

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ  
KHOA CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH  
HỌC KỲ I, NĂM HỌC 2025-2026

# SỬ DỤNG SPRING BOOT XÂY DỰNG WEBSITE BÁN MỸ PHẨM

*Giảng viên hướng dẫn:*

TS. Đoàn Phước Miền

*Sinh viên thực hiện:*

Họ tên: Lâm Tinh Tú

MSSV: 110122249

Lớp: DA22TTD

*Vĩnh Long, tháng 12 năm 2025*

TRƯỜNG KỸ THUẬT VÀ CÔNG NGHỆ  
KHOA CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CHUYÊN NGÀNH  
HỌC KỲ I, NĂM HỌC 2025-2026

# SỬ DỤNG SPRING BOOT XÂY DỰNG WEBSITE BÁN MỸ PHẨM

*Giảng viên hướng dẫn:*

TS. Đoàn Phước Miền

*Sinh viên thực hiện:*

Họ tên: Lâm Tinh Tú

MSSV: 110122249

Lớp: DA22TTD

*Vĩnh Long, tháng 12 năm 2025*

## This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

**Giáo viên hướng dẫn**  
(Ký tên và ghi rõ họ tên)

## This image shows a full page of a handwriting practice worksheet. It consists of approximately 28 horizontal rows. Each row is defined by two parallel dotted lines, creating a series of uniform gaps for letter height. The entire page is otherwise blank, with no margins, text, or other markings.

**Thành viên hội đồng**  
(Ký tên và ghi rõ họ tên)

## LỜI CẢM ƠN

Trước tiên, em xin gửi lời cảm ơn chân thành và sâu sắc đến thầy Đoàn Phước Miên.

Trong suốt quá trình thực hiện đồ án, thầy đã tận tình giảng dạy, hướng dẫn và đồng hành cùng em. Thầy không chỉ kiên nhẫn giải đáp mọi thắc mắc mà còn tận tâm chỉ dẫn từng bước đi cần thiết, giúp em hoàn thành tốt đề tài của mình. Nhờ sự dìu dắt và truyền đạt tận tình của thầy, em đã lĩnh hội được nhiều kiến thức và kỹ năng quý báu, làm nền tảng vững chắc cho sự phát triển sau này. Em vô cùng biết ơn thầy vì tất cả những gì thầy đã mang đến trong suốt thực hiện đề tài. Kính chúc thầy luôn mạnh khỏe, hạnh phúc và tiếp tục gặt hái thêm nhiều thành tựu trong sự nghiệp giáo dục cao quý của mình.

Em cũng xin gửi lời cảm ơn chân thành đến tất cả các bạn đã giúp đỡ và hỗ trợ em khi gặp những trở ngại và khó khăn trong quá trình thực hiện đồ án.

Dù đã nỗ lực hết mình, nhưng do kinh nghiệm còn hạn chế, báo cáo này khó tránh khỏi những thiếu sót. Em rất mong nhận được những góp ý, chỉ dẫn từ thầy và những người đã và sẽ đọc bài báo cáo này để có thể bổ sung, hoàn thiện hơn và nâng cao kiến thức, phục vụ tốt hơn cho các dự án, báo cáo trong tương lai.

Trân trọng cảm ơn thầy và mọi người!

Sinh viên thực hiện

Lâm Tinh Tú

## MỤC LỤC

<b>CHƯƠNG 1. Tổng quan .....</b>	<b>15</b>
1.1. Đặt vấn đề .....	15
1.2. Mục đích nghiên cứu.....	16
1.3. Giải pháp thực hiện .....	16
<b>CHƯƠNG 2. Nghiên cứu lý thuyết .....</b>	<b>19</b>
2.1. Tổng quan về Spring Boot .....	19
2.1.1. Khái niệm.....	19
2.1.2. Các tính năng chính .....	19
2.2. ReactJS .....	21
2.2.1. Khái niệm ReactJS.....	21
2.2.2. Ứng dụng của ReactJS.....	22
2.2.3. Ưu nhược điểm của ReactJS.....	22
2.3. PostgreSQL .....	23
2.3.1. Khái niệm về PostgreSQL .....	23
2.3.2. Chức năng của PostgreSQL.....	24
2.3.3. Ưu nhược điểm của PostgreSQL .....	25
2.3.3.1 <i>Ưu điểm của PostgreSQL</i> .....	25
2.3.3.2 <i>Nhược điểm của PostgreSQL</i> .....	25
2.4. Tổng quan về kiến trúc hệ thống.....	26
2.4.1. Giới thiệu về RESTful API.....	26
2.4.2. Kiến trúc Client – Server .....	27
2.4.3. Kiến trúc ba lớp trong Spring Boot .....	27
2.4.4. Giới thiệu về Spring Security và JWT .....	28

2.4.5. Giới thiệu về Spring Data JPA và Hibernate.....	29
<b>CHƯƠNG 3. thực HIỆN hóa nghiên cứu.....</b>	<b>30</b>
3.1. Mô tả bài toán.....	30
3.2. Đặc tả các yêu cầu.....	30
3.2.1. Yêu cầu chức năng.....	30
3.2.1.1 <i>Quản lý sản phẩm (Mỹ phẩm)</i> .....	30
3.2.1.2 <i>Quản lý khách hàng</i> .....	30
3.2.1.3 <i>Quản lý đơn hàng</i> .....	31
3.2.1.4 <i>Quản lý tài khoản</i> .....	31
3.2.1.5 <i>Quản lý thương hiệu</i> .....	31
3.2.1.6 <i>Quản lý bài viết đánh giá</i> .....	32
3.2.2. Yêu cầu phi chức năng .....	32
3.3. Thiết kế dữ liệu .....	33
3.3.1. Mô hình dữ liệu .....	33
3.3.2. Danh sách các thực thể và mối kết hợp .....	33
3.3.3. Chi tiết các thực thể .....	34
3.3.4. Danh sách các ràng buộc toàn vẹn.....	41
3.4. Thiết kế xử lý .....	42
3.4.1. Sơ đồ phân cấp chức năng .....	42
3.5. Sơ đồ Use Case .....	44
3.5.1. Use Case tổng quát .....	44
3.5.2. Use case tác nhân khách hàng .....	45
3.5.3. Use case tác nhân quản trị .....	49
3.5.4. Use case tác nhân nhân viên .....	53

3.6. Thiết kế API Endpoint .....	55
3.6.1. Các quy ước chung .....	55
3.6.2. Các endpoint xác thực và ủy quyền.....	57
3.6.3. Các endpoint liên quan đến quản lý người dùng .....	58
3.6.4. Các endpoint liên quan đến quản lý đơn hàng.....	59
3.6.5. Các endpoint liên quan đến quản lý vai trò .....	61
3.6.6. Các endpoint liên quan đến quản lý đánh giá.....	61
3.6.7. Các endpoint liên quan đến quản lý sản phẩm .....	62
3.6.8. Các endpoint liên quan đến bảng điều khiển.....	63
3.6.9. Các endpoint liên quan đến quản lý thương hiệu .....	64
3.6.10. Các endpoint liên quan đến quản lý hình ảnh.....	64
<b>CHƯƠNG 4. Kết quả nghiên cứu .....</b>	<b>66</b>
4.1. Giao diện đăng nhập và đăng ký .....	66
4.2. Giao diện người dùng.....	67
4.2.1. Giao diện trang chủ.....	67
4.2.2. Giao diện về chúng tôi.....	69
4.2.3. Giao diện trang trang điểm .....	70
4.2.4. Giao diện trang dưỡng da .....	71
4.2.5. Giao diện trang chăm sóc tóc.....	72
4.2.6. Giao diện tìm kiếm .....	73
4.2.7. Giao diện chi tiết sản phẩm .....	73
4.2.8. Giao diện viết đánh giá sản phẩm.....	74
4.2.9. Giao diện giỏ hàng.....	74
4.2.10. Giao diện đơn hàng của tôi.....	75



4.3. Giao diện quản trị.....	75
4.3.1. Giao diện tổng quan.....	75
4.3.2. Giao diện nhà sản xuất.....	76
4.3.3. Giao diện sản phẩm .....	77
4.3.4. Giao diện đơn hàng.....	77
4.3.5. Giao diện tài khoản người dùng .....	78
4.3.6. Giao diện bài viết đánh giá .....	79
4.4. Giao diện nhân viên .....	80
4.4.1. Giao diện tổng quan.....	80
4.4.2. Giao diện đơn hàng.....	81
4.4.3. Giao diện sản phẩm .....	81
4.4.4. Giao diện tra cứu khách hàng .....	82
4.5. Tài liệu API trên Swagger.....	82
4.5.1. API RESTful để quản lý các hoạt động bán lẻ sản phẩm, bao gồm:.....	82
4.5.2. Xác thực .....	83
4.5.3. Vai trò ủy quyền .....	83
4.5.4. Quy ước API.....	83
<b>CHƯƠNG 5. Kết luận và hướng phát triển.....</b>	<b>84</b>
5.1. Kết luận .....	84
5.1.1. Kết quả.....	84
5.1.2. Hạn chế .....	84
5.2. Hướng phát triển .....	85
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>86</b>

## DANH MỤC HÌNH ẢNH

Hình 3.1 Mô hình quan niệm dữ liệu của hệ thống .....	33
Hình 3.2 Sơ đồ phân cấp chức năng .....	42
Hình 3.3 Mô hình use case tổng quát .....	44
Hình 3.4 Mô hình use case tác nhân khách hàng .....	45
Hình 3.5 Mô hình use case tác nhân quản trị .....	49
Hình 3.6 Mô hình use case tác nhân nhân viên .....	53
Hình 4.1 Giao diện đăng nhập .....	66
Hình 4.2 Giao diện đăng ký.....	66
Hình 4.3 Giao diện quên mật khẩu .....	67
Hình 4.4 Giao diện trang chủ .....	67
Hình 4.5: Giao diện trang chủ (Sản phẩm mới) .....	68
Hình 4.6 Giao diện trang chủ (Sản phẩm bán chạy) .....	69
Hình 4.7 Giao diện về chúng tôi.....	69
Hình 4.8 Giao diện trang Trang điểm.....	70
Hình 4.9 Giao diện trang Trang điểm.....	70
Hình 4.10 Giao diện trang Dưỡng da .....	71
Hình 4.11 Giao diện trang Dưỡng da .....	71
Hình 4.12 Giao diện trang Chăm sóc tóc .....	72
Hình 4.13 Giao diện trang Chăm sóc tóc .....	72
Hình 4.14 Giao diện tìm kiếm .....	73
Hình 4.15 Giao diện chi tiết sản phẩm .....	73
Hình 4.16 Giao diện viết đánh giá sản phẩm .....	74
Hình 4.17 Giao diện giỏ hàng.....	74

Hình 4.18 Giao diện đơn hàng của tôi.....	75
Hình 4.19 Giao diện tổng quan của quản trị.....	75
Hình 4.20 Giao diện nhà sản xuất trong quản trị.....	76
Hình 4.21 Giao diện sản phẩm .....	77
Hình 4.22 Giao diện đơn hàng.....	77
Hình 4.23 Giao diện tài khoản người dùng .....	78
Hình 4.24 Giao diện bài viết đánh giá.....	79
Hình 4.25 Giao diện tổng quan của nhân viên .....	80
Hình 4.26 Giao diện đơn hàng.....	81
Hình 4.27 Giao diện sản phẩm .....	81
Hình 4.28 Giao diện tra cứu khách hàng.....	82

## DANH MỤC BẢNG BIỂU

Bảng 3.1: Danh sách các thực thể và mối kết hợp .....	34
Bảng 3.2: Chi tiết thực thể User .....	34
Bảng 3.3: Chi tiết thực thể Role .....	35
Bảng 3.4: Chi tiết thực thể Product .....	36
Bảng 3.5: Chi tiết thực thể Brand .....	37
Bảng 3.6 Chi tiết thực thể Images .....	37
Bảng 3.7 Chi tiết thực thể Reviews .....	38
Bảng 3.8 Chi tiết thực thể Orders .....	39
Bảng 3.9 Chi tiết thực thể Orders_items .....	40
Bảng 3.10: Danh sách các ràng buộc toàn vẹn.....	41
Bảng 3.11 Mô tả chi tiết use case tác nhân khách hàng .....	46
Bảng 3.12 Mô tả chi tiết use case tác nhân quản trị .....	50
Bảng 3.13 Mô tả chi tiết use case tác nhân nhân viên.....	53
Bảng 3.14 Các phương thức HTTP được sử dụng trong hệ thống website.....	55
Bảng 3.15 Các mã và thông báo phản hồi quy định chung .....	56
Bảng 3.16 Danh sách các endpoint xác thực và ủy quyền .....	57
Bảng 3.17 Danh sách các endpoint liên quan đến quản lý người dùng.....	58
Bảng 3.18 Danh sách các endpoint liên quan đến quản lý đơn hàng .....	59
Bảng 3.19 Danh sách các endpoint liên quan đến quản lý vai trò.....	61
Bảng 3.20 Danh sách các endpoint liên quan đến quản lý đánh giá .....	61
Bảng 3.21 Danh sách các endpoint liên quan đến quản lý sản phẩm.....	62
Bảng 3.22 Danh sách các endpoint liên quan đến bảng điều khiển .....	63
Bảng 3.23 Danh sách các endpoint liên quan đến quản lý thương hiệu.....	64

Bảng 3.24 Danh sách các endpoint liên quan đến quản lý hình ảnh .....	64
---	----

# **TÓM TẮT ĐỒ ÁN CHUYÊN NGÀNH**

## **SỬ DỤNG SPRING BOOT**

### **XÂY DỰNG WEBSITE BÁN MỸ PHẨM**

Mục tiêu:

Đề tài “Sử dụng Spring Boot xây dựng website bán mỹ phẩm” hướng đến việc phát triển một hệ thống thương mại điện tử chuyên cung cấp và kinh doanh các sản phẩm mỹ phẩm, giúp người dùng dễ dàng tìm kiếm, xem thông tin chi tiết, thêm sản phẩm vào giỏ hàng và thực hiện thanh toán trực tuyến. Mục tiêu chính của đề tài là ứng dụng công nghệ Spring Boot để xây dựng phần backend có khả năng xử lý nghiệp vụ, quản lý dữ liệu hiệu quả; đồng thời kết hợp ReactJS và Tailwind CSS trong phần frontend để mang lại giao diện hiện đại, thân thiện và phản hồi nhanh. Cơ sở dữ liệu của hệ thống được thiết kế và quản lý bằng PostgreSQL nhằm đảm bảo tính toàn vẹn, an toàn và khả năng mở rộng trong tương lai.

Hướng tiếp cận:

- Nghiên cứu lý thuyết: Tìm hiểu các công nghệ liên quan như Spring Boot, ReactJS, Tailwind CSS và PostgreSQL, cùng các tài liệu hướng dẫn thiết kế hệ thống thương mại điện tử.

- Nghiên cứu thực nghiệm: Xây dựng hệ thống phát triển backend bằng Spring Boot, thiết kế giao diện bằng ReactJS kết hợp Tailwind CSS. Quản lý dữ liệu bằng PostgreSQL.

Kết quả đạt được: Sử dụng công nghệ Spring Boot xây dựng website bán mỹ phẩm thành công và đáp ứng các yêu cầu đặt ra.

## MỞ ĐẦU

### 1. Lý do chọn đề tài

Hiện nay, thương mại điện tử phát triển mạnh mẽ và trở thành xu hướng mua sắm phổ biến, đặc biệt trong lĩnh vực mỹ phẩm nơi người tiêu dùng ngày càng ưu tiên sự tiện lợi và nhanh chóng. Việc xây dựng website bán mỹ phẩm giúp doanh nghiệp mở rộng thị trường, đồng thời mang đến trải nghiệm mua sắm hiện đại cho khách hàng.

Bên cạnh đó, việc lựa chọn sử dụng Spring Boot, kết hợp ReactJS và Tailwind CSS, cho phép người thực hiện áp dụng kiến thức đã học vào thực tế, tìm hiểu sâu hơn về phát triển web theo mô hình frontend và backend tách biệt. Đề tài không chỉ giúp củng cố kỹ năng lập trình và quản lý cơ sở dữ liệu, mà còn mang tính ứng dụng cao, phù hợp với xu hướng công nghệ hiện nay.

### 2. Mục tiêu nghiên cứu

Mục tiêu của đề tài “Sử dụng Spring Boot xây dựng website bán mỹ phẩm” là phát triển một hệ thống thương mại điện tử hiện đại, hỗ trợ hiệu quả việc quản lý và kinh doanh các sản phẩm mỹ phẩm. Website giúp người dùng dễ dàng tìm kiếm, xem thông tin chi tiết, và mua hàng trực tuyến một cách thuận tiện. Đồng thời, hệ thống hỗ trợ quản trị viên trong việc quản lý sản phẩm, đơn hàng, khách hàng và khuyến mãi, góp phần nâng cao hiệu quả kinh doanh, tiết kiệm thời gian và chi phí vận hành.

### 3. Đối tượng nghiên cứu

Đối tượng nghiên cứu của đề tài “Sử dụng Spring Boot xây dựng website bán mỹ phẩm” tập trung vào việc phát triển và quản lý hệ thống thương mại điện tử phục vụ cho hoạt động mua bán mỹ phẩm trực tuyến. Cụ thể, hệ thống hướng đến hai nhóm người dùng chính:

Khách hàng: những người truy cập website để xem thông tin, tìm kiếm, và đặt mua sản phẩm mỹ phẩm.

Quản trị viên: những người chịu trách nhiệm quản lý sản phẩm, đơn hàng, thông tin khách hàng, cũng như các chương trình khuyến mãi. Nghiên cứu tập trung vào cách

tổ chức, vận hành và tối ưu hóa các chức năng của website nhằm mang lại trải nghiệm mua sắm tiện lợi, nhanh chóng và hiệu quả.

#### **4. Phạm vi nghiên cứu**

Phạm vi nghiên cứu của đề tài “Sử dụng Spring Boot xây dựng website bán mỹ phẩm” tập trung vào việc phát triển một website thương mại điện tử phục vụ việc bán và quản lý các sản phẩm mỹ phẩm. Hệ thống nghiên cứu bao gồm các chức năng chính: quản lý sản phẩm, danh mục và thương hiệu, quản lý đơn hàng và khách hàng, hiển thị sản phẩm, giỏ hàng, đặt hàng và thanh toán trực tuyến.

Website hướng đến hai nhóm người dùng chính: khách hàng thực hiện mua sắm trực tuyến, và quản trị viên quản lý toàn bộ dữ liệu, đơn hàng và thống kê doanh thu. Nghiên cứu tập trung vào việc xây dựng hệ thống hoạt động ổn định, giao diện thân thiện, dễ sử dụng, và có khả năng mở rộng trong tương lai.



## CHƯƠNG 1. TỔNG QUAN

### 1.1. Đặt vấn đề

Trong bối cảnh thương mại điện tử và công nghệ 4.0 phát triển mạnh mẽ, nhu cầu mua sắm trực tuyến ngày càng tăng cao, đặc biệt trong lĩnh vực mỹ phẩm – nơi người tiêu dùng cần tra cứu, so sánh và lựa chọn sản phẩm một cách nhanh chóng và thuận tiện.

Tuy nhiên, nhiều doanh nghiệp và cửa hàng mỹ phẩm hiện nay vẫn quản lý sản phẩm, đơn hàng và khách hàng một cách thủ công hoặc thông qua các hệ thống chưa tối ưu, dẫn đến những hạn chế sau:

Quản lý sản phẩm và đơn hàng thủ công: Việc lưu trữ thông tin sản phẩm, kiểm soát tồn kho và xử lý đơn hàng bằng giấy tờ hoặc bảng tính tốn nhiều thời gian, dễ sai sót và khó tra cứu khi cần.

Thiếu khả năng theo dõi khách hàng và lịch sử mua hàng: Khách hàng khó nắm được thông tin chi tiết về đơn hàng, sản phẩm đã mua và các chương trình khuyến mãi.

Giao diện và trải nghiệm người dùng chưa thân thiện: Nhiều hệ thống chưa cung cấp trải nghiệm mua sắm trực tuyến nhanh chóng, hiện đại, dẫn đến khó thu hút và giữ chân khách hàng.

Trước thực trạng này, việc xây dựng một website bán mỹ phẩm sử dụng Spring Boot trở nên cần thiết. Hệ thống hướng đến là một nền tảng thương mại điện tử hiện đại, dễ sử dụng, hiệu quả, giúp:

Quản lý sản phẩm, danh mục, thương hiệu, khách hàng và đơn hàng một cách tập trung, chính xác và nhanh chóng.

Cung cấp cho khách hàng trải nghiệm mua sắm trực tuyến tiện lợi, tra cứu sản phẩm và thanh toán nhanh chóng.

Hỗ trợ quản trị viên theo dõi tình trạng đơn hàng, thống kê doanh thu và các chương trình khuyến mãi, tiết kiệm thời gian và chi phí vận hành.

Website ra đời nhằm giải quyết các vấn đề hạn chế trong quản lý bán hàng truyền thống, đồng thời đáp ứng nhu cầu ứng dụng công nghệ trong kinh doanh mỹ phẩm, nâng cao hiệu quả và chất lượng dịch vụ, tạo lợi thế cạnh tranh trên thị trường.

## **1.2. Mục đích nghiên cứu**

Mục đích chính khi thực hiện đề tài “Sử dụng Spring Boot xây dựng website bán mỹ phẩm” là xây dựng một hệ thống thương mại điện tử hiện đại, giúp quản lý và kinh doanh sản phẩm mỹ phẩm hiệu quả. Hệ thống hỗ trợ khách hàng dễ dàng tra cứu, tìm kiếm, xem thông tin chi tiết sản phẩm, thêm sản phẩm vào giỏ hàng và thực hiện thanh toán trực tuyến một cách nhanh chóng và thuận tiện. Đồng thời, website giúp quản trị viên quản lý sản phẩm, danh mục, thương hiệu, khách hàng và đơn hàng một cách khoa học, tiết kiệm thời gian và chi phí vận hành.

Hệ thống hướng đến nâng cao trải nghiệm người dùng thông qua giao diện trực quan, thân thiện, đồng thời cung cấp các chức năng mở rộng như thống kê doanh thu và quản lý chương trình khuyến mãi. Mục tiêu cuối cùng là xây dựng một nền tảng thương mại điện tử hiện đại, chuyên nghiệp, góp phần hiện đại hóa quy trình bán hàng và quản lý kinh doanh mỹ phẩm.

## **1.3. Giải pháp thực hiện**

Để xây dựng hệ thống website bán mỹ phẩm sử dụng Spring Boot, các giải pháp cụ thể được triển khai như sau:

+ Phân tích các yêu cầu của hệ thống:

Xác định mục tiêu và các chức năng chính của website, bao gồm:

Chức năng quản lý người dùng và quyền truy cập:

- Khách hàng: đăng ký, đăng nhập, quản lý thông tin cá nhân.
- Quản trị viên: phân quyền, quản lý tài khoản khách hàng, quản lý thông tin sản phẩm và đơn hàng.

Chức năng hiển thị và tìm kiếm:

- Trang chủ, danh mục sản phẩm, chi tiết sản phẩm.

- Tìm kiếm sản phẩm theo tên, danh mục hoặc thương hiệu.

Chức năng quản lý dữ liệu:

- Thêm, sửa, xóa sản phẩm, đơn hàng, danh mục, thương hiệu.
- Quản lý giỏ hàng, theo dõi trạng thái đơn hàng.

Chức năng thanh toán và lịch sử giao dịch:

- Khách hàng tiến hành đặt hàng, thanh toán trực tuyến.
- Xem lịch sử mua hàng và trạng thái đơn hàng.

+ Thiết kế hệ thống:

*Thiết kế giao diện người dùng:*

Tạo giao diện thân thiện, trực quan và dễ sử dụng, bao gồm menu, form nhập liệu, bảng dữ liệu và giỏ hàng.

Sử dụng ReactJS kết hợp Tailwind CSS để xây dựng giao diện hiện đại, phản hồi nhanh và tương thích trên nhiều thiết bị.

*Thiết kế cơ sở dữ liệu:*

Xác định các bảng cần thiết trong PostgreSQL: sản phẩm, danh mục, thương hiệu, khách hàng, đơn hàng, chi tiết đơn hàng.

Thiết kế quan hệ giữa các bảng đảm bảo tính toàn vẹn dữ liệu và khả năng truy vấn nhanh.

*Thiết kế logic hệ thống:*

Phân tách rõ ràng các module chức năng: quản lý sản phẩm, quản lý khách hàng, quản lý đơn hàng, thống kê.

Backend được xây dựng bằng Spring Boot theo kiến trúc RESTful API, đảm bảo tách biệt frontend – backend, dễ bảo trì và mở rộng.

+ Phát triển phần mềm

Backend: Sử dụng Spring Boot để xây dựng và kết nối cơ sở dữ liệu, xử lý đăng nhập, thanh toán, quản lý sản phẩm và đơn hàng.

Frontend: Sử dụng ReactJS kết hợp Tailwind CSS để xây dựng giao diện.

Cơ sở dữ liệu: Sử dụng PostgreSQL để lưu trữ thông tin sản phẩm, người dùng, đơn hàng và các bảng liên quan.

+ Kiểm thử hệ thống

Kiểm thử các chức năng thêm, sửa, xóa, tìm kiếm sản phẩm, giỏ hàng, đơn hàng hoạt động chính xác.. Giao diện hoạt động tốt trên các trình duyệt khác nhau.

## CHƯƠNG 2. NGHIÊN CỨU LÝ THUYẾT

### 2.1. Tổng quan về Spring Boot

#### 2.1.1. Khái niệm

Spring Boot là một framework định hướng được xây dựng dựa trên Spring Framework, với mục tiêu giúp các nhà phát triển xây dựng các ứng dụng dựa trên Spring một cách nhanh chóng và dễ dàng. [1]

Spring Boot không phải là một framework hoàn toàn mới mà là một phần mở rộng của Spring Framework nhằm mục đích đơn giản hóa quá trình phát triển ứng dụng dựa trên Spring. [1]

#### 2.1.2. Các tính năng chính

##### Tính năng cốt lõi

- *Spring Boot Starters*

Spring Boot cung cấp các “starter” là những gói phụ thuộc được định nghĩa sẵn, giúp nhà phát triển dễ dàng khởi tạo ứng dụng theo từng mục đích cụ thể như web, JPA, hay security. Các starter này giúp giảm thiểu việc cấu hình thủ công và loại bỏ tình trạng xung đột phiên bản giữa các thư viện. [1]

- *Tự động cấu hình*

Cơ chế tự động cấu hình cho phép Spring Boot tự động nhận biết và cấu hình các thành phần cần thiết của ứng dụng dựa trên những thư viện có sẵn trong classpath.

Cơ chế tự động cấu hình cho phép Spring Boot tự động nhận biết và cấu hình các thành phần cần thiết của ứng dụng dựa trên những thư viện có sẵn trong classpath. Tính năng này giúp tiết kiệm thời gian, giảm lỗi cấu hình và được kích hoạt thông qua annotation `@EnableAutoConfiguration` (được bao gồm trong `@SpringBootApplication`).

##### Tính năng sẵn sàng cho môi trường sản xuất

- *Spring Boot Actuator*

Actuator cung cấp các chức năng sẵn sàng cho môi trường sản xuất như giám sát hệ thống, đo lường hiệu suất, và kiểm tra tình trạng hoạt động của ứng dụng. Một số endpoint phổ biến bao gồm:

/actuator/health: kiểm tra trạng thái hoạt động của ứng dụng (UP/DOWN);

/actuator/info: hiển thị thông tin ứng dụng;

/actuator/beans, /actuator/env, /actuator/mappings: hỗ trợ kiểm tra cấu hình nội bộ.

Nhờ đó, việc theo dõi và quản lý ứng dụng trong môi trường sản xuất trở nên dễ dàng và trực quan hơn. [1]

#### - ***Hỗ trợ Container Servlet Nhúng***

Spring Boot cho phép đóng gói ứng dụng thành một tệp JAR có khả năng chạy độc lập, mà không cần triển khai lên máy chủ ứng dụng bên ngoài. Container mặc định được nhúng là Tomcat, và người dùng có thể dễ dàng thay thế bằng Jetty hoặc Undertow nếu cần thiết.

### **Tính năng hỗ trợ phát triển và cấu hình**

#### - ***Quản lý cấu hình ngoại vi***

Spring Boot hỗ trợ quản lý cấu hình thông qua các tệp application.properties hoặc application.yml, giúp dễ dàng thay đổi các thông số mà không cần chỉnh sửa mã nguồn. Ngoài ra, cấu hình có thể được ghi đè bằng biến môi trường, tham số dòng lệnh hoặc các nguồn khác. Annotation @ConfigurationProperties giúp liên kết cấu hình này vào các bean một cách an toàn kiểu dữ liệu. [1]

#### - ***Cấu hình theo môi trường***

Profiles cho phép định nghĩa các tập cấu hình riêng biệt cho từng môi trường. Các tệp cấu hình thường được đặt tên theo định dạng application-{profile}.properties, giúp việc chuyển đổi giữa các môi trường trở nên linh hoạt và tiện lợi.

#### - ***Công cụ phát triển***

Mô-đun Devtools giúp tăng tốc quá trình phát triển bằng cách tự động khởi động lại ứng dụng khi có thay đổi trong mã nguồn. Đồng thời, nó cũng thiết lập sẵn các giá

trị mặc định tối ưu cho môi trường phát triển, chẳng hạn như tắt bộ nhớ đệm của giao diện.

### **Các tính năng mở rộng và hỗ trợ phát triển**

#### **- *Hỗ trợ lập trình phản ứng***

Spring Boot hỗ trợ phát triển các ứng dụng phản ứng thông qua Spring WebFlux, cho phép xử lý bất đồng bộ và đạt hiệu năng cao hơn. Trong mô hình này, container mặc định được sử dụng là Netty. [1]

#### **- *Hỗ trợ Native Image và AOT***

Spring Boot cung cấp khả năng biên dịch ứng dụng thành tệp thực thi gốc thông qua GraalVM, giúp tối ưu hóa thời gian khởi động và giảm chi phí vận hành. Công nghệ AOT (Ahead-of-Time) cho phép xử lý cấu hình ngay trong giai đoạn build thay vì runtime.

#### **- *Containerization Docker và Buildpacks***

Spring Boot hỗ trợ tích hợp trực tiếp với công nghệ container hóa như Docker. Các plugin Maven và Gradle có thể tự động tạo Docker image thông qua Cloud-Native Buildpacks, giúp quá trình triển khai trở nên nhanh chóng mà không cần viết Dockerfile thủ công. Ngoài ra, mô-đun spring-boot-docker-compose giúp dễ dàng quản lý các dịch vụ phụ trợ như cơ sở dữ liệu hoặc message broker.

#### **- *Mở rộng***

Spring Boot cho phép các nhà phát triển hoặc tổ chức tạo ra Custom Starter riêng để đóng gói các cấu hình và mô-đun dùng chung, giúp chuẩn hóa và tái sử dụng trong nhiều dự án khác nhau.

## **2.2. ReactJS**

### **2.2.1. Khái niệm ReactJS**

ReactJS là một thư viện JavaScript có tính hiệu quả và linh hoạt để xây dựng các thành phần giao diện người dùng (UI) có thể sử dụng lại. ReactJS giúp phân chia các UI phức tạp thành các thành phần nhỏ (được gọi là component). Nó được tạo ra bởi Jordan

Walke, một kỹ sư phần mềm tại Facebook. ReactJS ban đầu được phát triển và duy trì bởi Facebook và sau đó được sử dụng trong các sản phẩm của mình như WhatsApp & Instagram. [2]

ReactJS được dùng để xây dựng các ứng dụng [single page application] (SPA). Một trong những điểm hấp dẫn của ReactJS là nó không chỉ được xây dựng bên phía clients mà còn sử dụng được bên phía server.

### **2.2.2. Ứng dụng của ReactJS**

Ứng dụng Web hiện đại (SPA - Single Page Applications): Xây dựng các trang web có khả năng cập nhật nội dung mà không cần tải lại toàn bộ trang. Ví dụ: Facebook, Instagram, Twitter. [2]

Ứng dụng thương mại điện tử: Tạo giao diện tương tác nhanh chóng và mượt mà cho giỏ hàng, danh sách sản phẩm, tìm kiếm và bộ lọc. Ví dụ: Shopee, Amazon.

Ứng dụng quản lý: Quản lý dữ liệu lớn với giao diện động, biểu đồ và báo cáo. Ví dụ: Hệ thống quản lý doanh nghiệp, quản lý tài chính.

Ứng dụng di động: ReactJS là nền tảng cho React Native, giúp xây dựng ứng dụng di động cho cả Android và iOS từ một codebase duy nhất. Ví dụ: Uber Eats, Airbnb.

### **2.2.3. Ưu nhược điểm của ReactJS**

#### **Ưu điểm:**

Hiệu suất cao nhờ Virtual DOM: Virtual DOM giúp React cập nhật giao diện một cách tối ưu mà không cần thao tác trực tiếp lên DOM thực, làm tăng tốc độ và hiệu suất.

Tái sử dụng các Component: Component là các khối giao diện có thể tái sử dụng, giúp giảm công sức phát triển và bảo trì mã nguồn.

Hỗ trợ mạnh mẽ từ cộng đồng: ReactJS có một cộng đồng lớn, thư viện và công cụ phong phú, giúp giải quyết nhiều vấn đề phức tạp.

Khả năng kết hợp với các công cụ khác: ReactJS có thể tích hợp với các công cụ như Redux, React Router, và các thư viện bên ngoài để tăng cường chức năng.



Dễ học và sử dụng: ReactJS sử dụng cú pháp JSX (JavaScript XML), giúp nhà phát triển dễ dàng viết và hiểu mã nguồn.

SEO Friendly: với sự hỗ trợ từ ReactDOMServer hoặc Next.js, React có thể render phía server (Server-Side Rendering - SSR), cải thiện SEO.

Tính mô-đun hóa cao: React giúp tổ chức code theo từng module, dễ dàng quản lý và mở rộng.

### **Nhược điểm:**

Chỉ là thư viện UI: React chỉ tập trung vào giao diện người dùng, không phải framework toàn diện. Nhà phát triển phải kết hợp thêm các công cụ khác để quản lý state (như Redux) hoặc xử lý API.

Học tập phức tạp cho người mới: ReactJS yêu cầu hiểu biết về JavaScript hiện đại (ES6+) và các khái niệm như props, state, lifecycle, hooks, v.v., có thể khó khăn cho người mới.

Cập nhật thường xuyên: React liên tục thay đổi và cập nhật, khiến việc theo kịp các phiên bản mới và áp dụng các tính năng mới trở nên khó khăn.

JSX không quen thuộc: JSX có thể khiến người mới cảm thấy khó khăn vì nó kết hợp cả JavaScript và HTML trong cùng một file.

Quản lý state phức tạp: Dù React có nhiều thư viện quản lý state (như Redux, Context API), nhưng việc lựa chọn và triển khai đôi khi khiến ứng dụng trở nên phức tạp.

## **2.3. PostgreSQL**

### **2.3.1. Khái niệm về PostgreSQL**

PostgreSQL là một hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS) mã nguồn mở hỗ trợ SQL phổ biến, cùng với MySQL. PostgreSQL được xây dựng nhiều chức năng phục vụ việc truy xuất dữ liệu phù hợp với nhiều Framework lớn. [3]

### 2.3.2. Chức năng của PostgreSQL

PostgreSQL được sử dụng rộng rãi trong nhiều ngữ cảnh khác nhau trong phát triển phần mềm và quản lý dữ liệu: [3]

- Hệ thống cơ sở dữ liệu chính: PostgreSQL là một lựa chọn phổ biến cho các ứng dụng cấp doanh nghiệp và các ứng dụng web.
- Lưu trữ dữ liệu ứng dụng Spring Boot: PostgreSQL có thể được sử dụng như một máy chủ cơ sở dữ liệu cho các ứng dụng Spring Boot. Nó có các dependency driver JDBC được sử dụng trong các ứng dụng này.
- Hỗ trợ ORM và JPA: PostgreSQL hoạt động tốt với các framework Ánh xạ Quan hệ Đối tượng (ORM) như Hibernate thông qua Java Persistence API (JPA), cho phép lập trình viên Java ánh xạ các đối tượng Java tới các bản ghi SQL.
- Phát triển và thử nghiệm: PostgreSQL được sử dụng làm cơ sở dữ liệu cho môi trường phát triển và thử nghiệm. Các công cụ như Testcontainers có thể được sử dụng để khởi động các container PostgreSQL Docker để thử nghiệm.
- Hỗ trợ đa dạng dữ liệu: PostgreSQL là một trong số ít RDBMS hỗ trợ các loại dữ liệu phức tạp hơn, chẳng hạn như lưu trữ các loại JSON và key-value. Ngoài ra, công nghệ này còn hỗ trợ các loại dữ liệu địa lý và hình học thông qua tiện ích mở rộng PostGIS, cho phép truy vấn không gian phức tạp. Cuối cùng, PostgreSQL hỗ trợ các tính năng đối tượng-quan hệ (object-relational features).
- Triển khai Cloud và Container: PostgreSQL có thể được triển khai dễ dàng trên các nền tảng Cloud, và thường được sử dụng trong môi trường Docker/Kubernetes.

### 2.3.3. Ưu nhược điểm của PostgreSQL

#### 2.3.3.1 Ưu điểm của PostgreSQL

Đảm bảo tính toàn vẹn dữ liệu: PostgreSQL nổi tiếng vì hỗ trợ các giao dịch ACID (Atomicity, Consistency, Isolation, Durability). Tính năng này làm cho PostgreSQL đặc biệt phù hợp cho các tình huống mà sự nhất quán là yếu tố quan trọng.

Mô hình dữ liệu linh hoạt: PostgreSQL là lựa chọn lý tưởng cho các dự án yêu cầu schema được xác định trước và cấu trúc dữ liệu dễ dự đoán và tương đối đơn giản.

Độ trưởng thành và cộng đồng: PostgreSQL được xem là một hệ thống trưởng thành với một cộng đồng các nhà phát triển và đóng góp lớn. Các mô hình thiết kế, triển khai và vận hành đã được biết rõ giúp nó dễ dàng được áp dụng, phát triển và tích hợp.

Khả năng tương thích: Nó tương thích tốt với các framework ORM như Hibernate và JPA. Spring Boot cũng hỗ trợ PostgreSQL như một nền tảng cơ sở dữ liệu có thể sử dụng cho các tập lệnh khởi tạo cơ sở dữ liệu và tích hợp thông qua các starter modules.

#### 2.3.3.2 Nhược điểm của PostgreSQL

*Khả năng mở rộng hạn chế:*

Các cơ sở dữ liệu quan hệ như PostgreSQL không thể xử lý tốt việc mở rộng theo chiều ngang bằng cách thêm nhiều nút phân tán.

Chúng thường dựa vào mở rộng theo chiều dọc bằng cách nâng cấp CPU và bộ nhớ của một máy chủ lớn duy nhất, điều này có giới hạn vật lý và dẫn đến chi phí cao hơn cho việc nâng cấp phần cứng.

*Ưu tiên tính nhất quán hơn tính sẵn sàng:*

Mô hình RDBMS tuân thủ các đảm bảo ACID. Để duy trì tính nhất quán nghiêm ngặt này, các cơ sở dữ liệu quan hệ thường phải đánh đổi tính sẵn sàng và khả năng chịu phân vùng một khía cạnh của định lý CAP.

Việc đảm bảo ACID khiến cho RDBMS không phù hợp với các kịch bản cần tính sẵn sàng cực cao vốn thường chấp nhận tính nhất quán cuối cùng để đạt được hiệu suất tốt hơn.

*Độ phức tạp trong vận hành cụm:*

Thiết lập các cấu hình phức tạp như nhân bản và chuyển đổi tự động trong RDBMS thường phức tạp hơn và đòi hỏi sự phối hợp cao hơn.

*Hạn chế trong lập trình phản ứng:*

Các driver JDBC truyền thống cho cơ sở dữ liệu quan hệ thường là blocking, gây ra vấn đề trong các môi trường lập trình phản ứng hiện đại vì các nhà cung cấp cơ sở dữ liệu quan hệ truyền thống chưa cung cấp các trình điều khiển không chặn tương ứng.

## **2.4. Tổng quan về kiến trúc hệ thống**

### **2.4.1. Giới thiệu về RESTful API**

REST (Representational State Transfer) là phong cách kiến trúc do Roy Fielding đề xuất năm 2000, nhằm thiết kế các dịch vụ web có hiệu suất cao, dễ mở rộng, đơn giản và tương thích trên nhiều nền tảng. Các dịch vụ tuân theo phong cách này được gọi là RESTful API.

RESTful API dựa trên 6 ràng buộc kiến trúc, trong đó quan trọng nhất là mô hình Client–Server và tính không trạng thái (stateless), giúp hệ thống dễ mở rộng và giảm phụ thuộc giữa các thành phần.

Các API hoạt động thông qua endpoint được định danh bằng URI, sử dụng các phương thức HTTP tiêu chuẩn:

GET: lấy dữ liệu

POST: tạo dữ liệu mới

PUT: cập nhật dữ liệu

DELETE: xóa dữ liệu

Dữ liệu trao đổi chủ yếu ở định dạng JSON vì nhẹ, dễ đọc và xử lý. Việc mô tả API thường dùng OpenAPI, hỗ trợ kiểm thử và phát triển tự động.

Để bảo mật, RESTful API thường sử dụng các cơ chế xác thực như Token, Cookie hoặc API Key.

Thách thức hiện nay của RESTful API là việc kiểm thử, do thiếu thông tin nội bộ, tài liệu không đồng bộ với triển khai thực tế và sự phức tạp trong các API yêu cầu xác thực. [4]

### 2.4.2. Kiến trúc Client – Server

Kiến trúc Client–Server là mô hình phần mềm gồm hai thành phần chính: Client (bên gửi yêu cầu) và Server (bên xử lý và cung cấp dịch vụ). Client gửi yêu cầu, Server xử lý dữ liệu (thường là các tác vụ nặng) và trả kết quả lại cho Client, giúp tối ưu hiệu suất hệ thống.

Client và Server giao tiếp thông qua các giao thức tiêu chuẩn như HTTP, FTP, SMTP. Trong RESTful API, kiến trúc Client–Server là ràng buộc cốt lõi, nơi Client sử dụng các phương thức HTTP (GET, POST, PUT, DELETE) để truy cập tài nguyên trên Server thông qua URI.

Về cấu trúc, kiến trúc này thường được triển khai dưới:

Mô hình 2 tầng (2-tier): Client chứa cả giao diện và logic nghiệp vụ, kết nối trực tiếp với Server cơ sở dữ liệu.

Mô hình 3 tầng (3-tier): Bổ sung tầng trung gian (Application Server), Client chỉ đảm nhiệm giao diện, còn logic nghiệp vụ được xử lý ở máy chủ ứng dụng.

Lợi ích của kiến trúc Client–Server là khả năng chia sẻ tài nguyên, quản lý dữ liệu tập trung và dễ mở rộng. Mô hình này được ứng dụng rộng rãi trong ATM, email và các dịch vụ điện toán đám mây. [5]

### 2.4.3. Kiến trúc ba lớp trong Spring Boot

Kiến trúc 3-tier trong Spring Boot là mô hình thiết kế giúp hệ thống dễ mở rộng, nâng cao hiệu suất và thuận tiện cho việc bảo trì. Mô hình gồm ba lớp chính: Presentation – Application – Data.

Tầng trình diễn (Presentation Layer – Controller)

Là lớp tiếp xúc trực tiếp với người dùng, sử dụng các `@RestController` để tiếp nhận yêu cầu HTTP (GET, POST, PUT, DELETE), điều hướng xử lý và trả dữ liệu phản hồi (thường ở dạng JSON). Lớp này chỉ tập trung vào logic trình bày.

### Tầng nghiệp vụ (Application Layer – Service)

Là lớp trung gian thực hiện các logic nghiệp vụ của hệ thống. Các Service nhận dữ liệu từ Controller, xử lý theo quy tắc kinh doanh và gọi đến tầng dữ liệu khi cần. Spring Boot sử dụng Dependency Injection (DI) để giảm sự phụ thuộc giữa các thành phần.

### Tầng truy cập dữ liệu (Data Layer – Repository)

Chịu trách nhiệm làm việc trực tiếp với cơ sở dữ liệu thông qua các Repository, thực hiện các thao tác CRUD (thêm, sửa, xóa, truy vấn). Việc tách riêng tầng này giúp quản lý dữ liệu tập trung và hiệu quả.

Spring Boot hoạt động như một IoC Container, tự động quét các component, quản lý vòng đời Bean và liên kết các lớp với nhau thông qua Application Context, đảm bảo hệ thống vận hành thống nhất và linh hoạt. [6]

## 2.4.4. Giới thiệu về Spring Security và JWT

Spring Security là framework bảo mật mạnh mẽ cho ứng dụng Java, xuất hiện từ năm 2003, chuyên xử lý xác thực (authentication) và phân quyền (authorization) nhằm bảo vệ ứng dụng trước các mối đe dọa an ninh phổ biến.

JWT (JSON Web Token) là cơ chế trao đổi thông tin an toàn dưới dạng JSON, có cấu trúc gọn nhẹ gồm Header – Payload – Signature, hỗ trợ chữ ký số để đảm bảo tính toàn vẹn dữ liệu. JWT đặc biệt phù hợp với cơ chế xác thực không trạng thái (stateless), rất hiệu quả trong RESTful API và kiến trúc microservices.

Khi kết hợp Spring Security với JWT, hệ thống có thể:

Thực hiện xác thực và phân quyền không cần session, giúp dễ mở rộng và tối ưu hiệu suất.

Kiểm tra tính hợp lệ của token thông qua xác minh chữ ký và các thông tin (claims).

Triển khai thuận tiện bằng các thư viện như spring-security-jwt hoặc spring-security-oauth2.

Ưu điểm của mô hình này là tính linh hoạt, khả năng mở rộng cao và phù hợp với nhiều loại ứng dụng hiện đại. [7]

#### **2.4.5. Giới thiệu về Spring Data JPA và Hibernate**

Trong hệ sinh thái Spring Boot, Spring Data JPA và Hibernate là hai công nghệ cốt lõi dùng để xây dựng tầng truy cập dữ liệu (Data Layer), giúp ứng dụng làm việc với cơ sở dữ liệu hiệu quả và gọn nhẹ.

Spring Data JPA cung cấp lớp trừu tượng trên JPA, giúp giảm đáng kể mã nguồn lặp lại. Nhà phát triển chỉ cần định nghĩa Repository Interface, Spring sẽ tự động tạo phần cài đặt. Ngoài ra, Spring Data JPA hỗ trợ tự sinh câu truy vấn dựa trên tên phương thức, giúp việc truy vấn dữ liệu nhanh và thuận tiện.

Hibernate là framework ORM mạnh mẽ, thường được dùng làm implementation của JPA trong Spring Boot. Hibernate chịu trách nhiệm ánh xạ giữa đối tượng Java và bảng trong cơ sở dữ liệu, đồng thời hỗ trợ tự động tạo và cập nhật cấu trúc bảng từ các Entity.

Trong kiến trúc 3-tier, Spring Data JPA và Hibernate nằm ở Data Layer, được Spring IoC Container quản lý dưới dạng Bean. Thông qua Dependency Injection, các Repository được tiêm vào Service, giúp tách biệt rõ ràng giữa logic nghiệp vụ và logic truy cập dữ liệu, từ đó tăng tính linh hoạt và khả năng bảo trì của hệ thống. [6]

## CHƯƠNG 3. THỰC HIỆN HÓA NGHIÊN CỨU

### 3.1. Mô tả bài toán

Hệ thống website bán mỹ phẩm là một ứng dụng giúp các cửa hàng kinh doanh mỹ phẩm có thể quản lý toàn bộ hoạt động liên quan đến việc nhập hàng, bán hàng và theo dõi sản phẩm. Hệ thống hỗ trợ các chức năng như: quản lý sản phẩm (thêm, sửa, xóa), quản lý nhập hàng (nhập các sản phẩm mỹ phẩm từ nhà cung cấp), quản lý đơn hàng (bán sản phẩm cho khách), quản lý khách hàng và quản lý tài khoản người dùng.

Website góp phần giúp việc kinh doanh mỹ phẩm trở nên dễ dàng, nhanh chóng và chính xác hơn thông qua việc tự động hóa quy trình và cung cấp thông tin kịp thời cho người quản lý.

### 3.2. Đặc tả các yêu cầu

#### 3.2.1. Yêu cầu chức năng

##### 3.2.1.1 *Quản lý sản phẩm (Mỹ phẩm)*

- Quản lý thông tin thương hiệu: Quản lý các thương hiệu mỹ phẩm được cửa hàng kinh doanh.
- Xem danh sách sản phẩm: Hiển thị danh sách các sản phẩm mỹ phẩm hiện có trong hệ thống.
- Thêm sản phẩm mới: Thêm mỹ phẩm mới với đầy đủ thông tin như tên, thương hiệu, giá, mô tả, số lượng và hình ảnh.
- Cập nhật thông tin sản phẩm: Chỉnh sửa thông tin của sản phẩm khi có thay đổi.
- Xóa sản phẩm: Xóa các sản phẩm không còn kinh doanh hoặc hết hàng.

##### 3.2.1.2 *Quản lý khách hàng*

- Xem danh sách khách hàng: Hiển thị danh sách các khách hàng đã đăng ký tài khoản trên hệ thống.
- Xem thông tin khách hàng: Xem chi tiết thông tin cá nhân và lịch sử mua hàng của khách hàng.



- Cập nhật thông tin khách hàng: Chỉnh sửa thông tin khách hàng khi cần thiết.
- Xóa khách hàng: Xóa các tài khoản khách hàng không còn sử dụng hoặc vi phạm quy định.

#### **3.2.1.3 Quản lý đơn hàng**

- Xem danh sách đơn hàng: Hiển thị toàn bộ các đơn hàng đã được khách hàng đặt trên hệ thống.
- Xem chi tiết đơn hàng: Xem thông tin chi tiết của từng đơn hàng bao gồm sản phẩm, số lượng, tổng tiền và thông tin khách hàng.
- Cập nhật trạng thái đơn hàng: Thay đổi trạng thái đơn hàng như đang xử lý, đã giao, hoàn thành hoặc hủy.
- Theo dõi tình trạng đơn hàng: Hỗ trợ quản trị viên kiểm soát quá trình xử lý và giao hàng.

#### **3.2.1.4 Quản lý tài khoản**

- Xem danh sách tài khoản: Hiển thị danh sách các tài khoản người dùng trong hệ thống.
- Thêm tài khoản: Tạo mới tài khoản cho quản trị viên hoặc nhân viên.
- Cập nhật tài khoản: Chỉnh sửa thông tin và phân quyền cho tài khoản.
- Xóa tài khoản: Xóa các tài khoản không còn sử dụng.

#### **3.2.1.5 Quản lý thương hiệu**

- Xem danh sách thương hiệu: Hiển thị các thương hiệu mỹ phẩm hiện có trong hệ thống.
- Thêm thương hiệu: Thêm mới thương hiệu mỹ phẩm.
- Cập nhật thương hiệu: Chỉnh sửa thông tin thương hiệu.
- Xóa thương hiệu: Xóa các thương hiệu không còn kinh doanh.

### **3.2.1.6 Quản lý bài viết đánh giá**

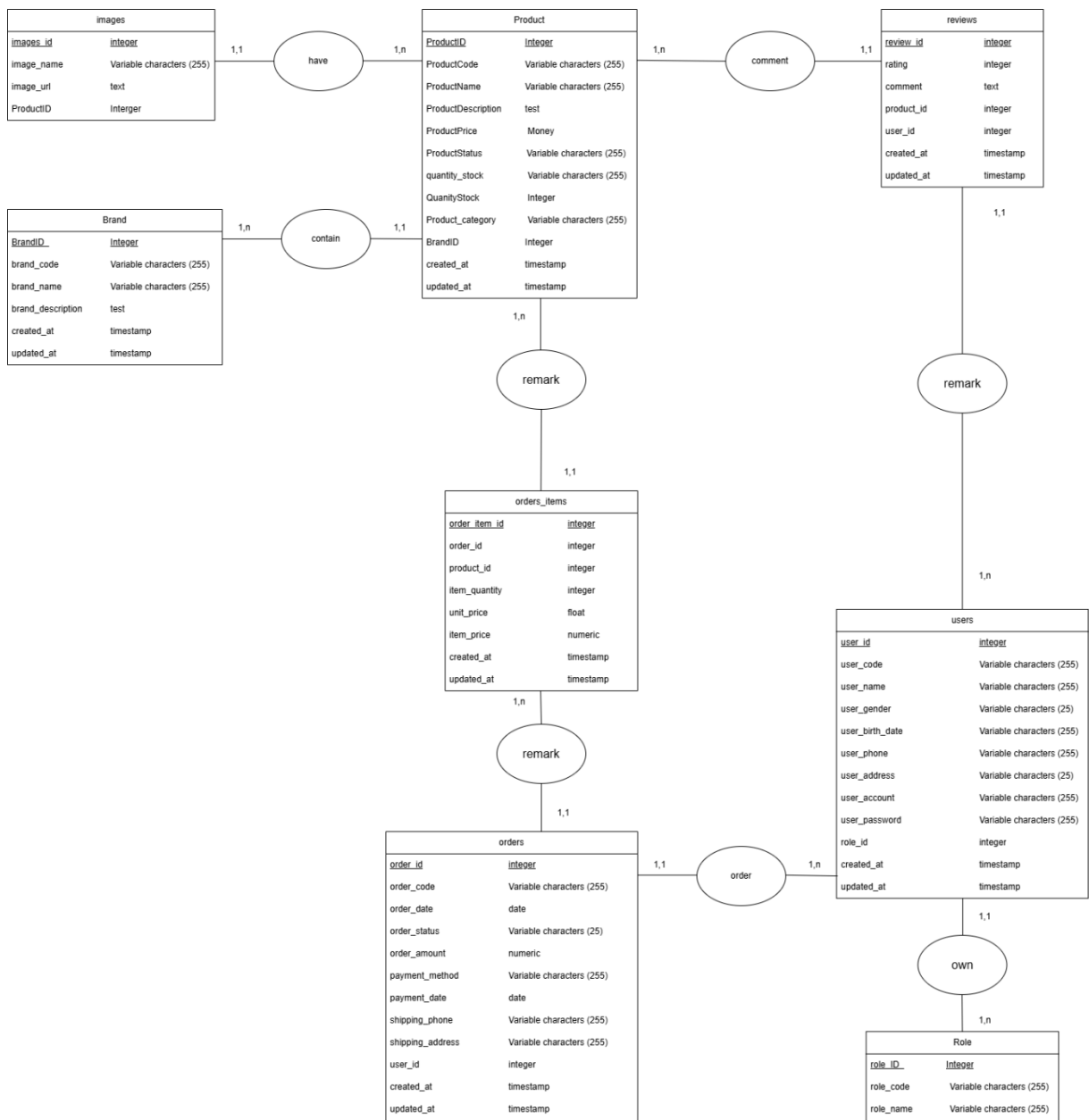
- Xem danh sách bài viết đánh giá: Hiển thị các bài viết đánh giá của khách hàng.
- Duyệt bài viết: Kiểm tra và phê duyệt nội dung bài viết.
- Xóa bài viết đánh giá: Xóa các bài viết không phù hợp với quy định.

### **3.2.2. Yêu cầu phi chức năng**

- Tính bảo mật: Hệ thống phải đảm bảo an toàn thông tin khách hàng, các giao dịch tài chính và dữ liệu sản phẩm.
- Khả năng mở rộng: Hệ thống cần dễ dàng mở rộng về quy mô để thêm nhiều sản phẩm hoặc tính năng trong tương lai.
- Tính khả dụng: Hệ thống cần đảm bảo tính khả dụng cao, có thể xử lý nhiều đơn hàng đồng thời mà không bị gián đoạn.
- Tối ưu hiệu suất: Hệ thống phải xử lý nhanh các yêu cầu người dùng, đặc biệt là trong các tác vụ tìm kiếm sản phẩm và xử lý đơn hàng.
- Giao diện thân thiện: Hệ thống cần có giao diện dễ sử dụng cho người quản lý và nhân viên bán hàng.

### 3.3. Thiết kế dữ liệu

#### 3.3.1. Mô hình dữ liệu



Hình 3.1 Mô hình quan niệm dữ liệu của hệ thống

#### 3.3.2. Danh sách các thực thể và mối kết hợp

Mô tả:

Danh sách các thực thể và mối kết hợp.

*Bảng 3.1 Danh sách các thực thể và mối kết hợp*

<i>STT</i>	<i>Tên thực thể/Mối kết hợp</i>	<i>Diễn giải</i>	<i>Ghi chú</i>
1	users	Thông tin người dùng trong hệ thống	
2	Role	Thông tin vai trò của người dùng	
3	Product	Thông tin về các sản phẩm mỹ phẩm	
4	Brand	Thông tin thương hiệu sản phẩm	
5	images	Hình ảnh của sản phẩm	
6	reviews	Đánh giá sản phẩm của người dùng	
7	orders	Thông tin đơn hàng	
8	orders_items	Chi tiết sản phẩm trong đơn hàng	

**3.3.3. Chi tiết các thực thể**

Thực thể: User

Mô tả: đây là bảng dữ liệu chứa thông tin của người dùng trong hệ thống.

Chi tiết thực thể:

*Bảng 3.2 Chi tiết thực thể User*

<i>Tên thuộc tính</i>	<i>Diễn giải</i>	<i>Loại giá trị</i>	<i>Kiểu dữ liệu</i>	<i>Miền giá trị</i>	<i>Chiều dài kí tự</i>	<i>Ghi chú</i>
user_id	Mã định danh duy nhất của người dùng	BB	Số nguyên	Khóa chính		
User_code	Mã người dùng	BB	Chuỗi			
user_name	Họ và tên người dùng	BB	Chuỗi	RBHT		

user_gender	Giới tính người dùng	BB	Chuỗi	RBSĐT		
user_birth_date	Ngày sinh người dùng	BB	Ngày	RBNS		
user_phone	Số điện thoại người dùng	BB	Chuỗi			
user_address	Địa chỉ người dùng	BB	Chuỗi			
user_account	Tên tài khoản đăng nhập	BB	Chuỗi			
user_password	Mật khẩu đăng nhập	BB	Chuỗi			
role_id	Mã vai trò người dùng	BB	Số nguyên			
created_at	Thời điểm tạo tài khoản	BB	Ngày giờ			
updated_at	Thời điểm cập nhật gần nhất	BB	Ngày giờ			

Thực thể Role

Mô tả: đây là bảng dữ liệu chứa thông tin vai trò của người dùng.

Chi tiết thực thể:

*Bảng 3.3 Chi tiết thực thể Role*

<i>Tên thuộc tính</i>	<i>Diễn giải</i>	<i>Loại giá trị</i>	<i>Kiểu dữ liệu</i>	<i>Miền giá trị</i>	<i>Chiều dài ký tự</i>	<i>Ghi chú</i>
role_id	Mã định danh duy nhất của vai trò	BB	Số nguyên	Khóa chính		
role_code	Mã vai trò	BB	Chuỗi			
role_name	Tên vai trò	BB	Chuỗi	RBHT		

### Thực thể Product

Mô tả: đây là bảng dữ liệu chứa thông tin về các sản phẩm mỹ phẩm.

Chi tiết thực thể:

*Bảng 3.4 Chi tiết thực thể Product*

<i>Tên thuộc tính</i>	<i>Diễn giải</i>	<i>Loại giá trị</i>	<i>Kiểu dữ liệu</i>	<i>Miền giá trị</i>	<i>Chiều dài ký tự</i>	<i>Ghi chú</i>
product_id	Mã định danh duy nhất của sản phẩm	BB	Số nguyên	Khóa chính		
product_code	Mã sản phẩm	BB	Số nguyên	Khóa chính		
product_name	Tên sản phẩm	BB	Chuỗi			
product_description	Mô tả sản phẩm	BB	Chuỗi			
product_price	Giá sản phẩm	BB	Money	RBGTN		
quantity_stock	Số lượng tồn	BB	Số nguyên	RBSL		
product_status	Trạng thái sản phẩm	BB	Chuỗi			
product_category	Danh mục sản phẩm	BB	Chuỗi			
brand_id	Mã thương hiệu	BB	Số nguyên			
created_at	Thời điểm tạo sản phẩm	BB	Ngày giờ			
updated_at	Thời điểm cập nhật gần nhất	BB	Ngày giờ			

### Thực thể Brand

Mô tả: đây là bảng dữ liệu chứa thông tin thương hiệu sản phẩm.

Chi tiết thực thể:

Bảng 3.5 Chi tiết thực thể Brand

Tên thuộc tính	Diễn giải	Loại giá trị	Kiểu dữ liệu	Miền giá trị	Chiều dài ký tự	Ghi chú
brand_id	Mã định danh duy nhất của thương hiệu	BB	Số nguyên	Khóa chính		
brand_code	Mã thương hiệu	BB	Ngày tháng			
brand_name	Tên thương hiệu	BB	Chuỗi			
brand_description	Mô tả chi tiết thương hiệu	BB	Số thực			
created_at	Thời điểm tạo thương hiệu	BB	Ngày giờ			
updated_at	Thời điểm cập nhật gần nhất	BB	Ngày giờ			

Thực thể Images

Mô tả: đây là bảng dữ liệu chứa hình ảnh của sản phẩm.

Chi tiết thực thể:

Bảng 3.6 Chi tiết thực thể Images

Tên thuộc tính	Diễn giải	Loại giá trị	Kiểu dữ liệu	Miền giá trị	Chiều dài ký tự	Ghi chú
image_id	Mã định danh duy nhất của hình ảnh	BB	Số nguyên	Khóa chính		
image_url	Đường dẫn tới hình ảnh	BB	Ngày tháng			
brand_name	Tên thương hiệu	BB	Chuỗi			
image_alt	Văn bản mô tả hình ảnh	BB	Chuỗi			
image_type	Loại hình ảnh	BB	Chuỗi			

product_id	Mã sản phẩm sở hữu hình ảnh	BB	Số thực			
created_at	Thời điểm thêm hình ảnh	BB	Ngày giờ			

### Thực thể Reviews

Mô tả: đây là bảng dữ liệu chứa đánh giá sản phẩm của người dùng.

Chi tiết thực thể:

*Bảng 3.7 Chi tiết thực thể Reviews*

<i>Tên thuộc tính</i>	<i>Diễn giải</i>	<i>Loại giá trị</i>	<i>Kiểu dữ liệu</i>	<i>Miền giá trị</i>	<i>Chiều dài ký tự</i>	<i>Ghi chú</i>
review_id	Mã định danh duy nhất của đánh giá	BB	Số nguyên	Khóa chính		
rating	Số sao đánh giá sản phẩm	BB	Số nguyên	RBSL		
review_content	Nội dung đánh giá	BB	Chuỗi			
review_date	Ngày đánh giá sản phẩm	BB	Ngày giờ			
user_id	Mã người dùng đánh giá	BB	Số nguyên			
product_id	Mã sản phẩm được đánh giá	BB	Số nguyên			

### Thực thể Orders

Mô tả: đây là bảng dữ liệu chứa thông tin đơn hàng.

Chi tiết thực thể:



Bảng 3.8 Chi tiết thực thể Orders

Tên thuộc tính	Diễn giải	Loại giá trị	Kiểu dữ liệu	Miền giá trị	Chiều dài ký tự	Ghi chú
order_id	Mã định danh duy nhất của hình ảnh	BB	Số nguyên	Khóa chính		
order_code	Mã đơn hàng	BB	Ngày tháng			
order_date	Thời điểm đặt hàng	BB	Chuỗi			
order_status	Trạng thái đơn hàng	BB	Chuỗi			
Order_amount	Giá trị đơn hàng	BB	Chuỗi			
payment_method	Phương thức thanh toán	BB	Chuỗi			
payment_status	Trạng thái thanh toán	BB	Chuỗi	RBTTĐH		
shipping_address	Địa chỉ giao hàng	BB	Chuỗi			
user_id	Mã người dùng đặt đơn hàng	BB	Số nguyên			
created_at	Thời điểm tạo đơn hàng	BB	Ngày giờ			

updated_at	Thời điểm cập nhật đơn hàng	BB	Ngày giờ			
------------	-----------------------------	----	----------	--	--	--

Thực thể Orders\_items

Mô tả: đây là bảng dữ liệu chi tiết sản phẩm trong đơn hàng.

Chi tiết thực thể:

*Bảng 3.9 Chi tiết thực thể Orders\_items*

<i>Tên thuộc tính</i>	<i>Diễn giải</i>	<i>Loại giá trị</i>	<i>Kiểu dữ liệu</i>	<i>Miền giá trị</i>	<i>Chiều dài ký tự</i>	<i>Ghi chú</i>
order_item_id	Mã định danh duy nhất của chi tiết đơn hàng	BB	Số nguyên	Khóa chính		
order_id	Mã đơn hàng	BB	Số nguyên			
product_id	Mã sản phẩm trong đơn hàng	BB	Số nguyên			
quantity	Văn bản mô tả hình ảnh	BB	Số nguyên			
unit_price	Loại hình ảnh	BB	Money	RBGTN		
total_price	Mã sản phẩm sở hữu hình ảnh	BB	Money	RBGTN		

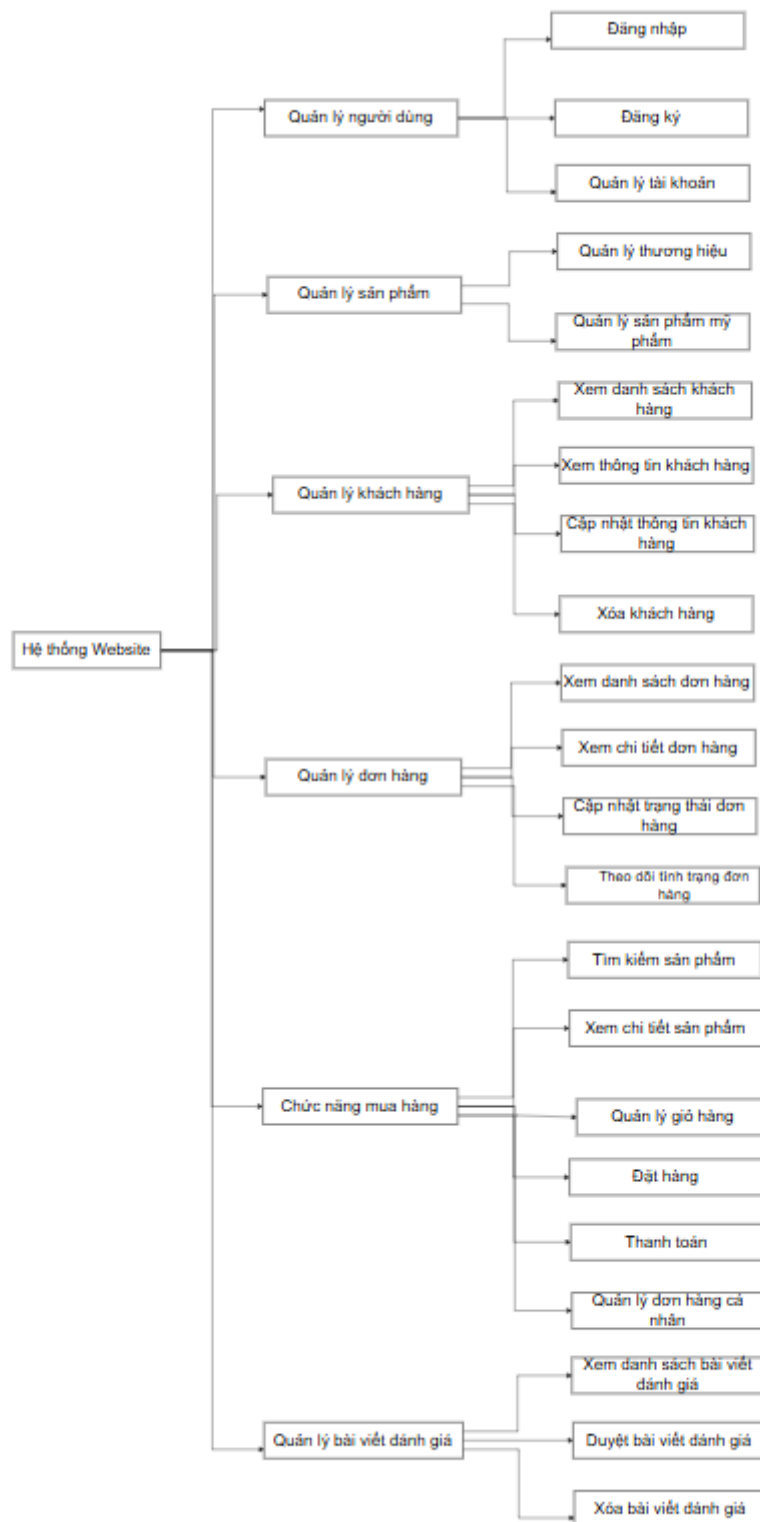
### 3.3.4. Danh sách các ràng buộc toàn vẹn

*Bảng 3.10: Danh sách các ràng buộc toàn vẹn*

<i>Mã ràng buộc</i>	<i>Diễn giải</i>	<i>Ghi chú</i>
RBHT	Họ tên khách hàng chỉ chứa các ký tự chữ cái	
RBGTH	Giới tính là một trong các giá trị: “Nam”, “Nữ”, “Khác”	
RBSĐT	Số điện thoại phải hợp lệ và đảm bảo đúng 10 hoặc 11 chữ số	
RBGTN	Giá tiền phải là số dương	
RBSL	Số lượng không được nhỏ hơn 0	
RBTTĐH	Trạng thái đơn hàng là một trong các giá trị: “Đã đặt”, “Đang xử lý”, “Đã giao”, “Đã hủy”	

### 3.4. Thiết kế xử lý

#### 3.4.1. Sơ đồ phân cấp chức năng



Hình 3.2 Sơ đồ phân cấp chức năng

### Mô tả chi tiết:

**Quản lý người dùng:** Chịu trách nhiệm xác thực và quản lý các tài khoản trong hệ thống. Chức năng này cho phép người dùng đăng nhập, đăng ký và cho phép quản trị viên quản lý tài khoản thông qua các thao tác thêm, cập nhật và xóa tài khoản.

**Quản lý sản phẩm:** Thực hiện quản lý toàn bộ dữ liệu liên quan đến sản phẩm mỹ phẩm. Nhóm chức năng này bao gồm quản lý thương hiệu mỹ phẩm và quản lý thông tin sản phẩm như xem danh sách, thêm mới, cập nhật và xóa sản phẩm, nhằm đảm bảo thông tin sản phẩm luôn chính xác và đầy đủ.

**Quản lý khách hàng:** Hỗ trợ quản trị viên theo dõi và quản lý thông tin khách hàng đã đăng ký hệ thống. Chức năng này cho phép xem danh sách khách hàng, xem chi tiết thông tin, cập nhật hoặc xóa khách hàng khi cần thiết.

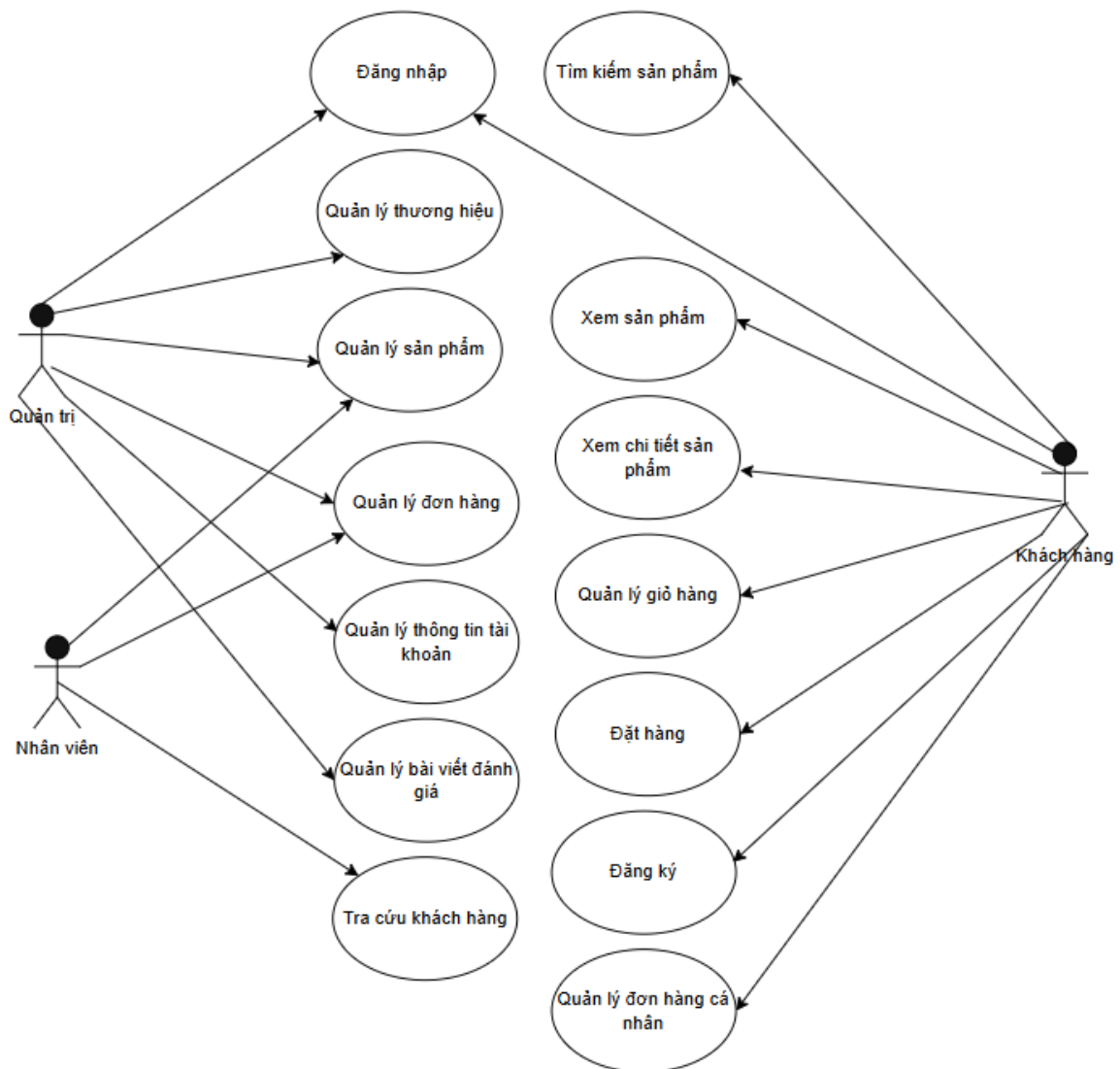
**Quản lý đơn hàng:** Đảm nhiệm việc theo dõi và xử lý các đơn hàng phát sinh trong hệ thống. Bao gồm các chức năng xem danh sách đơn hàng, xem chi tiết đơn hàng, cập nhật trạng thái và theo dõi quá trình xử lý đơn hàng.

**Chức năng mua hàng:** Hỗ trợ khách hàng trong quá trình mua sắm trực tuyến. Nhóm chức năng này bao gồm tìm kiếm sản phẩm, xem danh sách và chi tiết sản phẩm, quản lý giỏ hàng, đặt hàng, thanh toán và quản lý các đơn hàng cá nhân.

**Quản lý bài viết đánh giá:** Cho phép quản trị viên quản lý các bài viết đánh giá của khách hàng về sản phẩm. Chức năng này bao gồm xem danh sách bài viết, duyệt nội dung và xóa các bài viết không phù hợp với quy định của hệ thống.

### 3.5. Sơ đồ Use Case

#### 3.5.1. Use Case tổng quát



Hình 3.3 Mô hình use case tổng quát

Mô tả:

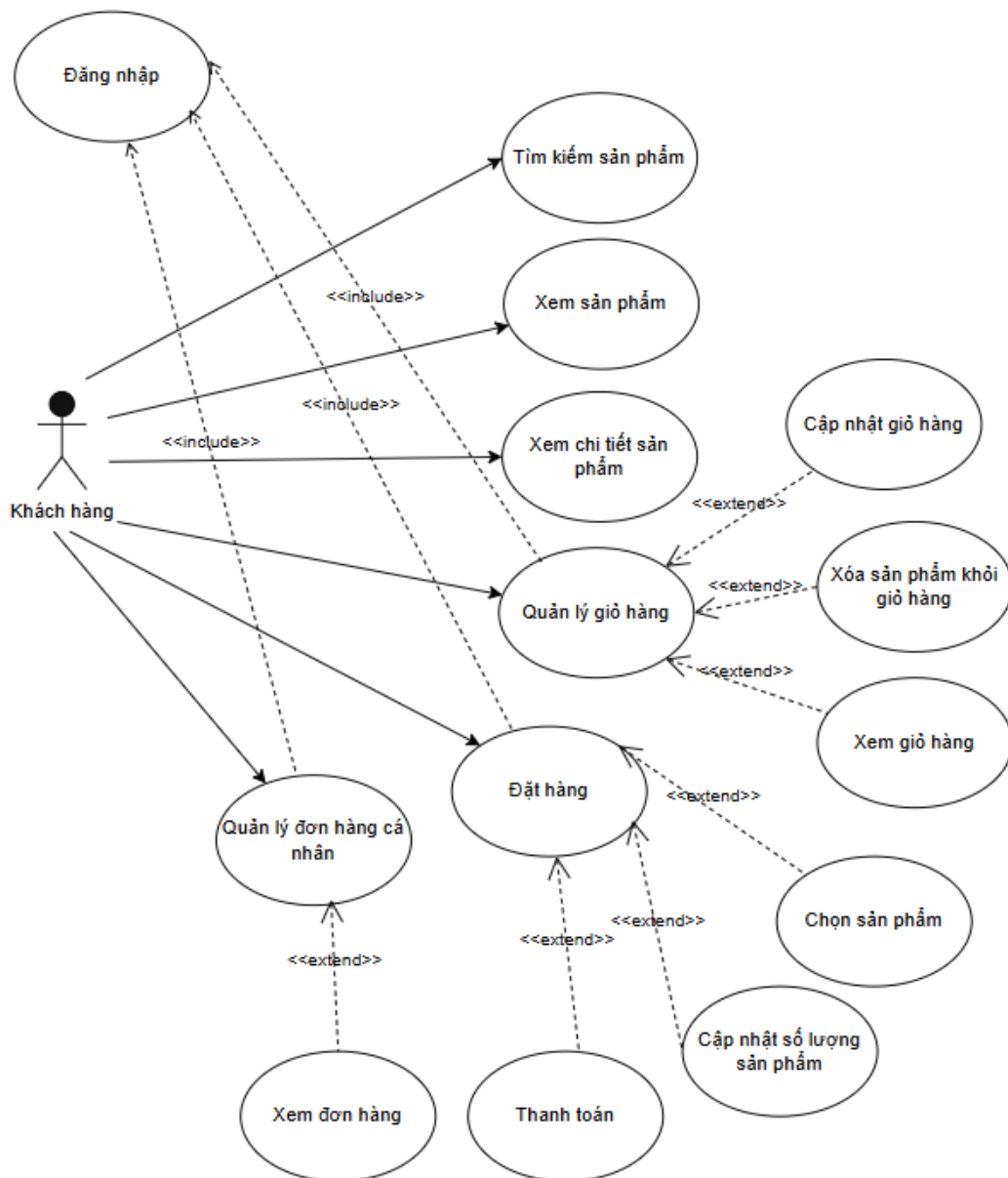
Hệ thống bao gồm hai tác nhân chính: Quản trị, Nhân viên và Khách hàng.

Quản trị chịu trách nhiệm quản lý toàn bộ dữ liệu hệ thống như sản phẩm, thương hiệu, đơn hàng, quản lý bài viết đánh giá và tài khoản.

Nhân viên có các chức năng như quản lý sản phẩm, quản lý đơn hàng và tra cứu khách hàng.

Khách hàng có thể đăng ký và đăng nhập tài khoản, tìm kiếm, xem, đặt mua sản phẩm và quản lý giỏ hàng của mình.

### 3.5.2. Use case tác nhân khách hàng



Hình 3.4 Mô hình use case tác nhân khách hàng

*Mô tả chi tiết:*

*Bảng 3.11 Mô tả chi tiết use case tác nhân khách hàng*

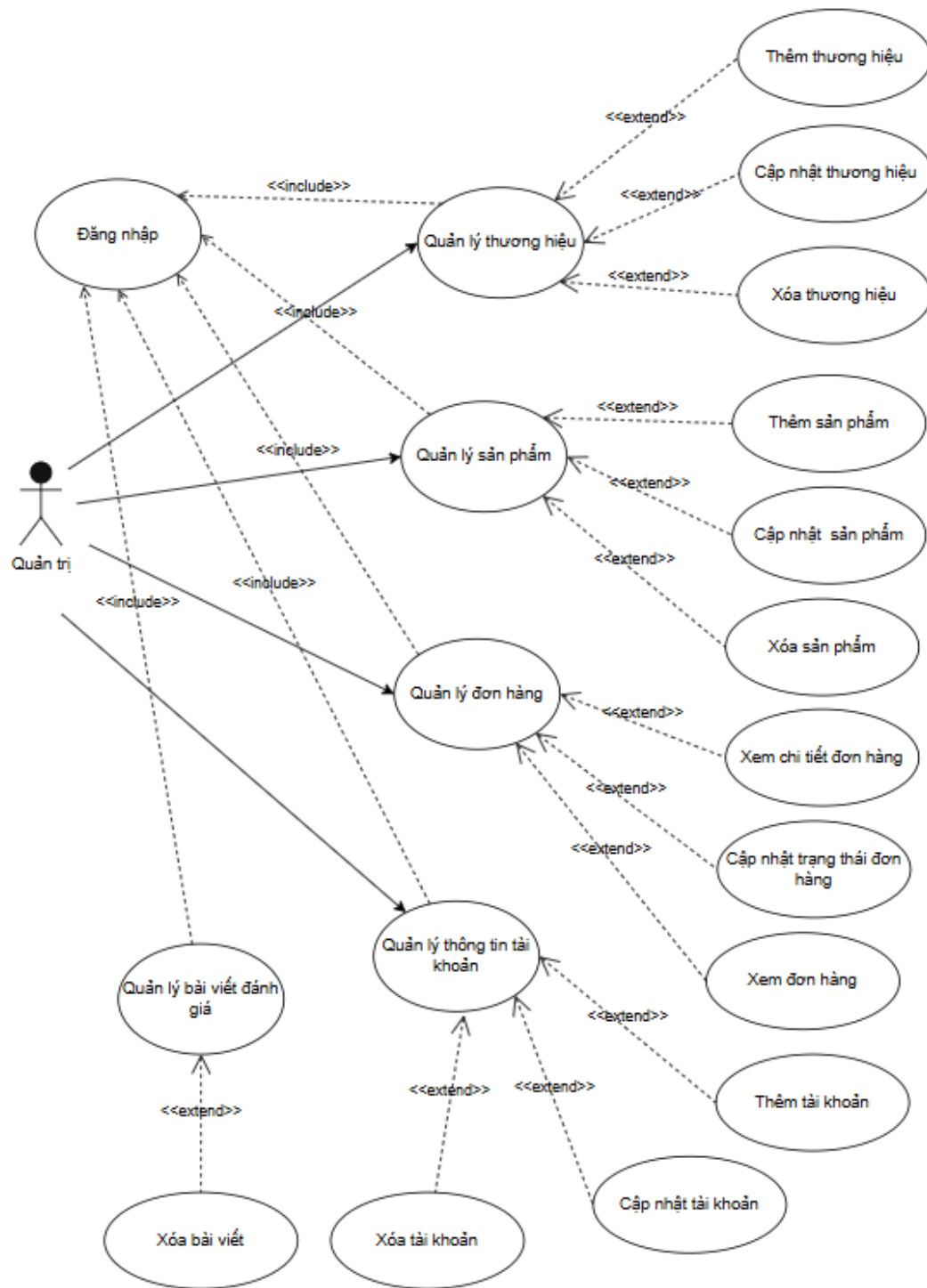
<i>Tên use case</i>	<i>Chức năng</i>	<i>Dữ liệu đầu vào</i>	<i>Kết quả đầu ra</i>	<i>Mô tả</i>
Đăng nhập	Xác thực tài khoản khách hàng	Tên đăng nhập, mật khẩu	Đăng nhập thành công / thất bại	Khách hàng đăng nhập để sử dụng các chức năng mua hàng
Tìm kiếm sản phẩm	Tìm sản phẩm theo từ khóa	Tên sản phẩm, từ khóa	Danh sách sản phẩm phù hợp	Giúp khách hàng nhanh chóng tìm được sản phẩm mong muốn
Xem sản phẩm	Xem danh sách sản phẩm	Xem sản phẩm	Danh sách sản phẩm	Hiển thị các sản phẩm đang kinh doanh
Xem chi tiết sản phẩm	Xem thông tin chi tiết	Mã sản phẩm	Thông tin chi tiết sản phẩm	Hiển thị giá, mô tả, hình ảnh, tồn kho
Quản lý giỏ hàng	Quản lý sản phẩm trong giỏ	Thông tin sản phẩm	Giỏ hàng được cập nhật	Chức năng tổng để thao tác với giỏ hàng
<i>Tên use case</i>	<i>Chức năng</i>	<i>Dữ liệu đầu vào</i>	<i>Kết quả đầu ra</i>	<i>Mô tả</i>



Xem giỏ hàng	Xem nội dung giỏ hàng	Giỏ hàng hiện tại	Danh sách sản phẩm trong giỏ	Khách hàng kiểm tra các sản phẩm đã chọn
Cập nhật giỏ hàng	Điều chỉnh thông tin giỏ	Số lượng sản phẩm	Giỏ hàng mới	Thay đổi số lượng sản phẩm trong giỏ
Xóa sản phẩm khỏi giỏ hàng	Loại bỏ sản phẩm	Mã sản phẩm	Giỏ hàng sau khi xóa	Xóa sản phẩm không còn nhu cầu mua
Đặt hàng	Tạo đơn hàng mới	Thông tin giỏ hàng	Đơn hàng được tạo	Khách hàng xác nhận mua các sản phẩm
Chọn sản phẩm	Chọn sản phẩm để mua	Mã sản phẩm	Sản phẩm được thêm vào đơn	Thêm sản phẩm vào giỏ/đơn hàng
Cập nhật số lượng sản phẩm	Thay đổi số lượng mua	Số lượng mới	Số lượng được cập nhật	Điều chỉnh số lượng trước khi thanh toán
Thanh toán	Thanh toán đơn hàng	Phương thức thanh toán	Thanh toán thành công / thất bại	Khách hàng hoàn tất quá trình mua hàng
<i>Tên use case</i>	<i>Chức năng</i>	<i>Dữ liệu đầu vào</i>	<i>Kết quả đầu ra</i>	<i>Mô tả</i>

Quản lý đơn hàng cá nhân	Theo dõi các đơn đã đặt	Tài khoản khách hàng	Danh sách đơn hàng	Quản lý lịch sử mua hàng
Xem đơn hàng	Xem chi tiết đơn hàng	Mã đơn hàng	Thông tin chi tiết đơn	Xem trạng thái, sản phẩm, tổng tiền

### 3.5.3. Use case tác nhân quản trị



Hình 3.5 Mô hình use case tác nhân quản trị

Mô tả chi tiết:

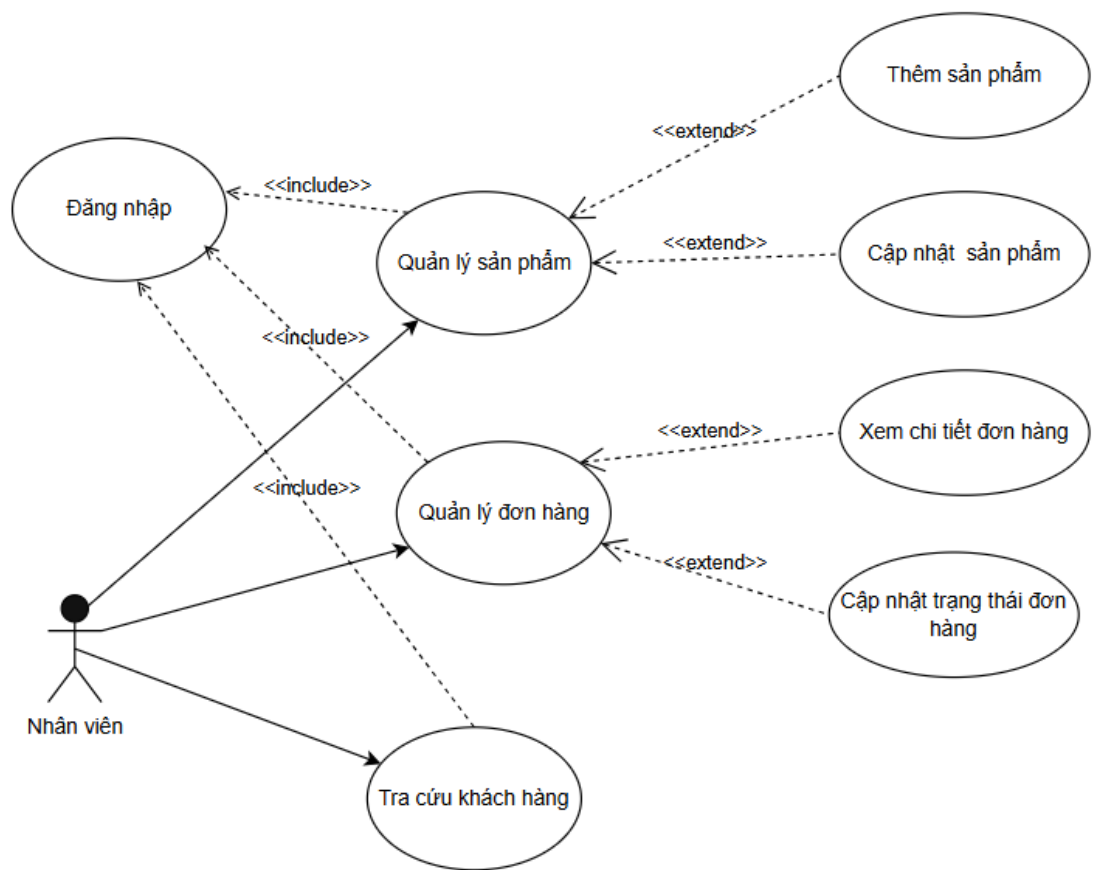
*Bảng 3.12 Mô tả chi tiết use case tác nhân quản trị*

<i>Tên use case</i>	<i>Chức năng</i>	<i>Dữ liệu đầu vào</i>	<i>Kết quả đầu ra</i>	<i>Mô tả</i>
Đăng nhập	Xác thực quản trị viên	Tên đăng nhập, mật khẩu	Đăng nhập thành công / thất bại	Quản trị đăng nhập để sử dụng hệ thống
Quản lý thương hiệu	Quản lý thông tin thương hiệu	Thông tin thương hiệu	Danh sách thương hiệu cập nhật	Chức năng tổng để thao tác với thương hiệu
Thêm thương hiệu	Thêm thương hiệu mới	Tên thương hiệu, mô tả	Thương hiệu mới	Thêm thương hiệu vào hệ thống
Cập nhật thương hiệu	Chỉnh sửa thương hiệu	ID thương hiệu, thông tin mới	Thông tin thương hiệu cập nhật	Cập nhật thông tin thương hiệu
Xóa thương hiệu	Xóa thương hiệu	ID thương hiệu	Thương hiệu bị xóa	Loại bỏ thương hiệu không còn sử dụng
Quản lý sản phẩm	Quản lý sản phẩm	Thông tin sản phẩm	Danh sách sản phẩm cập nhật	Chức năng tổng để quản lý sản phẩm

<i>Tên use case</i>	<i>Chức năng</i>	<i>Dữ liệu đầu vào</i>	<i>Kết quả đầu ra</i>	<i>Mô tả</i>
Thêm sản phẩm	Thêm sản phẩm mới	Tên, giá, mô tả, tồn kho	Sản phẩm mới	Thêm sản phẩm vào hệ thống
Cập nhật sản phẩm	Chỉnh sửa sản phẩm	ID sản phẩm, thông tin mới	Sản phẩm cập nhật	Cập nhật thông tin sản phẩm
Xóa sản phẩm	Xóa sản phẩm	ID Sản phẩm	Sản phẩm bị xóa	Loại bỏ sản phẩm khỏi hệ thống
Quản lý đơn hàng	Quản lý đơn đặt hàng	Thông tin đơn hàng	Danh sách đơn hàng	Theo dõi và xử lý đơn hàng
Xem đơn hàng	Xem danh sách đơn	Xem danh sách đơn	Danh sách đơn hàng	Hiển thị toàn bộ đơn hàng
Xem chi tiết đơn hàng	Xem chi tiết đơn	Mã đơn hàng	Thông tin chi tiết đơn	Xem sản phẩm, khách hàng, tổng tiền
Cập nhật trạng thái đơn hàng	Thay đổi trạng thái	Mã đơn, trạng thái mới	Trạng thái cập nhật	Cập nhật xử lý, giao hàng, hoàn tất
Quản lý tài khoản	Quản lý thông tin tài khoản	Thông tin tài khoản	Danh sách tài khoản	Quản lý người dùng hệ thống

<i>Tên use case</i>	<i>Chức năng</i>	<i>Dữ liệu đầu vào</i>	<i>Kết quả đầu ra</i>	<i>Mô tả</i>
Thêm tài khoản	Tạo tài khoản mới	Tên đăng nhập, vai trò	Tài khoản mới	Thêm tài khoản quản trị / nhân viên
Cập nhật tài khoản	Chỉnh sửa tài khoản	ID tài khoản, thông tin mới	Tài khoản cập nhật	Cập nhật thông tin tài khoản
Xóa tài khoản	Xóa tài khoản	ID tài khoản	Tài khoản bị xóa	Loại bỏ tài khoản không còn sử dụng
Quản lý bài viết đánh giá	Quản lý nội dung đánh giá	Nội dung bài viết	Danh sách bài viết	Quản lý các bài viết đánh giá
Xóa bài viết	Xóa bài đánh giá	ID bài viết	Bài viết bị xóa	Xóa bài viết không phù hợp

3.5.4. Use case tác nhân nhân viên



Hình 3.6 Mô hình use case tác nhân nhân viên

Mô tả chi tiết:

Bảng 3.13 Mô tả chi tiết use case tác nhân nhân viên

Tên use case	Chức năng	Dữ liệu đầu vào	Kết quả đầu ra	Mô tả
Đăng nhập	Xác thực nhân viên	Tên đăng nhập, mật khẩu	Đăng nhập thành công / thất bại	Nhân viên đăng nhập vào hệ thống để sử dụng các chức năng quản lý

<i>Tên use case</i>	<i>Chức năng</i>	<i>Dữ liệu đầu vào</i>	<i>Kết quả đầu ra</i>	<i>Mô tả</i>
Quản lý sản phẩm	Quản lý thông tin sản phẩm	Thông tin sản phẩm	Danh sách sản phẩm được cập nhật	Chức năng tổng cho phép thêm và cập nhật sản phẩm
Thêm sản phẩm	Thêm mới sản phẩm	Tên sản phẩm, giá, mô tả, số lượng,...	Sản phẩm mới được thêm	Nhân viên thêm sản phẩm mới vào hệ thống
Cập nhật sản phẩm	Chỉnh sửa sản phẩm	ID sản phẩm, thông tin cần sửa	Thông tin sản phẩm được cập nhật	Nhân viên cập nhật thông tin sản phẩm đã có
Quản lý đơn hàng	Quản lý các đơn đặt hàng	Thông tin đơn hàng	Danh sách đơn hàng	Chức năng tổng để theo dõi và xử lý đơn hàng
Xem chi tiết đơn hàng	Xem thông tin chi tiết	Mã đơn hàng	Thông tin chi tiết đơn hàng	Nhân viên xem chi tiết từng đơn hàng
Cập nhật trạng thái đơn hàng	Thay đổi trạng thái đơn	Mã đơn hàng, trạng thái mới	Trạng thái đơn hàng được cập nhật	Nhân viên cập nhật trạng thái (đang xử lý, hoàn thành, hủy...)



<i>Tên use case</i>	<i>Chức năng</i>	<i>Dữ liệu đầu vào</i>	<i>Kết quả đầu ra</i>	<i>Mô tả</i>
Tra cứu khách hàng	Tìm kiếm thông tin khách hàng	Tên / SĐT / mã khách hàng	Thông tin khách hàng	Nhân viên tra cứu thông tin khách hàng trong hệ thống

### 3.6. Thiết kế API Endpoint

#### 3.6.1. Các quy ước chung

- Đường dẫn cơ sở: /api/v1
- Xác thực: sử dụng JSON Web Token (JWT) dành cho những chức năng cần xác thực người dùng, nằm trong phần header của giao thức HTTP gửi đi.
- Các phương thức HTTP được sử dụng:

*Bảng 3.14 Các phương thức HTTP được sử dụng trong hệ thống website*

<i>Phương thức</i>	<i>Mô tả</i>
GET	Truy xuất dữ liệu
POST	Tạo mới dữ liệu
PUT	Cập nhật toàn bộ đối tượng
PATCH	Cập nhật một phần đối tượng ( ví dụ như trạng thái, mật khẩu...)
DELETE	Xóa dữ liệu (thường là xóa mềm - soft delete)

- Các mã và thông báo phản hồi quy định chung:

*Bảng 3.15 Các mã và thông báo phản hồi quy định chung*

<i>Mã phản hồi</i>	<i>Thông báo</i>	<i>Ý nghĩa</i>
200	OK	Yêu cầu thành công và có dữ liệu trả về. thường dùng cho các phương thức GET hoặc PUT/PATCH
201	Created	Tạo mới tài nguyên thành công, thường dùng cho phương thức POST
204	No Content	Xử lý thành công nhưng không có nội dung trả về (thường dùng cho DELETE)
400	Bad Request	Yêu cầu không hợp lệ về mặt cú pháp hoặc tham số cơ bản (không phải lỗi xác thực biểu mẫu)
401	Unauthorized	Người dùng chưa đăng nhập hoặc Token hết hạn/không hợp lệ
403	Forbidden	Đã đăng nhập nhưng không có quyền thực hiện hành động này
404	Not Found	Tài nguyên muốn tìm không tồn tại trong hệ thống
413	Payload Too Large	Dữ liệu gửi lên quá lớn so với quy định server
415	Unsupported Media Type	Định dạng file không được hỗ trợ
422	Unprocessable Entity	Lỗi Validate dữ liệu (Rất quan trọng). Dữ liệu đúng cú pháp nhưng sai logic nghiệp vụ
429	Too Many Requests	Gửi quá nhiều yêu cầu trong một khoảng thời gian (Rate Limiting)
500	Internal Server Error	Lỗi không xác định từ phía Server (Code bị crash, lỗi kết nối Database)

<i>Mã phản hồi</i>	<i>Thông báo</i>	<i>Ý nghĩa</i>
503	Service Unavailable	Server đang bảo trì hoặc quá tải

- Cấu trúc phản hồi chung:

```
{
    "status": 200,
    "message": "Success",
    "Data": {...}
}
```

Trong đó:

- + Trường "status" chứa mã phản hồi.
- + Trường "message" chứa thông báo phản hồi.
- + Trường "data" chứa dữ liệu phản hồi.

### 3.6.2. Các endpoint xác thực và ủy quyền

*Bảng 3.16 Danh sách các endpoint xác thực và ủy quyền*

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Đăng nhập hệ thống	POST	/auth/login	Công khai	Xác thực email/mật khẩu và trả về JWT.
Đăng ký người dùng	POST	/auth/register	Công khai	Tạo tài khoản người dùng mới.
Quên mật khẩu	POST	/auth/forgot-password	Xác thực	Yêu cầu số điện thoại để xác thực.
Đổi mật khẩu	PATCH	/auth/change-password	Xác thực	Yêu cầu mật khẩu cũ để xác thực.

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Đăng xuất tất cả thiết bị	POST	/auth/logout-all	Xác thực	Vô hiệu hóa các token cũ.

### 3.6.3. Các endpoint liên quan đến quản lý người dùng

Bảng 3.17 Danh sách các endpoint liên quan đến quản lý người dùng

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Lấy thông tin người dùng theo ID	GET	/api/v1/users/{userId}	Yêu cầu xác thực	Lấy thông tin chi tiết của người dùng theo ID
Cập nhật người dùng	PUT	/api/v1/users/{userId}	Yêu cầu xác thực	Cập nhật thông tin người dùng
Xóa người dùng	DELETE	/api/v1/users/{userId}	Yêu cầu xác thực	Xóa người dùng theo ID
Lấy danh sách người dùng (phân trang)	GET	/api/v1/users	Yêu cầu xác thực	Lấy danh sách người dùng có phân trang
Tạo người dùng mới	POST	/api/v1/users	Yêu cầu xác thực	Tạo mới một người dùng
Kiểm tra số điện thoại tồn tại	GET	/api/v1/users/check-phone/{userPhone}	Yêu cầu xác thực	Kiểm tra số điện thoại đã tồn tại hay chưa

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Lấy thông tin người dùng theo tài khoản	GET	/api/v1/users/account/{userAccount}	Yêu cầu xác thực	Lấy thông tin người dùng theo account
Kiểm tra tài khoản tồn tại	GET	/api/v1/users/check-account/{userAccount}	Yêu cầu xác thực	Kiểm tra tài khoản người dùng đã tồn tại hay chưa

### 3.6.4. Các endpoint liên quan đến quản lý đơn hàng

Bảng 3.18 Danh sách các endpoint liên quan đến quản lý đơn hàng

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Cập nhật trạng thái đơn hàng	PUT	/api/v1/orders/{orderId}/status	Yêu cầu xác thực	Cập nhật trạng thái của đơn hàng
Cập nhật thông tin thanh toán	PUT	/api/v1/orders/{orderId}/payment	Yêu cầu xác thực	Cập nhật thông tin thanh toán của đơn hàng
Cập nhật trạng thái thanh toán	PUT	/api/v1/orders/{orderId}/payment-status	Yêu cầu xác thực	Cập nhật trạng thái thanh toán (đã thanh toán / chưa thanh toán)
Hủy đơn hàng	PUT	/api/v1/orders/{orderId}/cancel	Yêu cầu xác thực	Hủy đơn hàng
Lấy danh sách đơn	GET	/api/v1/orders	Yêu cầu xác thực	Lấy danh sách đơn hàng có phân trang

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
hàng (phân trang)				
Tạo đơn hàng mới	POST	/api/v1/orders	Yêu cầu xác thực	Tạo mới một đơn hàng
Lấy đơn hàng theo ID	GET	/api/v1/orders/{orderId}	Yêu cầu xác thực	Lấy chi tiết đơn hàng theo ID
Xóa đơn hàng	DELETE	/api/v1/orders/{orderId}	Yêu cầu xác thực	Xóa đơn hàng
Lấy đơn hàng theo người dùng	GET	/api/v1/orders/user/{userId}	Yêu cầu xác thực	Lấy danh sách đơn hàng theo user
Lấy đơn hàng theo trạng thái	GET	/api/v1/orders/status/{status}	Yêu cầu xác thực	Lấy danh sách đơn hàng theo trạng thái
Lấy đơn hàng theo mã	GET	/api/v1/orders/code/{orderId}	Yêu cầu xác thực	Lấy thông tin đơn hàng theo mã đơn
Tính tổng tiền đơn hàng	GET	/api/v1/orders/calculate-amount	Yêu cầu xác thực	Tính tổng tiền của đơn hàng

### 3.6.5. Các endpoint liên quan đến quản lý vai trò

Bảng 3.19 Danh sách các endpoint liên quan đến quản lý vai trò

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Lấy vai trò theo ID	GET	/api/v1/roles/{roleId}	Yêu cầu xác thực	Lấy thông tin chi tiết role theo ID
Cập nhật vai trò	PUT	/api/v1/roles/{roleId}	Yêu cầu xác thực	Cập nhật thông tin role
Xóa vai trò	DELETE	/api/v1/roles/{roleId}	Yêu cầu xác thực	Xóa role khỏi hệ thống
Lấy danh sách vai trò	GET	/api/v1/roles	Yêu cầu xác thực	Lấy toàn bộ danh sách role
Tạo vai trò mới	POST	/api/v1/roles	Yêu cầu xác thực	Tạo mới một role
Lấy vai trò theo mã	GET	/api/v1/roles/code/{roleCode}	Yêu cầu xác thực	Lấy thông tin role theo mã role

### 3.6.6. Các endpoint liên quan đến quản lý đánh giá

Bảng 3.20 Danh sách các endpoint liên quan đến quản lý đánh giá

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Cập nhật đánh giá	PUT	/api/v1/reviews/{reviewId}	Yêu cầu xác thực	Cập nhật nội dung đánh giá
Xóa đánh giá	DELETE	/api/v1/reviews/{reviewId}	Yêu cầu xác thực	Xóa đánh giá

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Lấy tất cả đánh giá	GET	/api/v1/reviews	Yêu cầu xác thực	Lấy danh sách tất cả đánh giá
Tạo đánh giá mới	POST	/api/v1/reviews	Yêu cầu xác thực	Tạo đánh giá sản phẩm
Lấy đánh giá theo sản phẩm	GET	/api/v1/reviews/product/{productId}	Yêu cầu xác thực	Lấy danh sách đánh giá theo sản phẩm
Lấy điểm đánh giá trung bình	GET	/api/v1/reviews/product/{productId}/rating	Yêu cầu xác thực	Lấy điểm đánh giá trung bình của sản phẩm
Lấy đánh giá của người dùng	GET	/api/v1/reviews/my-reviews	Yêu cầu xác thực	Lấy danh sách đánh giá của người dùng hiện tại

### 3.6.7. Các endpoint liên quan đến quản lý sản phẩm

*Bảng 3.21 Danh sách các endpoint liên quan đến quản lý sản phẩm*

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Xem sản phẩm theo ID	GET	/api/v1/products/{productId}	Công khai	Lấy thông tin chi tiết sản phẩm theo ID
Cập nhật sản phẩm	PUT	/api/v1/products/{productId}	Yêu cầu xác thực	Cập nhật thông tin sản phẩm
Xóa sản phẩm	DELETE	/api/v1/products/{productId}	Yêu cầu xác thực	Xóa sản phẩm khỏi hệ thống



<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Danh sách sản phẩm	GET	/api/v1/products	Công khai	Lấy danh sách sản phẩm, hỗ trợ phân trang và lọc
Tạo sản phẩm mới	POST	/api/v1/products	Yêu cầu xác thực	Tạo mới một sản phẩm
Sản phẩm bán chạy	GET	/api/v1/products/best-sellers	Công khai	Lấy danh sách sản phẩm bán chạy

### 3.6.8. Các endpoint liên quan đến bảng điều khiển

*Bảng 3.22 Danh sách các endpoint liên quan đến bảng điều khiển*

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Top sản phẩm bán chạy	GET	/api/v1/dashboard/top-products	Yêu cầu xác thực	Lấy danh sách sản phẩm bán chạy nhất
Thống kê dashboard	GET	/api/v1/dashboard/stats	Yêu cầu xác thực	Lấy dữ liệu thống kê tổng quan
Đơn hàng gần đây	GET	/api/v1/dashboard/recent-orders	Yêu cầu xác thực	Lấy danh sách các đơn hàng gần nhất

### 3.6.9. Các endpoint liên quan đến quản lý thương hiệu

Bảng 3.23 Danh sách các endpoint liên quan đến quản lý thương hiệu

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Xem brand theo ID	GET	/api/v1/brands/{brandId}	Công khai	Lấy thông tin brand theo ID
Cập nhật brand	PUT	/api/v1/brands/{brandId}	Yêu cầu xác thực	Cập nhật thông tin brand
Xóa brand	DELETE	/api/v1/brands/{brandId}	Yêu cầu xác thực	Xóa brand
Danh sách brand	GET	/api/v1/brands	Công khai	Lấy danh sách tất cả brand
Tạo brand mới	POST	/api/v1/brands	Yêu cầu xác thực	Tạo mới một brand

### 3.6.10. Các endpoint liên quan đến quản lý hình ảnh

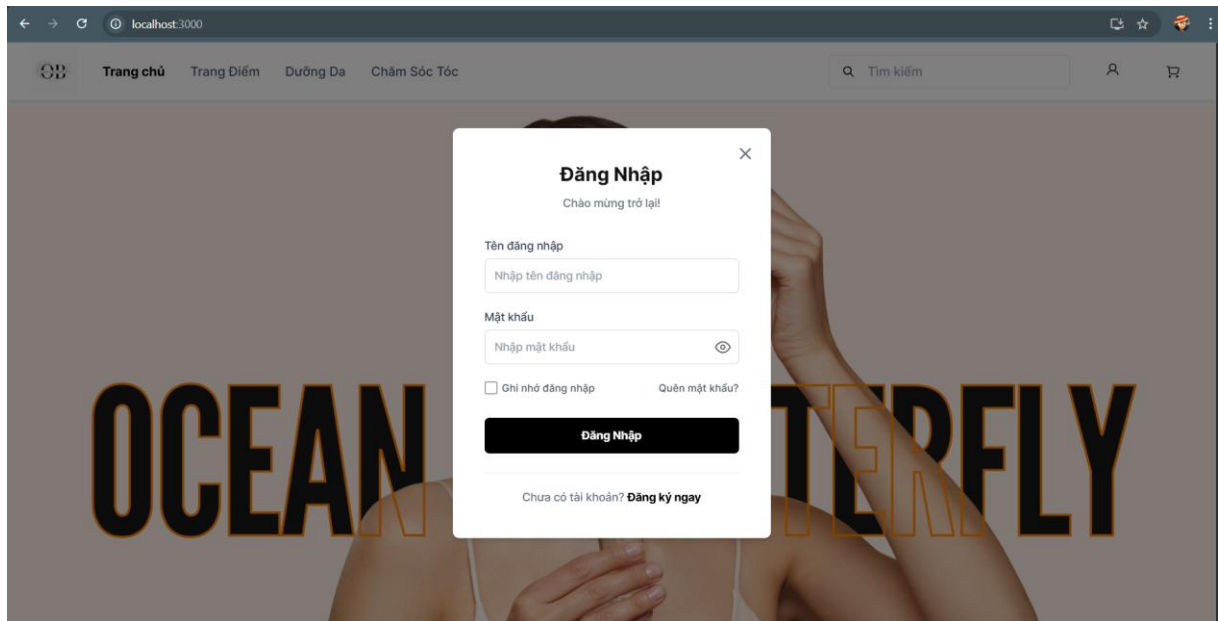
Bảng 3.24 Danh sách các endpoint liên quan đến quản lý hình ảnh

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Lấy hình ảnh theo ID	GET	/api/v1/images/{imageId}	ADMIN / STAFF	Lấy thông tin chi tiết hình ảnh theo ID
Cập nhật hình ảnh	PUT	/api/v1/images/{imageId}	ADMIN / STAFF	Cập nhật thông tin hình ảnh
Xóa hình ảnh	DELETE	/api/v1/images/{imageId}	ADMIN / STAFF	Xóa một hình ảnh khỏi hệ thống

<i>Chức năng</i>	<i>Phương thức</i>	<i>Endpoint</i>	<i>Quyền hạn</i>	<i>Mô tả</i>
Lấy tất cả hình ảnh	GET	/api/v1/images	ADMIN / STAFF	Lấy danh sách tất cả hình ảnh
Tải lên hình ảnh cho sản phẩm	POST	/api/v1/images	ADMIN / STAFF	Upload file hình ảnh gắn với sản phẩm
Tạo hình ảnh bằng JSON (phiên bản cũ)	POST	/api/v1/images/json	ADMIN / STAFF	Tạo hình ảnh mới bằng dữ liệu JSON (legacy)
Lấy hình ảnh theo ID sản phẩm	GET	/api/v1/images/product/{productId}	ADMIN / STAFF	Lấy danh sách hình ảnh theo ID sản phẩm
Xóa tất cả hình ảnh theo ID sản phẩm	DELETE	/api/v1/images/product/{productId}	ADMIN / STAFF	Xóa toàn bộ hình ảnh của một sản phẩm

## CHƯƠNG 4. KẾT QUẢ NGHIÊN CỨU

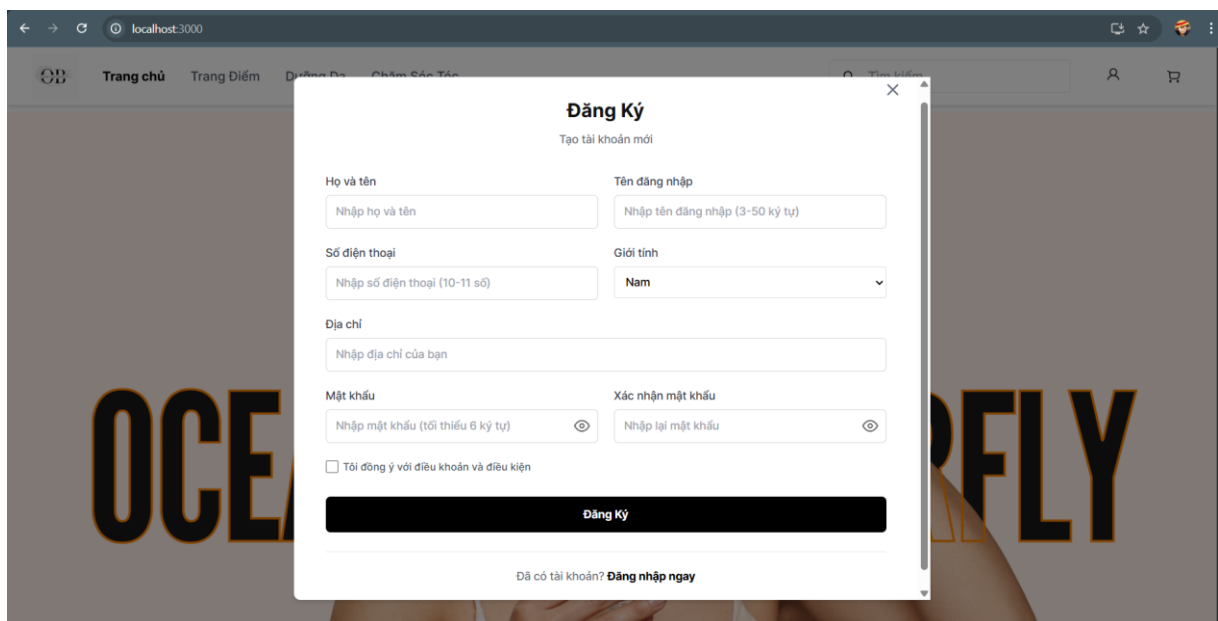
### 4.1. Giao diện đăng nhập và đăng ký



Hình 4.1 Giao diện đăng nhập

Mô tả:

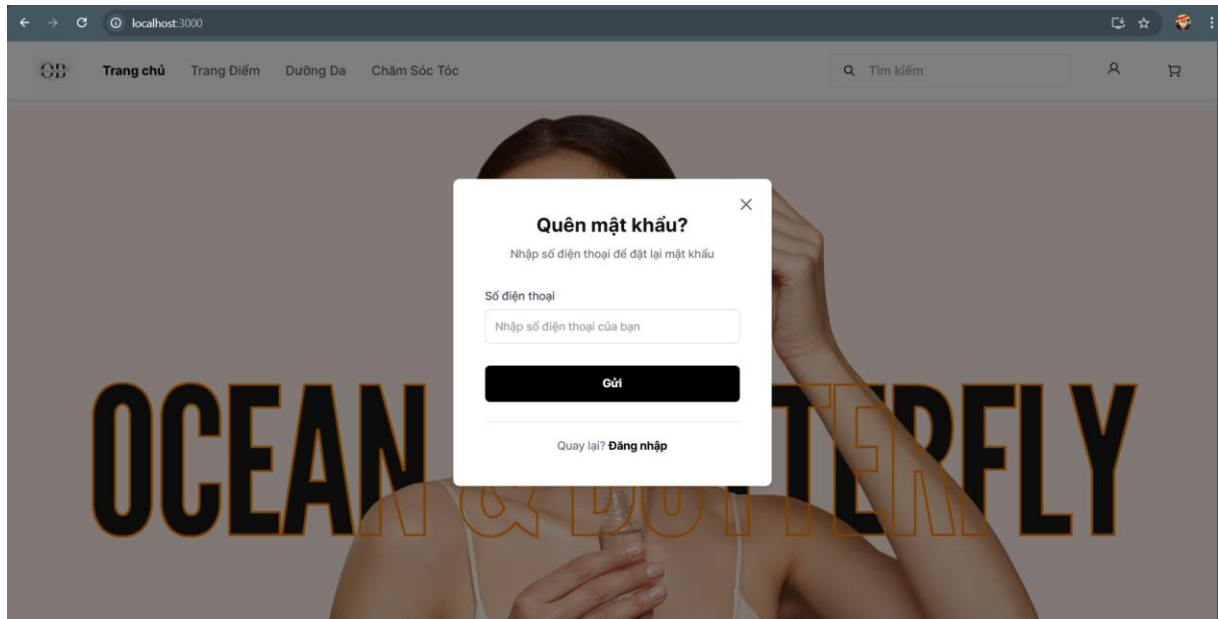
Giao diện đăng nhập bao gồm các trường như: Tên đăng nhập, mật khẩu và nút đăng nhập để gửi yêu cầu cho hệ thống.



Hình 4.2 Giao diện đăng ký

Mô tả:

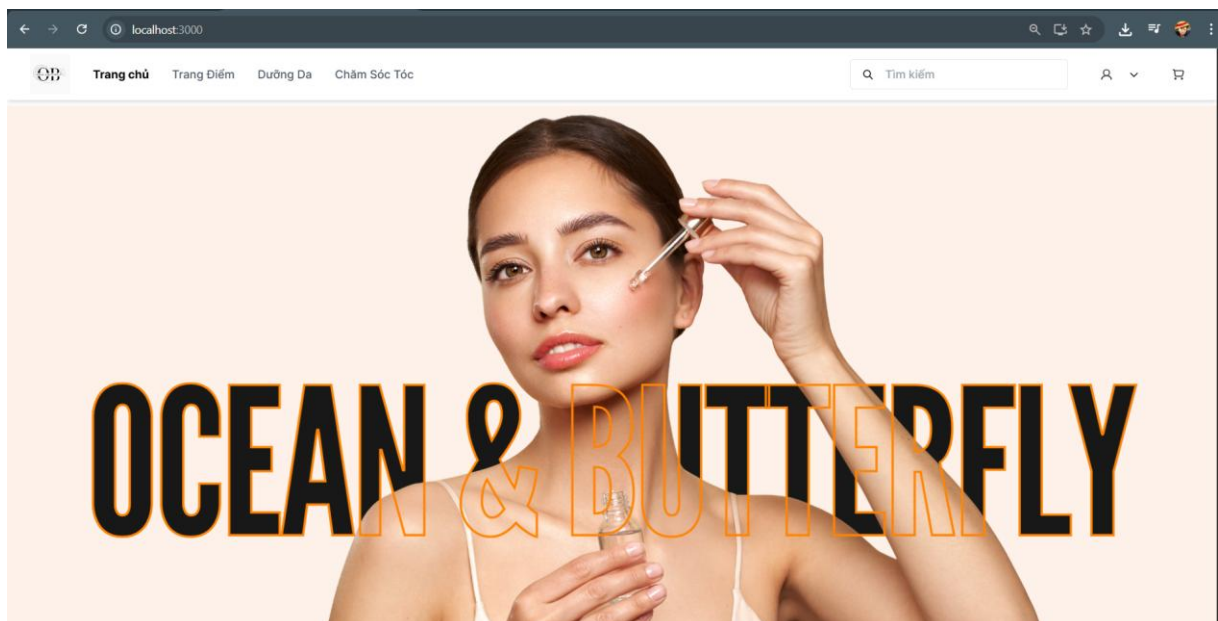
Giao diện đăng ký bao gồm các trường như: Họ và tên, tên đăng nhập, số điện thoại, giới tính, địa chỉ, mật khẩu, xác nhận lại mật khẩu và nút đăng ký để gửi yêu cầu cho hệ thống.



Hình 4.3 Giao diện quên mật khẩu

## 4.2. Giao diện người dùng

### 4.2.1. Giao diện trang chủ



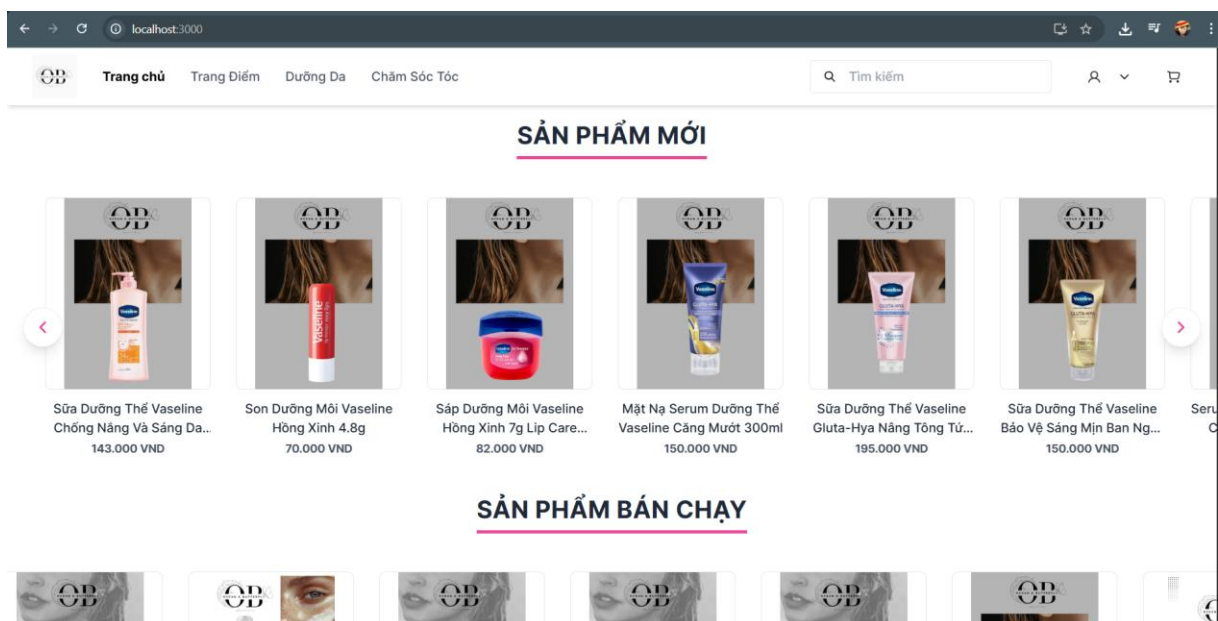
Hình 4.4 Giao diện trang chủ

Mô tả: Giao diện trang chủ bao gồm thanh điều hướng, thanh điều hướng có các mục như:

- Logo của shop: khi click vào logo sẽ hiển thị trang giới thiệu sơ lược về chúng tôi
- Trang chủ
- Trang điểm: Điều hướng đến trang chứa các sản phẩm liên quan về trang điểm.
- Dưỡng da: Điều hướng đến trang chứa các sản phẩm liên quan về dưỡng da.
- Chăm sóc tóc: Điều hướng đến trang chứa các sản phẩm liên quan về chăm sóc tóc.
- Thanh tìm kiếm: Có thể tìm kiếm các sản phẩm theo tên.
- Icon tài khoản: Khi click vào icon sẽ đưa qua trang đăng nhập nếu như đã đăng nhập thì sẽ hiển thị tên tài khoản, đơn hàng cá nhân và đăng xuất.
- Icon giỏ hàng: Điều hướng đến trang giỏ hàng để tiến hành đặt hàng.

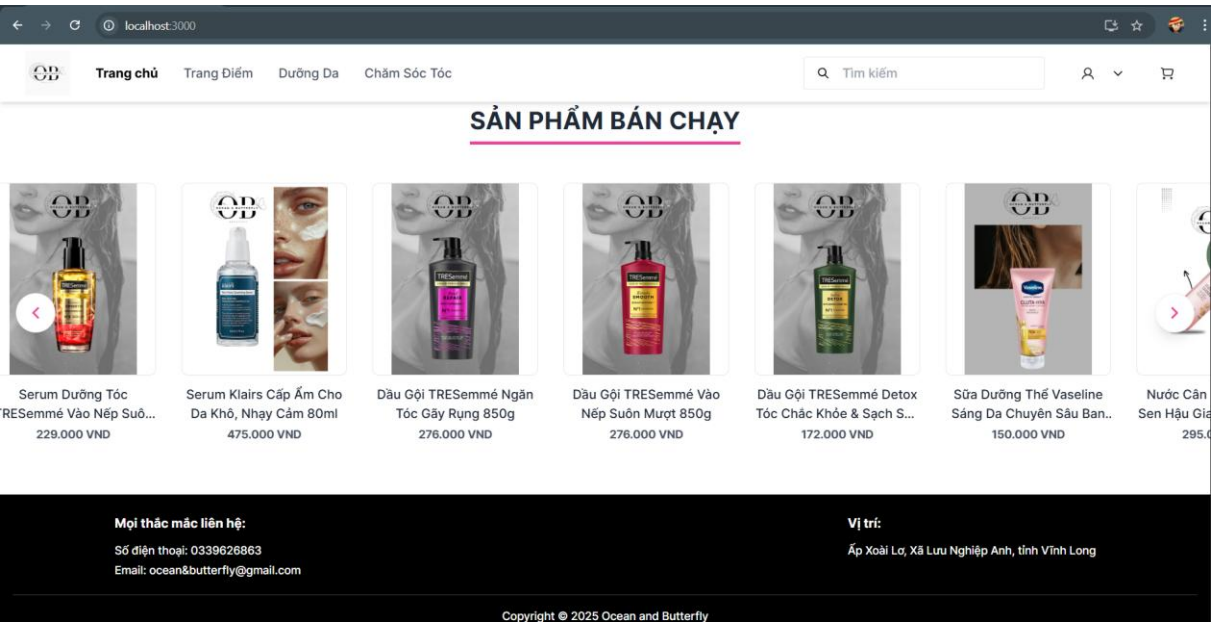
Thanh điều hướng dùng chung cho tất cả các trang giao diện người dùng.

Phía dưới thanh điều hướng là banner chứa tên shop Ocean & Butterlfy.



Hình 4.5: Giao diện trang chủ (Sản phẩm mới)

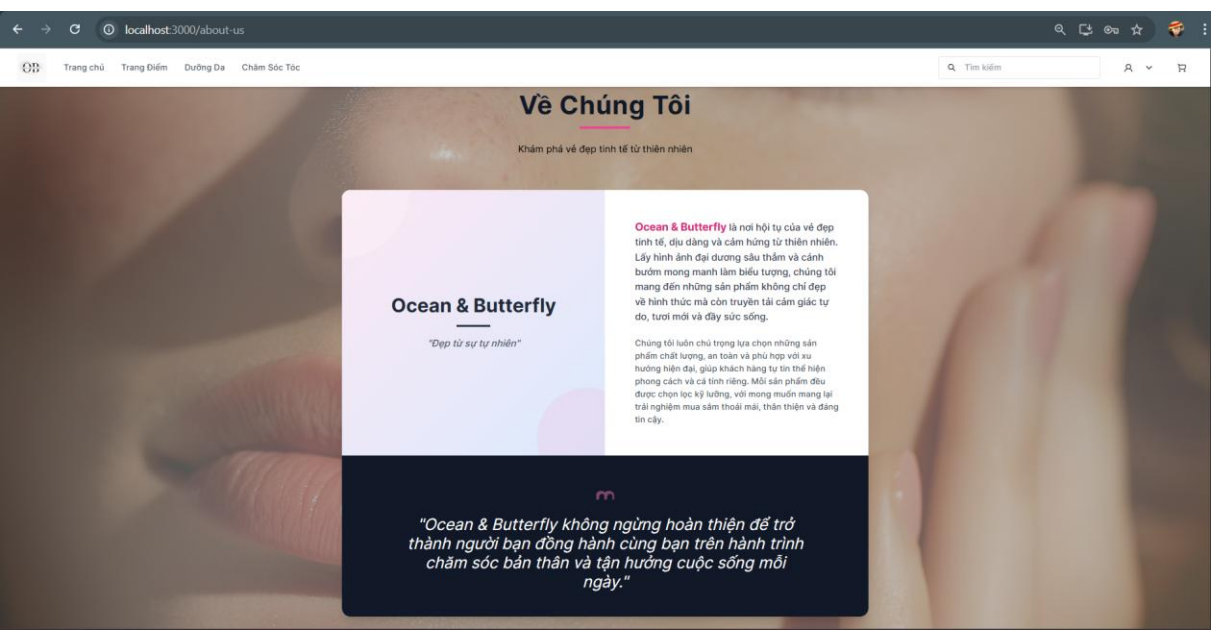
Mô tả: Khi có sản phẩm mới được nhân viên hoặc quản trị thêm vào thì ở mục sản phẩm mới sẽ hiển thị các sản phẩm vừa được thêm vào.



Hình 4.6 Giao diện trang chủ (Sản phẩm bán chạy)

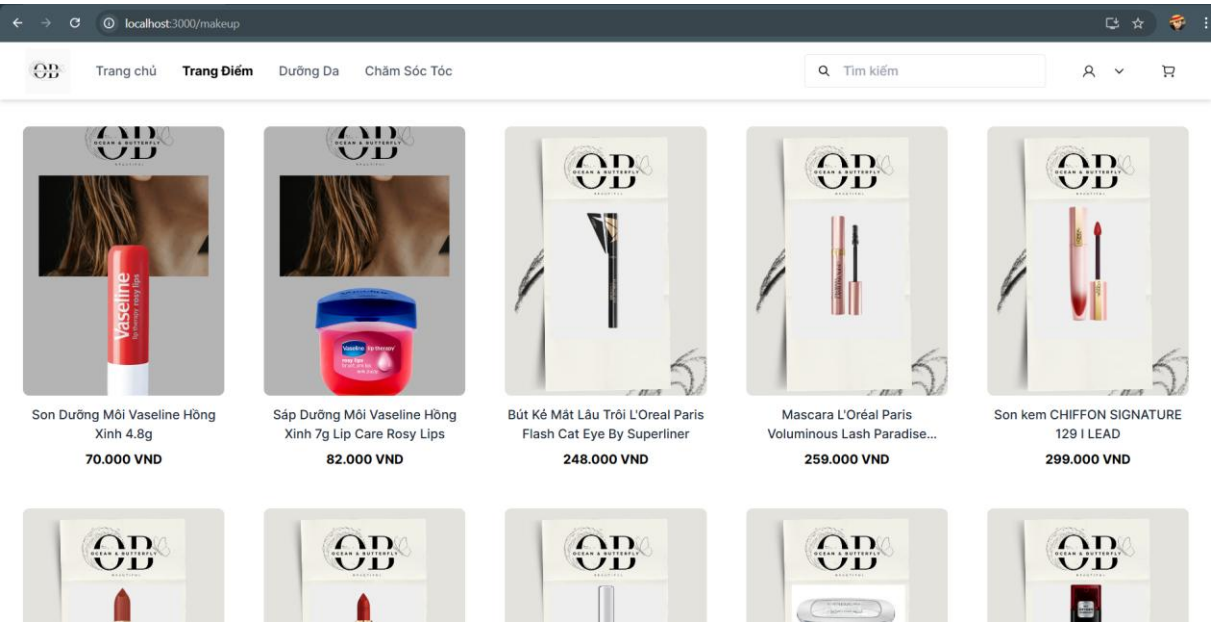
Mô tả: Sản phẩm có số lượng đặt hàng cao sẽ hiển thị trong mục sản phẩm bán chạy, phía dưới cùng là footer chứa thông tin liên hệ, bao gồm số điện thoại, email và vị trí của shop.

#### 4.2.2. Giao diện về chúng tôi

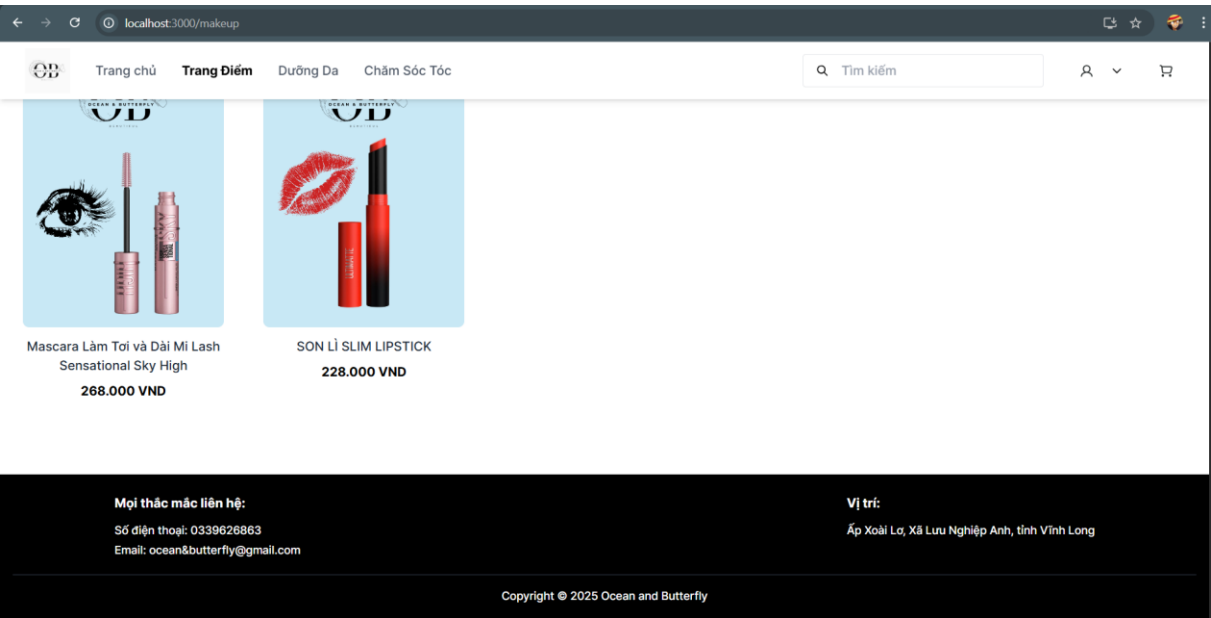


Hình 4.7 Giao diện về chúng tôi

4.2.3. Giao diện trang trang điểm



Hình 4.8 Giao diện trang Trang điểm

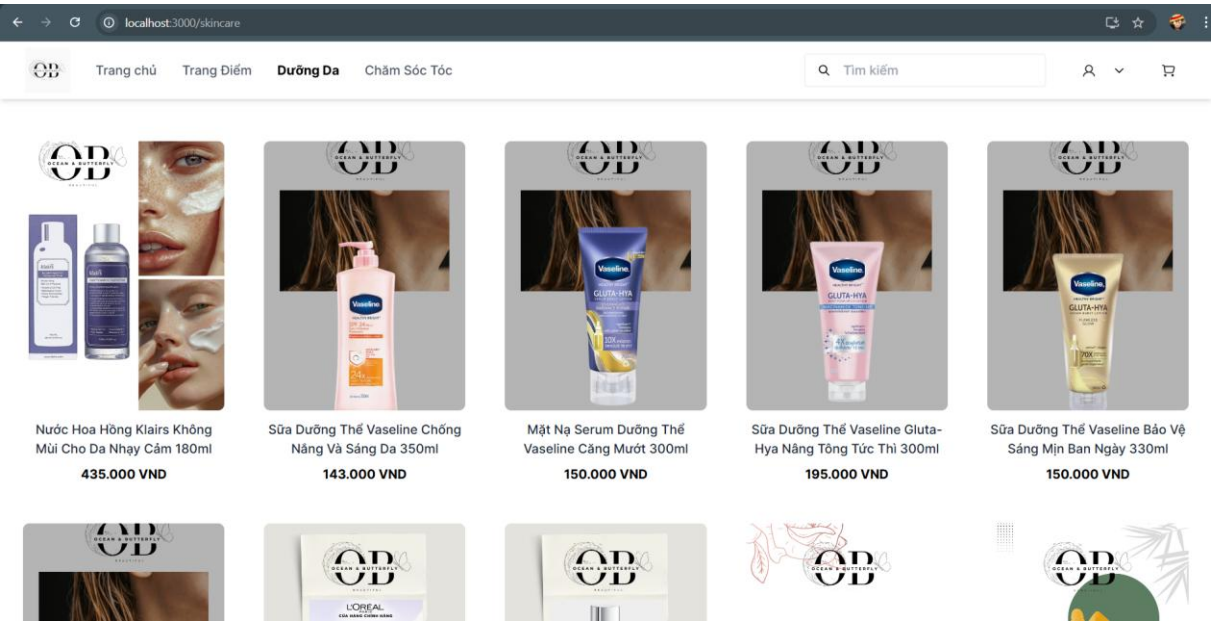


Hình 4.9 Giao diện trang Trang điểm

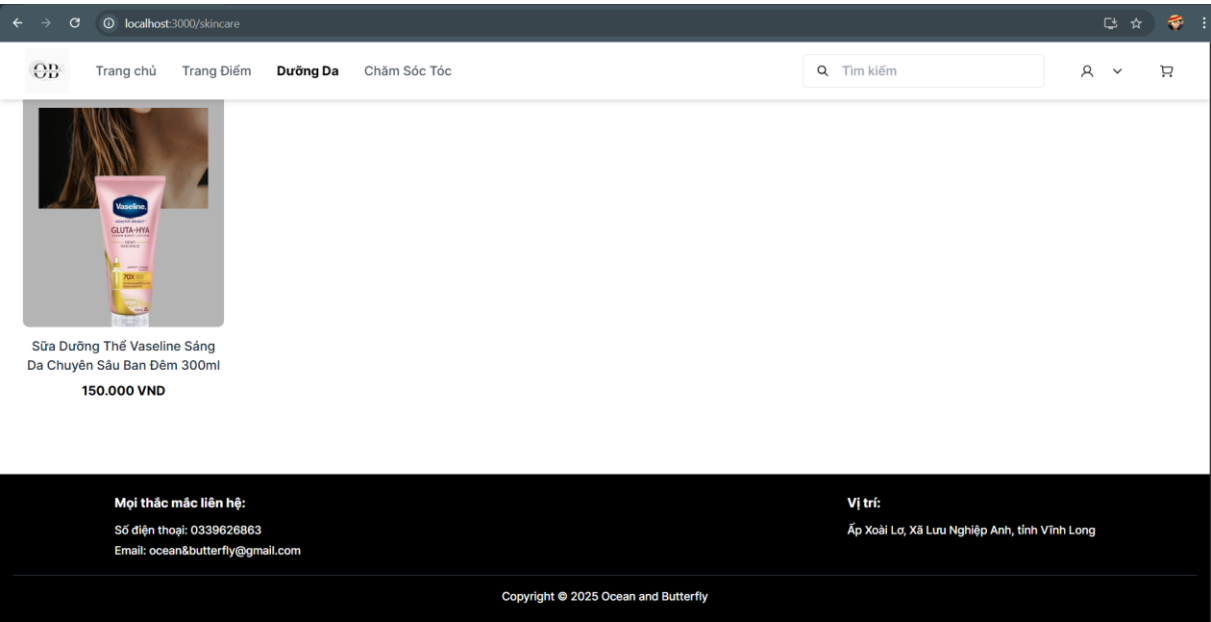
Mô tả: Chứa các sản phẩm thuộc phân loại trang điểm.



4.2.4. Giao diện trang dưỡng da



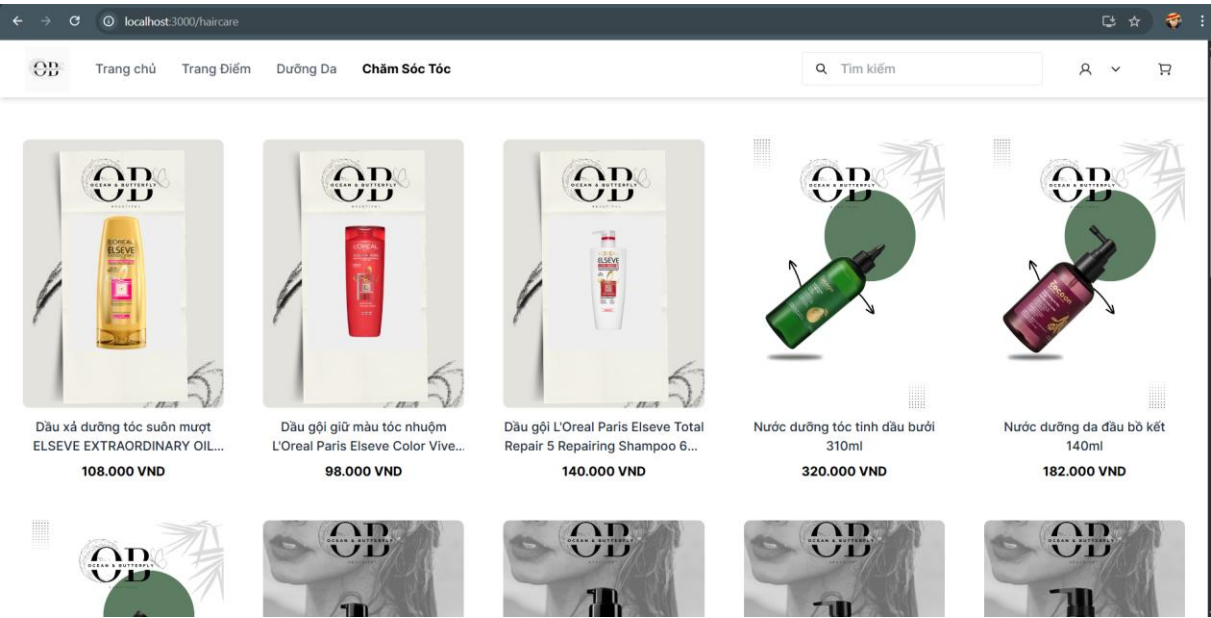
Hình 4.10 Giao diện trang Dưỡng da



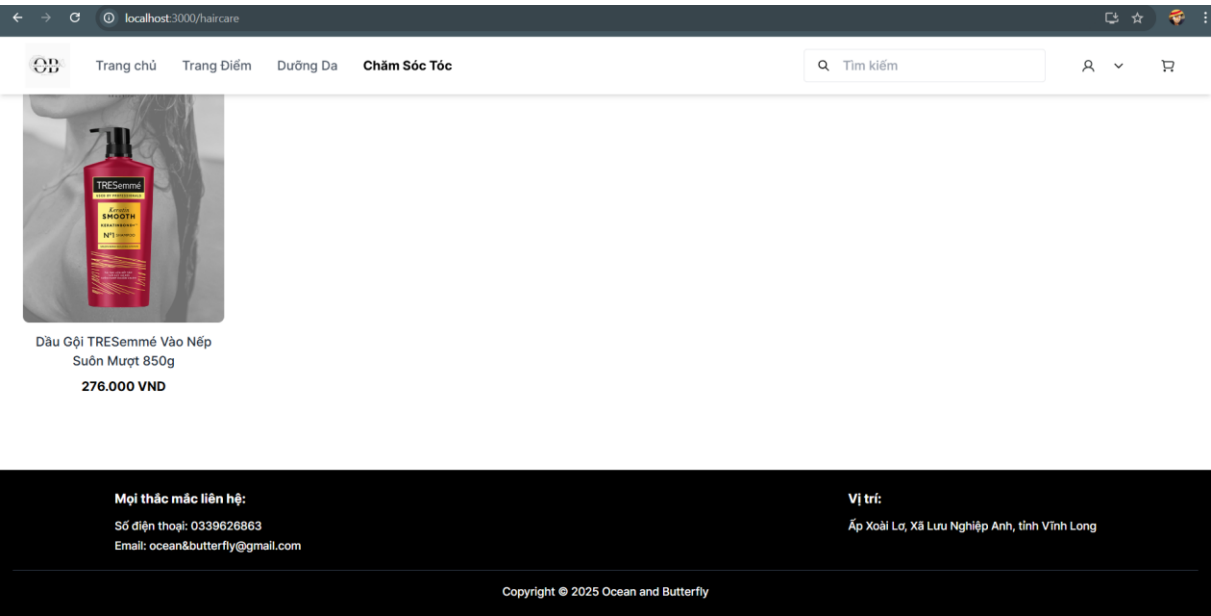
Hình 4.11 Giao diện trang Dưỡng da

Mô tả: Chứa các sản phẩm thuộc phân loại dưỡng da.

4.2.5. Giao diện trang chăm sóc tóc



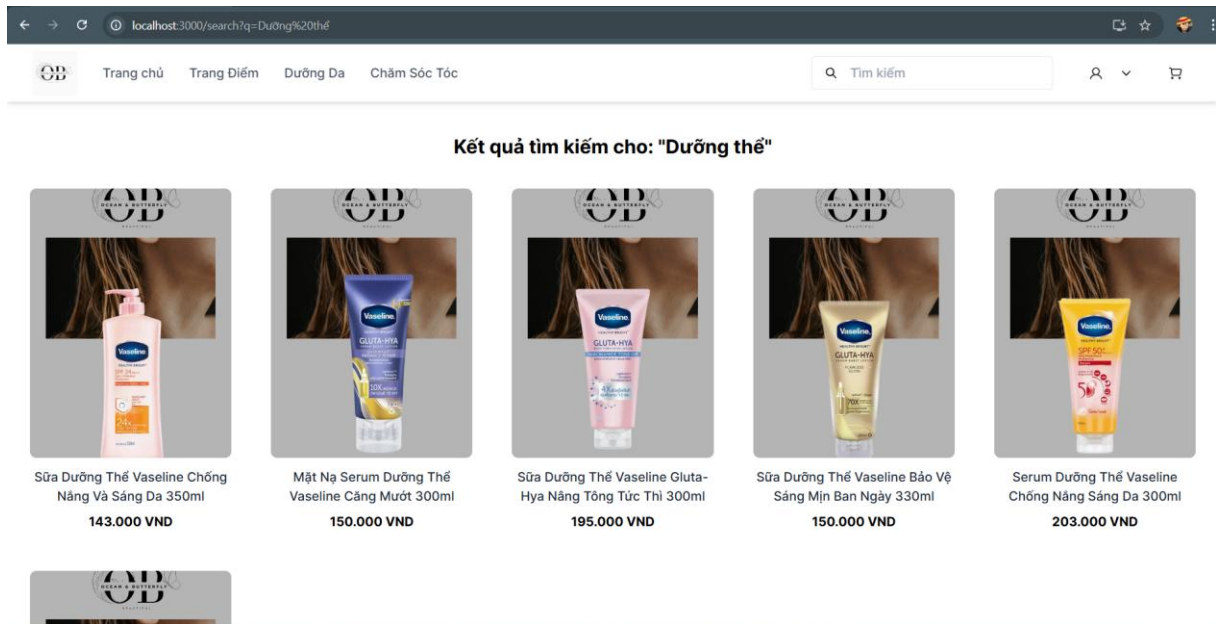
Hình 4.12 Giao diện trang Chăm sóc tóc



Hình 4.13 Giao diện trang Chăm sóc tóc

Mô tả: Chứa các sản phẩm thuộc phân loại chăm sóc tóc.

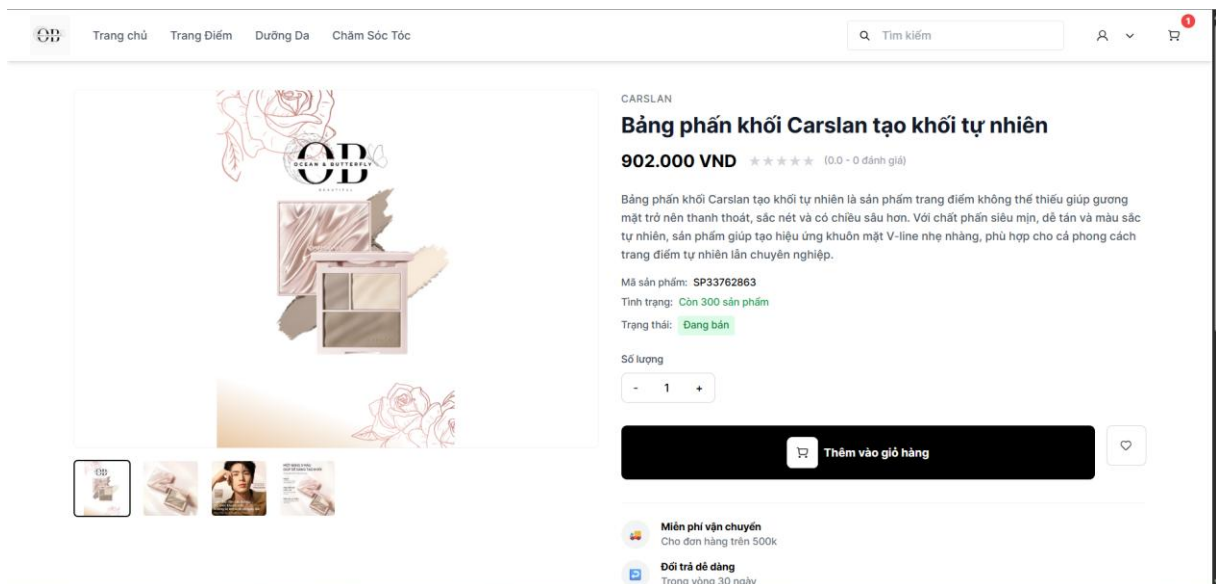
## 4.2.6. Giao diện tìm kiếm



Hình 4.14 Giao diện tìm kiếm

Mô tả: Khi tìm kiếm từ khóa giao diện sẽ cho ra kết quả đúng với từ khóa tìm kiếm mong muốn.

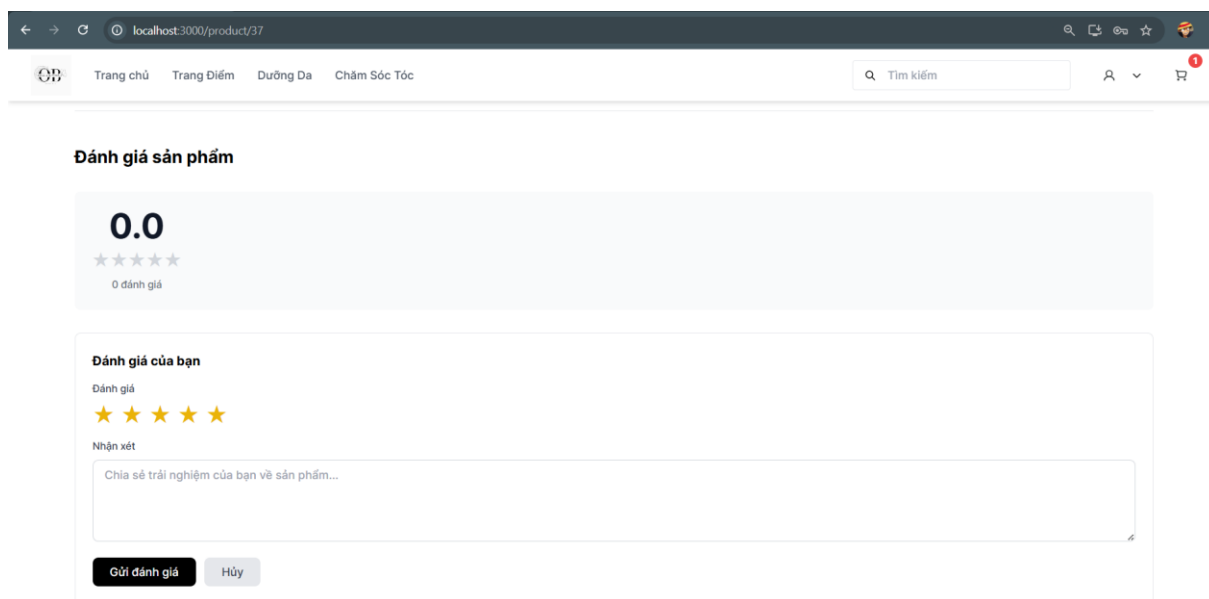
## 4.2.7. Giao diện chi tiết sản phẩm



Hình 4.15 Giao diện chi tiết sản phẩm

Mô tả: Giao diện chi tiết sản phẩm chứa các nội dung như hình ảnh chi tiết sản phẩm, thương hiệu, tên sản phẩm, giá sản phẩm, mô tả sản phẩm, mã sản phẩm, số lượng sản phẩm, trạng thái và thêm vào giỏ hàng.

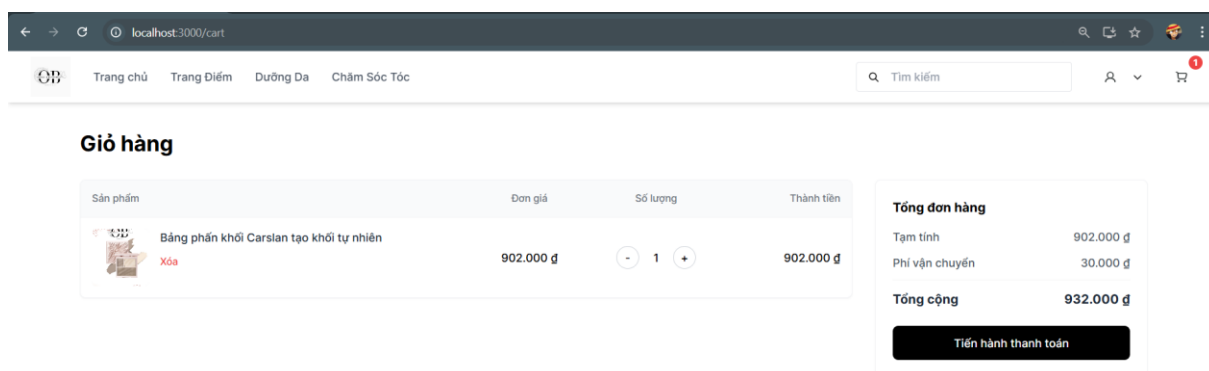
#### 4.2.8. Giao diện viết đánh giá sản phẩm




Hình 4.16 Giao diện viết đánh giá sản phẩm

Mô tả: Người dùng có thể viết đánh giá sản phẩm khi đăng nhập và đã mua sản phẩm, có thể xem các đánh giá của người khác nhưng không thể xóa, chỉ có thể xóa bài viết của chính mình.

#### 4.2.9. Giao diện giỏ hàng

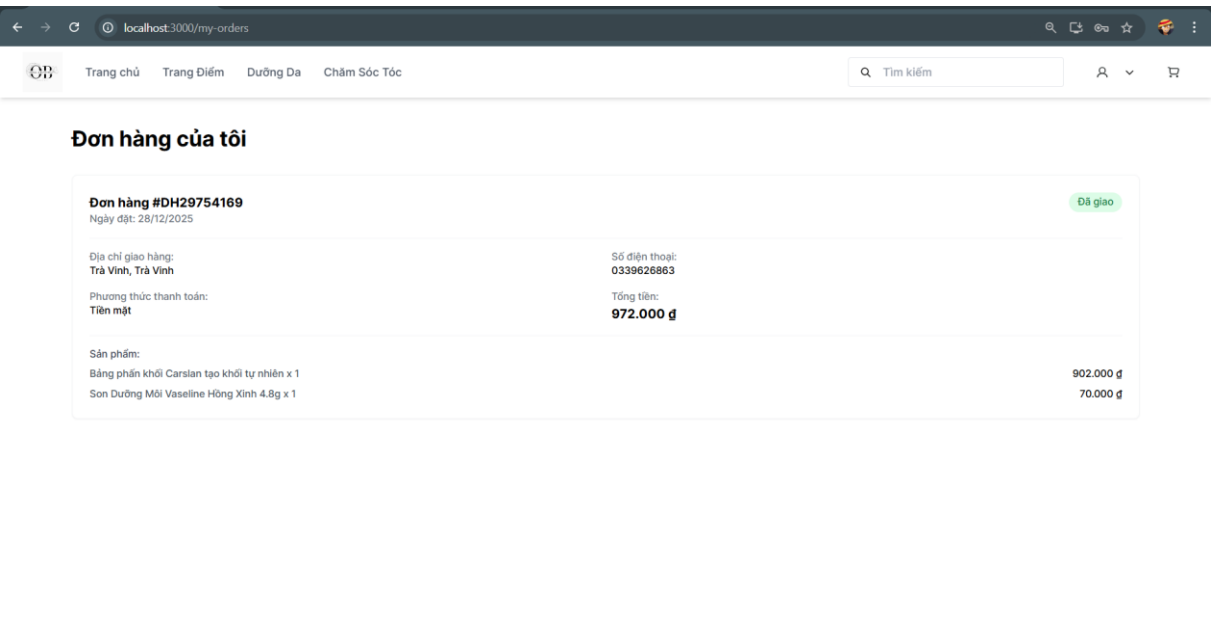


Sản phẩm	Đơn giá	Số lượng	Thành tiền
 Bảng phấn khối Carslan tạo khối tự nhiên <a href="#">Xóa</a>	902.000 đ	- 1 +	902.000 đ

**Tổng đơn hàng**  
Tạm tính: 902.000 đ  
Phí vận chuyển: 30.000 đ  
**Tổng cộng: 932.000 đ**  
[Tiến hành thanh toán](#)

Hình 4.17 Giao diện giỏ hàng

4.2.10. Giao diện đơn hàng của tôi

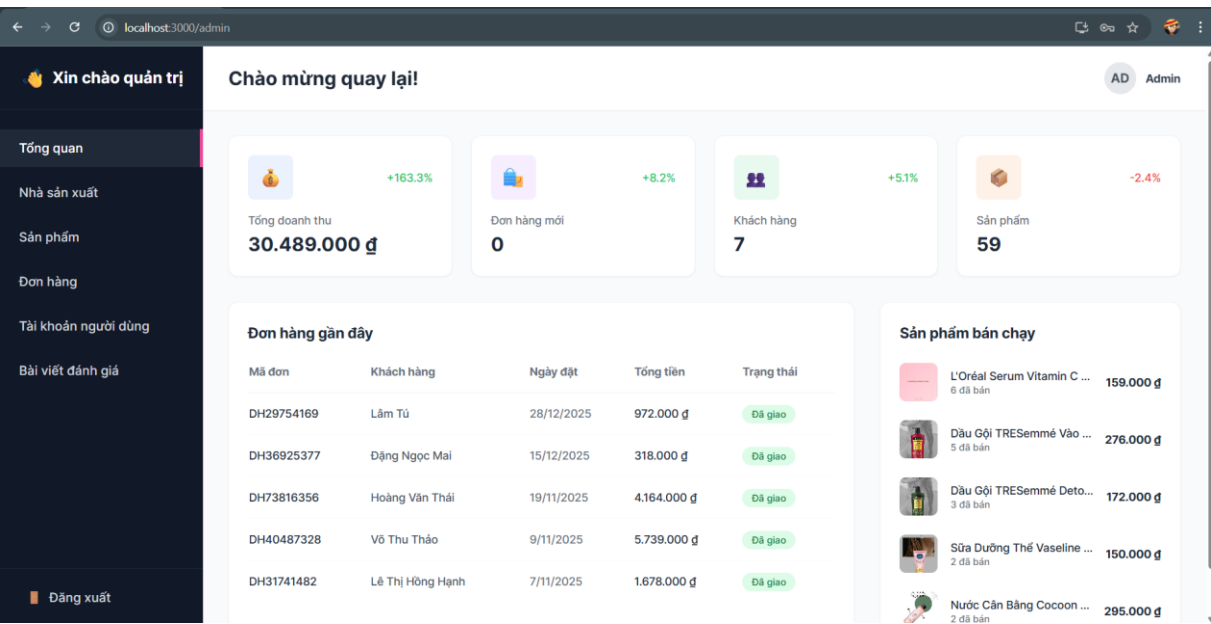


Hình 4.18 Giao diện đơn hàng của tôi

Mô tả: Ở đây người dùng có thể theo dõi được quá trình đơn hàng cũng như chi tiết đơn hàng của mình.

4.3. Giao diện quản trị

4.3.1. Giao diện tổng quan

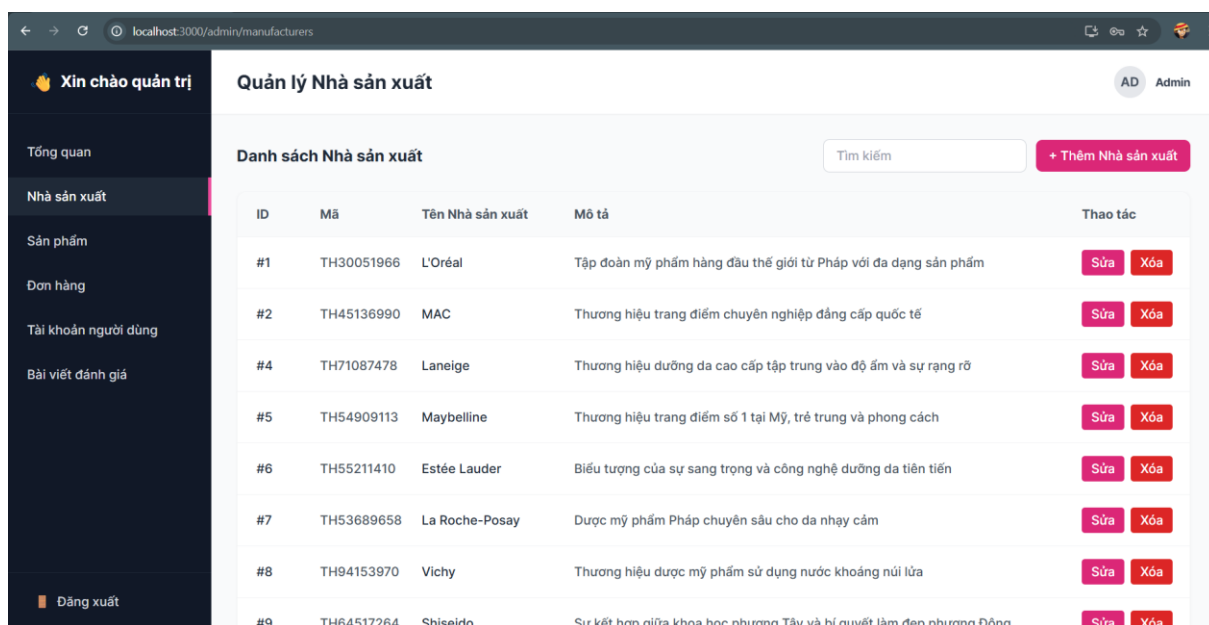


Hình 4.19 Giao diện tổng quan của quản trị

Mô tả: Giao diện hiển thị bao gồm doanh thu, đơn hàng mới, khách hàng, sản phẩm, đơn hàng gần đây và sản phẩm bán chạy.

Bên trái là thanh điều hướng bao gồm: Tổng quan, nhà sản xuất, sản phẩm, đơn hàng, tài khoản người dùng và bài viết đánh giá, đăng xuất.

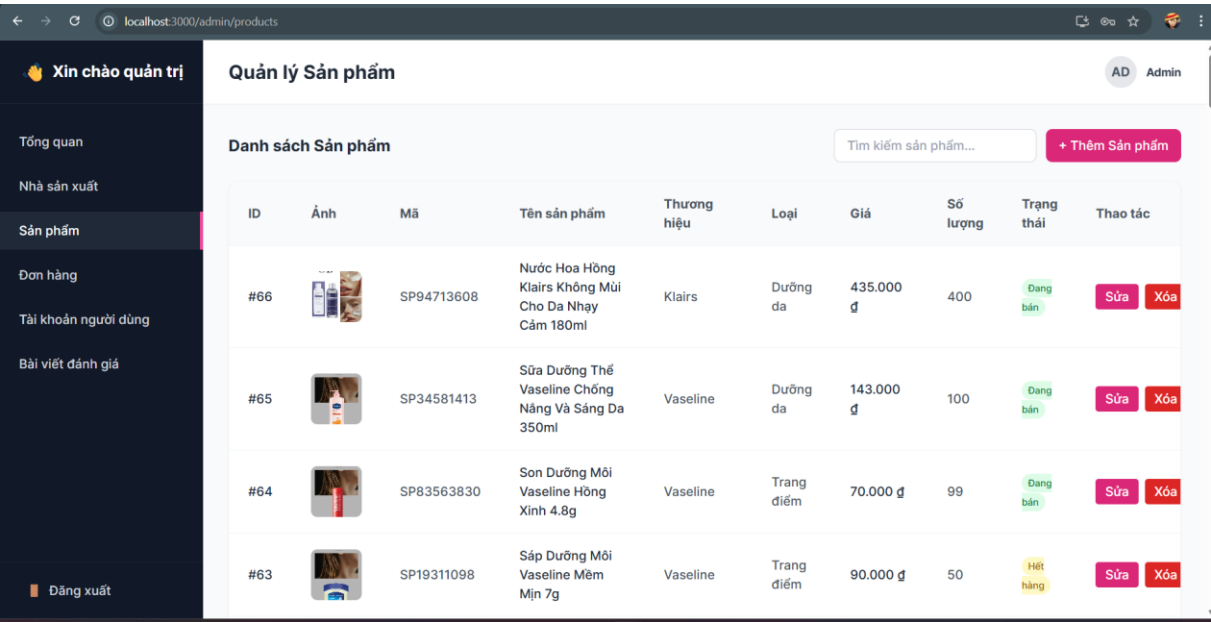
#### 4.3.2. Giao diện nhà sản xuất



Hình 4.20 Giao diện nhà sản xuất trong quản trị

Mô tả: Trong giao diện nhà sản xuất bao gồm danh sách nhà sản xuất hiển thị các thông tin cơ bản và các chức năng như thêm, sửa, xóa nhà sản xuất, tìm kiếm các thông tin có trong danh sách nhà sản xuất.

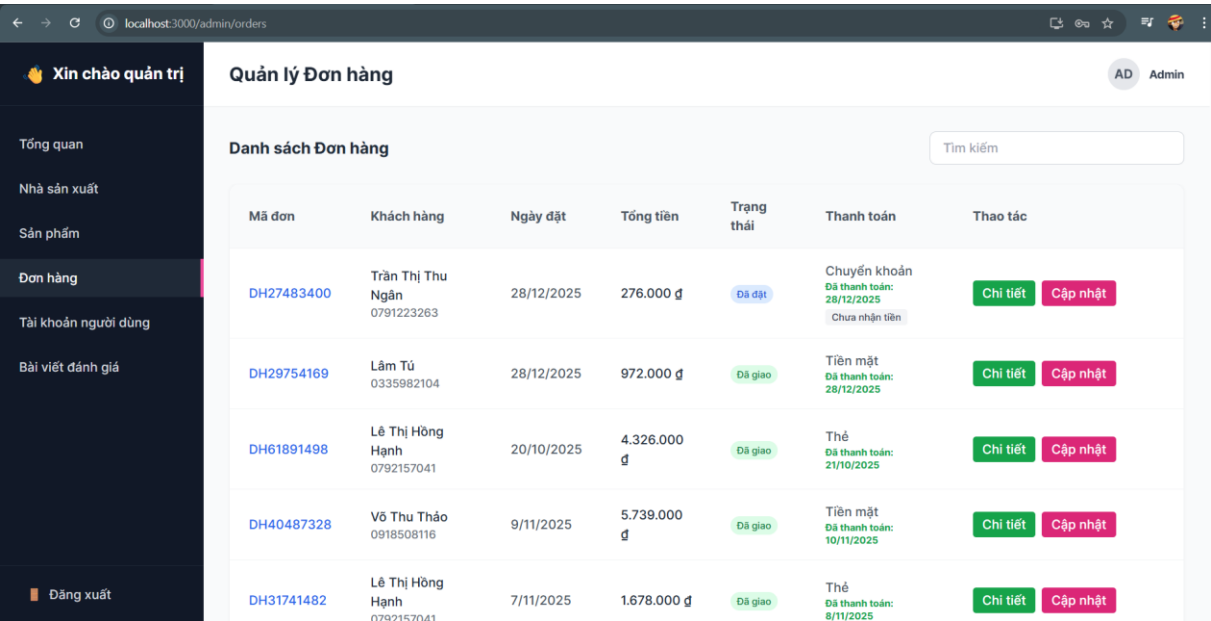
4.3.3. Giao diện sản phẩm



Hình 4.21 Giao diện sản phẩm

Mô tả: Trong giao diện sản phẩm bao gồm danh sách sản phẩm hiển thị các thông tin cơ bản và các chức năng như thêm, sửa, xóa sản phẩm, tìm kiếm các thông tin có trong danh sách sản phẩm.

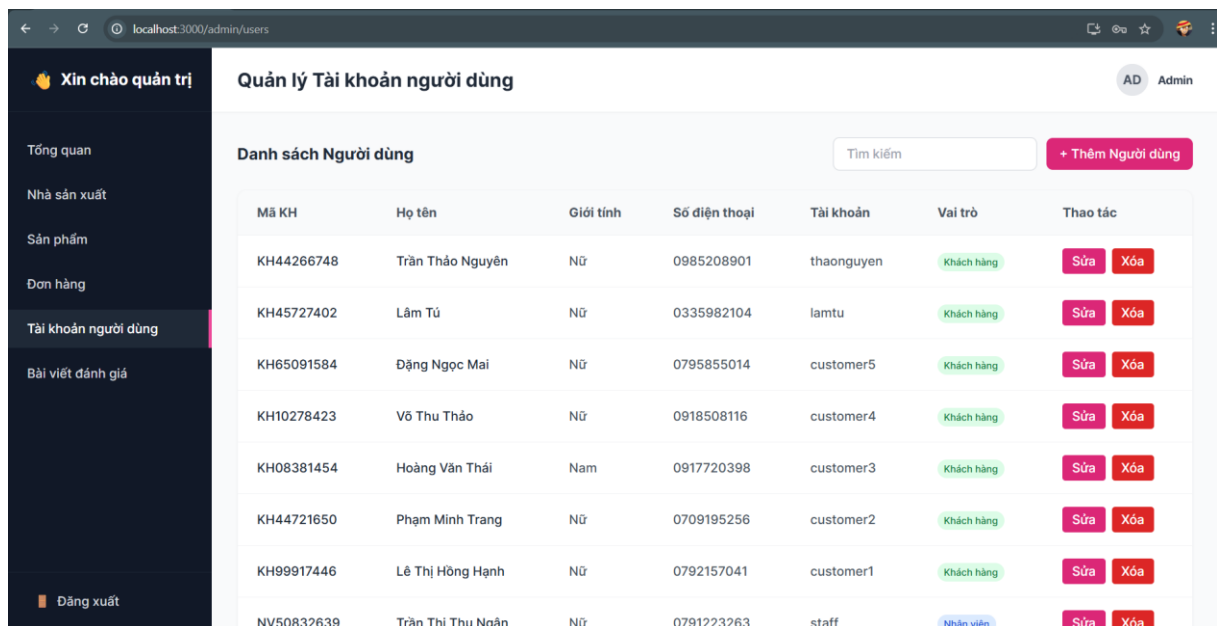
4.3.4. Giao diện đơn hàng



Hình 4.22 Giao diện đơn hàng

Mô tả: Trong giao diện đơn hàng bao gồm danh sách đơn hàng hiển thị các thông tin cơ bản và các chức năng như xem chi tiết, cập nhật và xóa đơn hàng, tìm kiếm các thông tin có trong danh sách đơn hàng.

#### 4.3.5. Giao diện tài khoản người dùng



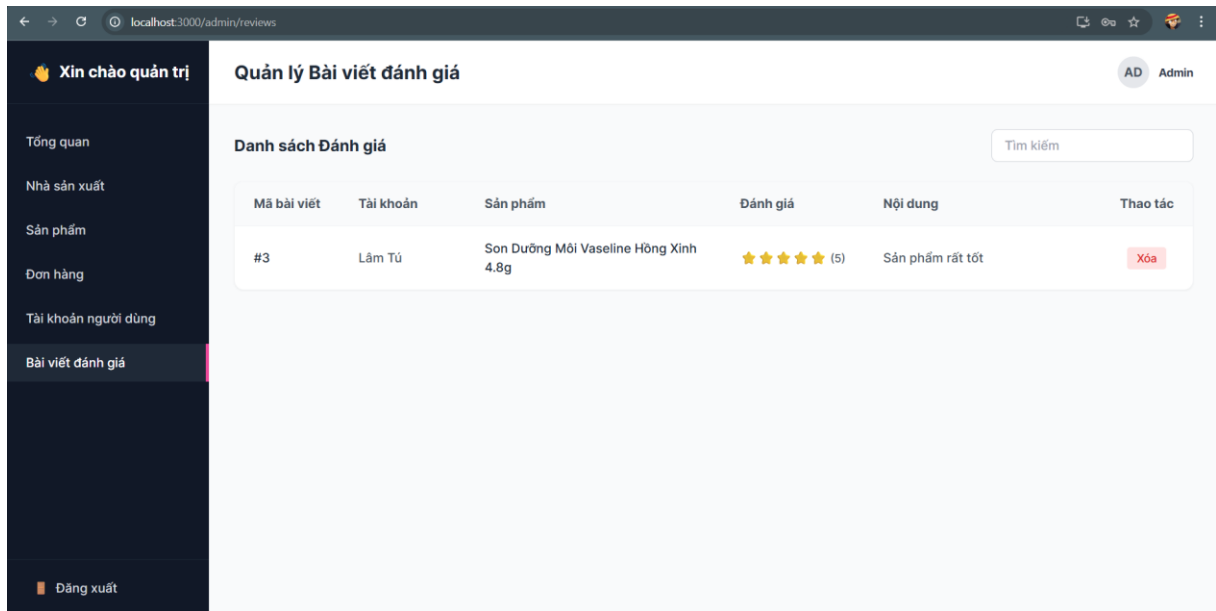
Mã KH	Họ tên	Giới tính	Số điện thoại	Tài khoản	Vai trò	Thao tác
KH44266748	Trần Thảo Nguyên	Nữ	0985208901	thaonguyen	Khách hàng	<a href="#">Sửa</a> <a href="#">Xóa</a>
KH45727402	Lâm Tú	Nữ	0335982104	lamtu	Khách hàng	<a href="#">Sửa</a> <a href="#">Xóa</a>
KH65091584	Đặng Ngọc Mai	Nữ	0795855014	customer5	Khách hàng	<a href="#">Sửa</a> <a href="#">Xóa</a>
KH10278423	Võ Thu Thảo	Nữ	0918508116	customer4	Khách hàng	<a href="#">Sửa</a> <a href="#">Xóa</a>
KH08381454	Hoàng Văn Thái	Nam	0917720398	customer3	Khách hàng	<a href="#">Sửa</a> <a href="#">Xóa</a>
KH44721650	Phạm Minh Trang	Nữ	0709195256	customer2	Khách hàng	<a href="#">Sửa</a> <a href="#">Xóa</a>
KH99917446	Lê Thị Hồng Hạnh	Nữ	0792157041	customer1	Khách hàng	<a href="#">Sửa</a> <a href="#">Xóa</a>
NV50832639	Trần Thị Thu Ngân	Nữ	0791223263	staff	Nhân viên	<a href="#">Sửa</a> <a href="#">Xóa</a>

Hình 4.23 Giao diện tài khoản người dùng

Mô tả: Trong giao diện tài khoản người dùng bao gồm danh sách tài khoản người dùng hiển thị các thông tin cơ bản và các chức năng như thêm, xóa, sửa tài khoản người dùng, tìm kiếm các thông tin có trong danh sách tài khoản người dùng.



### 4.3.6. Giao diện bài viết đánh giá

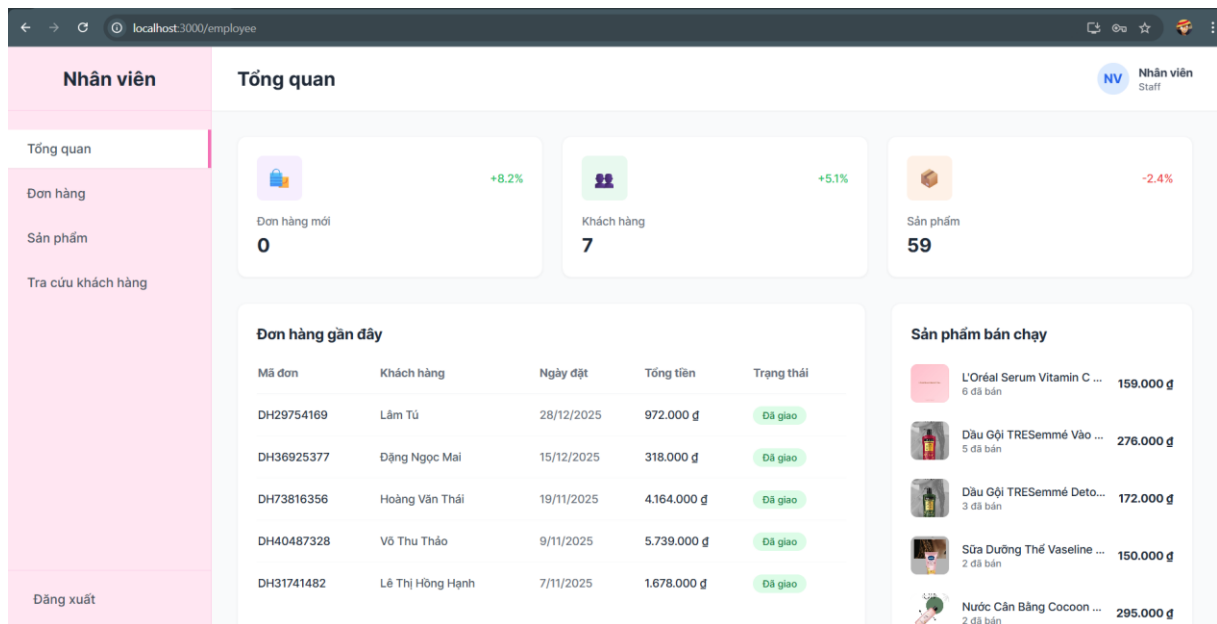


Hình 4.24 Giao diện bài viết đánh giá

Mô tả: Trong giao diện bài viết đánh giá bao gồm danh sách bài viết đánh giá hiển thị các thông tin cơ bản và chức năng xóa bài viết vi phạm quy tắc cộng đồng tìm kiếm các thông tin có trong danh sách bài viết đánh giá.

## 4.4. Giao diện nhân viên

### 4.4.1. Giao diện tổng quan



Hình 4.25 Giao diện tổng quan của nhân viên

Mô tả: Giao diện hiển thị bao gồm đơn hàng mới, khách hàng, sản phẩm, đơn hàng gần đây và sản phẩm bán chạy.

Bên trái là thanh điều hướng bao gồm: Tổng quan, đơn hàng, sản phẩm, tra cứu khách hàng, đăng xuất.

4.4.2. Giao diện đơn hàng

Nhân viên

Tổng quan

Đơn hàng

Sản phẩm

Trợ giúp khách hàng

Đăng xuất

Quản lý Đơn hàng

Danh sách Đơn hàng

Tìm kiếm

Mã đơn	Khách hàng	Ngày đặt	Tổng tiền	Trạng thái	Thanh toán	Thao tác
DH27483400	Trần Thị Thu Ngân 0791223263	28/12/2025	276.000 đ	Đã đặt	Chuyển khoản Đã thanh toán: 28/12/2025 Chưa nhận tiền	Chi tiết Cập nhật
DH29754169	Lâm Tú 0335982104	28/12/2025	972.000 đ	Đã giao	Tiền mặt Đã thanh toán: 28/12/2025	Chi tiết Cập nhật
DH61891498	Lê Thị Hồng Hạnh 0792157041	20/10/2025	4.326.000 đ	Đã giao	Thẻ Đã thanh toán: 21/10/2025	Chi tiết Cập nhật
DH40487328	Võ Thu Thảo 0918508116	9/11/2025	5.739.000 đ	Đã giao	Tiền mặt Đã thanh toán: 10/11/2025	Chi tiết Cập nhật
DH31741482	Lê Thị Hồng Hạnh 0792157041	7/11/2025	1.678.000 đ	Đã giao	Thẻ Đã thanh toán: 8/11/2025	Chi tiết Cập nhật
DH88213172	Đặng Ngọc Mai 0795855014	7/11/2025	1.412.000 đ	Đã hủy	Tiền mặt Đã thanh toán: 8/11/2025	Chi tiết Cập nhật

Hình 4.26 Giao diện đơn hàng

Mô tả: Trong giao diện đơn hàng bao gồm danh sách đơn hàng hiển thị các thông tin cơ bản và các chức năng như xem chi tiết, cập nhật đơn hàng, tìm kiếm các thông tin có trong danh sách đơn hàng.

4.4.3. Giao diện sản phẩm

Nhân viên

Tổng quan

Đơn hàng

Sản phẩm

Trợ giúp khách hàng

Đăng xuất

Quản lý Sản phẩm

Danh sách Sản phẩm

Tìm kiếm sản phẩm...

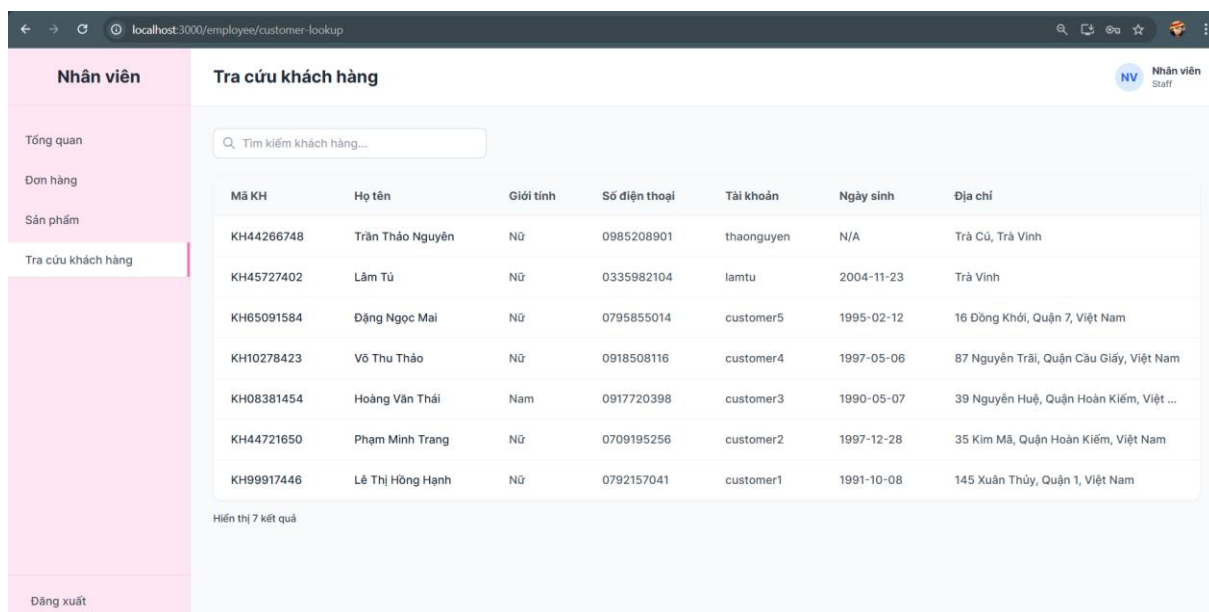
+ Thêm Sản phẩm

ID	Ảnh	Mã	Tên sản phẩm	Thương hiệu	Loại	Giá	Số lượng	Trạng thái	Thao tác
#66		SP94713608	Nước Hoa Hồng Klairs Không Mùi Cho Da Nhạy Cảm 180ml	Klairs	Dưỡng da	435.000 đ	400	Đang bán	Sửa
#65		SP34581413	Sữa Dưỡng Thể Vaseline Chống Nắng Và Sáng Da 350ml	Vaseline	Dưỡng da	143.000 đ	100	Đang bán	Sửa
#64		SP83563830	Son Dưỡng Môi Vaseline Hồng Xinh 4.8g	Vaseline	Trang điểm	70.000 đ	99	Đang bán	Sửa
#63		SP19311098	Sáp Dưỡng Môi Vaseline Mềm Mịn 7g	Vaseline	Trang điểm	90.000 đ	50	Hết hàng	Sửa

Hình 4.27 Giao diện sản phẩm

Mô tả: Trong giao diện sản phẩm bao gồm danh sách sản phẩm hiển thị các thông tin cơ bản và các chức năng như thêm, sửa, xóa sản phẩm, tìm kiếm các thông tin có trong danh sách sản phẩm.

#### 4.4.4. Giao diện tra cứu khách hàng



Hình 4.28 Giao diện tra cứu khách hàng

Mô tả: Trong giao diện tra cứu khách hàng bao gồm danh sách tài khoản người dùng hiển thị các thông tin cơ bản và tìm kiếm các thông tin có trong danh sách tra cứu khách hàng.

#### 4.5. Tài liệu API trên Swagger

##### 4.5.1. API RESTful để quản lý các hoạt động bán lẻ sản phẩm, bao gồm:

Quản lý sản phẩm : Thực hiện các thao tác CRUD cho các sản phẩm có liên kết thương hiệu.

Quản lý thương hiệu : Quản lý các thương hiệu sản phẩm và nhà sản xuất.

Quản lý đơn hàng : Tạo và theo dõi đơn hàng của khách hàng kèm cập nhật trạng thái.

Quản lý người dùng : Quản lý người dùng quản trị với cơ chế kiểm soát truy cập dựa trên vai trò.

Xác thực : Xác thực dựa trên JWT với ủy quyền dựa trên vai trò

#### 4.5.2. Xác thực

API này sử dụng xác thực JWT Bearer Token . Để truy cập các điểm cuối được bảo vệ:

Đăng ký hoặc đăng nhập qua `/api/v1/auth/register` hoặc `/api/v1/auth/login`

Sao chép `accessToken` từ phản hồi

Nhấp vào nút Xác thực ở góc trên bên phải.

Nhập: `Bearer <your-token>` và nhấp vào Xác thực

#### 4.5.3. Vai trò ủy quyền

QUẢN TRỊ VIÊN : Truy cập toàn hệ thống (Quản lý người dùng, Doanh thu, Cài đặt)

NHÂN VIÊN : Quản lý đơn hàng và sản phẩm

KHÁCH HÀNG : Xem sản phẩm, Đặt hàng, Xem lịch sử đơn hàng

KHÁCH : Xem sản phẩm, Đăng ký, Đăng nhập

#### 4.5.4. Quy ước API

Tất cả các điểm cuối đều được đặt tiền tố bằng `/api/v1`

Tất cả các phản hồi đều tuân theo định dạng JSON chuẩn: `{status, message, data}`

Mã được tạo tự động: Thương hiệu (TH), Sản phẩm (SP), Đơn hàng (DH), Người dùng (AD/NV/KH)

Độ chính xác BigDecimal cho tất cả các giá trị tiền tệ

## CHƯƠNG 5. KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1. Kết luận

#### 5.1.1. Kết quả

Hệ thống website bán mỹ phẩm được xây dựng nhằm hỗ trợ hiệu quả cho hoạt động quản lý và kinh doanh mỹ phẩm, giúp người quản trị và khách hàng thực hiện các chức năng theo đúng quyền hạn được phân công. Hệ thống góp phần tiết kiệm thời gian, giảm thiểu các thao tác thủ công, nâng cao trải nghiệm mua sắm của khách hàng, đồng thời tăng hiệu quả trong công tác quản lý sản phẩm, đơn hàng và khách hàng.

Website được thiết kế theo kiến trúc ba lớp gồm: giao diện người dùng thân thiện, trực quan; lớp nghiệp vụ đảm nhiệm việc xử lý các chức năng như quản lý sản phẩm, giỏ hàng, đơn hàng và thanh toán; cùng lớp dữ liệu đảm bảo lưu trữ thông tin an toàn và truy xuất nhanh chóng. Hệ thống hoạt động trên nền tảng web, cho phép người dùng dễ dàng truy cập mọi lúc, mọi nơi.

Sau khi triển khai, hệ thống sẽ được đánh giá và cải tiến định kỳ nhằm phát hiện các hạn chế, tối ưu hiệu năng và bổ sung các tính năng mới phù hợp với nhu cầu thực tế. Qua đó, website không chỉ đáp ứng yêu cầu kinh doanh hiện tại mà còn góp phần nâng cao chất lượng dịch vụ, xây dựng uy tín thương hiệu và tăng khả năng cạnh tranh của cửa hàng mỹ phẩm trong môi trường thương mại điện tử ngày càng phát triển.

#### 5.1.2. Hạn chế

Chương trình chưa thực sự đầy đủ các tính năng như mong muốn. Bên cạnh, việc khai thác thông tin chưa thật sự triệt để, chưa áp dụng nhiều vào thực tế.

Kiến thức thực tế và kiến thức lập trình còn hạn chế, nên tính chuyên nghiệp của chương trình chưa cao, tính năng bắt lỗi chưa được tối ưu.

## 5.2. Hướng phát triển

Trong tương lai, hệ thống website bán mỹ phẩm có thể được tiếp tục phát triển theo một số hướng sau:

- Tích hợp đăng nhập nhanh bằng các tài khoản phổ biến như Google hoặc Microsoft.
- Mở rộng triển khai hệ thống cho nhiều cửa hàng mỹ phẩm, đáp ứng nhu cầu kinh doanh ở quy mô lớn hơn.

## TÀI LIỆU THAM KHẢO

- [1] S. U. K. Siva Prasad Reddy, in *Beginning Spring Boot 3: Build Dynamic Cloud-Native Java* , 2023.
- [2] E. P. Alex Banks, "Learning React: Modern Pattern for Developing React Apps," 2020.
- [3] R. E. v. S. Navathe, "Fundamentals of database systems, Seventh edition. Boston Munich: Pearson," 2016.
- [4] A. Ehsan1, "RESTful API Testing Methodologies: Rationale, Challenges, and Solution Directions".
- [5] H. S. Oluwatosin, "Client-Server Model".
- [6] F. •. H. •. J. •. Pfleiderer, "Spring Boot 3".
- [7] \*. Nikola Dimitrijević1, "Advanced Security Mechanisms in the Spring Framework:".