i có khớp với sản phẩm không.
- Weight_i là trọng số của tiêu chí thứ i mà người dùng đã nhập vào.

b. Các bước tính toán

Bước 1: Xác định các tiêu chí tìm kiếm

Người dùng có thể nhập vào nhiều tiêu chí tìm kiếm như tên sách, tác giả, thể loại, nhà xuất bản, năm xuất bản, tag, đánh giá sao, và giá. Mỗi tiêu chí sẽ đi kèm với một trọng số.

Ví dụ:

- Tiêu chí: Tên sách (title), Trọng số: 5
- Tiêu chí: Tác giả (author), Trọng số: 3
- Tiêu chí: Thể loại (category), Trọng số: 2

Bước 2: Kiểm tra sự khớp (Match)

Đối với mỗi tiêu chí, kiểm tra xem sản phẩm có khớp với giá trị tiêu chí đó không. Nếu có, giá trị của Match sẽ là 1, nếu không sẽ là 0.

Bước 3: Tính Điểm ưu tiên (Priority Score)

Sử dụng công thức tổng quát, tính điểm ưu tiên cho mỗi sản phẩm:

$$\text{Score} = (\text{Match}_{\text{title}} \times \text{Weight}_{\text{title}}) + (\text{Match}_{\text{author}} \times \text{Weight}_{\text{author}}) + \dots$$

Ví dụ:

- Nếu sản phẩm có tên sách khớp và trọng số của tiêu chí tên sách là 5.
- Tác giả khớp và trọng số của tiêu chí tác giả là 3.
- Thể loại không khớp và trọng số của tiêu chí thể loại là 2.

Khi đó:

$$\text{Score} = (1 \times 5) + (1 \times 3) + (0 \times 2) = 8$$

2.2. Công nghệ sử dụng trong phát triển PM

2.2.1. Spring boot



Hình 2.1. Giới thiệu Spring boot

Spring Boot là một framework Java mã nguồn mở được phát triển bởi Pivotal, giúp xây dựng các ứng dụng Java độc lập và có khả năng mở rộng cao.

Đặc điểm:

- Convention over Configuration: Spring Boot giảm thiểu cấu hình cần thiết bằng cách sử dụng các thiết lập mặc định hợp lý.
- Embedded Servers: Spring Boot tích hợp các máy chủ ứng dụng nhúng (embedded servers) như Tomcat, Jetty, giúp triển khai ứng dụng dễ dàng.

- Production-ready features: Spring Boot cung cấp các tính năng sẵn sàng cho sản xuất như giám sát, kiểm tra và cấu hình ngoài.

Ứng dụng: Trong đề tài, Spring Boot được sử dụng để phát triển các dịch vụ web và RESTful API, hỗ trợ giao tiếp giữa frontend và backend.

2.2.2. Reactjs

ReactJS là một thư viện JavaScript mã nguồn mở do Facebook phát triển, được sử dụng để xây dựng giao diện người dùng (UI).

Đặc điểm:

- Component-based: ReactJS cho phép xây dựng UI dưới dạng các thành phần độc lập (components), giúp tái sử dụng và quản lý mã nguồn dễ dàng.
- Virtual DOM: ReactJS sử dụng Virtual DOM để cập nhật giao diện người dùng một cách hiệu quả, giúp tăng hiệu suất ứng dụng.
- Declarative: ReactJS cho phép phát triển UI một cách mô tả, giúp dễ dàng hiểu và bảo trì mã nguồn.

Ứng dụng: Trong đề tài, ReactJS sẽ được sử dụng để xây dựng giao diện người dùng, bao gồm các trang tìm kiếm, giỏ hàng, thanh toán và theo dõi đơn hàng.

2.2.3. MySQL



Hình 2.2. Giới thiệu MySQL

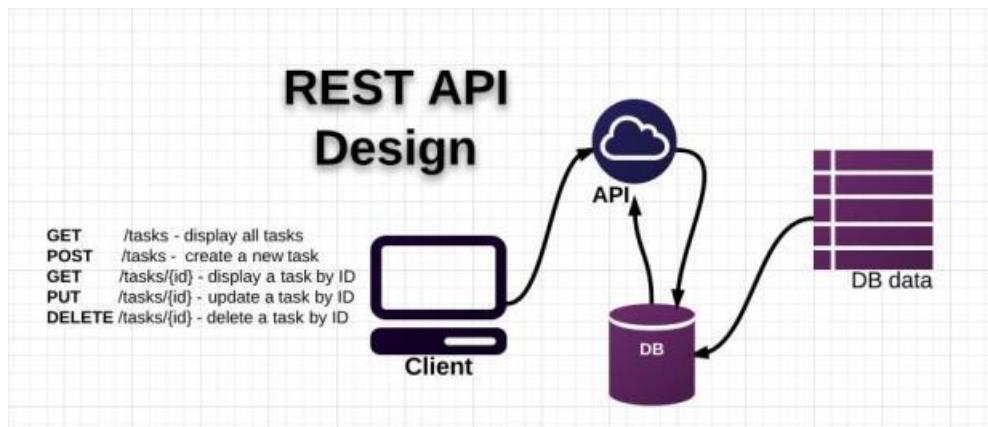
MySQL là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng. Vì MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyen, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh. Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet. Người dùng có thể tải về MySQL miễn phí từ trang chủ. MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows,

Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SGI Irix, Solaris, SunOS,..

MySQL là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

2.2.4. RESTful API

RESTful API là một tiêu chuẩn dùng trong việc thiết kế API cho các ứng dụng web (thiết kế Web services) để tiện cho việc quản lý các resource. Nó chú trọng vào tài nguyên hệ thống (tệp văn bản, ảnh, âm thanh, video, hoặc dữ liệu động...), bao gồm các trạng thái tài nguyên được định dạng và được truyền tải qua HTTP.



Hình 2.3. Giới thiệu RESTful API

Điễn giải các thành phần:

API (Application Programming Interface) là một tập các quy tắc và cơ chế mà theo đó, một ứng dụng hay một thành phần sẽ tương tác với một ứng dụng hay thành phần khác. API có thể trả về dữ liệu mà bạn cần cho ứng dụng của mình ở những kiểu dữ liệu phổ biến như JSON hay XML.

REST (Representational State Transfer) là một dạng chuyển đổi cấu trúc dữ liệu, một kiểu kiến trúc để viết API. Nó sử dụng phương thức HTTP đơn giản để tạo cho giao tiếp giữa các máy. Vì vậy, thay vì sử dụng một URL cho việc xử lý một số thông tin người dùng, REST gửi một yêu cầu HTTP như GET, POST, DELETE, vv đến một URL để xử lý dữ liệu.

RESTful API là một tiêu chuẩn dùng trong việc thiết kế các API cho các ứng dụng web để quản lý các resource. RESTful là một trong những kiểu thiết kế API được sử dụng phổ biến ngày nay để cho các ứng dụng (web, mobile...) khác nhau giao tiếp với nhau.

Chức năng quan trọng nhất của REST là quy định cách sử dụng các HTTP method (như GET, POST, PUT, DELETE...) và cách định dạng các URL cho ứng dụng web để quản

các resource. RESTful không quy định logic code ứng dụng và không giới hạn bởi ngôn ngữ lập trình ứng dụng, bất kỳ ngôn ngữ hoặc framework nào cũng có thể sử dụng để thiết kế một RESTful API. JavaScript là một ngôn ngữ lập trình của HTML và WEB. Nó là nhẹ và được sử dụng phổ biến nhất như là một phần của các trang web, mà sự thi hành của chúng cho phép Client-Side script tương tác với người sử dụng và tạo các trang web động. Nó là một ngôn ngữ chương trình thông dịch với các khả năng hướng đối tượng.

2.3. Ứng dụng công nghệ trong đề tài

- Tìm kiếm: Sử dụng mô hình Boolean để xây dựng chức năng tìm kiếm sách dựa trên các tiêu chí và trọng số do người dùng nhập vào.
- Frontend: Sử dụng ReactJS để xây dựng giao diện người dùng, đảm bảo trải nghiệm tốt nhất cho người dùng.
- Backend: Sử dụng Spring Boot để phát triển các dịch vụ web và RESTful API, hỗ trợ giao tiếp giữa frontend và backend.
- Cơ sở dữ liệu: Sử dụng MySQL để lưu trữ và quản lý dữ liệu sách, người dùng, đơn hàng và các thông tin liên quan.
- Giao tiếp: Sử dụng RESTful API để truyền tải dữ liệu giữa frontend và backend, đảm bảo hiệu suất và bảo mật.

Việc kết hợp các công nghệ này và áp dụng phương pháp phân tích và thiết kế theo hướng đối tượng sẽ giúp xây dựng một ứng dụng web tìm kiếm và đặt mua sách trực tuyến hiệu quả, đáp ứng tốt nhu cầu của người dùng và hỗ trợ hoạt động kinh doanh của nhà sách và nhà xuất bản.

CHƯƠNG 3. PHÂN TÍCH HỆ THỐNG

3.1. Ngữ cảnh của hệ thống

3.1.1. Bối cảnh và mục tiêu của đề tài

Hiện nay các hệ thống bán sách trực tuyến thông thường gặp một số vấn đề như sau:

- Khách hàng khó tìm kiếm sách phù hợp dựa trên nhiều tiêu chí khác nhau.
- Quy trình mua sách và thanh toán phức tạp, không thuận tiện.
- Thiếu chính sách ưu đãi và phân loại khách hàng rõ ràng.
- Việc quản lý đơn hàng và giao hàng còn gặp nhiều khó khăn.

Mục tiêu của đề tài là xây dựng một PM hỗ trợ tìm kiếm và mua sách trực tuyến nhằm:

- Cung cấp công cụ tìm kiếm sách hiệu quả dựa trên nhiều tiêu chí và mô hình Boolean.
- Đơn giản hóa quy trình đặt hàng và thanh toán.
- Áp dụng chính sách ưu đãi theo phân loại khách hàng.
- Hỗ trợ quản lý đơn hàng và giao hàng một cách hiệu quả.

3.1.2. Hiện trạng trước khi sử dụng PM

Trước khi có PM, các hoạt động tìm kiếm và mua bán sách được thực hiện thủ công và gặp nhiều hạn chế. Các đối tượng liên quan và sự cộng tác giữa chúng có thể được mô tả như sau:

-Khách hàng

- Tìm kiếm sách thông qua danh mục có sẵn trên website hoặc cửa hàng.
- Gửi yêu cầu mua sách qua email hoặc gọi điện thoại.
- Thanh toán thủ công qua chuyển khoản ngân hàng hoặc thanh toán khi nhận hàng.

-Nhân viên bán hàng

- Tiếp nhận yêu cầu mua sách từ khách hàng.
- Kiểm tra tồn kho và xác nhận đơn hàng.
- Xử lý đơn hàng và giao sách cho khách hàng.

-Nhân viên kho

- Quản lý tồn kho sách.
- Kiểm tra số lượng sách tồn và cập nhật thông tin lên hệ thống

Mô hình mô tả hiện trạng trước khi có PM:

- Khách hàng gửi yêu cầu mua sách đến nhân viên bán hàng.
- Nhân viên bán hàng tiếp nhận yêu cầu, xử lý đơn hàng và thanh toán.
- Nhân viên kho kiểm tra tồn kho và cập nhật thông tin.
- Shipper nhận thông tin và thực hiện giao hàng.

Nhận xét:

- Quy trình hiện tại phức tạp và tốn nhiều thời gian.
- Khách hàng gặp khó khăn trong việc tìm kiếm sách phù hợp.
- Thiếu chính sách ưu đãi và phân loại khách hàng rõ ràng.
- Việc quản lý đơn hàng và giao hàng còn gặp nhiều khó khăn.

3.1.3. Đề xuất giải pháp của đề tài

Đề giải quyết những vấn đề trên, đề tài đề xuất xây dựng một PM bán sách trực tuyến với các tính năng sau:

- Tìm kiếm sách dựa trên mô hình Boolean với nhiều tiêu chí tìm kiếm.
- Đặt mua sách trực tuyến với quy trình thanh toán đơn giản.
- Áp dụng chính sách ưu đãi theo phân loại khách hàng.
- Hỗ trợ quản lý đơn hàng và giao hàng hiệu quả.

Mô hình vận hành mới có thêm PM:

- Khách hàng tìm kiếm sách và đặt mua trực tuyến thông qua PM.
- PM tự động xác nhận đơn hàng và gửi đơn hàng cho kho hàng.
- Nhân viên bán hàng chỉ cần kiểm tra tồn kho và cập nhật thông tin.
- Shipper nhận thông tin giao hàng từ PM và thực hiện giao hàng.

Nhận xét về giá trị sử dụng của PM

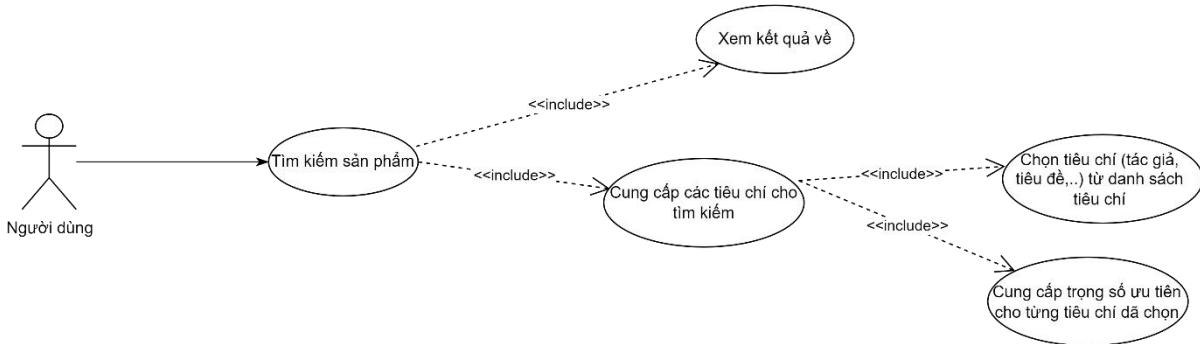
- Hiệu quả hoạt động: PM tự động hóa quy trình tìm kiếm và quản lý thông tin sách, giảm thiểu sự phụ thuộc vào nhân viên bán hàng và quản lý kho.
- Độ chính xác: Cung cấp thông tin chính xác và kịp thời cho khách hàng về tình trạng sách và đơn hàng.
- Trải nghiệm khách hàng: Cải thiện trải nghiệm khách hàng bằng cách cho phép tra cứu và đặt mua sách trực tuyến nhanh chóng và tiện lợi.

3.2. Mô hình usecase

Dưới đây là lược đồ use case cho hệ thống tìm kiếm và đặt mua sách trực tuyến:

3.2.1. Mô tả các use case:

a. Use case 1: Tìm kiếm sản phẩm



Sơ đồ 3.1. Usecase tìm sản phẩm

- **ID:** UC01
- **Ý nghĩa:** Khách hàng cần hệ thống hỗ trợ tìm kiếm sách dựa trên các tiêu chí như tên sách, tác giả, thể loại, nhà xuất bản, năm xuất bản, từ khóa tùy chỉnh, số lượng mua và đánh giá sao.
- **Actors:** Khách hàng
- **Điều kiện:** Khách hàng đã truy cập vào trang tìm kiếm sách.
- **Tương tác:**
 1. Khách hàng sẽ chọn các tiêu chí để tìm kiếm theo mong muốn từ danh sách tiêu chí tìm kiếm mà hệ thống cung cấp gồm (tên sách, tác giả, thể loại, nhà xuất bản, năm xuất bản, từ khóa tùy chỉnh, số lượng mua và đánh giá sao.)
 2. Hệ thống trả về form chứa các tiêu chí tìm kiếm theo danh sách tiêu chí mà người dùng đã chọn, gồm có ô để nhập tiêu chí và ô trọng số ưu tiên tương ứng với tiêu chí
 3. Khách hàng nhập các thông tin tìm kiếm cho các tiêu chí và cung cấp trọng số ưu tiên mong muốn đối với từng tiêu chí, trọng số ưu tiên mặc định ban đầu là 1
 4. Hệ thống tìm kiếm các sản phẩm có phù hợp các tiêu chí, và tính toán độ ưu tiên để sắp xếp danh sách trả về dựa theo độ ưu tiên của từng tiêu chí mà khách hàng cung cấp. Độ ưu tiên ban đầu của tất cả sản phẩm bằng 0,

nếu sản phẩm có thông tin phù hợp với tiêu chí thì độ ưu tiên sẽ cộng thêm trọng số độ ưu tiên của tiêu chí đó mà khách hàng cung cấp. Sau khi duyệt qua tất cả sách sẽ trả về danh sách sản phẩm có độ ưu tiên giảm dần, và chỉ lấy những sản phẩm có độ ưu tiên >0 . Công thức

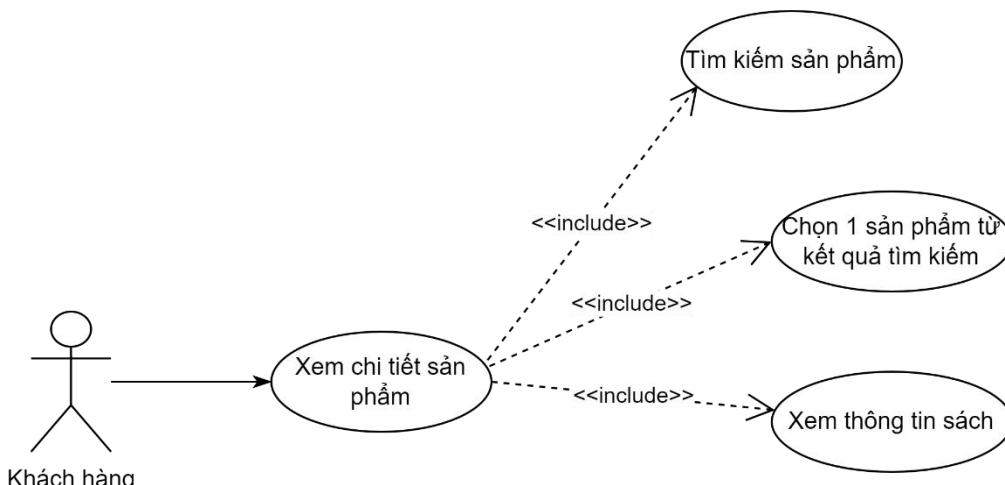
$$\text{Score} = \sum_{i=1}^n (\text{Match}_i \times \text{Weight}_i)$$

Trong đó:

- Score là điểm ưu tiên của sản phẩm do khách hàng chọn.
- n là số lượng tiêu chí tìm kiếm.
- Match_i là giá trị boolean (0 hoặc 1) cho biết tiêu chí thứ i có khớp với sản phẩm không.
- Weight_i là trọng số của tiêu chí thứ i mà người dùng đã nhập vào.

5. Khách hàng xem kết quả tìm kiếm và tìm sản phẩm theo mong muốn

b. Use case 2: Xem thông tin chi tiết sản phẩm



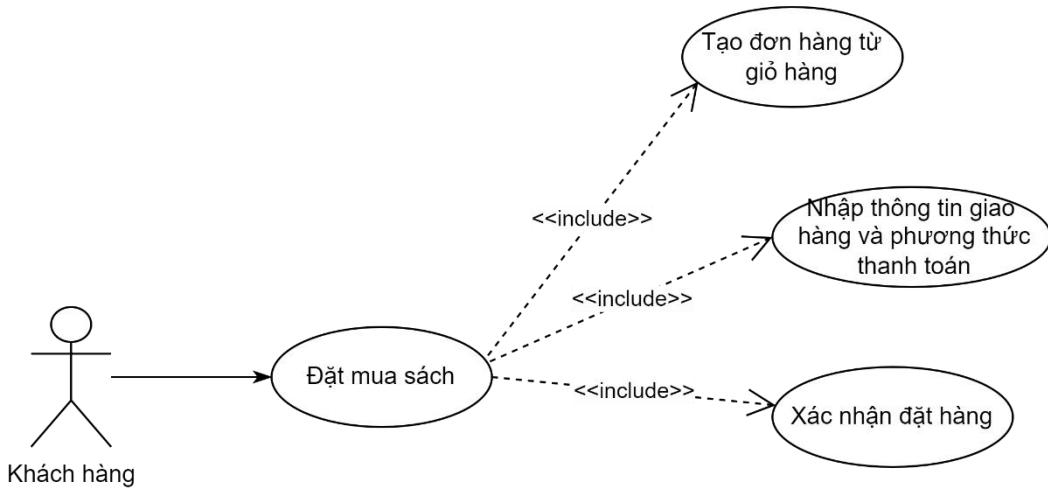
Sơ đồ 3.2. Usecase xem chi tiết sản phẩm

- **ID:** UC02
- **Ý nghĩa:** Khách hàng muốn xem chi tiết thông tin của một cuốn sách cụ thể.
- **Actors:** Khách hàng
- **Điều kiện:** Khách hàng đã tìm thấy sách từ kết quả tìm kiếm.
- **Tương tác:**

1. Khách hàng chọn một cuốn sách từ danh sách kết quả tìm kiếm.

2. Hệ thống hiển thị thông tin chi tiết của cuốn sách bao gồm tên sách, tác giả, mô tả, đánh giá, và thông tin khác.
3. Khách hàng có thể xem các đánh giá từ người dùng khác và thêm sách vào giỏ hàng.

c. Use case 3: Đặt mua sách



Sơ đồ 3.3. Usecase đặt mua sách

- **ID:** UC03
- **Ý nghĩa:** Khách hàng muốn đặt mua một hoặc nhiều cuốn sách đã chọn.
- **Actors:** Khách hàng, Hệ thống thanh toán
- **Điều kiện:** Khách hàng đã chọn sách và thêm vào giỏ hàng.
- **Tương tác:**

1. Khách hàng vào giỏ hàng, nhập địa chỉ gồm tỉnh, huyện, xã và xác nhận đơn hàng.

2. Hệ thống tính toán tổng chi phí, bao gồm phí giao hàng và các ưu đãi. Phí giao hàng tính theo trọng lượng của sản phẩm và khoảng cách giao hàng. Phí giao hàng được chia theo khu vực và khối lượng đơn hàng. (khu vực nội tỉnh, nội vùng, liên vùng, liên vùng tỉnh, liên vùng đặc biệt), và theo trọng lượng đơn hàng (từ 0-0,5kg, >3kg).

Ví dụ một đơn hàng nội thành Hà Nội nặng 2kg có cước phí là 21.000 VNĐ + (2,500 x 3) = 28.500 VNĐ/ đơn.

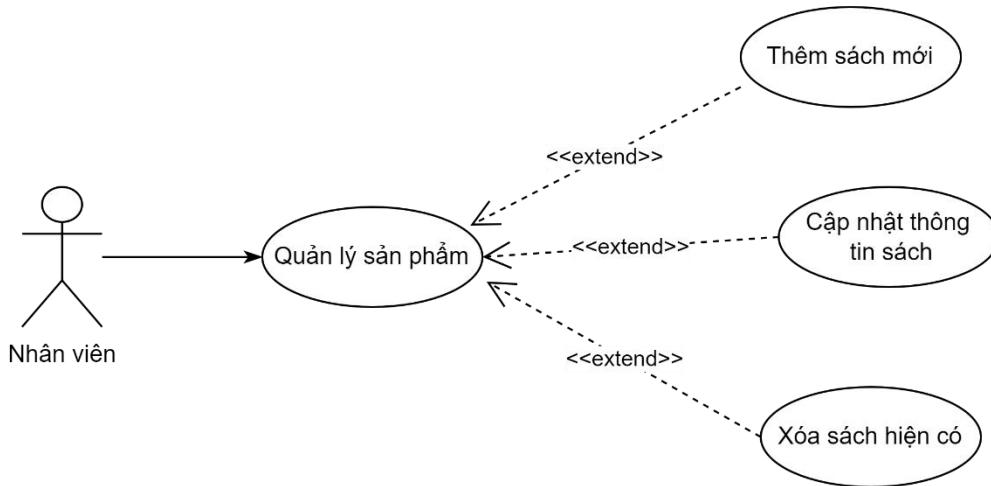
Ưu đãi: Khách hàng thân thiết: Giảm 10% phí giao hàng. Nếu đơn hàng có tổng giá trị từ 2.000.000 trở lên được giảm 2% giá trị đơn hàng.

Khách hàng VIP: Giảm 30% phí giao hàng. Nếu đơn hàng có tổng giá trị từ 2.000.000 trở lên được giảm 3% giá trị đơn hàng.

3. Khách hàng nhập thông tin giao hàng và phương thức thanh toán.

4. Hệ thống chuyển thông tin đến hệ thống thanh toán để xử lý giao dịch.
5. Hệ thống thanh toán xác nhận giao dịch thành công và thông báo cho hệ thống.
6. Hệ thống gửi xác nhận đơn hàng và thông báo giao hàng cho khách hàng.

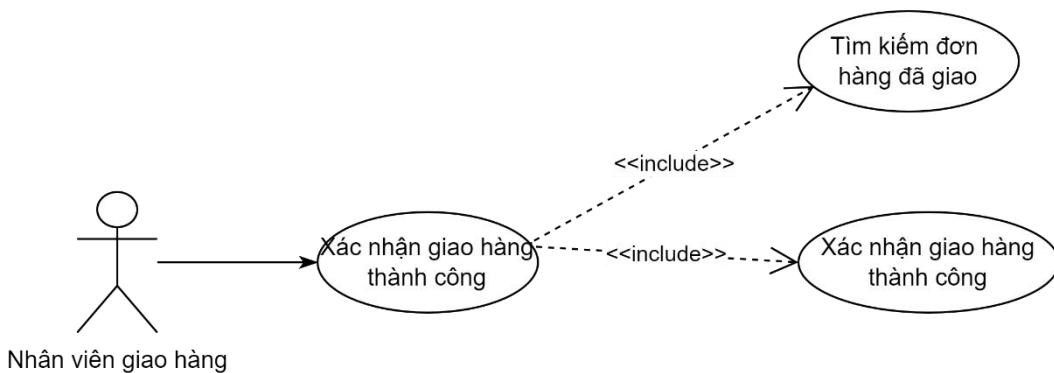
d. Use case 4: Quản lý sản phẩm



Sơ đồ 3.4. Usecase quản lý sản phẩm

- **ID:** UC04
- **Ý nghĩa:** Nhân viên bán hàng cần cập nhật thông tin sách, thêm sách mới, hoặc chỉnh sửa thông tin sách hiện có.
- **Actors:** Nhân viên bán hàng
- **Điều kiện:** Nhân viên đã đăng nhập vào hệ thống quản lý.
- **Tương tác:**
 1. Nhân viên chọn chức năng quản lý sách.
 2. Hệ thống hiển thị danh sách sách hiện có.
 3. Nhân viên có thể thêm sách mới, chỉnh sửa thông tin sách, hoặc xóa sách khỏi hệ thống.
Sách chỉ có thể xóa khi chưa được có trong đơn hàng của người dùng nào
 4. Hệ thống lưu lại các thay đổi và cập nhật thông tin sách.

e. Use case 5: Xác nhận giao hàng

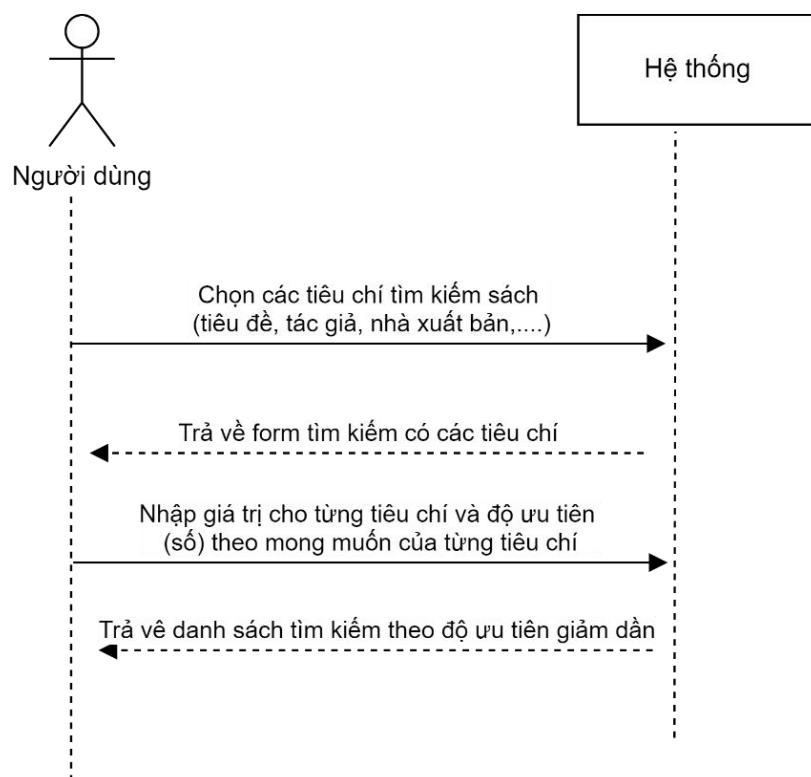


Sơ đồ 3.5. Usecase xác nhận giao hàng thành công

- **ID:** UC05
- **Ý nghĩa:** Nhân viên giao hàng cần xác nhận đã giao hàng thành công cho khách hàng.
- **Actors:** Nhân viên giao hàng
- **Điều kiện:** Đơn hàng đã được xử lý và giao cho nhân viên giao hàng.
- **Tương tác:**
 1. Nhân viên giao hàng truy cập vào hệ thống và chọn đơn hàng cần giao.
 2. Hệ thống hiển thị chi tiết đơn hàng và thông tin khách hàng.
 3. Nhân viên giao hàng xác nhận đã giao hàng thành công.
 4. Hệ thống cập nhật trạng thái đơn hàng và thông báo cho khách hàng.

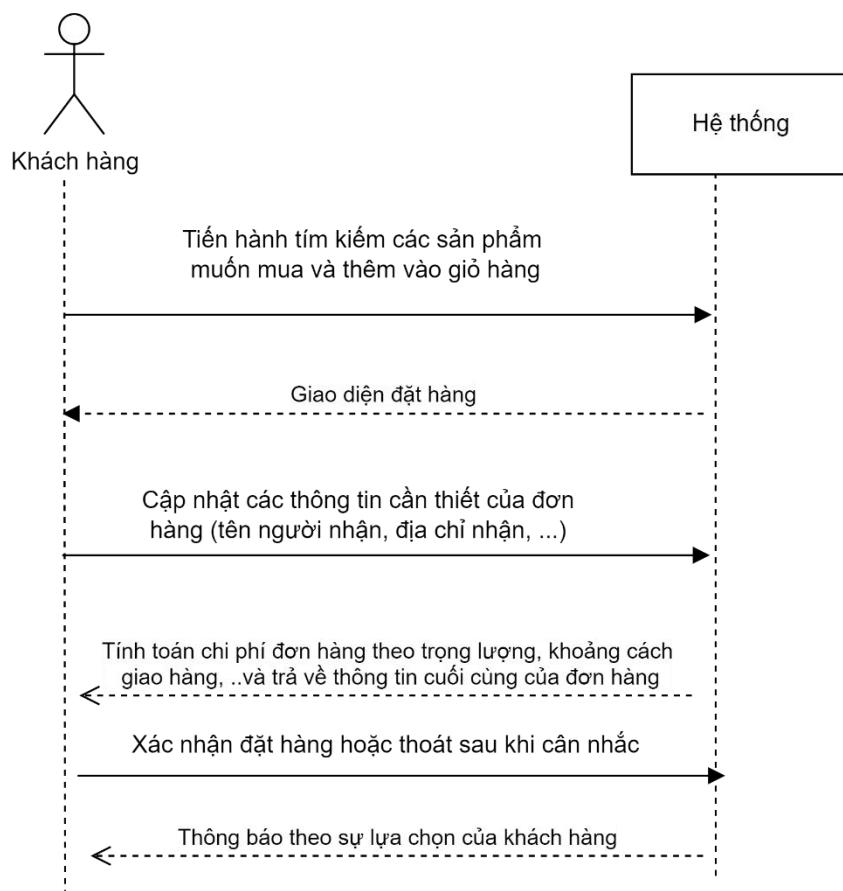
3.2.2. Biểu đồ tuần tự tương ứng với usecase

a. Biểu đồ tuần tự tìm kiếm sách



Sơ đồ 3.6. Biểu đồ tuần tự tìm kiếm sách

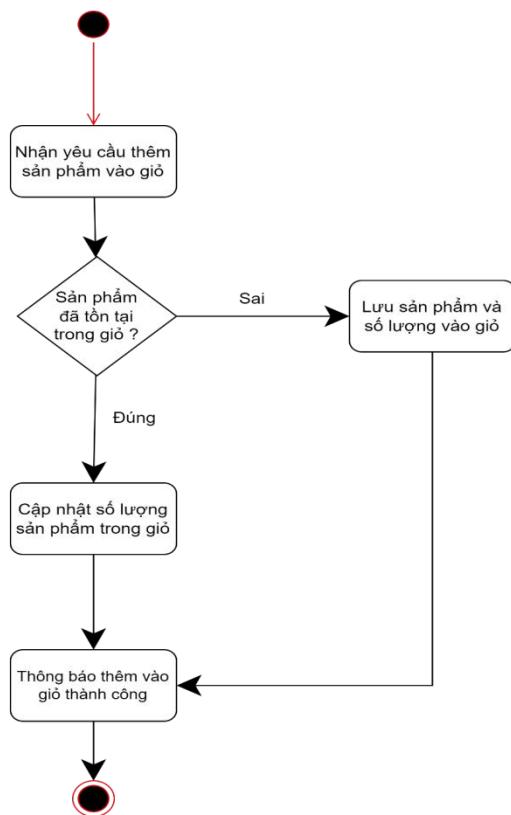
b. Biểu đồ tuần tự đặt mua sách



Sơ đồ 3.7. Biểu đồ tuần tự đặt mua sách

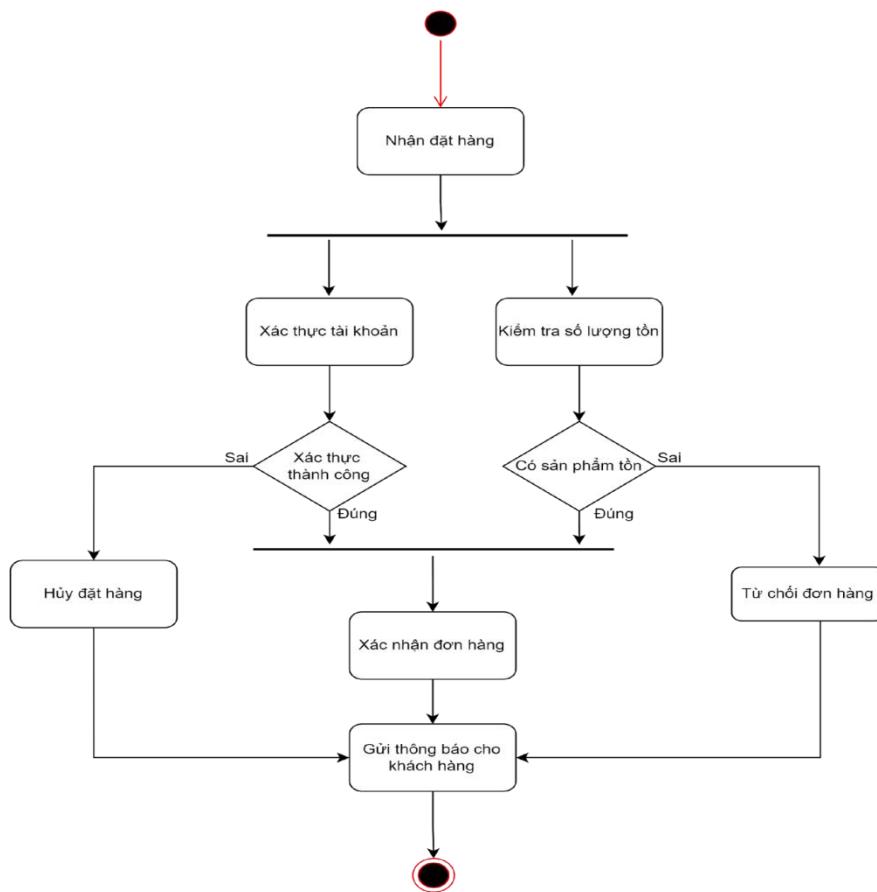
3.2.3. Biểu đồ hoạt động

a. Biểu đồ hoạt động Thêm sản phẩm vào giỏ hàng



Sơ đồ 3.8. Biểu đồ hoạt động thêm sản phẩm vào giỏ hàng

b. Biểu đồ hoạt động đặt mua sách



Sơ đồ 3.9. Biểu đồ hoạt động đặt mua sách

3.3. Các yêu cầu đối với PM:

3.3.1. Từ mô trường nghiệp vụ

UseCase	Req-ID	Nội dung yêu cầu	Stack-Holder
U01	F01.01	Hệ thống phải cho phép tìm kiếm sách dựa trên nhiều tiêu chí	Khách hàng
U02	F02.01	Hệ thống phải hỗ trợ quy trình đặt mua sách	Khách hàng
U03	F03.01	Hệ thống phải hiển thị thông tin chi tiết về sách	Khách hàng
U04	F04.01	Hệ thống phải hỗ trợ quản lý đơn hàng	Nhà sách
U05	F05.01	Hệ thống phải cho phép khách hàng đánh giá sách	Khách hàng

*Bảng 3.1. Yêu cầu đối với phần mềm từ môi trường nghiệp vụ***3.3.2. Từ môi trường vận hành của PM**

UseCase	ID	Nội dung yêu cầu	Stack-Holder
U01	F01.01	Hệ thống phải trả về kết quả tìm kiếm trong vòng 2 giây	Khách hàng
U02	F02.01	Hệ thống phải xác nhận đơn hàng trong vòng 5 giây	Khách hàng
U03	F03.01	Hệ thống phải hiển thị chi tiết sách trong vòng 2 giây	Khách hàng
U04	F04.01	Hệ thống phải cập nhật trạng thái đơn hàng ngay lập tức	Nhà sách
U05	F05.01	Hệ thống phải lưu đánh giá và cập nhật điểm đánh giá ngay lập tức	Khách hàng

*Bảng 3.2. Yêu cầu đối với phần mềm từ môi trường vận hàng của PM***3.3.3. Từ môi trường phát triển**

ID	Đối tượng	Nội dung yêu cầu	Stack-Holder
TA01	Application	Server: Intel, Windows 10, cpu I5, ssd256G, mem 8G	Khách hàng
TD02	Software	Viết code trên Visual Studio Code	Dev team
TD03	Software	Dùng ReactJS cho frontend, Spring Boot cho backend, MySQL cho cơ sở dữ liệu	Dev team

Bảng 3.3. Yêu cầu đối với phần mềm từ môi trường phát triển

CHƯƠNG 4. THIẾT KẾ PHẦN MỀM

4.1. Phân quyền và chức năng

- Phần mềm có 3 nhóm người dùng: khách vãng lai, người dùng có tài khoản admin.

- Khách vãng lai:

- Xem sản phẩm
- Tìm kiếm, lọc các sản phẩm
- Đăng ký tài khoản

- Người dùng đã đăng ký tài khoản

- Chính sửa thông tin cá nhân
- Xem sản phẩm
- Tìm kiếm, lọc các sản phẩm
- Có thể thêm sản phẩm vào giỏ hàng và tạo đơn hàng
- Xem lịch sử đơn hàng
- Đánh giá, bình luận về sản phẩm đã mua

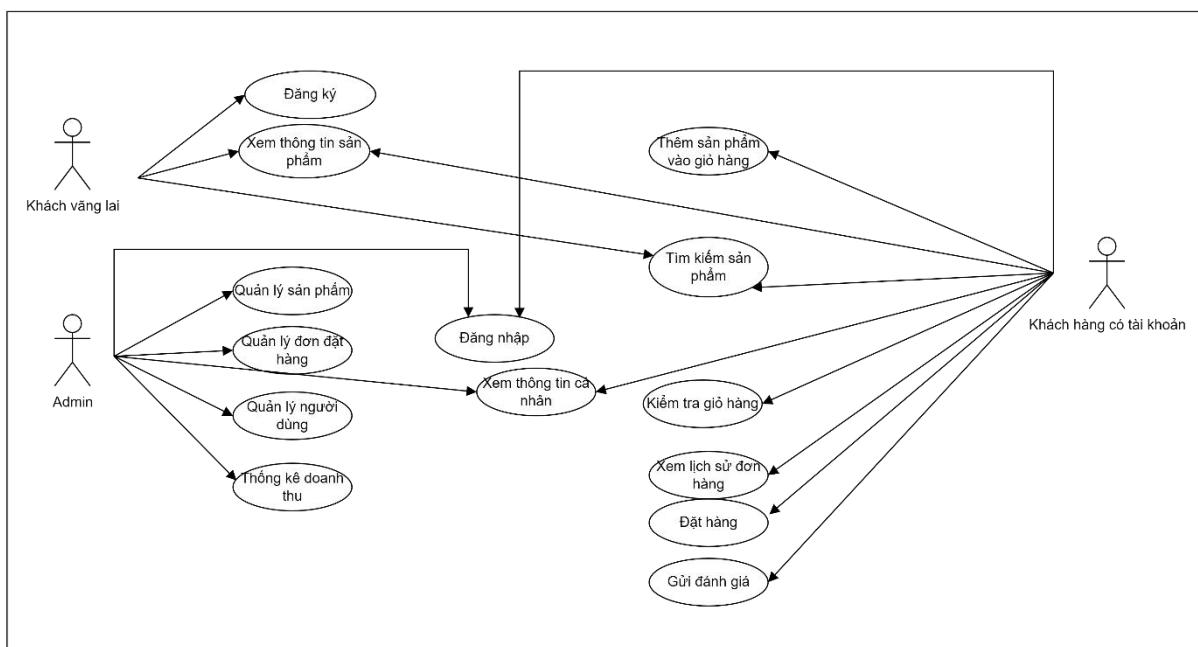
- Admin

- Xem thống kê doanh số, đơn hàng
- Quản lý sản phẩm và các thông tin liên quan đến sản phẩm
- Quản lý đơn hàng

4.2. Lược đồ usecase cho thiết kế phần mềm

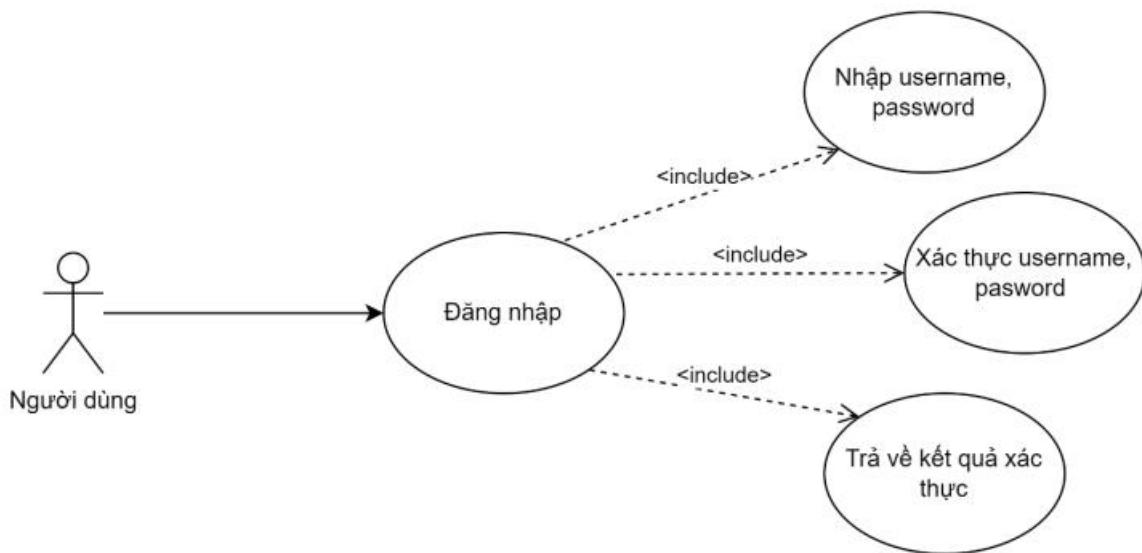
4.2.1. Các Usecase trong hệ thống

a. Usecase Tổng quát



Sơ đồ 4.1. Biểu đồ usecase tổng quát

b. Usecase Đăng nhập



Sơ đồ 4.2.Biểu đồ usecase đăng nhập

Tên Use Case	Đăng nhập
Tác nhân chính	Admin, khách hàng
Điều kiện trước	Tác nhân chưa đăng nhập vào hệ thống
Đảm bảo tối thiểu	Hệ thống hiển thị thông báo cho người dùng trong cả 2 trường hợp thành công hoặc thất bại.
Điều kiện sau	Hệ thống xác thực đúng user và chuyển đến giao diện tương ứng với Actor đó.
Chuỗi sự kiện chính	
Bước 1: Hệ thống hiển thị trang đăng nhập, yêu cầu actor nhập tên đăng nhập và mật khẩu Bước 2: Actor nhập vào tên đăng nhập và mật khẩu Bước 3: Hệ thống kiểm tra tên đăng nhập và mật khẩu có trùng khớp với thông tin trong cơ sở dữ liệu. Kiểm tra quyền người dùng.	

Bước 4: Tùy theo quyền người dùng, hệ thống hiển thị giao diện và các chức năng tương ứng.

Chuỗi sự kiện phụ

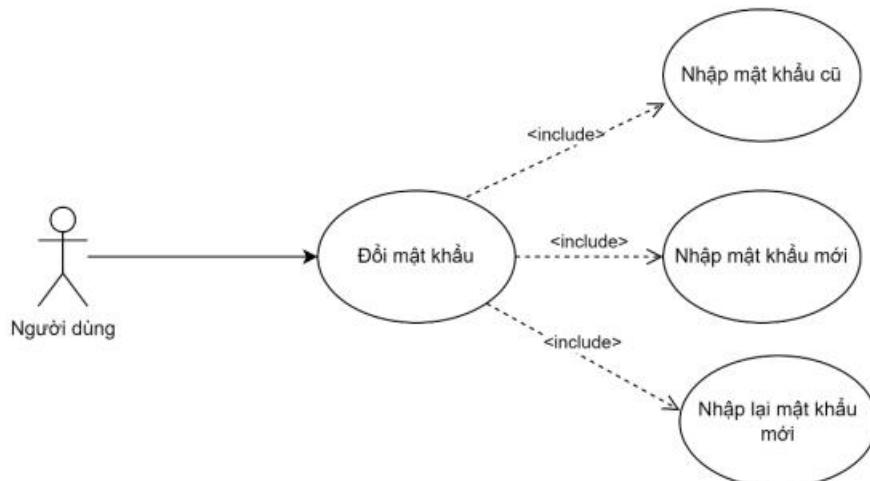
TH1: Tên đăng nhập và mật khẩu không trùng khớp với thông tin trong CSDL

1. Hiển thị thông báo là thông tin không trùng khớp
2. Giữ nguyên thông tin tên đăng nhập và mật khẩu trên trang hiện tại cho phép người dùng điều chỉnh.

TH2: Actor không nhập đủ thông tin

1. Hiển thị thông báo là thông tin không được để trống
2. Giữ nguyên thông tin mà người dùng đã nhập cho phép người dùng nhập thêm hoặc điều chỉnh.

c. Usecase đổi mật khẩu



Sơ đồ 4.3 Biểu đồ usecase đổi mật khẩu

Tên Use Case	Đổi mật khẩu
Tác nhân chính	Admin, khách hàng
Điều kiện trước	Tác nhân đã đăng nhập thành công vào hệ thống.
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng

Điều kiện sau	Hệ thống đổi mật khẩu thành công cho tác nhân.
Chuỗi sự kiện chính	

1. Hệ thống hiển thị trang đổi mật khẩu

2. Actor nhập mật khẩu hiện tại, nhập mật khẩu mới và nhập lại mật khẩu mới 1 lần nữa để xác nhận

3. Actor nhấn nút đổi mật khẩu.

4. Hệ thống kiểm tra mật khẩu hiện tại có trùng khớp với thông tin trong cơ sở dữ liệu.

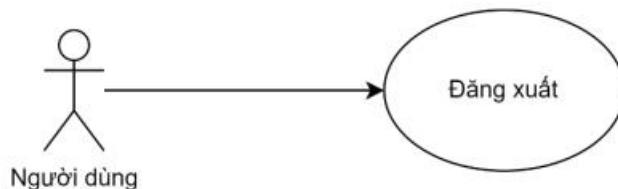
5. Hệ thống lưu lại thông tin mật khẩu mới và thông báo thành công.

Chuỗi sự kiện phụ

TH1: Actor nhập thiếu thông tin, hệ thống thông báo không được bỏ trống

TH2: Actor nhập lại mật khẩu mới không khớp với mật khẩu mới bên trên: Hệ thống thông báo xác nhận mật khẩu không đúng.

d. Usecase đăng xuất



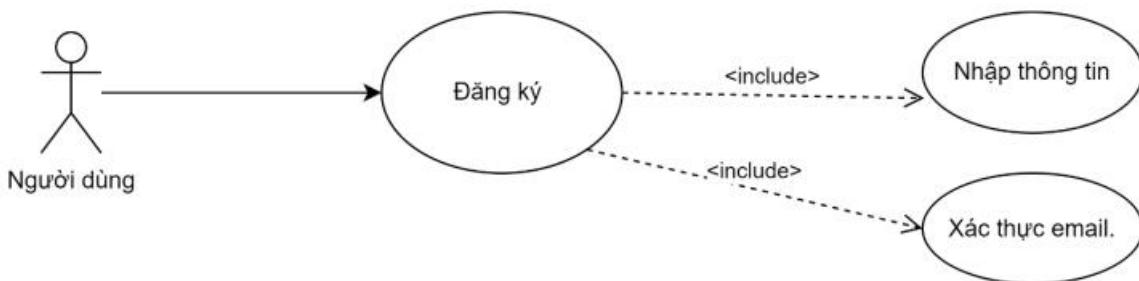
Sơ đồ 4.4.Biểu đồ usecase đăng xuất

Tên Use Case	Đăng xuất
Tác nhân chính	Admin, khách hàng
Điều kiện trước	Tác nhân đã đăng nhập thành công vào hệ thống.
Đảm bảo tối thiểu	Đăng xuất tác nhân khỏi hệ thống.
Điều kiện sau	Tác nhân được đăng xuất khỏi hệ thống, trở về giao diện trang chủ lúc chưa đăng nhập.

Chuỗi sự kiện chính

1. Hệ thống hiển thị giao diện làm việc của Actor
2. Actor nhấn nút Đăng xuất
3. Hệ thống đăng xuất tài khoản và trở về trang chủ

e. Usecase Đăng ký



Sơ đồ 4.5.Biểu đồ usecase đăng ký

Tên Use Case	Đăng ký
Tác nhân chính	Khách hàng chưa có tài khoản
Điều kiện trước	Khách hàng chưa đăng nhập, và chưa có tài khoản trong hệ thống.
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng.
Điều kiện sau	Khách hàng đăng ký được tài khoản để login vào hệ thống.

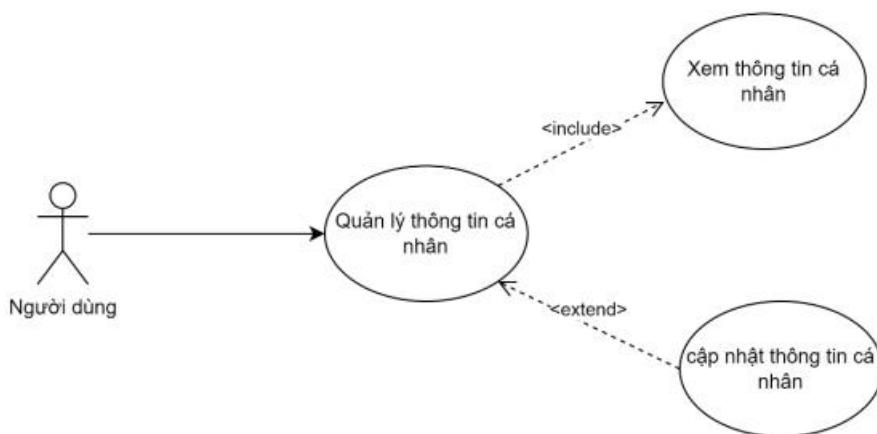
Chuỗi sự kiện chính

1. Hệ thống hiển thị giao diện đăng ký
2. Actor nhập email, username, password, nhập lại password, họ tên, ngày sinh, số điện thoại, địa chỉ. Sau đó nhấn nút đăng ký
3. Kiểm tra email đăng ký, nếu email đã đăng ký rồi thì khách hàng không được đăng ký tài khoản. Nếu email chưa đăng ký thì hệ thống sẽ ghi nhận thông tin khách hàng và gửi mã OTP về mail, hiển thị giao diện xác thực OTP cho khách hàng.
4. Người dùng nhập mã otp đúng và nhấn nút xác nhận. Hệ thống kiểm tra mã OTP chính xác và chuyển đến trang đặt mật khẩu mới.
5. Người dùng nhập mật khẩu mới và xác nhận mật khẩu sau đó nhấn hoàn tất

Chuỗi sự kiện phụ

Sau bước 3, người dùng có thể nhấn gửi lại mã OTP nhiều lần rồi thực hiện bước 4.

f. Usecase Quản lý thông tin cá nhân



Sơ đồ 4.6.Biểu đồ usecase quản lý thông tin cá nhân

Tên Use Case	Quản lý thông tin cá nhân
---------------------	---------------------------

Tác nhân chính	Admin, Khách hàng
Điều kiện trước	Actor đã đăng nhập thành công vào hệ thống.
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng.
Điều kiện sau	Hệ thống hiển thị thông tin cá nhân cho actor.

Chuỗi sự kiện chính

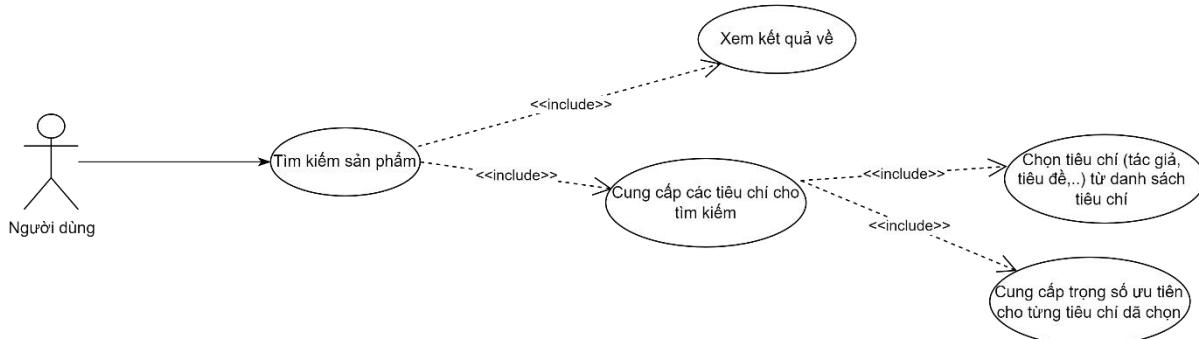
Bước 1: Hệ thống hiển thị thông tin cá nhân của actor.

Bước 2: Người dùng thay đổi thông tin và nhấn nút cập nhật thông tin.

Chuỗi sự kiện phụ

Người dùng có thể không thực hiện bước 2.

g. Usecase Tìm kiếm sản phẩm



Sơ đồ 4.7.Biểu đồ usecase Tìm kiếm sản phẩm

Tên Usecase	Lọc tìm sản phẩm
Tác nhân chính	Khách hàng, Khách vãng lai
Điều kiện trước	Người dùng đã truy cập vào trang web

Đảm bảo tối thiểu	Hệ thống phải tìm kiếm và hiển thị sản phẩm phù hợp với từ khóa nhập vào. Hệ thống phải phản hồi nhanh chóng và không gặp lỗi trong quá trình tìm kiếm
Điều kiện sau	Danh sách sản phẩm phù hợp với từ khóa được hiển thị cho người dùng

Chuỗi sự kiện chính

- Khách hàng sẽ chọn các tiêu chí để tìm kiếm theo mong muốn từ danh sách tiêu chí tìm kiếm mà hệ thống cung cấp gồm (tên sách, tác giả, thể loại, nhà xuất bản, năm xuất bản, từ khóa tùy chỉnh, số lượng mua và đánh giá sao.)
- Hệ thống trả về form chứa các tiêu chí tìm kiếm theo danh sách tiêu chí mà người dùng đã chọn, gồm có ô để nhập tiêu chí và ô trọng số ưu tiên tương ứng với tiêu chí.
- Khách hàng nhập các thông tin tìm kiếm cho các tiêu chí và cung cấp trọng số ưu tiên mong muốn đối với từng tiêu chí, trọng số ưu tiên mặc định ban đầu là 1.
- Hệ thống tìm kiếm các sản phẩm có phù hợp các tiêu chí, và tính toán độ ưu tiên để sắp xếp danh sách trả về dựa theo độ ưu tiên của từng tiêu chí mà khách hàng cung cấp. Độ ưu tiên ban đầu của tất cả sản phẩm bằng 0, nếu sản phẩm có thông tin phù hợp với tiêu chí đó thì độ ưu tiên sẽ cộng thêm trọng số độ ưu tiên của tiêu chí đó mà khách hàng cung cấp. Sau khi duyệt qua tất cả sách sẽ trả về danh sách sản phẩm có độ ưu tiên giảm dần, và chỉ lấy những sản phẩm có độ ưu tiên >0. Công thức:

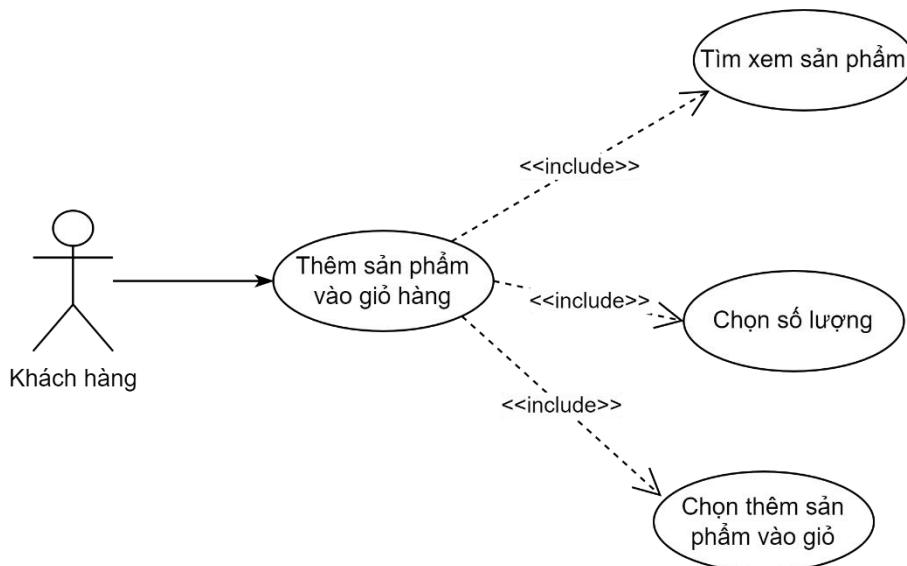
$$\text{Score} = \sum_{i=1}^n (\text{Match}_i \times \text{Weight}_i)$$

Trong đó:

- Score là điểm ưu tiên của sản phẩm do khách hàng chọn.
- n là số lượng tiêu chí tìm kiếm.
- Match_i là giá trị boolean (0 hoặc 1) cho biết tiêu chí thứ i có khớp với sản phẩm không.
- Weight_i là trọng số của tiêu chí thứ i mà người dùng đã nhập vào.

- Khách hàng xem kết quả tìm kiếm và tìm sản phẩm theo mong muốn.

h. Usecase Thêm sản phẩm vào giỏ hàng

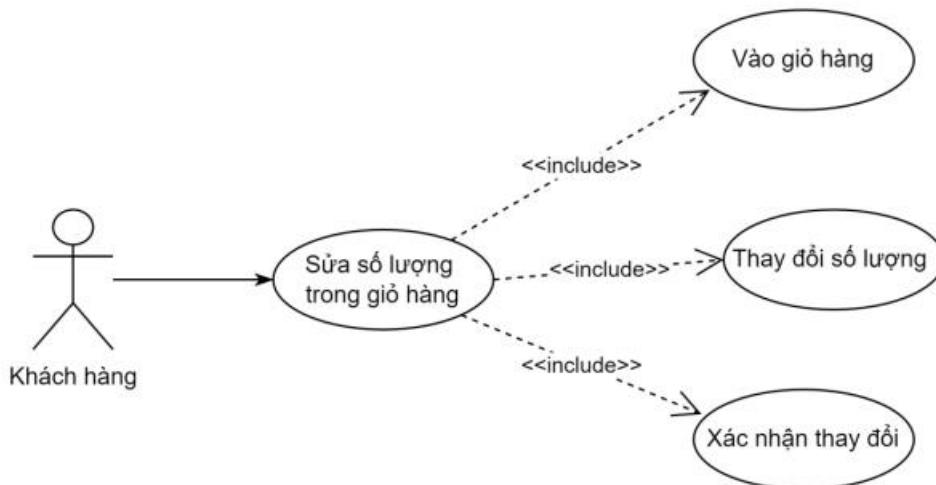


Sơ đồ 4.8. Biểu đồ usecase thêm sản phẩm vào giỏ hàng

Tên Usecase	Thêm sản phẩm vào giỏ hàng
Tác nhân chính	Khách hàng
Điều kiện trước	Khách hàng đã đăng nhập vào trang web
Đảm bảo tối thiểu	Hệ thống phải lưu trạng thái sản phẩm đã được thêm vào giỏ hàng, ngay cả khi có sự cố mạng hoặc người dùng đóng trình duyệt. Hệ thống phải thông báo cho người dùng nếu có lỗi xảy ra trong quá trình thêm sản phẩm vào giỏ hàng.
Điều kiện sau	Sản phẩm được thêm thành công vào giỏ hàng của người dùng.
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng tìm và chọn sản phẩm mà họ muốn thêm vào giỏ hàng. 2. Người dùng chọn số lượng.

3. Người dùng nhấp vào nút "Thêm vào giỏ hàng".
4. Hệ thống xác nhận rằng sản phẩm còn hàng và thông tin các thuộc tính đã chọn là hợp lệ.
5. Hệ thống cập nhật giỏ hàng của người dùng với sản phẩm đã chọn.
6. Hệ thống hiển thị thông báo xác nhận rằng sản phẩm đã được thêm vào giỏ hàng thành công.

Usecase sửa số lượng trong giỏ hàng



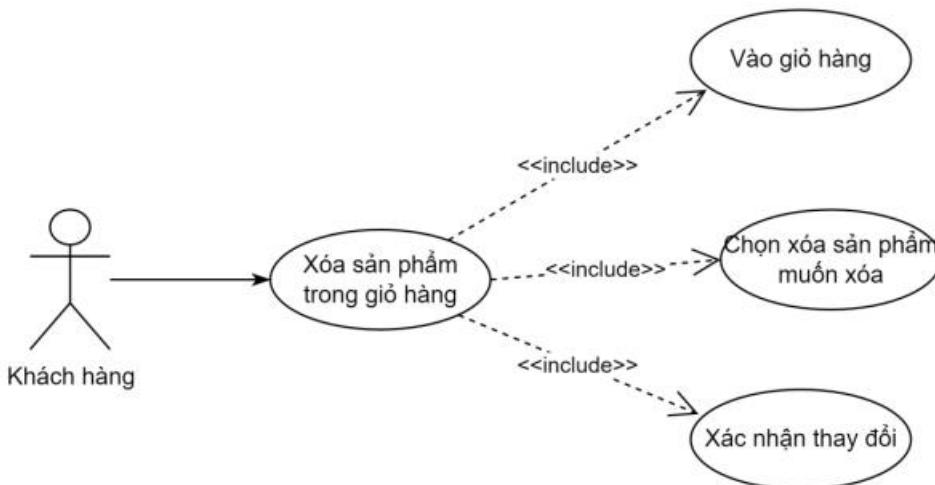
Sơ đồ 4.9.Biểu đồ usecase sửa số lượng trong giỏ hàng

Tên Usecase	Sửa số lượng trong giỏ hàng
Tác nhân chính	Khách hàng
Điều kiện trước	Khách hàng đã đăng nhập vào trang web Khách hàng có ít nhất một sản phẩm trong giỏ hàng
Đảm bảo tối thiểu	Hệ thống phải lưu lại thay đổi của người dùng và cập nhật giỏ hàng một cách chính xác.
Điều kiện sau	Giỏ hàng của người dùng được cập nhật với các thay đổi đã thực hiện

Chuỗi sự kiện chính

1. Người dùng truy cập vào trang giỏ hàng để xem các sản phẩm đã thêm vào.
2. Người dùng chọn sản phẩm cần sửa
3. Người dùng thay đổi số lượng sản phẩm.
4. Người dùng xác nhận thay đổi
5. Hệ thống cập nhật giỏ hàng với các thay đổi đã thực hiện và tính toán lại tổng giá trị giỏ hàng.
6. Hệ thống hiển thị lại giỏ hàng với thông tin sản phẩm đã được cập nhật cho người dùng xem.

i. Usecase Xóa sản phẩm trong giỏ hàng

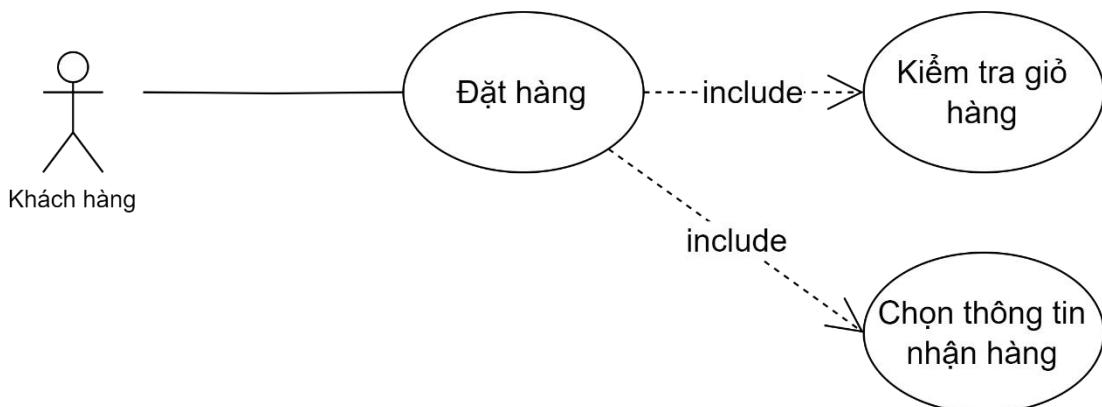


Sơ đồ 4.10.Biểu đồ usecase xóa sản phẩm trong giỏ hàng

Tên Usecase	Xóa sản phẩm trong giỏ hàng
Tác nhân chính	Khách hàng
Điều kiện trước	Khách hàng đã đăng nhập vào trang web Khách hàng có ít nhất một sản phẩm trong giỏ hàng
Đảm bảo tối thiểu	Hệ thống phải lưu lại thay đổi của người dùng và cập nhật

	giỎ hàng một cách chính xác.
Điều kiện sau	GiỎ hàng của người dùng được cập nhật với các thay đổi đã thực hiện
Chuỗi sự kiện chính	
<ol style="list-style-type: none"> 1. Người dùng truy cập vào trang giỏ hàng để xem các sản phẩm đã thêm vào. 2. Người dùng chọn sản phẩm cần xóa 3. Người dùng ấn xóa sản phẩm. 4. Người dùng xác nhận thay đổi 5. Hệ thống cập nhật giỏ hàng với các thay đổi đã thực hiện và tính toán lại tổng giá trị giỏ hàng. 6. Hệ thống hiển thị lại giỏ hàng với thông tin sản phẩm đã được cập nhật cho người dùng xem. 	

j. Usecase Đặt hàng



Sơ đồ 4.11. Biểu đồ usecase đặt hàng

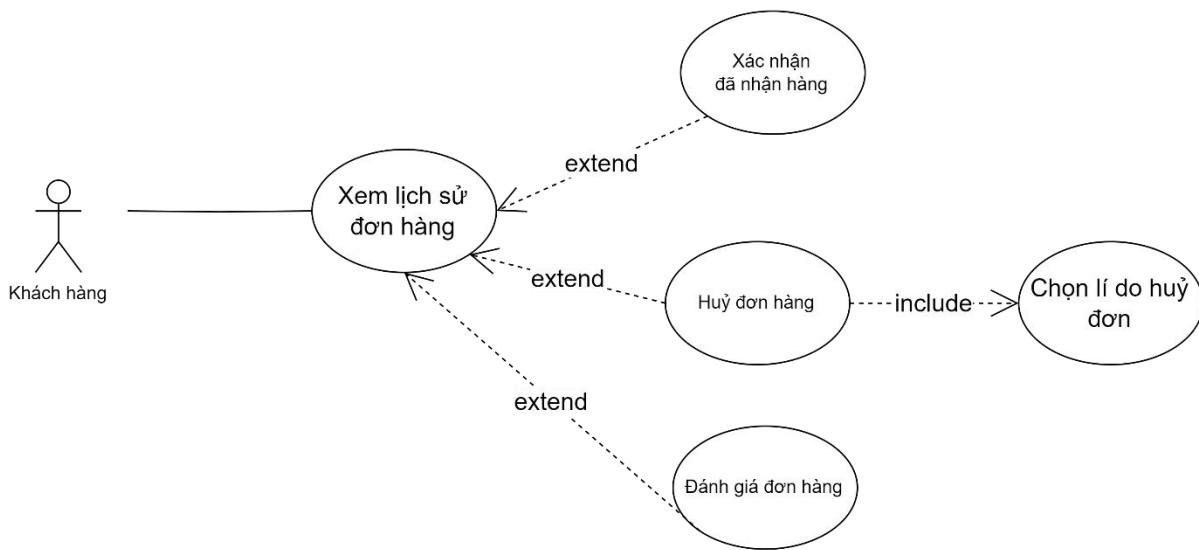
Tên Usecase	Đặt hàng
-------------	----------

Tác nhân chính	Khách hàng
Điều kiện trước	Khách hàng đã đăng nhập
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng
Điều kiện sau	Đặt hàng thành công và đơn hàng chờ duyệt

Chuỗi sự kiện chính

1. Truy cập giỏ hàng
2. Hệ thống hiển thị danh sách sản phẩm trong giỏ hàng của người dùng
3. Khách hàng bấm nút tạo đơn hàng
4. Hệ thống tính toán tổng chi phí, bao gồm phí giao hàng và các ưu đãi. Phí giao hàng tính theo trọng lượng của sản phẩm và khoảng cách giao hàng. Phí giao hàng được chia theo khu vực và khối lượng đơn hàng. (khu vực nội tỉnh, nội vùng, liên vùng, liên vùng tỉnh, liên vùng đặc biệt), và theo trọng lượng đơn hàng (từ 0-0,5kg, >3kg).
Ví dụ một đơn hàng nội thành Hà Nội nặng 2kg có cước phí là 21.000 VNĐ + (2,500 x 3) = 28.500 VNĐ/ đơn.
Ưu đãi: Khách hàng thân thiết: Giảm 10% phí giao hàng. Nếu đơn hàng có tổng giá trị từ 2.000.000 trở lên được giảm 2% giá trị đơn hàng.
Khách hàng VIP: Giảm 30% phí giao hàng. Nếu đơn hàng có tổng giá trị từ 2.000.000 trở lên được giảm 3% giá trị đơn hàng.
5. Khách hàng vào giỏ hàng, nhập địa chỉ gồm tỉnh, huyện, xã và xác nhận đơn hàng.
6. Chọn thông tin nhận hàng
7. Thông báo đặt hàng thành công và đơn hàng chờ duyệt

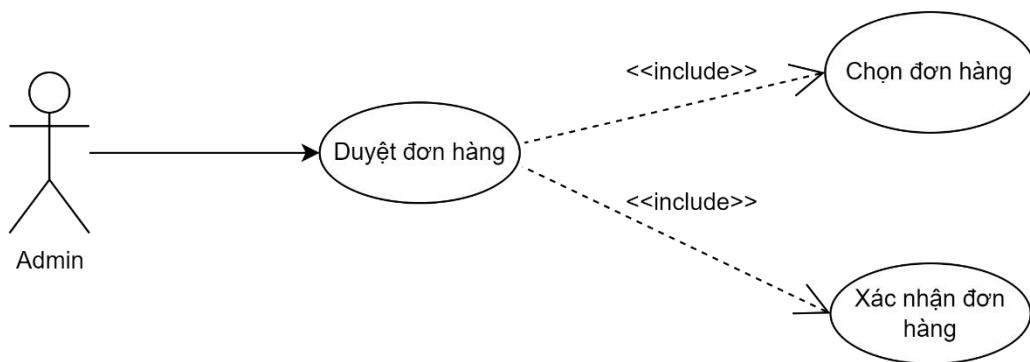
k. Usecase Xem lịch sử đơn hàng



Sơ đồ 4.12. Biểu đồ usecase xem lịch sử đơn hàng

Tên Usecase	Xem lịch sử đơn hàng
Tác nhân chính	Khách hàng
Điều kiện trước	Khách hàng đã đăng nhập
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng
Điều kiện sau	Hiển thị đầy đủ và chính xác lịch sử đơn hàng
Chuỗi sự kiện chính	
<ol style="list-style-type: none"> 1. Chọn chức năng xem lịch sử đơn hàng 2. Hệ thống hiển thị danh sách lịch sử đơn hàng 3. Người dùng xem chi tiết đơn hàng hoặc cập nhật trạng thái đơn hàng 	

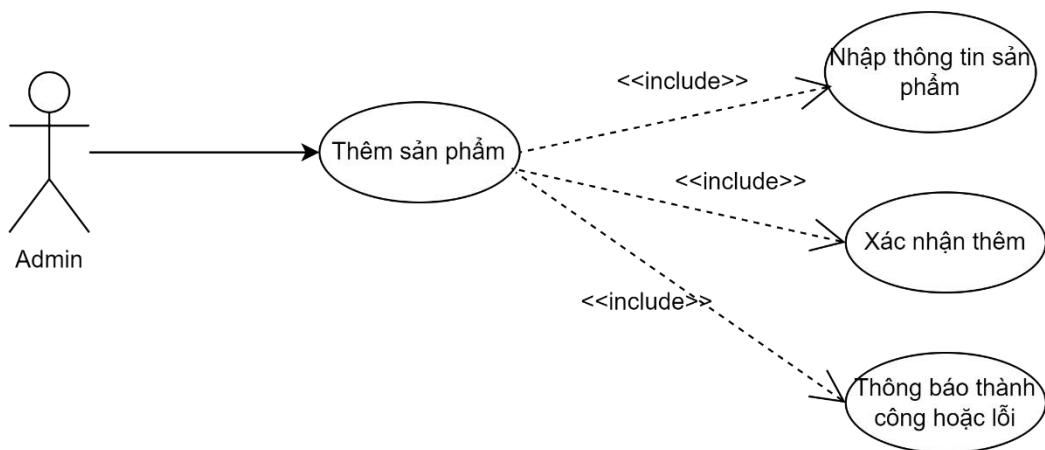
1. Usecase Duyệt đơn hàng



Sơ đồ 4.13. Biểu đồ usecase duyệt đơn hàng

Tên Use Case	Duyệt đơn hàng
Tác nhân chính	Admin
Điều kiện trước	Đăng nhập với tài khoản hoặc Admin
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng
Điều kiện sau	Đơn hàng được duyệt và chuyển sang trạng thái đang giao
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1.Chọn chức năng quản lý đơn hàng 2.Hệ thống hiển thị danh sách đơn hàng 3. Chọn đơn hàng cần duyệt và bấm nút “Duyệt” 4.Hệ thống hiển thị đã duyệt đơn hàng và chuyển đơn hàng sang “Đang giao”

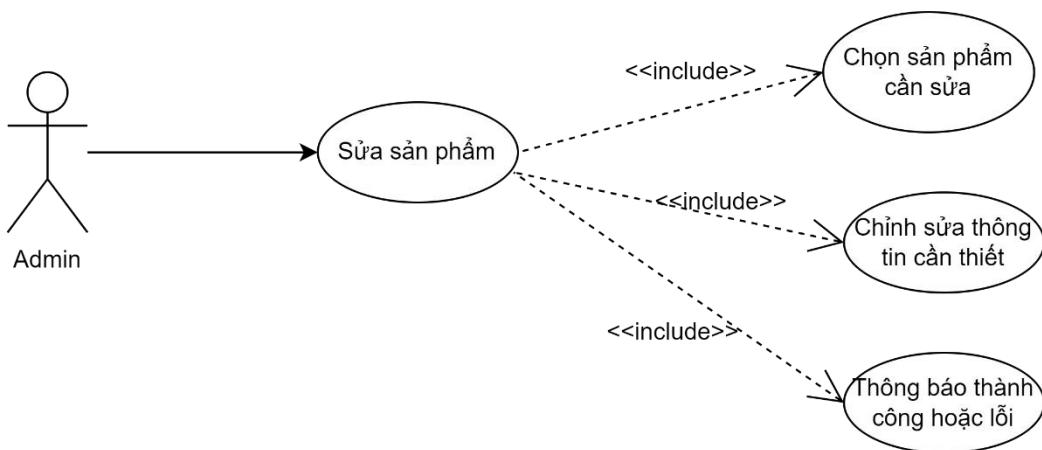
m. Usecase Thêm sản phẩm



Sơ đồ 4.14. Biểu đồ usecase thêm sản phẩm

Tên Use Case	Thêm sản phẩm
Tác nhân chính	Admin
Điều kiện trước	Đăng nhập với tài khoản Admin
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng
Điều kiện sau	Sản phẩm được thêm mới thành công
Chuỗi sự kiện chính	
1.Chọn chức năng quản lý sản phẩm 2.Hệ thống hiển thị danh sách sản phẩm 3. Nhấn nút “Thêm sản phẩm” để tới trang thêm sản phẩm mới 4. Nhập thông tin sản phẩm cần thiết và nhấn nút “Thêm” 5.Hệ thống hiển thị đã thêm thành công và cập nhật vào cơ sở dữ liệu hoặc nếu lỗi bắt đầu lại bước 4	

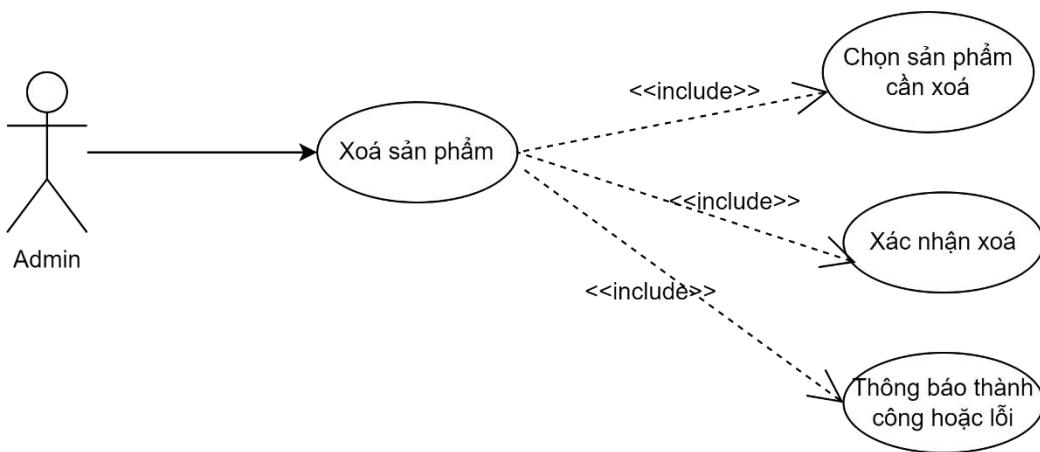
n. Usecase Sửa sản phẩm



Sơ đồ 4.15. Biểu đồ usecase sửa sản phẩm

Tên Use Case	Sửa sản phẩm
Tác nhân chính	Admin
Điều kiện trước	Đăng nhập với tài khoản Admin
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng
Điều kiện sau	Sản phẩm được sửa thành công
Chuỗi sự kiện chính	<ol style="list-style-type: none"> 1.Chọn chức năng quản lý sản phẩm 2.Hệ thống hiển thị danh sách sản phẩm 3. Chọn sản phẩm cần sửa 4. Sửa đổi các thông tin cần thiết và nhấn nút “Sửa” 5.Hệ thống hiển thị đã sửa thành công và cập nhật vào cơ sở dữ liệu hoặc nếu lỗi bắt đầu lại bước 4

o. Usecase Xoá sản phẩm

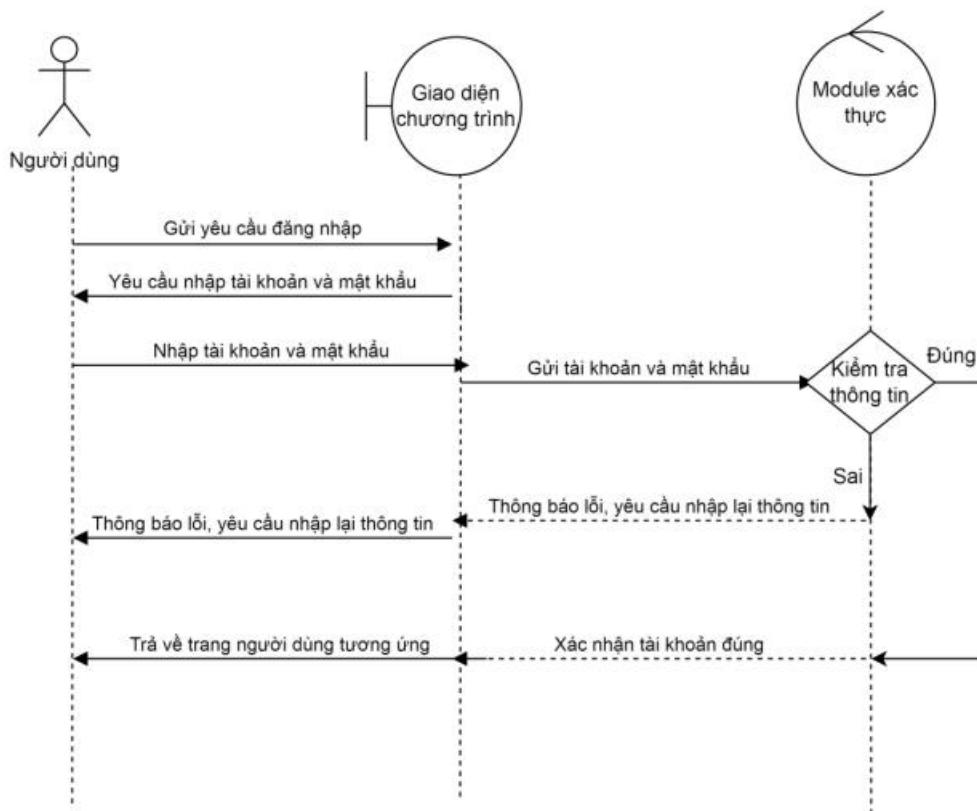


Sơ đồ 4.16. Biểu đồ usecase xóa sản phẩm

Tên Use Case	Xoá sản phẩm
Tác nhân chính	Admin
Điều kiện trước	Đăng nhập với tài khoản Admin
Đảm bảo tối thiểu	Hệ thống rollback lại nếu lỗi giữa chừng
Điều kiện sau	Sản phẩm bị xoá khỏi cơ sở dữ liệu
Chuỗi sự kiện chính	1.Chọn chức năng quản lý sản phẩm 2.Hệ thống hiển thị danh sách sản phẩm 3. Chọn sản phẩm cần xoá 4. Nhấn nút “Xoá” 5.Hệ thống hiển thị đã xoá thành công hoặc lỗi khi sản phẩm có liên kết khác (hệ thống đã nằm trong đơn hàng của người dùng)

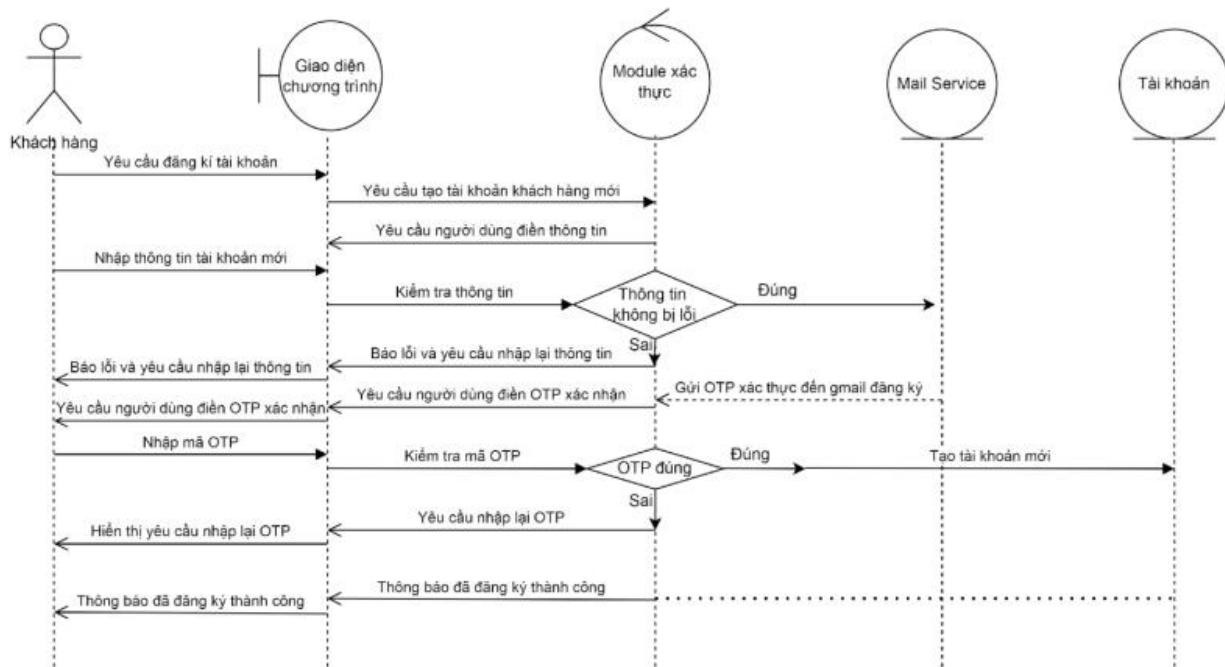
4.2.2. Biểu đồ tuần tự

a. Biểu đồ tuần tự Đăng nhập



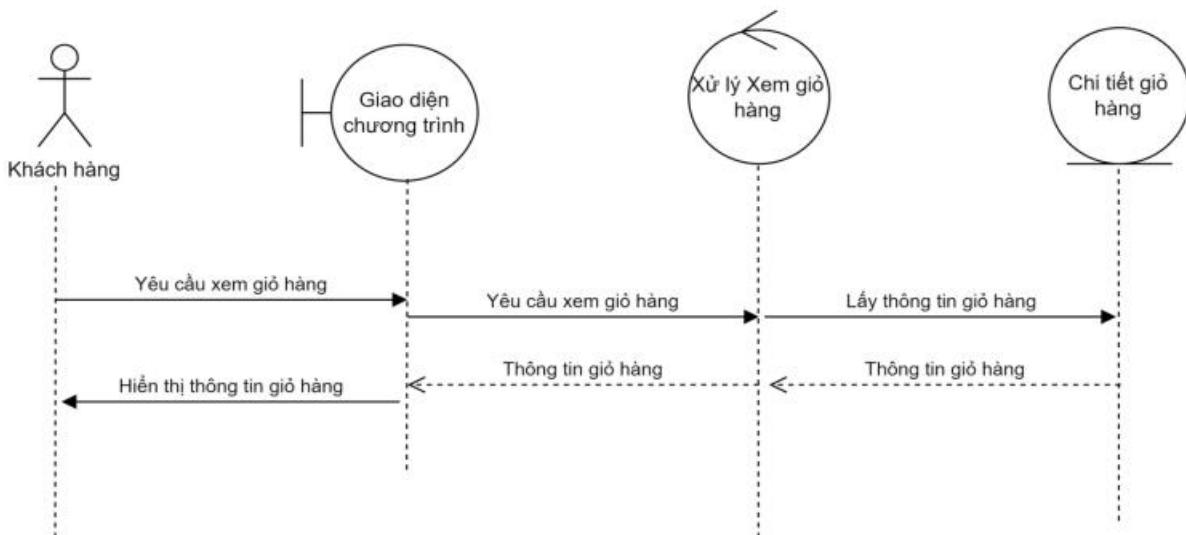
Sơ đồ 4.17. Biểu đồ tuần tự đăng nhập

b. Biểu đồ tuần tự Đăng ký



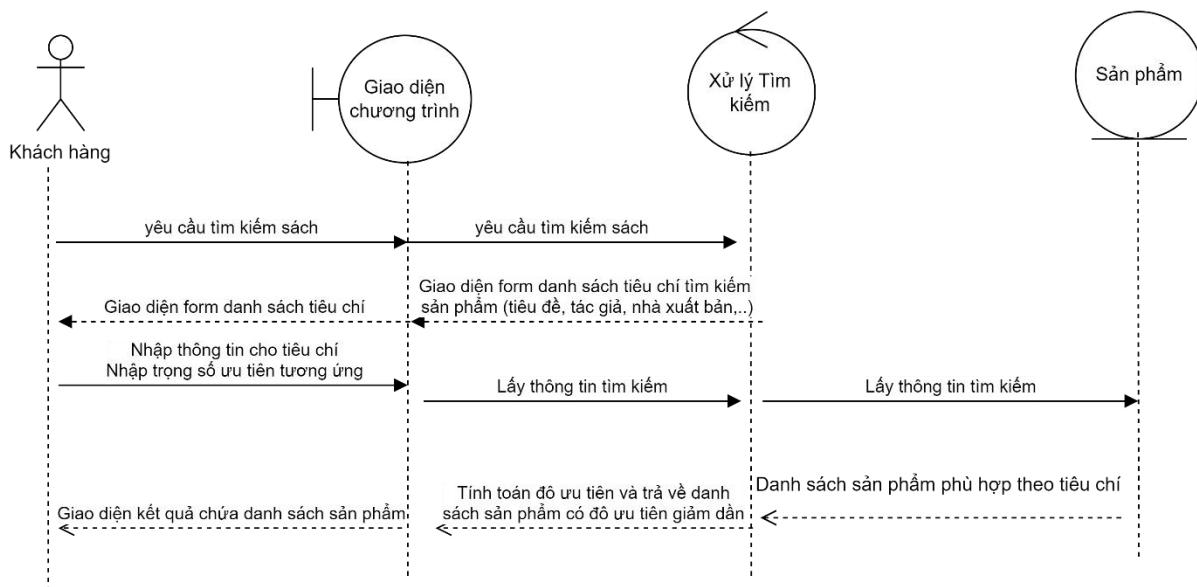
Sơ đồ 4.18. Biểu đồ tuần tự đăng ký

c. Biểu đồ tuần tự xem giỏ hàng

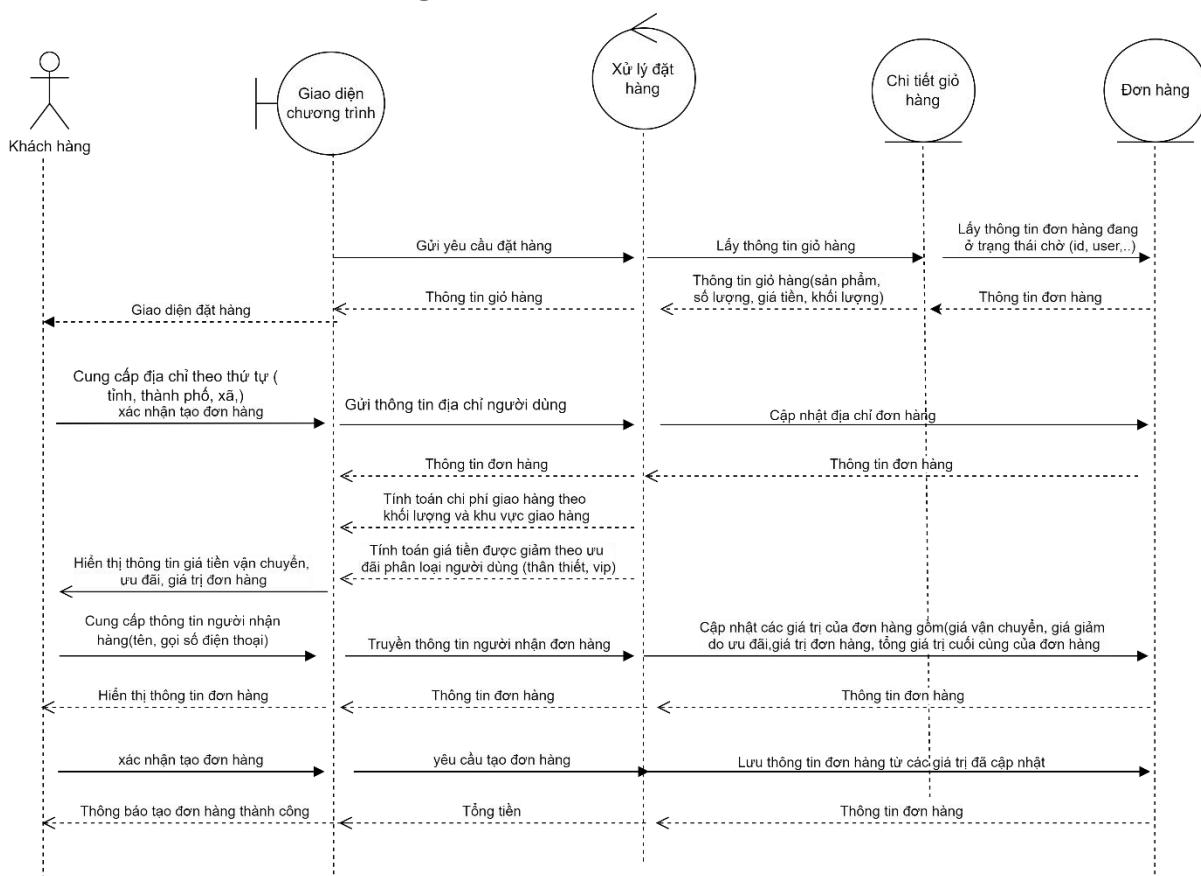


Sơ đồ 4.19. Biểu đồ tuần tự xem giỏ hàng

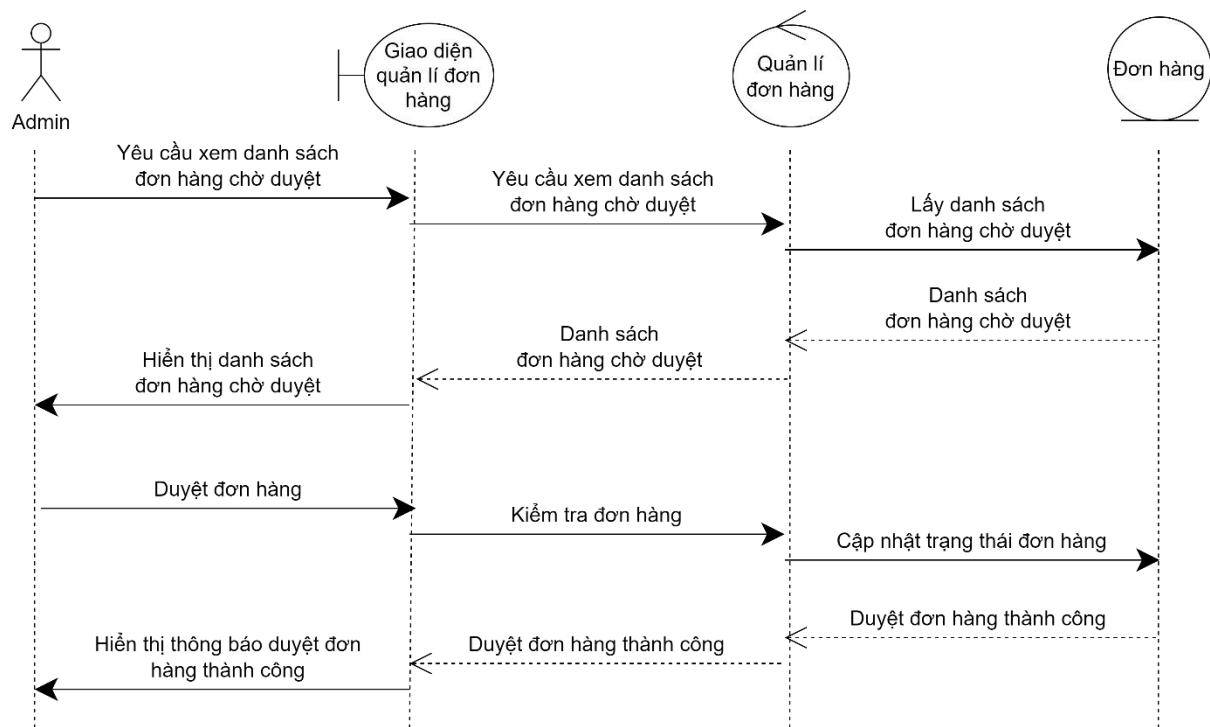
d. Biểu đồ tuần tự tìm kiếm sản phẩm



e. Biểu đồ tuần tự đặt hàng

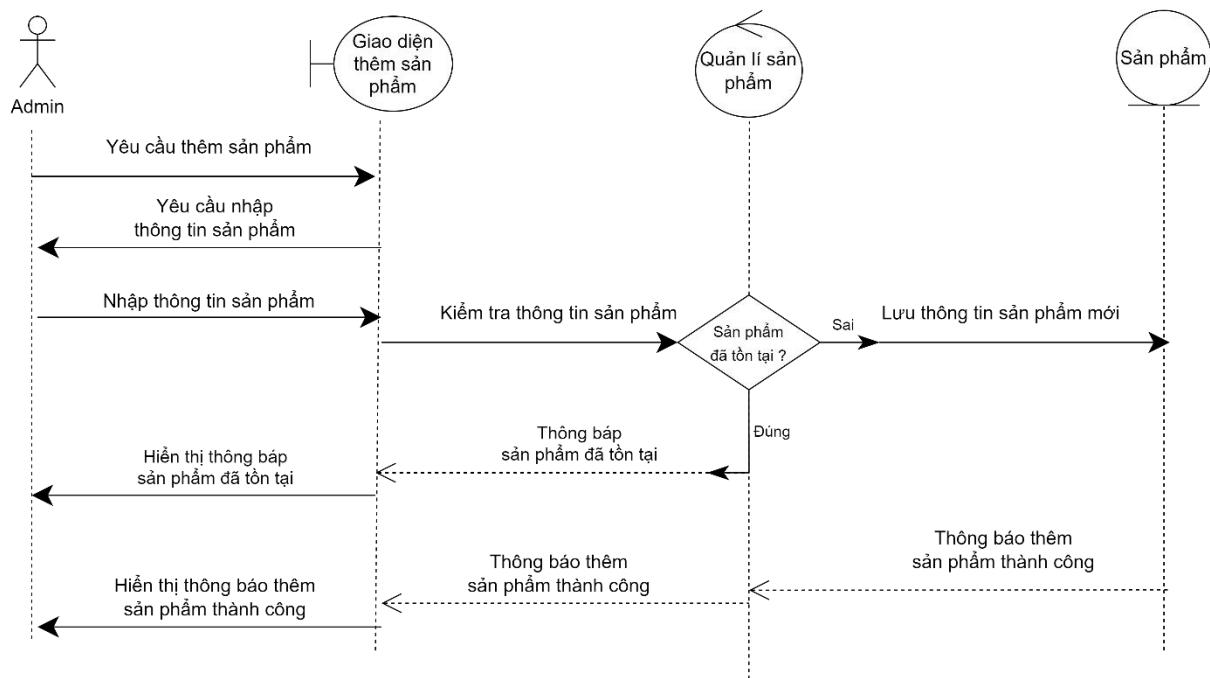


f. Biểu đồ tuần tự Duyệt đơn hàng



Sơ đồ 4.22. Biểu đồ tuần tự duyệt đơn hàng

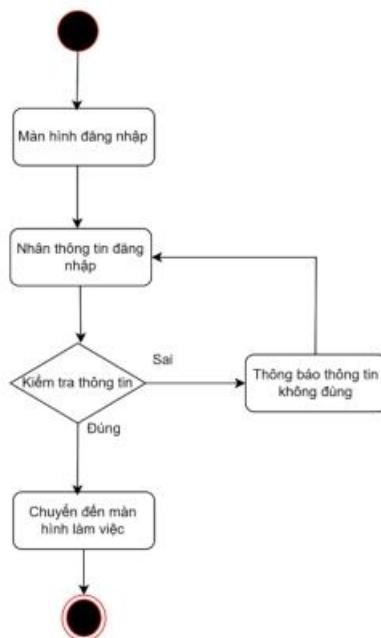
g. Biểu đồ tuần tự Thêm sản phẩm



Sơ đồ 4.23. Biểu đồ tuần tự thêm sản phẩm

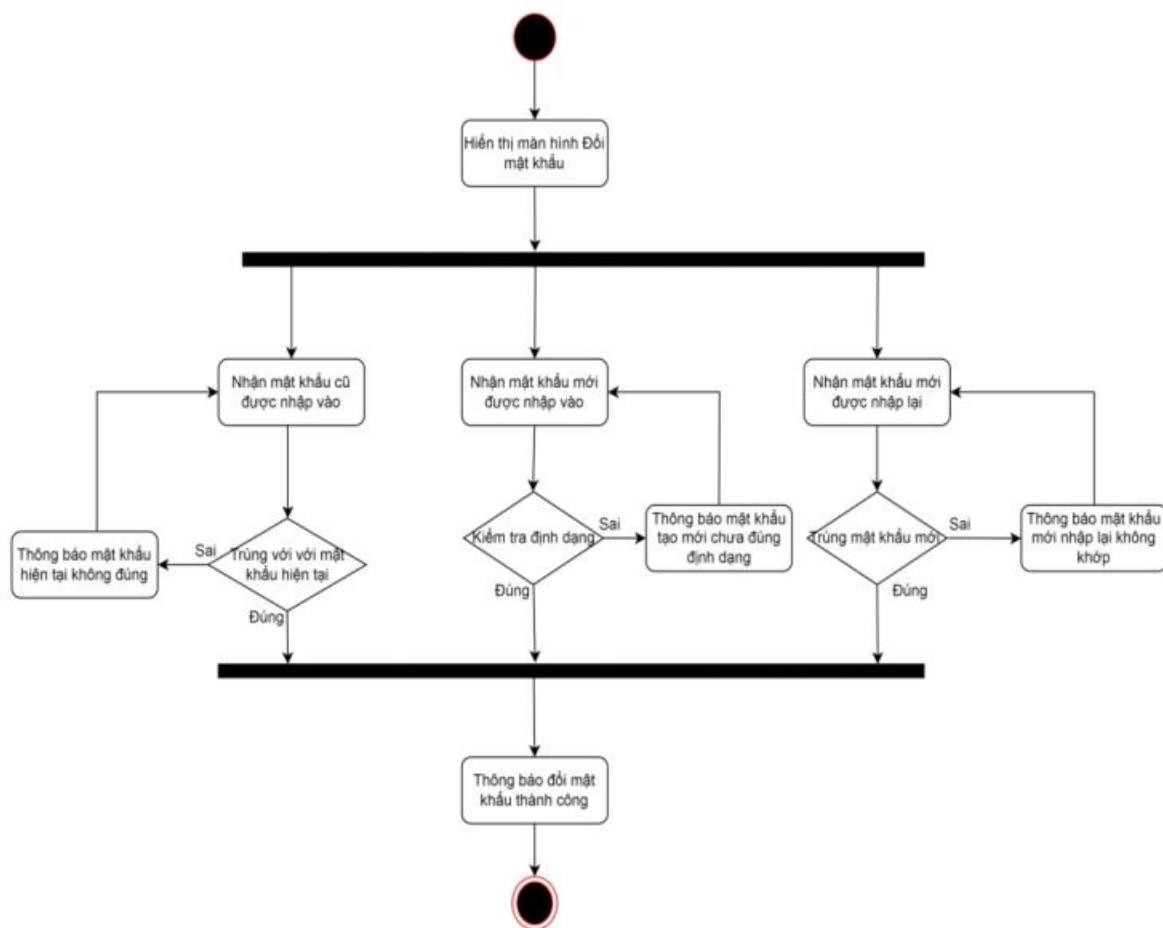
4.2.3. Biểu đồ hoạt động

a. Biểu đồ hoạt động Đăng nhập



Sơ đồ 4.24. Biểu đồ hoạt động đăng nhập

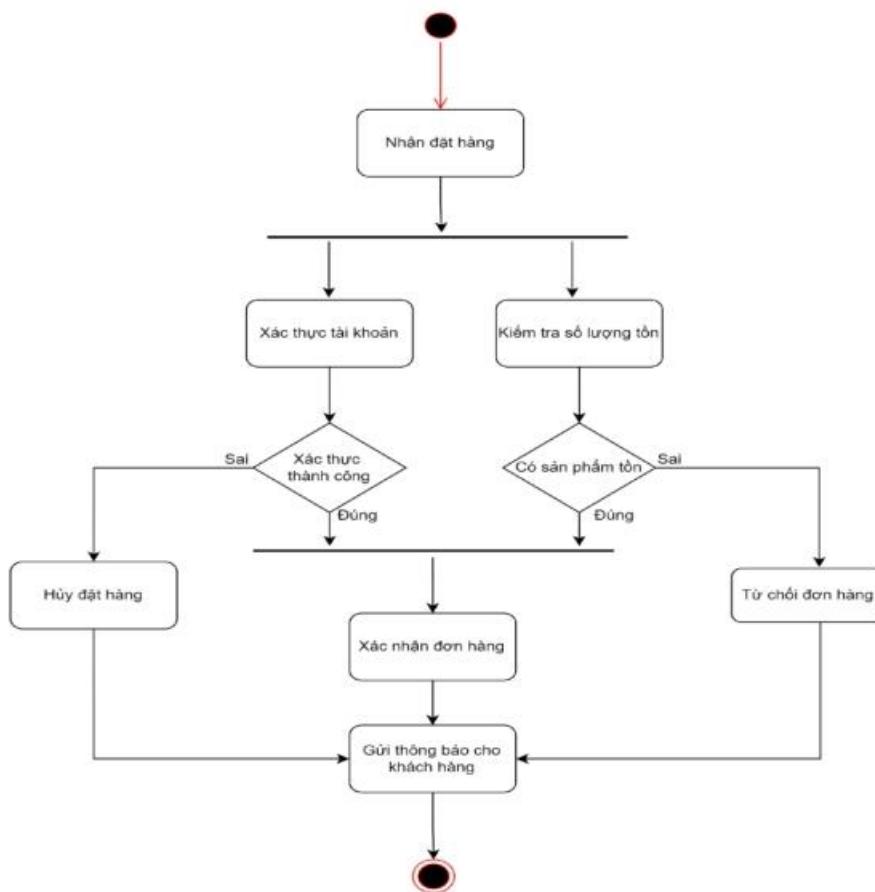
b. Biểu đồ hoạt động Đổi mật khẩu



Sơ đồ 4.25. Biểu đồ hoạt động đổi mật khẩu

c. Biểu đồ hoạt động thêm sản phẩm vào giỏ*Sơ đồ 4.26. Biểu đồ hoạt động thêm sản phẩm vào giỏ*

d. Biểu đồ hoạt động đặt hàng



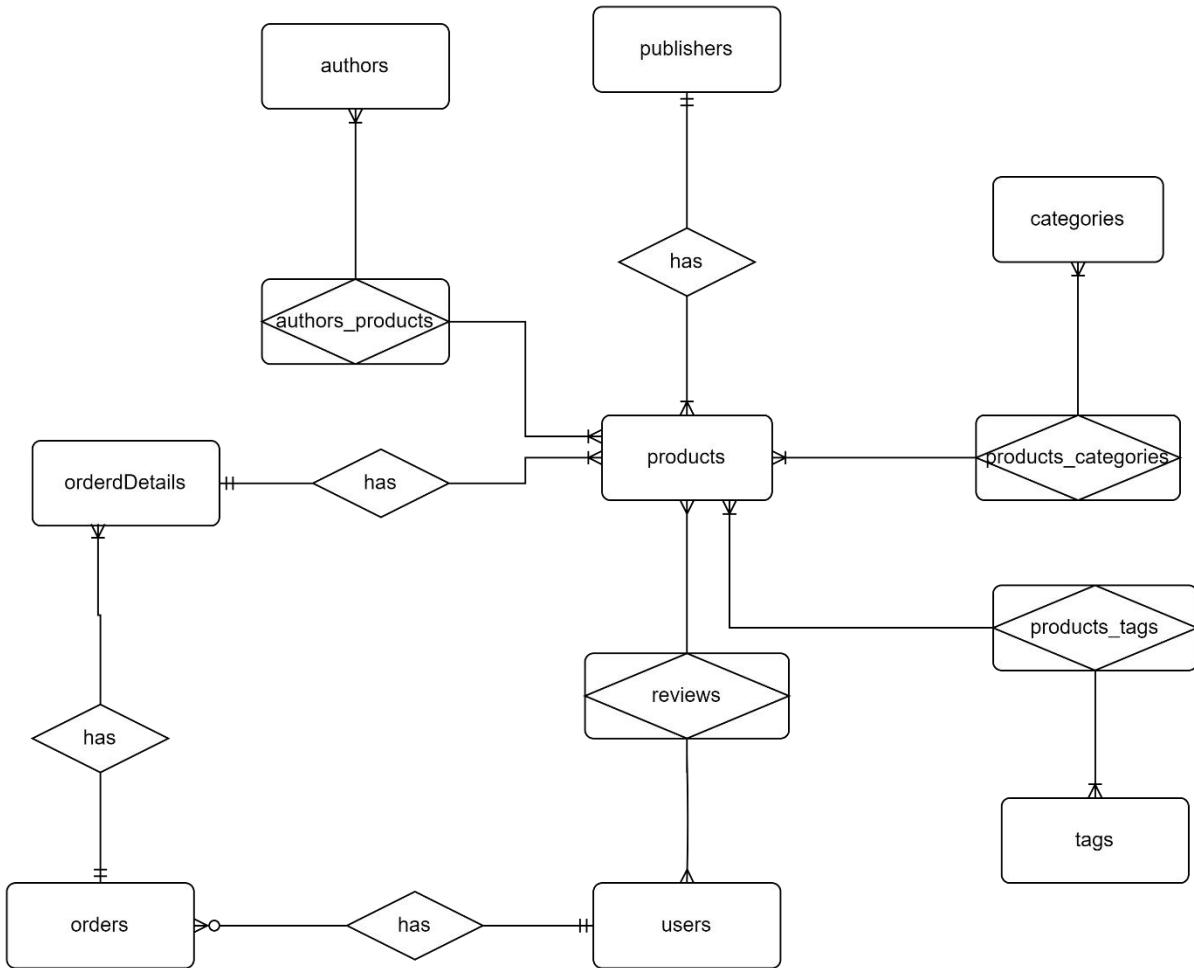
Sơ đồ 4.27. Biểu đồ hoạt động đặt hàng

4.3. Thiết kế cơ sở dữ liệu và ràng buộc

4.3.1. Xác định thực thể

- Products ([id](#), title, description, price, publication_year, sales_volume, star_rating, image, weight)
- Categories ([id](#), name)
- Authors ([id](#), name, biography)
- Tags ([id](#), name)
- Publishers (id, name, address, email)
- Orders ([id](#), order_date, status, shipping_fee, discount, receive_phone, receive_address, receive_name, create_at, update_at)
- Users ([id](#), username, password, email, role, avatar, fullname, phone, classification)

4.3.2. Lược đồ ERD



Sơ đồ 4.28. Lược đồ ERD

4.3.3. Mô hình cơ sở dữ liệu quan hệ

Quy ước: Khóa chính gạch chân, khóa ngoại in đậm

- Products (**id**, title, description, price, publication_year, sales_volume, star_rating, image, weight, status, **publisher_id**)
- Categories (**id**, name)
- Authors (**id**, name, biography)
- Tags (**id**, name)
- Publishers (id, name, address, email)
- Orders (**id**, order_date, status, shipping_fee, discount, receive_phone, receive_address, receive_name, create_at, update_at, **user_id**)
- Users (**id**, username, password, email, role, avatar, fullname, phone, classification)
- Products_categories (**id**, **product_id**, **category_id**)
- Authors_products (**id**, **author_id**, **product_id**)
- Products_tags (**id**, **product_id**, **tag_id**)

- OrderDetail (**id**, **order_id**, **product_id**, quantity, price, has_review)
- Review (**id**, **product_id**, **user_id**, rating, comment)

a. Bảng authors

STT	TÊN	LOẠI	KIỀU	RÀNG BUỘC	Ý NGHĨA
1	id	Primary key	Int	Not null,	Id tác giả
2	name		Varchar(255)	Not null	Tên tác giả
3	biography		Varchar(255)		Tiểu sử tác giả

Bảng 4.1.Bảng authors

b. Bảng authors_products

STT	TÊN	LOẠI	KIỀU	RÀNG BUỘC	Ý NGHĨA
1	id	Primary key	Int	Not null,	Id
2	author_id	Foreign key	Int	Not null	Id tác giả
3	product_id	Foreign key	Int	Not null	Id sản phẩm

Bảng 4.2. Bảng authors_products

c. Bảng categories

STT	TÊN	LOẠI	KIỀU	RÀNG BUỘC	Ý NGHĨA
1	id	Primary key	Int	Not null,	Id danh mục
2	name		Varchar(255)	Not null, Unique	Tên danh mục

Bảng 4.3.Bảng categories

d. Bảng orderdetails

STT	TÊN	LOẠI	KIỀU	RÀNG BUỘC	Ý NGHĨA

1	id	Primary key	Int	Not null,	Id chi tiết đơn hàng
2	order_id	Foreign key	Int		Id đơn hàng
3	product_id	Foreign key	Int		Id sản phẩm
4	quantity		Int	Not null	Số lượng
5	price		Int	>=0	Giá
6	has_review		Boolean	Default: 0	Trạng thái đánh giá sản phẩm khi đã nhận hàng

*Bảng 4.4.Bảng orderdetails***e. Bảng orders**

STT	TÊN	LOẠI	KIỂU	RÀNG BUỘC	Ý NGHĨA
1	id	Primary key	Int	Not null	Id đơn hàng
2	user_id	Foreign key	Int		Id người dùng
3	order_date		Datetime		Ngày đặt hàng
4	status		Varchar(30)		Trạng thái
5	total		Float	Not null	Tổng giá trị đơn hàng
6	shipping_fee		Float		Phí vận chuyển
7	discount		Float		Giảm giá
8	receive_phone		Varchar(15)	Not null	SDT người nhận
9	receive_address		Varchar(255)	Not null	Địa chỉ nhận
10	receive_name		Varchar(50)	Not null	Tên người nhận
11	create_at		DateTime	Not null	Thời gian tạo
12	update_at		DateTime	Not null	Thời gian cập nhật

*Bảng 4.5. Bảng orders***f. Bảng products**

STT	TÊN	LOẠI	KIỂU	RÀNG BUỘC	Ý NGHĨA

1	id	Primary key	Int	Not null,	Id sản phẩm
2	title		Varchar(255)	Not null	Tên sách
3	description		Text		Mô tả sách
4	price		Int	Not null	Giá sách
5	publication_year		Int		Năm xuất bản
6	publisher_id	Foreign key	Int		Id nhà xuất bản
7	sales_volume		Int		Số lượng bán
8	star_rating		Float		Đánh giá sao
9	image		Varchar(255)		Hình ảnh
10	weight		Float		Trọng lượng
11	status		Int	Default '1'	Trạng thái bán
12	priority		Float	Default '0'	Độ ưu tiên trong tìm kiếm boolean

Bảng 4.6. Bảng products

g. Bảng products_tags

STT	TÊN	LOẠI	KIỂU	RÀNG BUỘC	Ý NGHĨA
1	id	Primary key	Int	Not null	Id
2	product_id	Foreign key	Int		Id sản phẩm
3	tag_id	Foreign key	Int		Id tag

Bảng 4.7.Bảng products_tags

h. Bảng publishers

STT	TÊN	LOẠI	KIỂU	RÀNG BUỘC	Ý NGHĨA

1	id	Primary key	Int	Not null	Id nhà xuất bản
2	name		Varchar(255)	Not null	Tên nhà xuất bản
3	address		Varchar(255)		Địa chỉ
4	email		Varchar(255)		Email

*Bảng 4.8. Bảng publishers***i. Bảng reviews**

STT	TÊN	LOẠI	KIỂU	RÀNG BUỘC	Ý NGHĨA
1	id	Primary key	Int	Not null	Id đánh giá
2	product_id	Foreign key	Int		Id sản phẩm
3	user_id	Foreign key	Int		Id người dùng
4	rating		Int		Đánh giá
5	comment		Varchar(255)		Bình luận

*Bảng 4.9. Bảng reviews***j. Bảng tags**

STT	TÊN	LOẠI	KIỂU	RÀNG BUỘC	Ý NGHĨA
1	id	Primary key	Int	Not null	Id tag
2	name		Varchar(255)	Not null	Tên tag

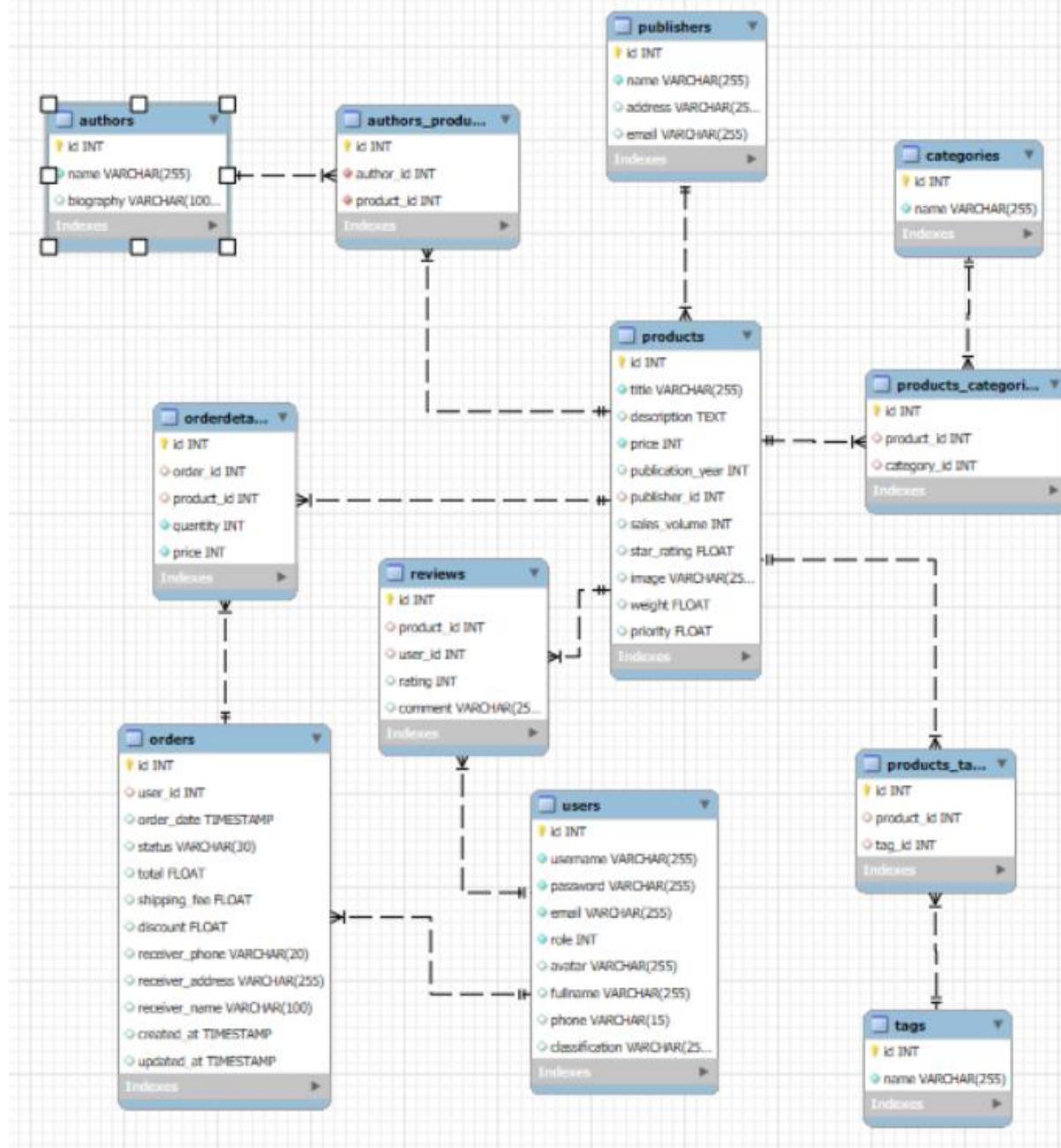
*Bảng 4.10. Bảng tags***k. Bảng users**

STT	TÊN	LOẠI	KIỂU	RÀNG BUỘC	Ý NGHĨA
1	id	Primary key	Int	Not null	Id người dùng

2	username		Varchar(255)	Not null, Unique	Tên đăng nhập
3	password		Varchar(255)	Not null	Mật khẩu
4	email		Varchar(255)	Not null, Unique	Email
5	role		Int	Not null	Vai trò
6	avatar		Varchar(255)		Ảnh đại diện
7	fullname		Varchar(255)		Họ tên
8	phone		Varchar(15)		Số điện thoại
9	classification		Varchar(255)		Loại khách hàng

Bảng 4.11. Bảng users

4.3.4. Diagram



Hình 4.1. Diagram trong hệ thống

4.4. Tạo API cho chương trình

4.4.1. Trang đăng nhập, đăng ký

a. Đăng nhập

Phương thức: POST

API: /api/auth/login

Mô tả: Đăng nhập người dùng

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
username	string	Tên người dùng
password	string	Mật khẩu

Output: Trả về thông tin người dùng và token phiên đăng nhập

b. Đăng ký

Phương thức: POST

API: /api/auth/register

Mô tả: Đăng ký người dùng mới

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
username	string	Tên người dùng
password	string	Mật khẩu
email	string	Địa chỉ email
fullname	string	Họ và tên đầy đủ
phone	string	Số điện thoại

Output: Trả về thông tin người dùng mới và token phiên đăng ký

4.4.2. Trang chủ

a. Lấy sản phẩm mới nhất

Phương thức: GET

API: /api/products/newest

Mô tả: Lấy danh sách sản phẩm mới nhất

Input: Không có input

Output: Mô tả: Danh sách các sản phẩm mới nhất

b. Lấy tất cả sản phẩm

Phương thức: GET

API: /api/products/get-all

Mô tả: Lấy danh sách tất cả sản phẩm

Input: Không có input

Output: Mô tả: Danh sách tất cả các sản phẩm

4.4.3. Trang chi tiết sản phẩm

a. Lấy sản phẩm theo ID

Phương thức: GET

API: /api/products/{id}

Mô tả: Lấy thông tin sản phẩm theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của sản phẩm

Output: Thông tin chi tiết của sản phẩm

b. Thêm sản phẩm vào giỏ

Phương thức: POST

API: /api/orderdetails/add-to-cart

Mô tả: Thêm sản phẩm vào giỏ hàng

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
productId	integer	ID của sản phẩm
quantity	integer	Số lượng sản phẩm

Output: Xác nhận sản phẩm đã được thêm vào giỏ hàng

c. Lấy sản phẩm cùng loại

Phương thức: GET

API: /api/products/category/{categoryId}

Mô tả: Lấy danh sách sản phẩm cùng loại theo danh mục

Tên thuộc tính	Kiểu giá trị	Mô tả
categoryId	integer	ID của danh mục

Output: Danh sách sản phẩm cùng loại

4.4.4. Trang tìm kiếm theo Boolean Search

a. Tìm kiếm theo Boolean search

Phương thức: POST

API: /api/products/BooleanSearch

Mô tả: Tìm kiếm sản phẩm theo tiêu chí Boolean

Input:

Tên thuộc tính	Kiểu dữ liệu	Mô tả
title	string	Tiêu đề sản phẩm
author	string	Tác giả
category	string	Danh mục
publisher	string	Nhà xuất bản
publicationYear	integer	Năm xuất bản
tag	string	Tag
minRating	number	Đánh giá tối thiểu
maxRating	number	Đánh giá tối đa
minPrice	integer	Giá tối thiểu
maxPrice	integer	Giá tối đa
titleWeight	number	Trọng số tiêu đề
authorWeight	number	Trọng số tác giả
categoryWeight	number	Trọng số danh mục
publisherWeight	number	Trọng số nhà xuất bản
yearWeight	number	Trọng số năm xuất bản
tagWeight	number	Trọng số tag
ratingWeight	number	Trọng số đánh giá
priceWeight	number	Trọng số giá

Output: Danh sách sản phẩm phù hợp với tiêu chí tìm kiếm sắp xếp theo độ ưu tiên giảm dần

4.4.5. Trang giỏ hàng của người dùng

a. Xem sản phẩm trong giỏ hàng

Phương thức: GET

API: /api/orderdetails/get-cart-items/{userId}

Mô tả: Lấy tất cả sản phẩm trong giỏ hàng của người dùng

Input:

Tên thuộc tính	Kiểu dữ liệu	Mô tả
userId	integer	ID của người dùng

Output: Danh sách sản phẩm trong giỏ hàng

b. Cập nhật số lượng chi tiết đơn hàng con

Phương thức: PUT

API: /api/orderdetails/update-quantity

Mô tả: Cập nhật số lượng của chi tiết đơn hàng

Input:

Tên thuộc tính	Kiểu dữ liệu	Mô tả
id	integer	ID của chi tiết đơn hàng
quantity	integer	Số lượng sản phẩm mới

c. Lấy thông tin chi tiết đơn hàng con của giỏ hàng

Phương thức: GET

API: /api/orderdetails/{id}

Mô tả: Lấy thông tin chi tiết đơn hàng theo ID

Input: Không có

Output: Thông tin chi tiết đơn hàng

d. Xóa chi tiết đơn hàng

Phương thức: DELETE

API: /api/orderdetails/delete/{id}

Mô tả: Xóa chi tiết đơn hàng theo ID

Input:

Tên thuộc tính	Kiểu dữ liệu	Mô tả
id	integer	ID của chi tiết đơn hàng

4.4.6. Trang thông tin cá nhân

a. Lấy thông tin cá nhân

Phương thức: GET

API: /api/users/{id}

Mô tả: Lấy thông tin cá nhân của người dùng theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của người dùng

Output: Thông tin cá nhân của người dùng

b. Cập nhật thông tin cá nhân

Phương thức: PUT

API: /api/users/update/{id}

Mô tả: Cập nhật thông tin cá nhân của người dùng theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả

id	integer	ID của người dùng
username	string	Tên người dùng
email	string	Địa chỉ email
fullname	string	Họ và tên đầy đủ
phone	string	Số điện thoại

Output: Xác nhận thông tin cá nhân đã được cập nhật

c. Đổi mật khẩu

Phương thức: PUT

API: /api/users/change-password/{id}

Mô tả: Đổi mật khẩu của người dùng theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của người dùng
oldPassword	string	Mật khẩu cũ
newPassword	string	Mật khẩu mới

Output: Xác nhận mật khẩu đã được đổi

4.4.7. Trang đơn hàng của người dùng

a. Lấy danh sách đơn hàng theo trạng thái

Phương thức: GET

API: /api/orders/get-by-user-and-status

Mô tả: Lấy danh sách đơn hàng của người dùng theo trạng thái

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
userId	integer	ID của người dùng
status	string	Trạng thái đơn hàng

Output: Danh sách đơn hàng theo trạng thái

b. Cập nhật trạng thái đơn hàng

Phương thức: PUT

API: /api/orders/update-status/{id}

Mô tả: Cập nhật trạng thái đơn hàng theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả

id	integer	ID của đơn hàng
status	string	Trạng thái mới

Output: Xác nhận trạng thái đơn hàng đã được cập nhật

4.4.8. Trang đặt hàng

a. Lấy thông tin tỉnh, thành phố

Phương thức: GET

API: <https://online-gateway.ghn.vn/shiip/public-api/master-data/province>

Mô tả: Lấy danh sách tất cả các tỉnh, thành phố của Việt Nam.

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
token	string	Xác thực API

Output: Danh sách tỉnh, thành phố của Việt Nam

b. Lấy thông tin quận, huyện

Phương thức: GET

API: <https://online-gateway.ghn.vn/shiip/public-api/master-data/district>

Mô tả: Lấy danh sách các quận, huyện dựa trên tỉnh, thành phố.

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
token	string	Token xác thực API
province_id	integer	ID của tỉnh, thành phố đã lấy trước đó

Output: Danh sách quận huyện theo tỉnh đã chọn.

c. Lấy thông tin phường, xã

Phương thức: GET

API: <https://online-gateway.ghn.vn/shiip/public-api/master-data/ward>

Mô tả: Lấy danh sách các phường, xã dựa trên quận, huyện.

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
token	string	Token xác thực API
district_id	integer	ID của quận, huyện đã lấy trước đó

Output: Danh sách phường, xã của quận huyện trước đó

d. Lấy thông tin gói dịch vụ

Phương thức: GET

API: <https://online-gateway.ghn.vn/shiip/public-api/v2/shipping-order/available-services>

Mô tả: Lấy danh sách các phương thức giao hàng khả dụng từ nơi gửi đến nơi nhận.

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
token	string	Token xác thực API
shop_id	integer	ID của shop trên GHN
from_district	integer	ID của quận, huyện người gửi
to_district	integer	ID của quận, huyện người nhận

Output: Thông tin gói dịch vụ

e. Tính giá cước vận chuyển

Phương thức: GET

API: <https://online-gateway.ghn.vn/shiip/public-api/v2/shipping-order/fee>

Mô tả: Tính toán giá cước vận chuyển dựa trên các thông tin vận chuyển cụ thể.

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
token	string	Token xác thực API
shop_id	integer	ID của shop
service_id	integer	ID của gói dịch vụ đã chọn
insurance_value	integer	Giá trị đơn hàng
to_district_id	integer	ID của Quận, Huyện người nhận
from_district_id	integer	ID của Quận, Huyện người gửi
weight	integer	Trọng lượng hàng hóa (gram)

f. Đặt đơn hàng

Phương thức: POST

API: /api/orders/place-order

Mô tả: Đặt một đơn hàng mới theo ID người dùng, hệ thống sẽ lấy đơn hàng đang ở trạng thái PENDING của người dùng, tức là giỏ hàng hiện tại để tiến hành đặt hàng, sau đặt hàng, đơn hàng sẽ chuyển sang trạng thái PLACED.

Input:

Tên thuộc tính	Kiểu dữ liệu	Mô tả
userId	integer	ID của người dùng

orderDate	string	Ngày đặt hàng
total	float	Tổng giá trị đơn hàng
shippingFee	number	Phí vận chuyển
discount	number	Giảm giá
receiverPhone	string	Số điện thoại người nhận
receiverAddress	string	Địa chỉ người nhận
receiverName	string	Tên người nhận

Output: Xác nhận đơn hàng đã được đặt

4.4.9. Trang chi tiết đơn hàng

a. Lấy thông tin đơn hàng

Phương thức: GET

API: /api/orders/{id}

Mô tả: Lấy thông tin chi tiết của đơn hàng theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của đơn hàng

Output: Thông tin chi tiết của đơn hàng

4.4.10. Trang thống kê

a. Lấy tất cả đơn hàng

Phương thức: GET

API: /api/orders/get-all

Mô tả: Lấy danh sách tất cả đơn hàng

Input: Không

Output: Danh sách tất cả các đơn hàng

b. Lấy danh sách đơn hàng theo trạng thái

Phương thức: GET

API: /api/orders/get-all-by-status

Mô tả: Lấy danh sách đơn hàng theo trạng thái

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
status	string	Trạng thái đơn hàng

Output:Danh sách đơn hàng theo trạng thái

4.4.11. Trang quản lý đơn hàng

a. Lấy danh sách đơn hàng theo trạng thái

Phương thức: GET

API: /api/orders/get-all-by-status

Mô tả: Lấy danh sách đơn hàng theo trạng thái

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
status	string	Trạng thái đơn hàng

Output:Danh sách đơn hàng theo trạng thái

b. Cập nhật trạng thái đơn hàng

Phương thức: PUT

API: /api/orders/update-status/{id}

Mô tả: Cập nhật trạng thái đơn hàng theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của đơn hàng
status	string	Trạng thái mới

Output: Xác nhận trạng thái đơn hàng đã được cập nhật

4.4.12. Trang quản lý sản phẩm

a. Lấy danh sách tất cả sản phẩm

Phương thức: GET

API: /api/products/get-all

Mô tả: Lấy danh sách tất cả sản phẩm

Input: Không có input

Output:Danh sách tất cả sản phẩm

b. Thêm sản phẩm mới

Phương thức: POST

API: /api/products/create

Mô tả: Thêm sản phẩm mới

Input:

Tên thuộc tính	Kiểu dữ liệu	Mô tả
id	integer	ID của sản phẩm
title	string	Tiêu đề sản phẩm

description	string	Mô tả sản phẩm
price	integer	Giá sản phẩm
publicationYear	integer	Năm xuất bản
image	string	URL hình ảnh sản phẩm
weight	float	Trọng lượng sản phẩm
category	List<category>	Danh sách thể loại
publishId	integer	Id nhà sản xuất
author	List<author>	danh sách tác giả
tag	List<tag>	Danh sách tag đặc trưng của sách

Output: Xác nhận sản phẩm đã được thêm

c. Cập nhật sản phẩm

Phương thức: PUT

API: /api/products/update/{id}

Mô tả: Cập nhật sản phẩm theo ID

Input:

Tên thuộc tính	Kiểu dữ liệu	Mô tả
id	integer	ID của sản phẩm
title	string	Tiêu đề sản phẩm
description	string	Mô tả sản phẩm
price	integer	Giá sản phẩm
publicationYear	integer	Năm xuất bản
image	string	URL hình ảnh sản phẩm
weight	float	Trọng lượng sản phẩm
category	List<category>	Danh sách thể loại
publishId	integer	Id nhà sản xuất
author	List<author>	danh sách tác giả
tag	List<tag>	Danh sách tag đặc trưng của sách

Output: Xác nhận tình trạng cập nhật của sản phẩm

d. Xóa sản phẩm

Phương thức: DELETE

API: /api/products/delete/{id}

Mô tả: Xóa sản phẩm theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của sản phẩm

Output: Tình trạng xóa sản phẩm

4.4.13. Trang quản lý thể loại

a. Lấy tất cả thể loại

Phương thức: GET

API: /api/categories/get-all

Mô tả: Lấy danh sách tất cả thể loại

Input: Không

Output: Danh sách tất cả thể loại

b. Thêm thể loại mới

Phương thức: POST

API: /api/categories/create

Mô tả: Thêm thể loại mới

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
name	string	Tên thể loại

Output: Xác nhận thể loại đã được thêm

c. Cập nhật thể loại

Phương thức: PUT

API: /api/categories/update/{id}

Mô tả: Cập nhật thể loại theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của thể loại
name	string	Tên thể loại

Output: Tình trạng cập nhật

d. Xóa thể loại

Phương thức: DELETE

API: /api/categories/delete/{id}

Mô tả: Xóa thể loại theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của thể loại

Output: Tình trạng xóa

4.4.14. Trang quản lý nhà xuất bản

a. Lấy danh sách nhà xuất bản

Phương thức: GET

API: /api/publishers/get-all

Mô tả: Lấy danh sách tất cả nhà xuất bản

Input: Không

Output: Danh sách tất cả nhà xuất bản

b. Thêm nhà xuất bản mới

Phương thức: POST

API: /api/publishers/create

Mô tả: Thêm nhà xuất bản mới

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
name	string	Tên nhà xuất bản
address	string	Địa chỉ
email	string	Địa chỉ email

Output: tình trạng thêm nhà xuất bản

c. Cập nhật nhà xuất bản

Phương thức: PUT

API: /api/publishers/update/{id}

Mô tả: Cập nhật nhà xuất bản theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của nhà xuất bản
name	string	Tên nhà xuất bản
address	string	Địa chỉ
email	string	Địa chỉ email

Output: Tình trạng cập nhật nhà xuất bản

d. Xóa nhà xuất bản

Phương thức: DELETE

API: /api/publishers/delete/{id}

Mô tả: Xóa nhà xuất bản theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của nhà xuất bản

Output: tình trạng xóa nhà xuất bản

4.4.15. Trang quản lý tác giả

a. Lấy danh sách tác giả

Phương thức: GET

API: /api/authors/get-all

Mô tả: Lấy danh sách tất cả tác giả

Input: Không

Output: Danh sách tất cả tác giả

b. Thêm tác giả mới

Phương thức: POST

API: /api/authors/create

Mô tả: Thêm tác giả mới

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
name	string	Tên tác giả
biography	string	Tiểu sử tác giả

Output: Tình trạng thêm tác giả

c. Cập nhật tác giả

Phương thức: PUT

API: /api/authors/update/{id}

Mô tả: Cập nhật tác giả theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của tác giả
name	string	Tên tác giả
biography	string	Tiểu sử tác giả

Output: Tình trạng cập nhật của tác giả

d. Xóa tác giả

Phương thức: DELETE

API: /api/authors/delete/{id}

Mô tả: Xóa tác giả theo ID

Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của tác giả

Output: tình trạng xóa tác giả

4.4.16. Trang quản lý người dùng

a. Lấy danh sách người dùng

Phương thức: GET

API: /api/users/get-all

Mô tả: Lấy danh sách tất cả người dùng

Input: Không

Output: danh sách người dùng

b. Lấy thông tin người dùng

Phương thức: GET

API: /api/users/{id}

Mô tả: Lấy thông tin người dùng theo ID

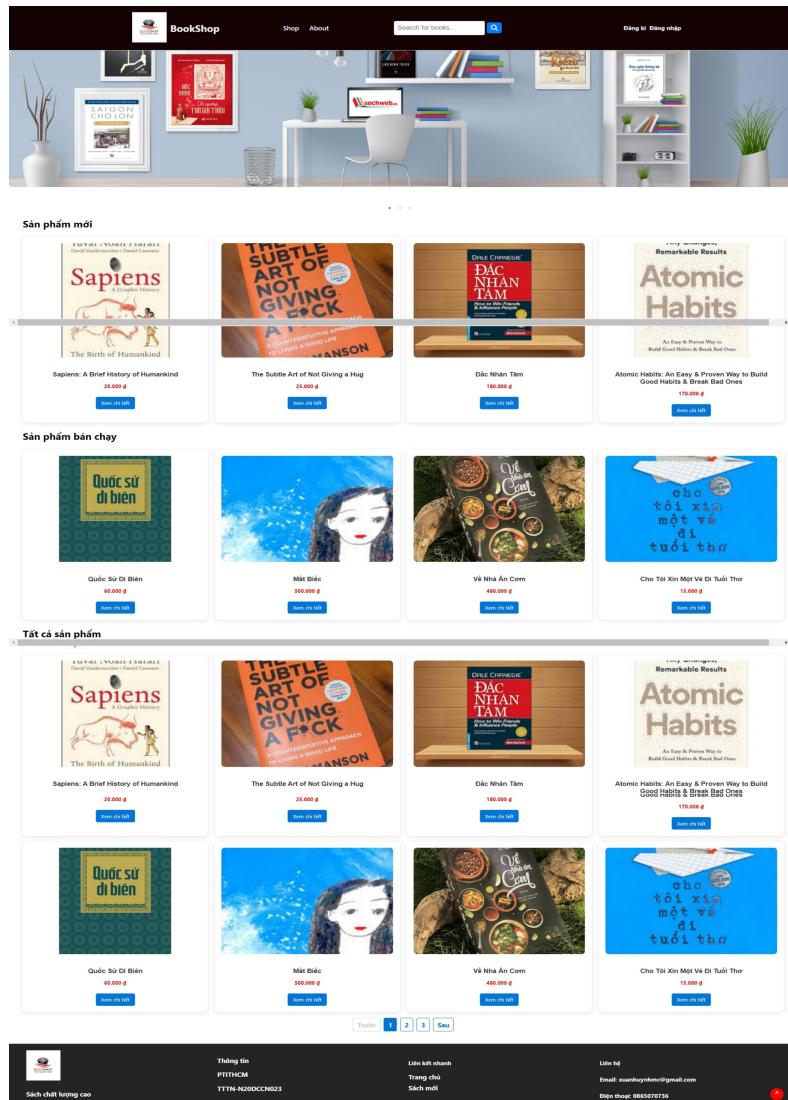
Input:

Tên thuộc tính	Kiểu giá trị	Mô tả
id	integer	ID của người dùng

Output: thông tin người dùng theo id

4.5. Thiết kế giao diện

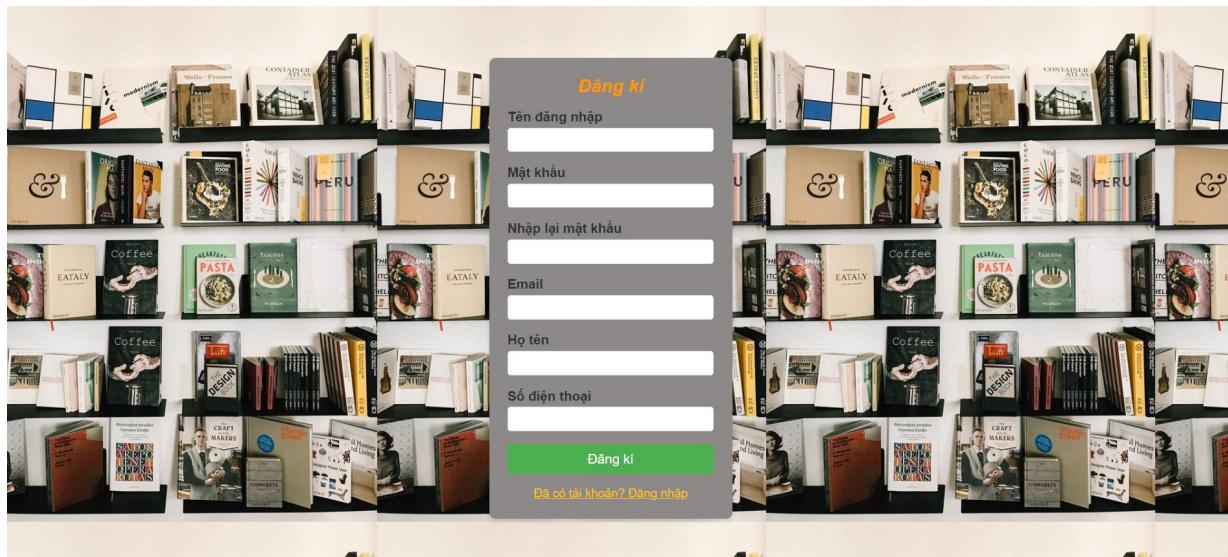
4.5.1. Trang chủ



Hình 4.2. Giao diện trang chủ

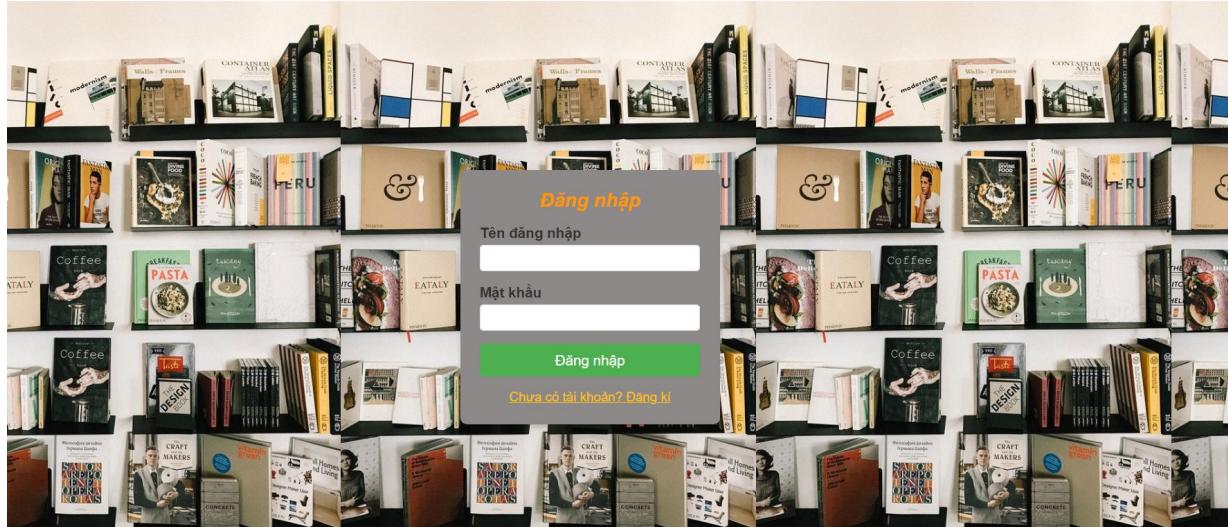
Trang chủ của website bán sách sẽ hiển thị danh sách sản phẩm mới, sản phẩm bán chạy, để người dùng có thể có thêm nhiều lựa chọn

4.5.2. Đăng kí

*Hình 4.3. Giao diện đăng ký*

Ở trang đăng ký, người dùng cần cung cấp các thông tin cần thiết để tạo một người dùng mới gồm tên đăng nhập, mật khẩu, email, họ tên và số điện thoại. Mật khẩu phải nhập lại để xác nhận. Tên đăng nhập và email nếu bị trùng với tài khoản đã có trong cơ sở dữ liệu sẽ được thông báo lỗi. Sau khi đăng ký thành công có thể tới trang đăng nhập để vào trang web thực hiện các tác vụ của người dùng như xem và thêm vào giỏ hàng, đặt hàng,..

4.5.3. Đăng nhập

*Hình 4.4. Giao diện đăng nhập*

Người dùng có tài khoản có thể đăng nhập vào trang web để thực hiện các nhu cầu cần thiết.

Nếu tài khoản có quyền Admin sẽ đến trang admin của trang web, trường hợp còn lại sẽ đến trang chủ của trang web.

4.5.4. Trang shop

The screenshot shows a website for a bookshop. At the top, there is a header with the logo "BookShop", a search bar containing "Search for books...", and a magnifying glass icon. To the right of the search bar are links for "Đăng kí" and "Đăng nhập". On the left side, there is a sidebar titled "Thể loại" (Categories) with the following options: Day nấu ăn, Giáo trình, Khoa học công nghệ, Kinh dị, bí ẩn, Lịch sử, Tâm lý, tình cảm, Thiếu nhi, Tiểu sử, tự truyện, Tiểu thuyết, Tôn giáo, Truyền cảm hứng, Văn hóa xã hội, and Văn học viễn tưởng. Below this sidebar is a yellow button labeled "Tìm kiếm nâng cao" (Advanced Search). The main content area displays a grid of eight book cards, each with a thumbnail image, title, price, and a "Xem chi tiết" (View details) button. The books shown are:

- Sapiens: A Brief History of Humankind by Yuval Noah Harari - 20.000 ₫
- The Subtle Art of Not Giving a F*ck by Mark Manson - 25.000 ₫
- Đắc Nhân Tâm by Dale Carnegie - 180.000 ₫
- Atomic Habits: An Easy & Proven Way to Build Good Habits & Break Bad Ones by James Clear - 170.000 ₫
- Quốc Sư Di Biên by various authors - 60.000 ₫
- Mắt Biếc by Nguyễn Huy Thiep - 500.000 ₫
- Về Nhà Ăn Cơm by various authors - 480.000 ₫
- Cho Tôi Xin Một Vé Đến Tuổi Thơ by various authors - 15.000 ₫

At the bottom of the page, there is a footer with sections for "Thông tin" (Information), "Liên kết nhanh" (Quick links), and "Liên hệ" (Contact). The "Thông tin" section includes the address "PTITHCM" and the code "TTTN-N20DCCN023". The "Liên kết nhanh" section includes links for "Trang chủ" and "Sách mới". The "Liên hệ" section includes an email address "Email: xuanhuynhmc@gmail.com" and a phone number "Điện thoại: 0865070736". There is also a small red circular icon with a white letter "A" in the bottom right corner.

Hình 4.5.Giao diện trang shop

Tại đây sẽ hiển thị tất cả sản phẩm và một danh sách thể loại để người dùng có thể lựa chọn tìm sách theo thể loại mình mong muốn, nếu người dùng muốn tìm kiếm nâng cao hơn có thể bấm vào “Tìm kiếm nâng cao” bên dưới các thể loại để đến trang tìm kiếm theo Boolean Search.

4.5.5. Chi tiết sản phẩm

Sapiens: A Brief History of Humankind

Planet Earth is 4.5 billion years old. In just a fraction of that time, one species among countless others has conquered it. Us. We are the most advanced and most destructive animals ever to have lived. What makes us brilliant? What makes us deadly? What makes us Sapiens?

Giá: 20000 đồng

Năm xuất bản: 2023

Còn lại: 10

Đánh giá:

Tác giả:

Yuval Noah Harari

Thể loại:

Lịch sử, Tiểu sử, tự truyện, Văn hóa xã hội, Văn học viễn tưởng

Tags:

thành công

Số lượng: 1

Thêm vào giỏ hàng

Bình luận

Chưa có bình luận nào.

Có thể bạn quan tâm

Sapiens: A Brief History of Humankind	The Subtle Art of Not Giving a Hug	Đặc Nhân Tâm	Atomic Habits
20.000 ₫	25.000 ₫	180.000 ₫	170.000 ₫

Hình 4.6. Giao diện trang chi tiết sản phẩm

Ở trang chi tiết sản phẩm sẽ hiển thị tất cả thông tin của sản phẩm như tiêu đề sách, số lượng trong kho, tác giả, thể loại, đánh giá và ô chọn số lượng và thêm vào giỏ hàng. Người dùng muốn mua hàng cần chọn số lượng cần mua và nhấn chọn “Thêm vào giỏ hàng”. Hệ thống sẽ không cho số lượng được chọn sẽ không thể vượt quá số lượng còn lại của sản phẩm. Nếu trạng thái của sách được admin cài đặt là “ngưng bán” sẽ không hiển thị ô nhập số lượng và nút “Thêm vào giỏ hàng” mà hiển thị “Sản phẩm tạm ngưng bán”

4.5.6. Giao diện form chọn tiêu chí tìm kiếm sách bằng Boolean Search

Tìm kiếm sách

Chọn tiêu chí tìm kiếm

Tên sách
 Tác giả
 Thể loại
 Nhà xuất bản
 Năm xuất bản
 Tag
 Đánh giá
 Giá

OK

Hình 4.7.Giao diện chọn tiêu chí tìm kiếm Boolean Search

Hệ thống sẽ cung cấp một số tiêu chí liên quan đến việc tìm kiếm sách theo mô hình Boolean, người dùng có thể chọn tiêu chí mà mình muốn liên quan đến việc tìm kiếm sách theo yêu cầu của bản thân.

4.5.7. Giao diện form tìm kiếm sách theo Boolean Search

Tìm kiếm sách

Tên sách:	<input type="text" value="mắt"/>	Độ ưu tiên:	<input type="text" value="3"/>
Tác giả:	<input type="text"/>	Độ ưu tiên:	<input type="text" value="1"/>
Thể loại:	<input type="text"/>	Độ ưu tiên:	<input type="text" value="1"/>
Nhà xuất bản:	<input type="text"/>	Độ ưu tiên:	<input type="text" value="1"/>
Năm xuất bản:	<input type="text"/>	Độ ưu tiên:	<input type="text" value="1"/>
Tag:	<input type="text"/>	Độ ưu tiên:	<input type="text" value="1"/>
Đánh giá (sao)từ:	<input type="text"/> - <input type="text"/>	Độ ưu tiên:	<input type="text" value="1"/>
Giá từ:	<input type="text"/> - <input type="text"/>	Độ ưu tiên:	<input type="text" value="1"/>



Mắt Biếc
 Tác giả: Nguyễn Nhật Ánh
 Thể loại:
 Nhà xuất bản: Tổng hợp TPHCM
 Năm xuất bản: 2022
 Tag:
 Đánh giá: 4.1 sao
 Giá: 500,000 VNĐ

Hình 4.8. Giao diện tìm kiếm sách Boolean Search

Từ các tiêu chí mà người dùng đã chọn trước đó, người dùng sẽ nhập các thông tin cho các tiêu chí để thực hiện việc tìm kiếm, bên cạnh từng tiêu chí sẽ có mức độ ưu tiên của mỗi tiêu chí đó, độ ưu tiên cho tiêu chí càng cao, việc sắp xếp danh sách có liên quan tới tiêu chí đó lên đầu sẽ càng lớn, thuận tiện theo ý muốn của người tìm kiếm. Khi sắp xếp, đối với mỗi tiêu chí, hệ thống sẽ kiểm tra xem sản phẩm có khớp với giá trị tiêu chí đó không. Nếu có, giá trị của Match_i sẽ là 1, nếu không sẽ là 0.

Công thức tổng quát, tính điểm ưu tiên cho mỗi sản phẩm:

$$\text{Score} = (\text{Match}_{\text{title}} \times \text{Weight}_{\text{title}}) + (\text{Match}_{\text{author}} \times \text{Weight}_{\text{author}}) + \dots$$

Ví dụ:

- Nếu sản phẩm có tên sách khớp và trọng số của tiêu chí tên sách là 5.
- Tác giả khớp và trọng số của tiêu chí tác giả là 3.
- Thể loại không khớp và trọng số của tiêu chí thể loại là 2.

Khi đó:

$$\text{Score} = (1 \times 5) + (1 \times 3) + (0 \times 2) = 8$$

4.5.8. Giỏ hàng

Sản phẩm	Đơn giá	Số lượng	Thành tiền
	25.000 đ	<input type="button" value="-"/> 1 <input type="button" value="+"/>	25.000 đ
	180.000 đ	<input type="button" value="-"/> 3 <input type="button" value="+"/>	540.000 đ
	480.000 đ	<input type="button" value="-"/> 1 <input type="button" value="+"/>	480.000 đ
	170.000 đ	<input type="button" value="-"/> 1 <input type="button" value="+"/>	170.000 đ

Tổng: 1.215.000 đ

[Tạo đơn hàng](#)

Hình 4.9. Giao diện giỏ hàng người dùng

Các sản phẩm được thêm vào giỏ hàng sẽ nằm ở trang giỏ hàng của người dùng. Hệ thống tính tổng số tiền của đơn hàng và hiển thị cho người dùng có thể dễ nhận biết. Tại đây vẫn có thể cập nhật số lượng mong muốn, và thực hiện xóa sản phẩm khỏi giỏ hàng khi không còn nhu cầu mong muốn, nếu cập nhật số lượng nhỏ hơn 1 hoặc bấm nút xóa sản phẩm khỏi đơn hàng sẽ có yêu cầu xác nhận xóa mới có thể xóa để phòng trường hợp lỡ tay xóa. Khi cảm thấy muốn mua sản phẩm trong giỏ hàng người dùng có thể đến trang tạo đơn hàng từ giỏ hàng từ nút Tạo đơn hàng

4.5.9. Trang đặt hàng

Sản phẩm	Đơn giá	Số lượng	Thành tiền
	20.000 ₫	23	460.000 ₫
	180.000 ₫	1	180.000 ₫

Tổng trọng lượng: 2400g

Thông tin vận chuyển

Tỉnh/Thành phố:

Quận/Huyện:

Phường/Xã:

Tên người nhận:

Số điện thoại:

Địa chỉ:

Tổng tiền sản phẩm: 640.000 ₫
Phi vận chuyển: 0 ₫
Giảm giá: 0 ₫

Tổng thanh toán: 640.000 ₫

Đặt hàng

Thông tin

PTITHCM

TTTN-N20DCCN023

Liên kết nhanh

Trang chủ

Sách mới

Liên hệ

Email: xuanhuynhmc@gmail.com

Điện thoại: 0865070736

Hình 4.10. Trang đặt hàng của người dùng

Khi người dùng bấm tạo đơn hàng từ trang giỏ hàng sẽ được đưa đến trang tạo đơn hàng. Tại đây sẽ hiển thị các sản phẩm được đặt hàng và số lượng tương tự như trong giỏ hàng.

Tổng khối lượng của các sản phẩm và số lượng của chúng cũng được tính và hiển thị ra giao diện.

Người dùng sẽ cung cấp thông tin địa chỉ từ tỉnh, thành phố đến quận/huyện đến phường/xã để cung cấp địa chỉ nhận hàng cho đơn hàng. Sau khi thông tin phường/xã được cập nhật thì hệ thống sẽ tự động tính giá tiền vận chuyển cho đơn hàng theo api cung cấp bởi hệ thống Giao hàng nhanh. Số tiền ưu đãi cũng sẽ được cập nhật theo phân loại khách hàng.

BẢNG GIÁ GIAO HÀNG

Đơn vị: VNĐ (Đã bao gồm 10% VAT)

TUYẾN	GÓI DỊCH VỤ	KHỐI LƯỢNG	NỘI THÀNH	HUYỆN/XÃ	THÊM 0.5 KG	THỜI GIAN GIAO
Nội Tỉnh (trừ Hà Nội, Hồ Chí Minh)	Đi bộ	0 - 3Kg	15.500	29.000	2.500	1 ngày
Nội Tỉnh (áp dụng cho Hà Nội, Hồ Chí Minh)	Đi bộ	0 - 0.5 Kg	21.000	29.000	2.500	1 ngày
Nội Vùng	Đi bộ	0 - 0.5 Kg	29.000	34.000	2.500	1 - 2 ngày
Nội Vùng Tỉnh	Đi bộ	0 - 0.5 Kg	29.000	34.000	2.500	2 - 3 ngày
Liên Vùng Đặc Biệt	Đi bộ	0 - 0.5 Kg	29.000	39.000	5.000	1 - 2 ngày
Liên Vùng	Đi bộ	0 - 0.5 Kg	29.000	39.000	5.000	2 - 3 ngày
Liên Vùng Tỉnh	Đi bộ	0 - 0.5 Kg	29.000	36.000	5.000	3 - 4 ngày

Ví dụ một đơn hàng nội thành Hà Nội nặng 2kg có cước phí là là 21.000 VNĐ + (2,500 x 3) = 28.500 VNĐ/ đơn.

Ưu đãi: Khách hàng thân thiết: Giảm 10% phí giao hàng. Nếu đơn hàng có tổng giá trị từ 2.000.000 trở lên được giảm 2% giá trị đơn hàng.

Khách hàng VIP: Giảm 30% phí giao hàng. Nếu đơn hàng có tổng giá trị từ 2.000.000 trở lên được giảm 3% giá trị đơn hàng.

Khách hàng trở thành thành viên “Thân thiết” sau 10 đơn hàng hoặc tổng giá trị mua hàng đạt 10 triệu đồng và trở thành thành viên “VIP” sau 25 đơn hàng hoặc tổng giá trị mua hàng đạt 25 triệu đồng.

4.5.10. Thông tin cá nhân

Hình 4.11.Giao diện thông tin cá nhân

Tại đây sẽ hiển thị thông tin người dùng đã cung cấp, người dùng có thể cập nhật thông tin và đổi mật khẩu.

4.5.11. Danh lịch sử đơn hàng

The screenshot displays a user interface for managing order history. At the top, it says "Đơn hàng đã đặt". Below this, there are two separate boxes, each representing a completed order (Mã đơn hàng: 3 and Mã đơn hàng: 4), showing details like date and total amount. At the bottom, there is a box for a canceled order (Mã đơn hàng: 1). The entire section is titled "Đơn hàng đã hủy".

Mã đơn hàng	Ngày đặt hàng	Tổng tiền
3	7/29/2024	1.100.000 đ
4	7/29/2024	10.000 đ
1		

Hình 4.12.Giao diện danh sách đơn hàng của người dùng

Trang lịch sử đơn hàng hiển thị tất cả đơn hàng của người dùng bao gồm đã đặt, đã hủy, đang giao và đã giao, để người dùng có thể nhận biết các đơn hàng đang có của bản thân

4.5.12. Giao diện trang Dashboard



Hình 4.13. Giao diện trang Dashboard

Khi vào trang Dashboard sẽ hiển thị biểu đồ thống kê đơn hàng theo tháng hiện tại, biểu đồ sẽ có thống kê số lượng và giá trị các loại đơn hàng gồm: đã đặt, đã hủy, đang giao và đã giao trong tháng . Doanh thu sẽ là tổng doanh thu của các đơn hàng đã giao trong tháng

4.5.13. Giao diện quản lý sản phẩm

ID	Hình ảnh	Tiêu đề	Trạng thái	Hành động
1		Sapiens: A Brief History of Humankind	Đang bán	<button>Sửa</button> <button>Xóa</button> <button>Ngừng bán</button>
2		The Subtle Art of Not Giving a Hug	Đang bán	<button>Sửa</button> <button>Xóa</button> <button>Ngừng bán</button>
3		Đắc Nhân Tâm	Đang bán	<button>Sửa</button> <button>Xóa</button> <button>Ngừng bán</button>
4		Atomic Habits: An Easy & Proven Way to Build Good Habits & Break Bad Ones	Đang bán	<button>Sửa</button> <button>Xóa</button> <button>Ngừng bán</button>
5		Quốc Sử Di Biên	Đang bán	<button>Sửa</button> <button>Xóa</button> <button>Ngừng bán</button>
6		Mắt Biếc	Đang bán	<button>Sửa</button> <button>Xóa</button> <button>Ngừng bán</button>
7		Về Nhà Ăn Cơm	Đang bán	<button>Sửa</button> <button>Xóa</button> <button>Ngừng bán</button>
8		Cho Tôi Xin Một Vé Đi Tuổi Thơ	Đang bán	<button>Sửa</button> <button>Xóa</button> <button>Ngừng bán</button>

Hình 4.14. Giao diện trang quản lý sản phẩm

Đây là trang quản lý của người dùng, người dùng có thể thêm sản phẩm qua form thêm sản phẩm, cần cung cấp các thông tin cần thiết cho sách mới (tiêu đề, mô tả, giá,...) từ nút “Thêm sản phẩm” ở đầu trang web.

Thêm sản phẩm

Tiêu đề:

Mô tả:

Giá:

Số lượng bán:

Năm xuất bản:

Nhà xuất bản:

Chọn nhà xuất bản

Thể loại:

Dạy nấu ăn
Giáo trình
Khoa học công nghệ
Kinh doanh
Lĩnh vực

Tác giả:

Yuval Noah Harari
Mark Manson
David Carnegie
James Clear
Nhà văn Nhật Anh

Tags:

Lịch sử loài người
phát triển nhân loại
tâm lý
xã hội loài người
Hàn lâm Khoa học

Cần năng:

Hình ảnh:

Choose File | No file chosen

Thêm sản phẩm

Hình 4.15. Form thêm mới sản phẩm

Khi cần cập nhật thông tin sản phẩm như thêm hình ảnh, thay đổi thể loại, cập nhật các thông tin sai sót có thể sửa sản phẩm qua form sửa sản phẩm từ nút Sửa

x

Sửa sản phẩm

Tiêu đề:

Mô tả:

Giá:

Số lượng bán:

Năm xuất bản:

Nhà xuất bản:

Thể loại:

Tác giả:

Tags:

Cân nặng:

Hình ảnh:

4.5.14. Giao diện quản lý thể loại

Tên thể loại	Hành động
Dạy nấu ăn	
Giáo trình	
Khoa học công nghệ	
Kinh dị, bí ẩn	
Lịch sử	
Tâm lý, tình cảm	
testđ	
Thiếu nhi	
Tiểu sử, tự truyện	
Tiểu thuyết	
Tôn giáo	
Truyền cảm hứng	
Văn hóa xã hội	
Văn học viễn tưởng	

Hình 4.16. Giao diện quản lý thể loại

Tại đây, ta có thể quản lý danh sách thể loại tương ứng với sách cho trang web, ta có thể thêm mới thể loại, sửa lại tên thể loại hoặc xóa, thể loại hiện có.

4.5.15. Giao diện quản lý nhà xuất bản

Tên	Địa Chỉ	Email	Hành Động
Trẻ	1618 Lý Chính Thắng, phường Võ Thị Sáu, Quận 3, TP. Hồ Chí Minh	hopthuabandoc@nvbtre.com.vn	
Kim Đồng	Địa chỉ: 55 Quang Trung, Hà Nội, Việt Nam	info@nvbkimdong.com.vn	
Tổng hợp TP.HCM	62 Nguyễn Thị Minh Khai, Phường Đa Kao, Quận 1, TP. HCM	lienhe@nxbcm.com	
Hội Nhà văn Việt Nam	6/86 Duy Tân, Cầu Giấy, Hà Nội	nstonhop@gmail.com	
Phụ nữ	39 Hàng Chuối, Quận Hai Bà Trưng, Hà Nội	truyenthongvapm@pn@gmail.com	
Lao động	6/86 Duy Tân, Cầu Giấy, Hà Nội	dongabooks.vn@gmail.com	
Nhà Nam	6/86 Duy Tân, Cầu Giấy, Hà Nội	dongabooks.vn@gmail.com	
Bách khoa Hà Nội	Số 1 Đường Đại Cồ Việt, Hai Bà Trưng, Hà Nội.	dongabooks.vn@gmail.com	
Chính trị Quốc gia Sự thật	6/86 Duy Tân, Cầu Giấy, Hà Nội	dongabooks.vn@gmail.com	
Công Thương	Tầng 4, Tòa nhà Bộ Công Thương, số 655 Phạm Văn Đồng, quận Bắc Từ Liêm, Hà Nội	dongabooks.vn@gmail.com	

Hình 4.17.Giao diện quản lý nhà xuất bản

Sách sẽ thuộc một nhà xuất bản sách, ở trang quản lý nhà xuất bản, ta có thể thêm mới nhà xuất bản, cập nhật thông tin nhà xuất bản và xóa nhà xuất bản nếu nó không có liên kết với sách

4.5.16. Giao diện quản lý tag

Tên	Hành động
lịch sử loài người	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
phát triển nhân loại	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
tiến hóa	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
xã hội loài người	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
tâm lý học tích cực	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
phát triển bản thân	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
quản lý cảm xúc	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
tiếp cận cuộc sống	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
thói quen	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
tâm lý học	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa
thành công	<input checked="" type="checkbox"/> Sửa <input type="checkbox"/> Xóa

Hình 4.18. Giao diện quản lý tag sách

Một quyền sách sẽ chứa các tag mang ý nghĩa cô đọng mà tác giả muốn người dùng biết đến, tại đây có thể thêm các tag mới để khi thêm sách mới có thể lựa chọn, và cũng có thể sửa hoặc xóa tag nếu cần thiết

4.5.17. Giao diện quản lý đơn hàng

a. Đơn hàng đã đặt

ĐƠN HÀNG ĐÃ ĐẶT				
Mã đơn	Ngày đặt	Tổng tiền	Người đặt	Hành động
5	8/8/2024, 1:59:19 PM	36.510 ₫	Trần Hải Nam	<input type="button" value="Đang giao"/> <input type="button" value="Hủy"/>
15	8/6/2024, 11:08:17 AM	45.000 ₫	Nguyễn hà hạnh	<input type="button" value="Đang giao"/> <input type="button" value="Hủy"/>
22	8/6/2024, 12:53:03 PM	266.500 ₫	Nguyễn hà hạnh	<input type="button" value="Đang giao"/> <input type="button" value="Hủy"/>

Hình 4.19.Giao diện quản lý đơn hàng đã đặt của khách

Trên giao diện quản lý đơn hàng đã đặt của khách hàng sẽ có thông tin sơ lược của đơn hàng bao gồm tổng tiền và 2 nút để chuyển đơn hàng sang đang giao hoặc hủy đơn, khi bấm vào đơn hàng sẽ hiển thị thông tin chi tiết đơn hàng

b. Đơn hàng đang giao

ĐƠN HÀNG ĐANG GIAO				
Mã đơn	Ngày đặt	Tổng tiền	Người đặt	Hành động
1	7/27/2024, 11:26:39 AM	120.000 ₫	Trần Hải Nam	<button>Đã giao</button> <button>Hủy</button>
3	7/29/2024, 9:56:42 PM	1.100.000 ₫	Huynh	<button>Đã giao</button> <button>Hủy</button>
7	8/6/2024, 10:36:05 AM	80.000 ₫	Nguyễn hà hạnh	<button>Đã giao</button> <button>Hủy</button>
23	8/6/2024, 12:56:47 PM	3.007.000 ₫	Nguyễn hà hạnh	<button>Đã giao</button> <button>Hủy</button>

Hình 4.20. Giao diện quản lý đơn hàng đang giao của khách

Trên giao diện quản lý đơn hàng đã đặt của khách hàng sẽ có thông tin sơ lược của đơn hàng bao gồm tổng tiền và 2 nút để chuyển đơn hàng sang đã giao hoặc hủy đơn, khi bấm vào đơn hàng sẽ hiển thị thông tin chi tiết đơn hàng

c. Đơn hàng đã hủy

ĐƠN HÀNG ĐÃ HỦY			
Mã đơn	Ngày đặt	Tổng tiền	Người đặt
2	7/28/2024, 11:26:39 AM	1.140.000 ₫	Nguyễn hà hạnh
25	8/7/2024, 10:16:12 AM	96.503 ₫	Mai Thanh Hải
27	8/7/2024, 10:48:38 AM	76.506 ₫	Mai Thanh Hải
28	8/7/2024, 11:22:28 AM	36.502 ₫	Mai Thanh Hải

Hình 4.21. Giao diện quản lý đơn hàng đã hủy của khách

Trên giao diện quản lý đơn hàng đã hủy của khách hàng sẽ có thông tin sơ lược của đơn hàng đã bị hủy bao gồm tổng tiền, khi bấm vào đơn hàng sẽ hiển thị thông tin chi tiết đơn hàng

d. Đơn hàng đã giao

ĐƠN HÀNG ĐÃ GIAO			
Mã đơn	Ngày đặt	Tổng tiền	Người đặt
8	8/7/2024, 6:30:26 AM	173.500 ₫	Mai Thanh Hải
26	8/7/2024, 10:44:44 AM	56.502 ₫	Mai Thanh Hải

Hình 4.22. Giao diện quản lý đơn hàng đã giao đến cho khách

Trên giao diện quản lý đơn hàng đã giao của khách hàng sẽ có thông tin sơ lược của đơn hàng đã giao bao gồm tổng tiền, khi bấm vào đơn hàng sẽ hiển thị thông tin chi tiết đơn hàng

e. Chi tiết đơn hàng

Chi tiết đơn hàng

Mã: 5

Người đặt: Trần Hải Nam

Số điện thoại người nhận: 0705685698

Địa chỉ người nhận: Xã Sập Vật, Huyện Tống Phan, Phù Cừ, Hưng Yên

Tên người nhận: Trần Hải Nam

Ngày đặt hàng: 8/8/2024, 1:59:19 PM

Trạng thái: Đã đặt hàng

Sản phẩm trong đơn hàng

Sản phẩm	Đơn giá	Số lượng	Thành tiền
 Atomic Habits: An Easy & Proven Way to Build Good Habits & Break Bad Ones James Clear	170.000 ₫	1	170.000 ₫

Tổng tiền sản phẩm: 170.000 ₫

Phí vận chuyển: 36.500 ₫

Giảm giá: 0 ₫

Tổng thanh toán: 206.500 ₫

Hình 4.23. Giao diện chi tiết đơn hàng khi admin xem thông tin đơn hàng

Ở trang chi tiết đơn hàng sẽ hiển thị thông tin của đơn hàng gồm các sách đã đặt cùng số lượng và giá cả, phí đơn hàng, phí vận chuyển và số tiền cần thanh toán cuối cùng

CHƯƠNG 5. KẾT LUẬN VÀ KIẾN NGHỊ

5.1. Kết luận

Mặc dù đã gặp không ít khó khăn nhưng với sự giúp đỡ của các thầy cô giáo, đặc biệt là thầy Nguyễn Anh Hào, em đã tìm hiểu thực tiễn hoạt động, giải quyết các vấn đề nghiệp vụ của trang web đặt mua sách trực tuyến và đã có gắng hoàn thành đồ án thực tập của mình theo thời gian yêu cầu đề ra.

Dự án xây dựng trang web thương mại điện tử sử dụng Java Spring Boot và ReactJS đã được triển khai thành công với đầy đủ các chức năng cần thiết để phục vụ các nhóm người dùng khác nhau. Trang web không chỉ cung cấp một nền tảng trực quan và dễ sử dụng cho khách vãng lai và người dùng đã đăng ký, mà còn cung cấp các công cụ quản lý mạnh mẽ cho các quản trị viên.

Tính năng nổi bật của hệ thống bao gồm:

- ✓ Khách vãng lai: Có khả năng xem và tìm kiếm sản phẩm, cũng như đăng ký tài khoản để trở thành người dùng chính thức.
- ✓ Người dùng đã đăng ký: Có thể thêm sản phẩm vào giỏ hàng, tạo đơn hàng, xem lịch sử đơn hàng, đánh giá và bình luận sản phẩm, cùng với việc chỉnh sửa thông tin cá nhân.
- ✓ Admin: Quản lý toàn bộ hệ thống bao gồm thông tin cá nhân, thống kê doanh số, quản lý sản phẩm và đơn hàng.
- ✓ Tìm kiếm sản phẩm bằng Boolean search và trả về danh sách sản phẩm với tiêu chí và trọng số ưu tiên do người dùng cung cấp
- ✓ Tính toán chi phí cho đơn hàng theo tổng khối lượng của vật phẩm trong đơn hàng và khu vực

Hệ thống đã được thiết kế với sự chú trọng vào trải nghiệm người dùng và tính linh hoạt trong quản lý, đảm bảo rằng người dùng có thể dễ dàng tương tác với các chức năng của trang web và các quản trị viên có đủ công cụ để quản lý hiệu quả.

5.2. Kiến nghị phát triển

- Cải thiện hiệu suất và tối ưu hóa: Để nâng cao hiệu suất của hệ thống, nên thực hiện các biện pháp tối ưu hóa hiệu suất, bao gồm việc tối ưu hóa truy vấn cơ sở dữ liệu, cải thiện tốc độ tải trang, và giảm thiểu kích thước tệp tin tĩnh.
- Bảo mật: Đảm bảo rằng các biện pháp bảo mật được cập nhật thường xuyên để bảo vệ thông tin người dùng và ngăn chặn các mối đe dọa từ bên ngoài. Mã hóa dữ liệu nhạy cảm và kiểm tra thường xuyên các lỗ hổng bảo mật.
- Giao diện người dùng: Tiếp tục cải thiện giao diện người dùng để mang lại trải nghiệm trực quan và dễ sử dụng hơn. Đánh giá phản hồi của người dùng và thực hiện các điều chỉnh dựa trên ý kiến đóng góp.

- Tính năng mở rộng: Xem xét việc tích hợp thêm các tính năng mới như hệ thống điểm thưởng cho người dùng, hỗ trợ nhiều ngôn ngữ, và các công cụ phân tích để giúp quản trị viên theo dõi hành vi của người dùng và hiệu suất của trang web.
- Hỗ trợ và bảo trì: Cung cấp tài liệu chi tiết cho người dùng và quản trị viên, cũng như xây dựng một hệ thống hỗ trợ khách hàng hiệu quả để giải quyết các vấn đề phát sinh và đảm bảo hệ thống hoạt động ổn định.
- Tích hợp thêm các hình thức thanh toán đa dạng, và tính toán chi phí hợp lý.
- Đánh giá và cập nhật: Thực hiện các đánh giá định kỳ về hệ thống và cập nhật các tính năng để đáp ứng nhu cầu thay đổi của người dùng và xu hướng công nghệ mới.

Việc thực hiện các kiến nghị này sẽ giúp hệ thống tiếp tục hoạt động hiệu quả, đáp ứng tốt hơn nhu cầu của người dùng và duy trì khả năng cạnh tranh trên thị trường.

DANH MỤC TÀI LIỆU THAM KHẢO

1. *Dinh Thị Mai.* (2020). Lập Trình Ứng Dụng Web với Spring Boot. Nhà Xuất Bản Đại học Bưu chính Viễn thông, Hà Nội
2. *Nguyễn Thị Lan.* (2020). Tìm Kiếm và Phân Tích Dữ Liệu với Boolean Search. Nhà Xuất Bản Đại học Bưu chính Viễn thông, Hà Nội.
3. *Richard Roe,* Advanced Boolean Search Techniques, Journal of Information Retrieval, Tập 10, trang 98-105, (2020)
4. *Nguyễn Thị Mai.* (2020). Hướng Dẫn Sử Dụng MySQL. Nhà Xuất Bản Khoa Học và Kỹ Thuật. ISBN: 978-604-78-0302-5.
5. *Nguyễn Minh Tuấn.* (2021). Tối Ưu Hóa Ứng Dụng Web với Spring Boot. Trong *Kỷ yếu Hội nghị Khoa học và Công nghệ Thông tin*, Đại học Bưu chính Viễn thông, Hà Nội.
6. <https://spring.io/learn>
7. <https://www.w3schools.com/react/>