

**TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG
TP.HCM**

KHOA CÔNG NGHỆ THÔNG TIN



**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC
NGÀNH CÔNG NGHỆ THÔNG TIN**

PHÁT TRIỂN ỨNG DỤNG MUA SẴM ĐỒ GIA DỤNG



Giảng viên hướng dẫn: **ThS. Phạm Trọng Huỳnh**

Sinh viên thực hiện : **Lê Quốc Việt**

Mã số sinh viên: **1050080083**

Lớp: **10_ĐH_CNPM1**

Khoá: **2021 - 2025**

TP. Hồ Chí Minh, tháng 12 năm 2025

TRƯỜNG ĐẠI HỌC TÀI NGUYÊN VÀ MÔI TRƯỜNG TP.HCM

KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC NGÀNH CÔNG NGHỆ THÔNG TIN

PHÁT TRIỂN ỨNG DỤNG MUA SẮM ĐỒ GIA DỤNG



Giảng viên hướng dẫn: **ThS. Phạm Trọng Huỳnh**

Sinh viên thực hiện: **Lê Quốc Việt**

Mã số sinh viên: **1050080083**

Lớp: **10_ĐH_CNPM1**

Khoá: **2021 - 2025**

TP. Hồ Chí Minh, tháng 12 năm 2025

LỜI CẢM ƠN

Lời đầu tiên, em xin được gửi lời cảm ơn chân thành nhất đến thầy Phạm Trọng Huynh người đã tận tình hướng dẫn và hỗ trợ em trong quá trình thực hiện đồ án tốt nghiệp. Trong suốt thời gian làm đồ án, em đã nhận được rất nhiều sự quan tâm, chỉ bảo tận tâm từ thầy, giúp em tích lũy thêm nhiều kiến thức quý báu về lĩnh vực này. Nhờ sự hướng dẫn nhiệt huyết của thầy, em đã có thể hoàn thành tốt đồ án với đề tài: Phát triển ứng dụng mua sắm đồ gia dụng .Cuối cùng, em xin kính chúc thầy Huynh, thật nhiều sức khỏe, hạnh phúc, một năm tràn đầy may mắn và thành công trong mọi nhiệm vụ được giao.

Em xin chân thành cảm ơn!

Sinh viên thực hiện

Lê Quốc Việt

LỜI CAM ĐOAN

Em xin cam đoan đề tài “Phát triển ứng dụng mua sắm đồ gia dụng” là công trình nghiên cứu do chính em thực hiện dưới sự hướng dẫn của ThS. Phạm Trọng Huỳnh. Toàn bộ nội dung nghiên cứu, sản phẩm và kết quả trình bày trong đồ án được hình thành từ quá trình nghiên cứu, lên ý tưởng và xây dựng của cá nhân em.

Các tài liệu, hình ảnh và nội dung được trích dẫn trong đồ án đều có nguồn gốc rõ ràng và được ghi chú đầy đủ theo quy định. Em xin cam đoan đồ án không sao chép từ bất kỳ tổ chức, cá nhân hay công trình nghiên cứu nào khác.

Em hoàn toàn chịu trách nhiệm về tính trung thực và chính xác của toàn bộ nội dung đồ án này.

Sinh viên thực hiện

(ký tên và ghi rõ họ tên)

Lê Quốc Việt

[illegible]

.....

(Ký và ghi rõ học hàm, học vị họ tên)

ThS. Phạm Trọng Huỳnh

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

.....

(Ký và ghi rõ học hàm, học vị họ tên)

LỊCH LÀM VIỆC

Tuần	Nội dung	Địa điểm	Nhận xét của GV hướng dẫn	Chữ ký của GV hướng dẫn
1	Lên ý tưởng, thu thập tài liệu	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
2	Vẽ sơ đồ chức năng và mô hình dữ liệu.	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
3	Thiết kế cơ sở dữ liệu, use case	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
4	Thiết kế giao diện ứng dụng và web	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
5	Thiết kế giao diện ứng dụng và web	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
6	Xây dựng back end	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
7	Xây dựng back end	Trường ĐH Tài Nguyên và Môi		

		Trường thành phố Hồ Chí Minh		
8	Xây dựng chức năng đăng nhập, đăng ký, xác minh qua email	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
9	Xây dựng chức năng tìm kiếm sản phẩm và đề xuất sản phẩm theo nhu cầu người dùng	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
10	Xây dựng chức năng xem chi tiết sản phẩm và bình luận	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
11	Xây dựng chức năng đặt hàng và thanh toán qua VNPAY	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
12	Xây dựng chức năng chat	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
13	Kiểm thử và đánh giá chức năng hệ thống.	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		
14	Kiểm thử và đánh giá chức năng hệ thống.	Trường ĐH Tài Nguyên và Môi		

		Trường thành phố Hồ Chí Minh		
15	Viết tài liệu báo cáo	Trường ĐH Tài Nguyên và Môi Trường thành phố Hồ Chí Minh		

TÓM TẮT ĐỒ ÁN

Đề tài xây dựng ứng dụng thương mại điện tử mua sắm đồ gia dụng dành cho hệ điều hành Android, được phát triển bằng ngôn ngữ Kotlin, cơ sở dữ liệu MySQL nhằm tăng tính bảo mật và hiệu quả trong quản lý dữ liệu người dùng. Ứng dụng hỗ trợ hai hình thức đăng nhập là Google và tài khoản riêng (username & password).

Ứng dụng cung cấp các chức năng chính như: xem và tìm kiếm sản phẩm, thêm vào giỏ hàng, đặt hàng và theo dõi đơn hàng, cùng với quản lý tài khoản cá nhân. Dữ liệu người dùng, sản phẩm và đơn hàng được lưu trữ trong hệ thống cơ sở dữ liệu quan hệ để đảm bảo tính toàn vẹn và an toàn.

Kết quả đạt được là một ứng dụng mẫu hoàn chỉnh, đáp ứng các chức năng đề ra, giao diện thân thiện và hoạt động ổn định. Hướng phát triển trong tương lai bao gồm phiên bản iOS, tối ưu hóa chức năng thanh toán trực tuyến và hệ thống gợi ý sản phẩm thông minh sử dụng AI nhằm nâng cao trải nghiệm người dùng.

MỤC LỤC

CHƯƠNG 1: TỔNG QUAN TÀI LIỆU	3
1.1 Cơ sở lý thuyết.....	3
1.1.1 Thuật toán	3
1.1.2 Kỹ thuật sử dụng.....	5
Chương 2: PHƯƠNG PHÁP THỰC HIỆN	18
2.1 Yêu cầu kỹ thuật	18
2.1.1 Yêu cầu chức năng.....	18
2.1.2 Yêu cầu hiệu suất.....	19
2.1.3 Yêu cầu bảo mật	19
2.2 Tổng thể về thiết kế hệ thống.....	20
2.2.1 Tầng Presentation (giao diện).....	20
2.2.2. Tầng Business logic (xử lý nghiệp vụ).....	21
2.2.3 Tầng Data (dữ liệu).....	21
2.3 Sự lựa chọn thiết kế và công nghệ.....	22
Chương 3: CÀI ĐẶT VÀ THỬ NGHIỆM	24
3.1 Phân tích hệ thống.....	24
3.1.1 Phía khách hàng.....	24
3.1.1 Phía quản trị viên.....	24
3.2 Thiết kế hệ thống.....	26
3.2.1 Mô tả ký hiệu UML	26
3.2.2 Đặc tả Actor	26
3.2.3 Sơ đồ Use Case	27
3.2.4 Sơ đồ trạng thái	50
3.2.5 Sơ đồ tuần tự.....	53
3.3 Thiết kế dữ liệu	60
3.3.1 Sơ đồ lớp (mức phân tích)	61
3.3.2 Giải thích từng thuộc tính và phương thức.....	61
3.4 Thiết kế mô hình đề xuất sản phẩm	71

3.4.1 Giới thiệu bài toán.....	71
3.4.2 Lý do chọn mô hình.....	72
3.4.3 Dữ liệu đầu vào của mô hình.....	72
3.4.4 Phân tích mô hình đề xuất NCF.....	73
3.4.5 Cài đặt mô hình NCF	74
3.4.6 Kết luận	76
3.5 Thiết kế giao diện ứng dụng.....	76
3.5.1 Giao diện khách hàng.....	76
3.5.2 Giao diện quản trị viên	77
3.6 Môi trường triển khai hệ thống	77
3.7 Thiết kế giao diện hệ thống.....	78
3.7.1 Giao diện trang đăng ký	80
3.7.2 Giao diện trang đăng nhập.....	80
3.7.3 Giao diện trang chủ.....	81
3.7.4 Giao diện trang tìm kiếm	81
3.7.5 Giao diện trang giỏ hàng.....	82
3.7.6 Giao diện trang quản lý đơn hàng	82
3.7.7 Giao diện trang chat.....	83
3.7.8 Giao diện trang thông tin cá nhân	83
3.7.9 Giao diện trang đổi mật khẩu.....	84
3.7.10 Giao diện trang chi tiết sản phẩm.....	84
3.7.11 Giao diện trang đánh giá sản phẩm	85
3.7.12 Giao diện trang đăng nhập admin	85
3.7.13 Giao diện trang thống kê.....	86
3.7.14 Giao diện trang quản lý khách hàng.....	86
3.7.15 Giao diện trang quản lý danh mục sản phẩm	87
3.7.17 Giao diện trang quản lý đơn hàng.....	88
3.7.18 Giao diện trang chat	88
3.7.19 Giao diện trang Kho	89

3.7.20 Giao diện trang xuất kho.....	89
3.7.21 Giao diện trang Nhập kho	90
3.7.22 Giao diện trang quản lý kho.....	91
3.7.22 Giao diện trang quản lý khuyến mãi	91
CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	92
4.1 Môi trường phát triển và môi trường triển khai	92
4.2 Kết quả đạt được	92
4.3 Hạn chế	93
4.4 Hướng phát triển	93
DANH MỤC TÀI LIỆU THAM KHẢO.....	95
Tiếng Việt:	95
Tiếng Anh:	100

DANH MỤC SƠ ĐỒ, HÌNH ẢNH

Hình 1.1 Logo Nodeks+Express	7
Hình 1.2 Logo MySQL.....	8
Hình 1.3 Logo Visual Studio Code	9
Hình 1.4 Logo Androi Studio.....	10
Hình 1.5 Logo Pycharm	11
Hình 1.6 Logo Postman.....	12
Hình 1.7 Logo git	13
Hình 1.8 Nguyên lý hoạt động mô hình MVVM	14
Hình 1.9 Nguyên lý hoạt động của NCF	16
Hình 2.1 Mô tả luồng dữ liệu trong hệ thống.....	22
Sơ đồ 3.1 Sơ đồ usecase tổng quát	27
Sơ đồ 3.2 Sơ đồ phân rã chức năng đăng ký	28
Sơ đồ 3.3 Sơ đồ phân rã UC đăng nhập	29
Sơ đồ 3.4 Sơ đồ phân rã UC đăng xuất	31
Sơ đồ 3.5 Sơ đồ phân rã UC tìm kiếm sản phẩm	32
Sơ đồ 3.6 Sơ đồ phân rã UC lọc sản phẩm.....	34
Sơ đồ 3.7 Sơ đồ phân rã UC quản lý danh sách yêu thích	35
Sơ đồ 3.8 Sơ đồ phân rã UC quản lý giỏ hàng	36
Sơ đồ 3.9 Sơ đồ phân rã UC thanh toán	38
Sơ đồ 3.10 Sơ đồ phân rã UC quản lý thông tin cá nhân	39
Sơ đồ 3.11 Sơ đồ phân rã UC quản lý đơn hàng	40
Sơ đồ 3.12 Sơ đồ phân rã quản lý danh mục.....	42
Sơ đồ 3.13 Sơ đồ phân rã UC quản lý sản phẩm.....	43
Sơ đồ 3.14 Sơ đồ phân rã UC thống kê	44
Sơ đồ 3.15 Sơ đồ phân rã UC quản lý đơn hàng	46
Sơ đồ 3.16 Sơ đồ phân rã UC quản lý khuyến mãi	47
Sơ đồ 3.17 Sơ đồ phân rã UC quản lý kho hàng	49
Sơ đồ 3.18 Sơ đồ trạng thái khách hàng.....	50

Sơ đồ 3.19 Sơ đồ trạng thái của quản trị viên	52
Sơ đồ 3.20 Sơ đồ tuần tự đăng ký	53
Sơ đồ 3.21 Sơ đồ tuần tự đăng nhập.....	54
Sơ đồ 3.22 Sơ đồ tuần tự đăng xuất.....	54
Sơ đồ 3.23 Sơ đồ tuần tự tìm kiếm sản phẩm.....	55
Sơ đồ 3.24 Sơ đồ tuần tự lọc sản phẩm	55
Sơ đồ 3.25 Sơ đồ tuần tự quản lý giỏ hàng của user	56
Sơ đồ 3.26 Sơ đồ tuần tự thanh toán	56
Sơ đồ 3.27 Sơ đồ quản lý đơn hàng của khách hàng.....	57
Sơ đồ 3.28 Sơ đồ quản lý sản phẩm của quản trị viên	57
Sơ đồ 3.29 Sơ đồ quản lý danh mục sản phẩm.....	58
Sơ đồ 3.30 Sơ đồ tuần tự thống kê	58
Sơ đồ 3.31 Sơ đồ tuần tự quản lý khuyến mãi.....	59
Sơ đồ 3.32 Sơ đồ tuần tự quản lý kho	59
Sơ đồ 3.33 Sơ đồ tuần tự tạo mới kho	60
Sơ đồ 3.34 Sơ đồ lớp ở mức phân tích	61
Sơ đồ 3.35 Sơ đồ luồng dữ liệu của mô hình	74
Sơ đồ 3.36 Sơ đồ liên kết giao diện khách hàng	77
Sơ đồ 3.37 Sơ đồ liên kết giao diện quản trị viên	77
Hình 3.1 Giao diện trang đăng ký	80
Hình 3.2 Giao diện đăng nhập.....	80
Hình 3.3 Giao diện trang chủ	81
Hình 3.4 Giao diện trang chủ đề.....	81
Hình 3.5 Giao diện trang giỏ hàng	82
Hình 3.6 Giao diện trang quản lý đơn hàng	82
Hình 3.7 Giao diện trang chat.....	83
Hình 3.8 Giao diện trang thông tin cá nhân	83
Hình 3.9 Giao diện trang đổi mật khẩu	84
Hình 3.10 Giao diện trang chi tiết sản phẩm	84

Hình 3.11 Giao diện trang đánh giá sản phẩm	85
Hình 3.12 Giao diện trang đăng nhập admin.....	85
Hình 3.13 Giao diện trang thống kê	86
Hình 3.14 Giao diện trang quản lý khách hàng	86
Hình 3.15 Giao diện trang quản lý danh mục sản phẩm	87
Hình 3.16 Giao diện trang quản lý sản phẩm	87
Hình 3.17 Giao diện trang quản lý đơn hàng	88
Hình 3.18 Giao diện trang chat.....	88
Hình 3.19 Giao diện trang Kho	89
Hình 3.20 Giao diện trang xuất kho	89
Hình 3.21 Giao diện trang Nhập kho	90
Hình 3.22 Giao diện trang lịch sử kho	90
Hình 3.23 Giao diện trang quản lý kho	91
Hình 3.24 Giao diện trang quản lý kho	91

DANH MỤC BẢNG

Bảng 3.1 Bảng mô tả ký hiệu UML	26
Bảng 3.2 Bảng mô tả Ký hiệu UML	26
Bảng 3.3 Kịch bản đăng ký	29
Bảng 3.4 Kịch bản đăng nhập.....	30
Bảng 3.5 Kịch bản đăng xuất	32
Bảng 3.6 Kịch bản UC tìm kiếm sản phẩm.....	33
Bảng 3.7 Kịch bản UC chi tiết sản phẩm	35
Bảng 3.8 Kịch bản UC quản lý danh sách yêu thích.....	36
Bảng 3.9 Kịch bản UC quản lý giỏ hàng.....	37
Bảng 3.10 Kịch bản UC thanh toán.....	39
Bảng 3.11 Kịch bản UC quản lý thông tin cá nhân.....	40
Bảng 3.12 Kịch bản UC quản lý đơn hàng.....	41
Bảng 3.13 Kịch bản UC quản lý danh mục	43
Bảng 3.14 Kịch bản UC quản lý sản phẩm	44
Bảng 3.15 Kịch bản UC thống kê.....	45
Bảng 3.16 Kịch bản UC quản lý đơn hàng.....	47
Bảng 3.17 Kịch bản quản lý khuyến mãi	48
Bảng 3.18 Kịch bản UC quản lý kho hàng.....	50
Bảng 3.19 Mô tả chi tiết thuộc tính và chi tiết	71
Bảng 3.20 Thuộc tính bảng tương tác người dùng.....	72
Bảng 3.21 Bảng thang điểm chuẩn hóa	73
Bảng 3.22 Bảng mô tả luồng sự kiện mô hình	74
Bảng 3.23 Danh sách màn hình của ứng dụng	79

DANH MỤC VIẾT TẮT

Kí hiệu viết tắt	Ý nghĩa
UC	Use case
AI	Artificial Intelligence (trí tuệ nhân tạo)
UI	User Interface (giao diện người dùng)
API	Application Programming Interface
MF	Matrix Factorization
MLP	Multilayer Perceptron

Mở Đầu

Trong thời đại công nghệ số phát triển mạnh mẽ, nhu cầu mua sắm trực tuyến ngày càng trở nên phổ biến và thiết yếu trong đời sống hàng ngày. Người tiêu dùng có xu hướng tìm kiếm các giải pháp mua sắm nhanh chóng, tiện lợi và an toàn ngay trên thiết bị di động. Tuy nhiên, hầu hết các ứng dụng thương mại điện tử hiện nay tập trung vào nhiều lĩnh vực khác nhau, chưa có nền tảng nào chuyên biệt cho mảng đồ gia dụng – một nhóm sản phẩm gắn liền với sinh hoạt của mọi gia đình.

Từ thực tế đó, đề tài “Xây dựng ứng dụng mua sắm đồ gia dụng dành cho Android” được thực hiện nhằm tạo ra một ứng dụng đơn giản, thân thiện và tiện lợi, giúp người dùng dễ dàng tìm kiếm, chọn mua và quản lý sản phẩm gia dụng ngay trên điện thoại.

Cơ sở khoa học của đề tài dựa trên các kiến thức về phát triển ứng dụng di động Android bằng ngôn ngữ Kotlin, kiến trúc client–server, cơ sở dữ liệu quan hệ MySQL.

Tính thực tiễn của đề tài thể hiện ở việc đáp ứng nhu cầu ngày càng tăng của người tiêu dùng về mua sắm tiện lợi và tiết kiệm thời gian, đồng thời tạo điều kiện cho các nhà bán lẻ đồ gia dụng mở rộng kênh phân phối trực tuyến, phù hợp với xu hướng chuyển đổi số hiện nay.

Mục tiêu của đề tài là xây dựng được một ứng dụng mẫu có các chức năng chính như:

- Đăng ký, đăng nhập (qua Google hoặc tài khoản riêng).
- Hiện thị và tìm kiếm sản phẩm, lọc sản phẩm.
- Quản lý giỏ hàng và đơn hàng (xem trạng thái đơn hàng).
- Xem chi tiết sản phẩm và thông tin liên quan.
- Quản lý sản phẩm yêu thích.
- Quản lý thông tin người dùng.

Phương pháp nghiên cứu được áp dụng gồm:

- Phân tích – tổng hợp tài liệu về thương mại điện tử và lập trình Android.
- Phương pháp mô hình hóa (sử dụng sơ đồ Use Case, ERD, sơ đồ lớp).
- Phương pháp thử nghiệm để kiểm tra và đánh giá kết quả thực tế của ứng dụng.

Đối tượng nghiên cứu của đề tài là người dùng ứng dụng di động có nhu cầu mua sắm đồ gia dụng.

Phạm vi nghiên cứu tập trung vào phát triển ứng dụng Android, với cơ sở dữ liệu lưu trữ người dùng, sản phẩm và đơn hàng; chưa bao gồm thanh toán trực tuyến hoặc vận chuyển thực tế.

Đề tài được tự tìm hiểu và nghiên cứu, kết hợp giữa lý thuyết đã học và thực hành phát triển phần mềm.

Báo cáo gồm các nội dung chính:

Chương 1: Giới thiệu tổng quan về các cơ sở lý thuyết, kỹ thuật liên quan tới đề tài như trình bày tóm tắt cơ sở lý thuyết sẽ sử dụng để giải quyết vấn đề như thuật toán, trình bày tóm tắt các kỹ thuật sẽ sử dụng để giải quyết vấn đề như ngôn ngữ, phần mềm sử dụng, framework, công cụ, phần cứng, v.v...

Chương 2: Nội dung của chương này thể hiện các thông tin về các yêu cầu kỹ thuật, xây dựng danh sách các yêu cầu kỹ thuật mà sản phẩm cần đạt được. Tổng thể về thiết kế hệ thống của sản phẩm: Các thành phần chính của sản phẩm, những mối liên hệ của các thành phần này như thế nào. Sự lựa chọn thiết kế và công nghệ để giải quyết vấn đề của đề tài: đưa ra các công nghệ, và phương án thiết kế được thực hiện có thể giải quyết được bài toán. Sau đó nêu rõ vì cách thức chọn phương án thiết kế và công nghệ để thực nghiệm cho vấn đề hay bài toán trong đề tài đang thực hiện đưa ra. Những thách thức về thuật toán để xây dựng các chức năng theo yêu cầu của sản phẩm: cần mô tả rõ về thuật toán đã có để giải quyết các vấn đề, những khó khăn của các thuật toán đó và có thể áp dụng các giải pháp thay thế nào khác.

Chương 3: Chương này mô tả chi tiết quá trình thực hiện, các thiết kế, các giải pháp để thực hiện phương án. Mô tả các kết quả đạt được.

Chương 4: Tổng kết các kết quả đạt được từ việc xây dựng ứng dụng, đề xuất các khả năng phát triển và cải tiến tiềm năng cho ứng dụng cũng như hệ thống quản lý sản phẩm trong tương lai.

CHƯƠNG 1: TỔNG QUAN TÀI LIỆU

1.1 Cơ sở lý thuyết

1.1.1 Thuật toán

Trong quá trình xây dựng và phát triển ứng dụng mua sắm đồ gia dụng dành cho Android, đề tài sử dụng một số thuật toán và phương pháp xử lý dữ liệu nhằm tối ưu hóa trải nghiệm người dùng, tăng độ chính xác trong gợi ý sản phẩm và đảm bảo hiệu năng của hệ thống.

a. Thuật toán sắp xếp

Ứng dụng cho phép người dùng tìm kiếm sản phẩm theo tên, loại sản phẩm hoặc từ khóa liên quan. Để thực hiện chức năng này, đề tài áp dụng thuật toán tìm kiếm chuỗi (String Matching Algorithm), cụ thể là:

- Thuật toán Tìm kiếm Tuyến tính (Linear Search): Duyệt qua danh sách sản phẩm, so sánh chuỗi ký tự nhập vào với tên sản phẩm trong cơ sở dữ liệu.
- Thuật toán này có độ phức tạp $O(n)$ và phù hợp với quy mô dữ liệu vừa phải (vài nghìn sản phẩm).
- Cải tiến: Sử dụng các kỹ thuật tìm kiếm gần đúng (Fuzzy Search) hoặc so khớp không phân biệt hoa/thường để tăng tính linh hoạt khi người dùng nhập từ khóa sai chính tả hoặc thiếu ký tự.

b. Thuật toán gợi ý sản phẩm

Để nâng cao trải nghiệm mua sắm và kích thích hành vi mua hàng, ứng dụng sử dụng thuật toán gợi ý sản phẩm tương tự (Content-Based Filtering), và thuật toán Neural Collaborative Filtering (NCF). (Nguyễn Văn Lộc, 2021; He et al., 2017).

c. Thuật toán sắp xếp sản phẩm

Để hiển thị danh sách sản phẩm theo thứ tự giá tăng dần, giảm dần, hoặc theo mức độ phổ biến, hệ thống sử dụng các thuật toán sắp xếp cơ bản:

- Thuật toán Quick Sort – Sắp xếp nhanh, hiệu quả với danh sách lớn (độ phức tạp trung bình $O(n \log n)$).
- Thuật toán Bubble Sort hoặc Insertion Sort – Áp dụng trong các trường hợp danh sách nhỏ hoặc khi cần xử lý tại client-side (React Native).

d. Thuật toán lọc dữ liệu

Ứng dụng cho phép người dùng lọc sản phẩm theo giá, đánh giá, dịch vụ khuyến mãi,...

e. Thuật toán xác thực và mã hóa dữ liệu

Để đảm bảo an toàn thông tin người sử dụng. Ứng dụng sử dụng các thuật toán bảo mật phổ biến:

- Thuật toán băm (Hash Function): Như SHA-256 để mã hóa mật khẩu người dùng khi lưu vào cơ sở dữ liệu.
- Google OAuth 2.0: Sử dụng để xác thực người dùng thông qua đăng nhập bằng google .

f. Thuật toán đề xuất sản phẩm

Hệ thống đề xuất (Recommender System) là một trong những thành phần quan trọng đối với các nền tảng thương mại điện tử hiện đại. Vai trò của hệ thống đề xuất là phân tích hành vi người dùng, xác định nhu cầu tiềm ẩn và gợi ý các sản phẩm phù hợp nhằm tăng trải nghiệm mua sắm, cải thiện tỷ lệ chuyển đổi (conversion rate) và tối ưu doanh thu.

Trong đồ án, mô hình NCF – Neural Collaborative Filtering được sử dụng để dự đoán mức độ yêu thích của khách hàng đối với từng sản phẩm dựa trên dữ liệu tương tác lịch sử. Đây là một mô hình dựa trên học sâu (Deep Learning), khắc phục nhiều hạn chế của phương pháp lọc cộng tác truyền thống.

g. Thuật toán chuẩn hóa dữ liệu(Data Normalization)

Trong hệ thống thương mại điện tử, dữ liệu đầu vào (giá sản phẩm, điểm đánh giá, số lượt mua, điểm tương tác) có thang đo khác nhau. Để đảm bảo mô hình học máy hoạt động hiệu quả, dữ liệu được chuẩn hóa trước khi huấn luyện.

Chuẩn hóa dữ liệu

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

Trong đó:

x : Giá trị ban đầu của dữ liệu

x_{min} :Giá trị nhỏ nhất của dữ liệu

x_{max} :Giá trị lớn nhất của dữ liệu

\hat{x} :Giá trị sau khi chuẩn hóa nằm trong khoảng (0,1]

h. Thuật toán giảm chiều dữ liệu (Embedding & Dimensionality Reduction)

Để giảm kích thước dữ liệu và tăng hiệu suất tính toán, hệ thống sử dụng Embedding Vector.

Embedding người dùng và sản phẩm:

$$\text{User} \rightarrow p_u \in R^k$$

$$\text{Product} \rightarrow q_i \in R^k$$

Với $K=32$

Giúp mô hình học được mối quan hệ tiềm ẩn giữa người dùng và sản phẩm

1.1.2 Kỹ thuật sử dụng

a. Tìm hiểu về ngôn ngữ kotlin

Kotlin là một ngôn ngữ lập trình hiện đại, được phát triển bởi JetBrains và ra mắt lần đầu tiên vào năm 2011. Kotlin được thiết kế để chạy trên nền tảng Java Virtual Machine (JVM), đồng thời cũng có thể được biên dịch sang mã JavaScript hoặc mã gốc (Native code) để phát triển ứng dụng đa nền tảng. Vào năm 2017, Google chính thức công nhận Kotlin là ngôn ngữ lập trình được hỗ trợ cho phát triển ứng dụng Android, bên cạnh Java. Kotlin được tạo ra nhằm khắc phục những hạn chế của Java, mang lại một cú pháp ngắn gọn, an toàn và dễ đọc hơn, giúp lập trình viên viết mã hiệu quả và ít lỗi hơn. Kotlin hoàn toàn tương thích với Java, nghĩa là lập trình viên có thể sử dụng mã Java trong dự án Kotlin hoặc ngược lại mà không gặp trở ngại.

Ưu điểm:

- Tương thích hoàn toàn với Java: Kotlin có thể sử dụng chung thư viện, framework và mã nguồn Java sẵn có. Điều này giúp việc chuyển đổi từ Java sang Kotlin diễn ra dễ dàng.
- Cú pháp ngắn gọn, dễ đọc: Kotlin giảm thiểu mã lặp, giúp lập trình viên viết ít hơn nhưng đạt cùng hiệu quả, từ đó tăng tốc độ phát triển.

- An toàn về Null (Null Safety): Kotlin được thiết kế để giảm thiểu lỗi null pointer – một lỗi phổ biến trong Java.
- Hỗ trợ lập trình hướng đối tượng và lập trình hàm: Kotlin kết hợp cả hai mô hình này, giúp linh hoạt hơn trong phát triển phần mềm.
- Dễ bảo trì và mở rộng: Mã nguồn rõ ràng và có cấu trúc tốt, thuận tiện cho việc mở rộng dự án hoặc làm việc nhóm.
- Được Google hỗ trợ chính thức: Là ngôn ngữ ưu tiên cho lập trình Android, có hệ sinh thái, công cụ và tài liệu phong phú.

Nhược điểm

- Hiệu năng biên dịch chậm hơn java: Mặc dù Kotlin chạy trên nền JVM giống Java, nhưng thời gian biên dịch (compiler) của Kotlin thường chậm hơn một chút, đặc biệt là đối với các dự án lớn. Điều này có thể ảnh hưởng đến năng suất khi lập trình viên phải biên dịch thường xuyên trong quá trình phát triển.
- Kích thước tập APK lớn: Khi phát triển ứng dụng Android bằng Kotlin, tệp APK (ứng dụng cài đặt) thường có kích thước lớn hơn một chút so với khi viết bằng Java, do phải tích hợp thêm thư viện Kotlin runtime.

b. Tìm hiểu về Reactjs

ReactJS là một thư viện JavaScript mã nguồn mở do Facebook (nay là Meta) phát triển, ra mắt lần đầu năm 2013. ReactJS được sử dụng để xây dựng giao diện người dùng (UI) cho các ứng dụng web.

React cho phép lập trình viên tạo ra các thành phần (component) có thể tái sử dụng, giúp quá trình phát triển nhanh chóng, dễ bảo trì và mở rộng.

Ưu điểm:

- Hiệu năng cao: Nhờ cơ chế Virtual DOM, React chỉ cập nhật phần giao diện thực sự thay đổi thay vì toàn bộ trang.
- Tái sử dụng component: Các thành phần giao diện có thể dùng lại nhiều nơi, giúp tiết kiệm thời gian và giảm lỗi.
- Cộng đồng lớn, tài liệu phong phú: React là một trong những thư viện front-end phổ biến nhất thế giới.

- Kết hợp linh hoạt: React có thể kết hợp với nhiều thư viện hoặc framework khác như Redux, Next.js, hoặc TailwindCSS.

- Hỗ trợ phát triển đa nền tảng: Với React Native, lập trình viên có thể dùng cùng cú pháp để phát triển ứng dụng di động (Android, iOS).

Nhược điểm

- Cần học thêm nhiều công cụ đi kèm (JSX, Redux, Router,...).
- Thay đổi nhanh: Các phiên bản và cách làm việc thường xuyên được cập nhật, khiến người mới khó theo kịp.
- Không phải là framework hoàn chỉnh: React chỉ tập trung vào UI, nên cần thêm các công cụ khác để xử lý routing, state management, API,...

c. Tìm hiểu về Node.js + Express

Node.js là một môi trường chạy mã JavaScript trên phía máy chủ (server-side), được xây dựng dựa trên V8 Engine của Google Chrome. Nó cho phép lập trình viên dùng JavaScript để viết ứng dụng backend, thay vì chỉ dùng cho frontend.



Hình 1.1: Logo Node.js + Express

Ưu điểm:

- Hiệu năng cao: Node.js sử dụng mô hình non-blocking I/O (xử lý bất đồng bộ), rất thích hợp cho các ứng dụng thời gian thực.
- Dễ học: Sử dụng cùng ngôn ngữ JavaScript cho cả frontend và backend.
- Thư viện phong phú: Cộng đồng mạnh mẽ, hàng nghìn module miễn phí qua npm.

- Express.js nhẹ và linh hoạt: Giúp thiết lập server, định tuyến (routing) và xử lý middleware một cách nhanh chóng.

- Phù hợp cho ứng dụng REST API, chat, game hoặc IoT.

Nhược điểm

- Không phù hợp cho xử lý tính toán nặng.

- Quản lý phụ thuộc (package) phức tạp khi dự án lớn.

d. Tìm hiểu về hệ quản trị cơ sở dữ liệu MySQL

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server. RDBMS là một phần mềm hay dịch vụ dùng để tạo và quản lý các cơ sở dữ liệu (Database) theo hình thức quản lý các mối liên hệ giữa chúng.



Hình 1.2: Logo MySQL

Ưu điểm:

- Miễn phí và mã nguồn mở.

- Hiệu năng ổn định, tốc độ cao khi làm việc với dữ liệu lớn.

- Hỗ trợ đa nền tảng: chạy được trên Windows, Linux, macOS,...

- Bảo mật tốt, dễ sao lưu và phục hồi dữ liệu.

- Tương thích với nhiều ngôn ngữ lập trình như PHP, Java, Python, Node.js,...

Nhược điểm

- Hạn chế trong xử lý giao dịch phức tạp.

- Khó mở rộng cho hệ thống cực lớn.

- Không hỗ trợ tốt cho NoSQL hoặc dữ liệu phi cấu trúc.

e. Tìm hiểu về công cụ hỗ trợ lập trình Visual Studio Code

Visual Studio Code (VS Code) là một trình soạn thảo mã nguồn mở do Microsoft phát triển, ra mắt vào năm 2015. VS Code được đánh giá cao bởi tính linh hoạt, hiệu năng nhẹ và khả năng mở rộng mạnh mẽ thông qua kho Extension phong phú. Đây là công cụ viết mã phổ biến nhất thế giới hiện nay, đặc biệt phù hợp cho lập trình web, Node.js, ReactJS và các công nghệ hiện đại.



Hình 1.3: Logo Visual Studio Code

Ưu điểm:

- Nhẹ, khởi động nhanh: VS Code tối ưu cho quá trình soạn thảo mã, không nặng như các IDE truyền thống.
- Kho mở rộng phong phú: Hỗ trợ hàng nghìn extension như Prettier, ESLint, GitLens, Live Server, MongoDB, MySQL...
- Tích hợp Git trực tiếp: Cho phép commit, push, pull mà không cần rời khỏi môi trường làm việc.
- Hỗ trợ đa ngôn ngữ: JavaScript, TypeScript, Python, Java, PHP, C#, Go,...
- Giao diện thân thiện, dễ tùy biến: Thay đổi theme, layout, phím tắt dễ dàng.
- Debugger tích hợp: Hỗ trợ debug Node.js, Python, Chrome Debug, React...

Nhược điểm:

- Không mạnh bằng IDE chuyên sâu cho từng ngôn ngữ (như Android Studio cho Android, PyCharm cho Python).
- Cần cài thêm extension để có đầy đủ tính năng, gây tốn thời gian cấu hình.
- Có thể tiêu tốn RAM khi mở nhiều extension.

f. Tìm hiểu về công cụ hỗ trợ lập trình Androi Studio

Android Studio là IDE chính thức do Google phát triển dành cho lập trình ứng dụng Android. Công cụ được xây dựng dựa trên IntelliJ IDEA và tích hợp đầy đủ các tính năng từ thiết kế giao diện đến quản lý project, build APK và kiểm thử.



Hình 1.4: Logo Androi Studio

Ưu điểm:

- Tích hợp đầy đủ công cụ cho Android: Layout Editor, AVD (Android Emulator), Logcat, Profiler.
- Hỗ trợ Kotlin và Java: Biên dịch tối ưu, gợi ý mã thông minh.
- Hệ thống Gradle mạnh mẽ: Kiểm soát build, phụ thuộc (dependencies), cấu hình đa môi trường.
- Thiết kế giao diện trực quan: Kéo thả component bằng Layout Editor.
- Tích hợp Firebase, Google Play Services: Dễ dàng thêm authentication, database, analytics,...

Nhược điểm:

- Nặng, tiêu thụ RAM cao: Thường yêu cầu máy tính cấu hình mạnh (RAM \geq 8GB).
- Thời gian build lâu: Đặc biệt với các project lớn.
- Khó dùng cho người mới: Do có nhiều công cụ và cấu hình phức tạp.

g. Tìm hiểu về công cụ hỗ trợ lập trình Pycharm

PyCharm là một IDE chuyên dụng dành cho ngôn ngữ Python, được phát triển bởi JetBrains. PyCharm cung cấp một môi trường mạnh mẽ để xây dựng, kiểm thử và tối ưu các mô hình học máy hoặc hệ thống backend sử dụng Python.



Hình 1.5: Logo Pycharm

Ưu điểm:

- Hỗ trợ Python chuyên sâu: Gợi ý mã thông minh, phân tích lỗi, quản lý thư viện ảo (Virtual Env).
- Điều hướng code tốt: Jump to definition, search, refactor nhanh.
- Tích hợp công cụ phân tích dữ liệu: Jupyter Notebook, SciPy, NumPy, Pandas.
- Debugger mạnh mẽ: Hỗ trợ debug từng dòng lệnh, theo dõi biến, biểu đồ bộ nhớ.
- Tích hợp với Git, Docker, Database.

Nhược điểm:

- Nặng hơn VS Code, cần cấu hình mạnh để chạy mượt.
- Bản miễn phí (Community) bị giới hạn chức năng so với bản Pro

h. Tìm hiểu về công cụ hỗ trợ kiểm thử postman

Postman là công cụ kiểm thử API phổ biến nhất hiện nay, hỗ trợ lập trình viên dễ dàng gửi yêu cầu (request) và xem phản hồi (response) từ server.



Hình 1.6: Logo Postman

Ưu điểm:

- Giao diện trực quan: Dễ dàng tạo GET, POST, PUT, DELETE...
- Hỗ trợ kiểm thử API RESTful: Gửi request kèm Body, Params, Headers, Authorization.
- Lưu trữ các bộ sưu tập API (Collection): Thuận tiện cho việc quản lý và kiểm thử nhiều API.
- Tự động hóa kiểm thử: Sử dụng Scripts để kiểm tra response hoặc chạy API theo chuỗi.
- Hỗ trợ nhiều định dạng: JSON, XML, Form-Data, x-www-form-urlencoded,...

Nhược điểm:

- Phiên bản miễn phí giới hạn số lượng Collection và Team workspace.
- Không phù hợp cho test UI hoặc kiểm thử nâng cao như Selenium.

i. Tìm hiểu về hệ thống quản lý git

Git là một hệ thống quản lý phiên bản phân tán (Distributed Version Control System — DVCS), được tạo bởi Linus Torvalds vào năm 2005. Git giúp lập trình viên theo dõi lịch sử thay đổi của mã nguồn, làm việc nhóm hiệu quả và quản lý các nhánh phát triển trong dự án.



Hình 1.7: Logo git

Khái niệm cơ bản:

- Repository (Repo): Kho lưu trữ mã nguồn, chứa toàn bộ lịch sử và các phiên bản.
- Commit: Một lần lưu thay đổi mã nguồn kèm chú thích.
- Branch: Nhánh phát triển độc lập, phục vụ thử nghiệm hoặc triển khai tính năng mới.
- Merge: Hợp nhất mã từ một nhánh vào nhánh khác.
- Clone: Sao chép toàn bộ repo từ server về máy cá nhân.
- Pull / Push: Lấy dữ liệu từ repo online về máy (pull), hoặc đẩy thay đổi từ máy lên repo (push).

Ưu điểm:

- Khá khó học đối với người mới: Quy trình commit, merge, resolve conflict cần rèn luyện.
- Xung đột merge (merge conflict) phức tạp: Với dự án lớn dễ xảy ra và khó xử lý.
- Quy trình làm việc team cần đồng nhất: Nếu không tuân thủ quy tắc Git Flow sẽ gây rối.

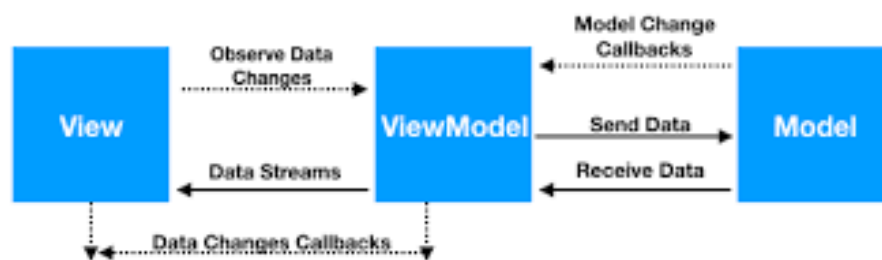
Nhược điểm:

- Khá khó học đối với người mới: Quy trình commit, merge, resolve conflict cần rèn luyện.
- Xung đột merge (merge conflict) phức tạp: Với dự án lớn dễ xảy ra và khó xử lý.

- Quy trình làm việc team cần đồng nhất: Nếu không tuân thủ quy tắc Git Flow sẽ gây rối.

j. Tìm hiểu về mô hình MVVM

MVVM (Model – View – ViewModel) là một mô hình kiến trúc phần mềm giúp tách biệt giao diện người dùng (UI) khỏi logic xử lý dữ liệu, được sử dụng phổ biến trong lập trình ứng dụng Android, iOS, và web hiện đại. (Nguyễn Văn Minh, 2021; Sommerville, 2018).



Hình 1.8: Nguyên lý hoạt động mô hình MVVM

Cấu trúc mô hình:

- Model: Là đối tượng cho phép bạn truy xuất dữ liệu và thao tác trên dữ liệu thật sự.
- View: View được hiểu là thành phần giao diện mô tả dữ liệu của một ứng dụng, đồng thời đây cũng là thành phần duy nhất cho phép người dùng tương tác trong chương trình. View trong MVVM được đánh giá là tích cực hơn so với các mô hình khác nhờ vào khả năng thực hiện hành vi và cung cấp phản hồi cho người dùng với một số tính năng nổi bật như Command, Binding,... (Lê Hoàng Long, 2023)
- ViewModel: Đóng vai trò trung gian và có nhiệm vụ đồng bộ hóa dữ liệu khi truyền tải từ Model lên View, đồng thời xử lý các hoạt động từ View để cập nhật Model. View sẽ được ánh xạ (binding) tới ViewModel, nhưng ViewModel lại không biết thông tin của View mà thông tin này sẽ được ẩn giấu bởi Data-binding cùng cơ chế hoạt động của mô hình Observer. Trong đó, một ViewModel có khả năng ánh xạ (binding) từ nhiều View.

Ưu điểm:

- Tách biệt rõ ràng giữa giao diện và xử lý dữ liệu, giúp dễ bảo trì và kiểm thử.
- Tái sử dụng được logic trong ViewModel.
- Hỗ trợ cập nhật giao diện tự động khi dữ liệu thay đổi (nhờ cơ chế data binding).
- Giảm thiểu lỗi do tương tác trực tiếp với UI.

Nhược điểm

- Cấu trúc ban đầu phức tạp hơn mô hình MVC.
- Cần hiểu rõ về data binding và vòng đời (Life Cycle) của View.
- Dễ gây nhầm lẫn nếu không quản lý tốt mối liên kết giữa View và ViewModel.

k. *Tìm hiểu về mô hình NCF-Neural Collaborative Filtering*

NCF (Neural Collaborative Filtering) là một mô hình học sâu (Deep Learning) được sử dụng trong các hệ thống gợi ý (Recommender System). Mô hình được phát triển nhằm khắc phục những hạn chế của phương pháp lọc cộng tác truyền thống (Collaborative Filtering), đặc biệt là mô hình phân rã ma trận (Matrix Factorization – MF), vốn chỉ mô hình hóa mối quan hệ tuyến tính giữa người dùng và sản phẩm. (He et al., 2017)

NCF sử dụng mạng nơ-ron nhiều lớp (Multilayer Perceptron – MLP) để học và mô tả các mối quan hệ phi tuyến, nhờ đó mang lại độ chính xác cao hơn trong các bài toán đề xuất sản phẩm, đặc biệt với dữ liệu đa dạng và phức tạp như hành vi người dùng trong ứng dụng mua sắm.

Nguyên lý hoạt động:

Trong mô hình NCF, mỗi người dùng (user) và sản phẩm (item) được biểu diễn dưới dạng một vector đặc trưng, gọi là embedding. Thay vì dùng nhân dot product tuyến tính như MF, mô hình NCF kết hợp embeddings của user và item rồi đưa qua hệ thống MLP để học mối quan hệ phi tuyến.

$$\tilde{y}_{ui} = f(MLP(Emb_u || Emb_i))$$

Trong đó:

\tilde{y}_{ui} : Điểm dự đoán mức độ yêu thích của người dùng đối với sản phẩm

Emb_u : Vector embedding của người dùng u

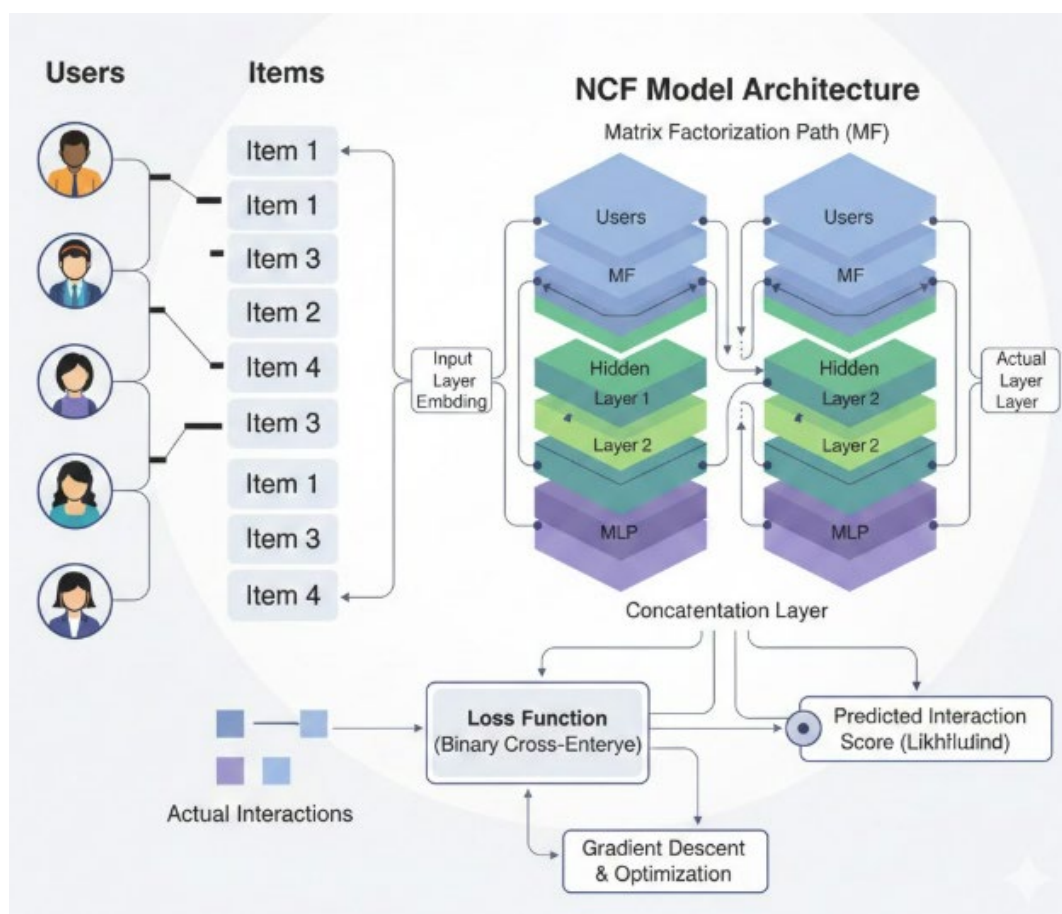
Emb_i : Vector embedding của sản phẩm i

$\|$: Phép nối của hai vector

MLP: Mạng neural gồm nhiều lớp ẩn

f : Hàm kích hoạt ở đầu ra

Nhờ tận dụng mạng neural mà mô hình có thể học được các đặc trưng phức tạp mà MF không thể biểu diễn.



Hình 1.8: Nguyên lý hoạt động của NCF

Cấu trúc mô hình: Gồm 3 phần chính

- Embedding Layer (Lớp biểu diễn)
- Thay thế userId và productId bằng vector liên tục (continuous vector).
- Tự động học các đặc điểm ẩn của người dùng và sản phẩm.
- MLP Layer (Mạng neural phi tuyến)

- Ghép hai embedding lại và đưa qua nhiều lớp Dense
- Mỗi lớp có hàm kích hoạt như ReLU
- Cho phép mô hình hóa nhiều dạng quan hệ phức tạp giữa user và item
- Output Layer (Lớp dự đoán)
- Sử dụng sigmoid để tạo ra xác suất trong khoảng $[0,1]$
- Giá trị càng cao \rightarrow mức độ phù hợp càng lớn \rightarrow ưu tiên gợi ý

Ưu điểm:

- Dự đoán chính xác hơn so với MF nhờ mô hình phi tuyến.
- Học được hành vi phức tạp của người dùng, ví dụ: xem – thêm giỏ – yêu thích – đánh giá – mua hàng.
- Linh hoạt, có thể kết hợp thêm đặc trưng phụ như độ tuổi, giá sản phẩm, danh mục,...
- Phù hợp cho ứng dụng thương mại điện tử, nơi hành vi khách hàng đa dạng và không tuyến tính.

Nhược điểm:

- Tốn tài nguyên tính toán hơn MF.
- Cần nhiều dữ liệu tương tác để mô hình học tốt.
- Việc tối ưu kiến trúc network (số lớp, số neuron) đòi hỏi kỹ năng kinh nghiệm.

Chương 2: PHƯƠNG PHÁP THỰC HIỆN

2.1 Yêu cầu kỹ thuật

2.1.1 Yêu cầu chức năng

a. Chức năng đăng nhập/đăng ký

Cho phép người dùng đăng nhập bằng tài khoản Google hoặc tài khoản riêng (username & password).

Xác thực bằng JWT và lưu trữ thông tin trong cơ sở dữ liệu MySQL (mã hóa mật khẩu).

b. Quản lý sản phẩm

Hiển thị danh mục và chi tiết sản phẩm (hình ảnh, giá, thông tin kỹ thuật, đánh giá).

Cho phép tìm kiếm, lọc theo giá, thương hiệu hoặc loại sản phẩm.

c. Giỏ hàng và đặt hàng

Người dùng có thể thêm, cập nhật, hoặc xóa sản phẩm trong giỏ hàng.

Hệ thống tính toán tổng giá trị đơn hàng, phí vận chuyển và xử lý đặt hàng.

d. Quản lý đơn hàng

Người dùng có thể xem lịch sử và trạng thái đơn hàng.

Cho phép người dùng hủy hoặc xác nhận đơn hàng.

e. Gợi ý sản phẩm

Ứng dụng đề xuất sản phẩm tương tự dựa trên lịch sử mua hoặc hành vi tìm kiếm.

f. Thanh toán

Người có thể chọn phương thức thanh toán khác nhau như: Momo, VNpay, Visa,..

g. Quản lý thông tin tài sản

Người dùng thêm thông tin .

Người dùng có thể thay đổi thông tin.

h. Tìm kiếm, lọc sản phẩm

Người dùng có thể tìm kiếm sản phẩm và lọc sản phẩm theo: khu vực giao, phương thức giao hàng (hỏa tốc, nhanh,..), đánh giá,.. sao cho phù hợp nhất với nhu cầu của mình.

i. Đánh giá sản phẩm

Người dùng có thể đánh giá bằng hình ảnh, văn bản hay video đối với các sản phẩm đã mua.

k. Hệ thống quản trị

Admin có thể quản lý sản phẩm, đơn hàng, người dùng, hình ảnh sản phẩm và danh mục.

Thông kê doanh thu và theo dõi lượng bán theo thời gian.

2.1.2 Yêu cầu hiệu suất

Tốc độ phản hồi nhanh: Ứng dụng phải đáp ứng yêu cầu tốc độ, đảm bảo các thao tác như tải danh mục sản phẩm, tìm kiếm sản phẩm, thanh toán... diễn ra mượt mà và nhanh chóng.

Khả năng hoạt động trên cả Android và iOS: Do sử dụng Kotlin, ứng dụng cần phải tối ưu hóa cho cả hai hệ điều hành này.

2.1.3 Yêu cầu bảo mật*a. Bảo mật người dùng*

Mật khẩu được mã hóa bằng thuật toán băm (bcrypt) trước khi lưu vào MySQL.

b. Bảo mật kết nối

Toàn bộ giao tiếp giữa ứng dụng và máy chủ được mã hóa bằng giao thức HTTPS (SSL/TLS).

Sử dụng Token xác thực (JWT – JSON Web Token) cho các API riêng tư.

c. Phân quyền người dùng

Hệ thống phân biệt quyền truy cập giữa người dùng thường và quản trị viên (Admin).

Admin có quyền cao nhất trong việc quản lý sản phẩm và đơn hàng.

d. Bảo mật dữ liệu giao dịch

Thông tin đơn hàng, lịch sử mua sắm và đánh giá được kiểm soát chặt chẽ, tránh rò rỉ dữ liệu.

Mỗi thao tác thêm, sửa, xóa dữ liệu đều được ghi log lại trên server.

e. Chống các tấn công phổ biến

Ngăn chặn SQL Injection bằng cách sử dụng câu truy vấn tham số hóa (Prepared Statements).

Chống Cross-Site Scripting (XSS) trên giao diện web bằng ReactJS.

2.2 Tổng thể về thiết kế hệ thống

Hệ thống sẽ được xây dựng theo kiến trúc Client-Server và sử dụng mô hình 3 tầng.

2.2.1 Tầng Presentation (giao diện)

Đây là tầng hiển thị và tương tác trực tiếp với người dùng, đóng vai trò quan trọng trong việc tạo ra trải nghiệm thân thiện, dễ sử dụng. Tầng này bao gồm hai phần chính: ứng dụng di động cho khách hàng và trang web quản trị (Web Admin).

a. Ứng dụng di động (Khách hàng)

Công nghệ sử dụng: Phát triển bằng kotlin trên nền tảng Android studio

Chức năng chính:

- Trang đăng nhập/đăng ký: Cho phép người tạo tài khoản và đăng nhập sử dụng ứng dụng.
- Trang chủ: Cung cấp danh mục sản phẩm, bộ lọc, tìm kiếm và đề xuất sản phẩm.
- Trang chi tiết sản phẩm: Hiển thị thông tin sản phẩm, hình ảnh, giá, đánh giá, bình luận của sản phẩm.
- Trang giỏ hàng và thanh toán: Cho phép người dùng thêm/xóa sản phẩm, đặt hàng và thanh toán sản phẩm.
- Trang lịch sử mua hàng: Người dùng có thể coi đơn hàng đã mua và trạng thái đơn hàng.
- Trang chat: Cho phép người nhắn tin với người bán.
- Trang thông tin cá nhân: Người dùng có thể thêm xóa sửa thông tin cá nhân.

b. Trang quản trị (Admin)

Công nghệ sử dụng: Xây dựng bằng reactjs, giúp giao diện mượt mà.

Chức năng chính:

- Quản lý sản phẩm: Thêm, xóa, sửa thông tin sản phẩm, hình ảnh và giá bán.

- Quản lý người dùng: Theo dõi số lượng người dùng, khóa hoặc mở quyền truy cập của người dùng.
- Quản lý đơn hàng: Xem danh sách đơn hàng, xác nhận, cập nhật trạng thái đơn hàng.
- Thống kê doanh thu: Tổng hợp doanh số, sản phẩm bán chạy, lượng đơn hàng trong một quý.
- Quản lý khuyến mãi: Tạo và cập nhật chương trình khuyến mãi, giá ưu đãi.

2.2.2. Tầng Business logic (xử lý nghiệp vụ)

Công nghệ sử dụng: Node.js kết hợp Express.js, giúp xử lý yêu cầu từ client, thực hiện các nghiệp vụ, và trả kết quả cho tầng Presentation.

Chức năng chính:

- Xử lý đăng nhập/đăng ký: Kiểm tra thông tin người dùng, mã hóa mật khẩu, xác thực qua token.
- Quản lý sản phẩm: Xử lý các yêu cầu thêm, sửa, xóa, và truy xuất danh sách sản phẩm từ cơ sở dữ liệu.
- Xử lý giỏ hàng và thanh toán: Tính toán tổng tiền, áp dụng khuyến mãi, xác nhận thanh toán và tạo đơn hàng mới.
- Quản lý đơn hàng: Cập nhật trạng thái đơn hàng, thông báo cho khách hàng và admin.
- Quản lý người dùng: Kiểm tra quyền truy cập, phân loại người dùng (khách hàng, admin).
- Thống kê & báo cáo: Tổng hợp dữ liệu bán hàng, số lượng người dùng hoạt động, doanh thu, sản phẩm bán chạy.
- Xử lý chat: Gửi và nhận tin nhắn giữa khách hàng và người bán, đảm bảo dữ liệu thời gian thực

2.2.3 Tầng Data (dữ liệu)

Tầng Data chịu trách nhiệm lưu trữ, truy xuất và quản lý dữ liệu của hệ thống.

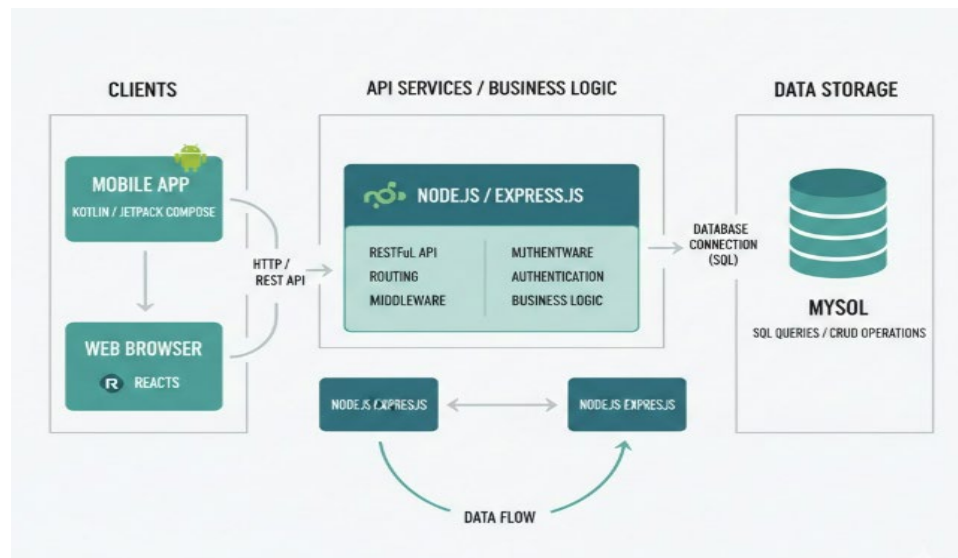
Công nghệ sử dụng: Cơ sở dữ liệu quan hệ: MySQL quản lý thông tin người dùng, sản phẩm, đơn hàng, khuyến mãi.

Chức năng chính:

- Lưu trữ thông tin người dùng: Tên, email, mật khẩu đã mã hóa, quyền truy cập, thông tin cá nhân.
- Lưu trữ sản phẩm: Thông tin sản phẩm, hình ảnh, giá, số lượng tồn kho.
- Lưu trữ đơn hàng: Chi tiết đơn hàng, trạng thái, lịch sử thanh toán, giao hàng.
- Lưu trữ chat: Tin nhắn, thời gian gửi, trạng thái đã đọc.
- Lưu trữ khuyến mãi: Chi tiết chương trình giảm giá, ngày bắt đầu/kết thúc, điều kiện áp dụng

Mối quan hệ giữa các thành phần:

- Kotlin sẽ gửi yêu cầu HTTP (Get,Post) đến node.js +Express.js .
- Node.js +Express.js nhận các yêu cầu, truy vấn cơ sở dữ liệu, xử lý logic và trả kết quả cho Kotlin dưới dạng JSON.
- Kotlin sẽ xử lý và hiển thị thông tin từ JSON cho người dùng.



Hình 2.1: Mô tả luồng dữ liệu trong hệ thống

2.3 Sự lựa chọn thiết kế và công nghệ

Kotlin (Frontend – Ứng dụng di động): Kotlin được lựa chọn làm công nghệ phát triển ứng dụng di động do khả năng hỗ trợ đa nền tảng thông qua Kotlin Multiplatform và Compose Multiplatform. Ngôn ngữ này cho phép chia sẻ phần lớn logic giữa hai nền tảng Android và iOS, đồng thời vẫn đảm bảo hiệu năng và trải nghiệm người dùng tương đương với ứng dụng native. Các thư viện như Ktor (giao tiếp API), Kotlinx Serialization

(xử lý dữ liệu), và Compose Multiplatform (xây dựng giao diện) giúp tối ưu hóa quy trình phát triển, nâng cao tính nhất quán và dễ dàng bảo trì hệ thống trong tương lai.

ReactJS (Frontend – Ứng dụng web): ReactJS được sử dụng để xây dựng giao diện người dùng trên nền tảng web nhờ khả năng linh hoạt, hiệu suất cao và cộng đồng hỗ trợ rộng lớn. Với cấu trúc hướng thành phần (component-based), ReactJS cho phép tái sử dụng mã nguồn, giảm thiểu thời gian phát triển và nâng cao khả năng mở rộng. Các công nghệ bổ trợ như Redux/Redux Toolkit (quản lý trạng thái), React Router (điều hướng), và Axios (giao tiếp API) giúp đảm bảo hệ thống frontend hoạt động ổn định, rõ ràng và dễ dàng mở rộng.

Node.js kết hợp Express.js (Backend): Backend được phát triển dựa trên Node.js kết hợp với Express.js nhằm tận dụng khả năng xử lý bất đồng bộ mạnh mẽ và hiệu năng cao của nền tảng này. Express.js cung cấp cơ chế routing linh hoạt, hệ thống middleware hiệu quả và khả năng xây dựng API RESTful một cách nhanh chóng. Kiến trúc backend được thiết kế rõ ràng, dễ mở rộng, đáp ứng tốt yêu cầu xử lý dữ liệu, phân quyền, xác thực và kết nối với cơ sở dữ liệu.

Cơ sở dữ liệu MySQL: MySQL được lựa chọn làm hệ quản trị cơ sở dữ liệu nhờ vào tính ổn định, hiệu năng cao và khả năng xử lý truy vấn mạnh mẽ. Hệ thống sử dụng MySQL để lưu trữ các dữ liệu quan trọng như thông tin người dùng, sản phẩm và dữ liệu nghiệp vụ. Cơ sở dữ liệu được kết nối với backend thông qua các thư viện như mysql2 hoặc Prisma ORM, đảm bảo khả năng truy xuất dữ liệu nhanh chóng, an toàn và dễ quản lý.

API RESTful: Đóng vai trò là cầu nối chính giữa frontend (Kotlin và ReactJS) và backend (Node.js/Express.js). (Phạm Trần Bích Thuận & cộng sự, 2023). Việc sử dụng chuẩn REST giúp các thành phần trong hệ thống giao tiếp với nhau một cách độc lập, rõ ràng và tiêu chuẩn hóa. Đồng thời, cấu trúc RESTful tạo điều kiện thuận lợi cho việc mở rộng chức năng, nâng cấp hoặc thay đổi từng phần trong hệ thống mà không ảnh hưởng đến các thành phần khác.

Chương 3: CÀI ĐẶT VÀ THỬ NGHIỆM

3.1 Phân tích hệ thống

3.1.1 Phía khách hàng

Tài khoản: Nếu khách hàng chưa có tài khoản, họ có thể tạo tài khoản mới với việc nhập tên, email và mật khẩu và nhấn vào nút tạo tài khoản.

- Nếu khách hàng đã có tài khoản, họ sẽ phải nhập email và mật khẩu đúng với tài khoản từng tạo trước đó và nhấn đăng nhập.
- Sau khi đăng nhập, khách hàng có thể thay đổi mật khẩu của mình.

Tìm kiếm: Khách hàng có nhu cầu tìm kiếm sản phẩm có thể nhập ký tự cụ thể vào thanh tìm kiếm với các lựa chọn tìm kiếm như:

- Tìm kiếm theo tên sản phẩm: Nhập vào tên sản phẩm khách hàng sẽ nhận về một danh sách tương ứng được tìm kiếm.
- Tìm kiếm theo giá cả: Chọn tìm kiếm theo giá cả, website sẽ hiển thị lựa chọn tìm theo giá từ cao đến thấp và giá từ thấp đến cao.

Xem sản phẩm: Khách hàng có thể xem sản phẩm hoặc xem chi tiết từng sản phẩm với hình ảnh và mô tả đầy đủ khi khách lướt xem hoặc tìm kiếm.

Đặt hàng: Sau khi xem và chọn sản phẩm, khách hàng có thể đưa sản phẩm vào giỏ hàng, thực hiện các thao tác tăng giảm số lượng sản phẩm trong giỏ hàng, thêm sửa xóa sản phẩm trong giỏ hàng và thanh toán.

Thanh toán: Khách hàng có thể thanh toán bằng 2 cách: một là thanh toán sau khi nhận được hàng, hai là thanh toán trực tuyến.

Đánh giá sản phẩm: Người dùng có thể gửi những ý kiến đánh giá, bình luận và cho điểm đối với sản phẩm đã mua.

3.1.1 Phía quản trị viên

Quản trị viên: Quản trị viên đăng nhập vào hệ thống bằng tài khoản riêng với quyền admin để thực hiện các chức năng quản lý toàn hệ thống. Sau khi đăng nhập, quản trị viên có thể truy cập vào bảng điều khiển quản trị với các chức năng chính như:

Quản lý người dùng: Xem danh sách người dùng, khóa/mở tài khoản, phân quyền vai trò (admin, customer, staff), xem lịch sử hoạt động của người dùng.

Quản lý sản phẩm: Thêm mới, chỉnh sửa, xóa sản phẩm; quản lý danh mục sản phẩm; cập nhật giá cả, mô tả, hình ảnh và trạng thái sản phẩm; thiết lập giảm giá theo sản phẩm hoặc theo danh mục.

Quản lý kho hàng: Theo dõi tồn kho theo từng kho, nhập/xuất hàng, điều chỉnh tồn kho, xem nhật ký thay đổi tồn kho và thiết lập mức tồn kho tối thiểu để cảnh báo.

Quản lý đơn hàng: Xem tất cả đơn hàng, cập nhật trạng thái đơn hàng (chờ xác nhận, đang giao, đã giao, đã hủy), in hóa đơn, hủy đơn hàng theo yêu cầu hoặc vi phạm.

Quản lý vận chuyển: Cập nhật thông tin vận chuyển, nhập mã theo dõi, thiết lập đơn vị vận chuyển và phí ship theo khu vực.

Quản lý thanh toán: Xác nhận thanh toán, xem lịch sử giao dịch, xử lý hoàn tiền khi có yêu cầu hoặc đơn hàng hủy.

Quản lý mã giảm giá: Tạo mã giảm giá mới, thiết lập điều kiện sử dụng, theo dõi số lần sử dụng và vô hiệu hóa mã khi cần.

Quản lý đánh giá: Duyệt hoặc xóa đánh giá không phù hợp, phản hồi bình luận của khách hàng, theo dõi điểm đánh giá trung bình của sản phẩm.




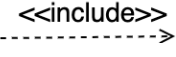
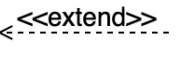
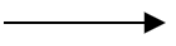
Quản lý thông báo: Gửi thông báo hệ thống đến người dùng, thiết lập thông báo tự động cho các sự kiện như đơn hàng mới, hàng sắp hết, khuyến mãi.

Hỗ trợ khách hàng: Trò chuyện trực tiếp với khách hàng qua chat hỗ trợ, xử lý khiếu nại, phản hồi yêu cầu hỗ trợ.

Báo cáo thống kê: Xem báo cáo doanh thu theo ngày/tuần/tháng, thống kê sản phẩm bán chạy, phân tích hành vi người dùng, xuất báo cáo dưới dạng Excel/PDF.

3.2 Thiết kế hệ thống

3.2.1 Mô tả ký hiệu UML

Ký hiệu	Mô tả
	Biểu diễn các tác nhân (Actor) \rightleftarrows Người hay hệ thống tương tác với hệ thống hiện tại
	Biểu diễn các Use Case (Ca sử dụng) biểu diễn hành động mà hệ thống thực hiện
	Biểu diễn mối quan hệ giữa Actor và Use Case
	Quan hệ <<include>> thể hiện bao gồm giữa hai ca sử dụng
	Quan hệ <<extend>> thể hiện quan hệ mở rộng giữa hai sử dụng
	Quan hệ khái quát hoá Actor hay Ca sử dụng

Bảng 3.1: Bảng mô tả ký hiệu UML

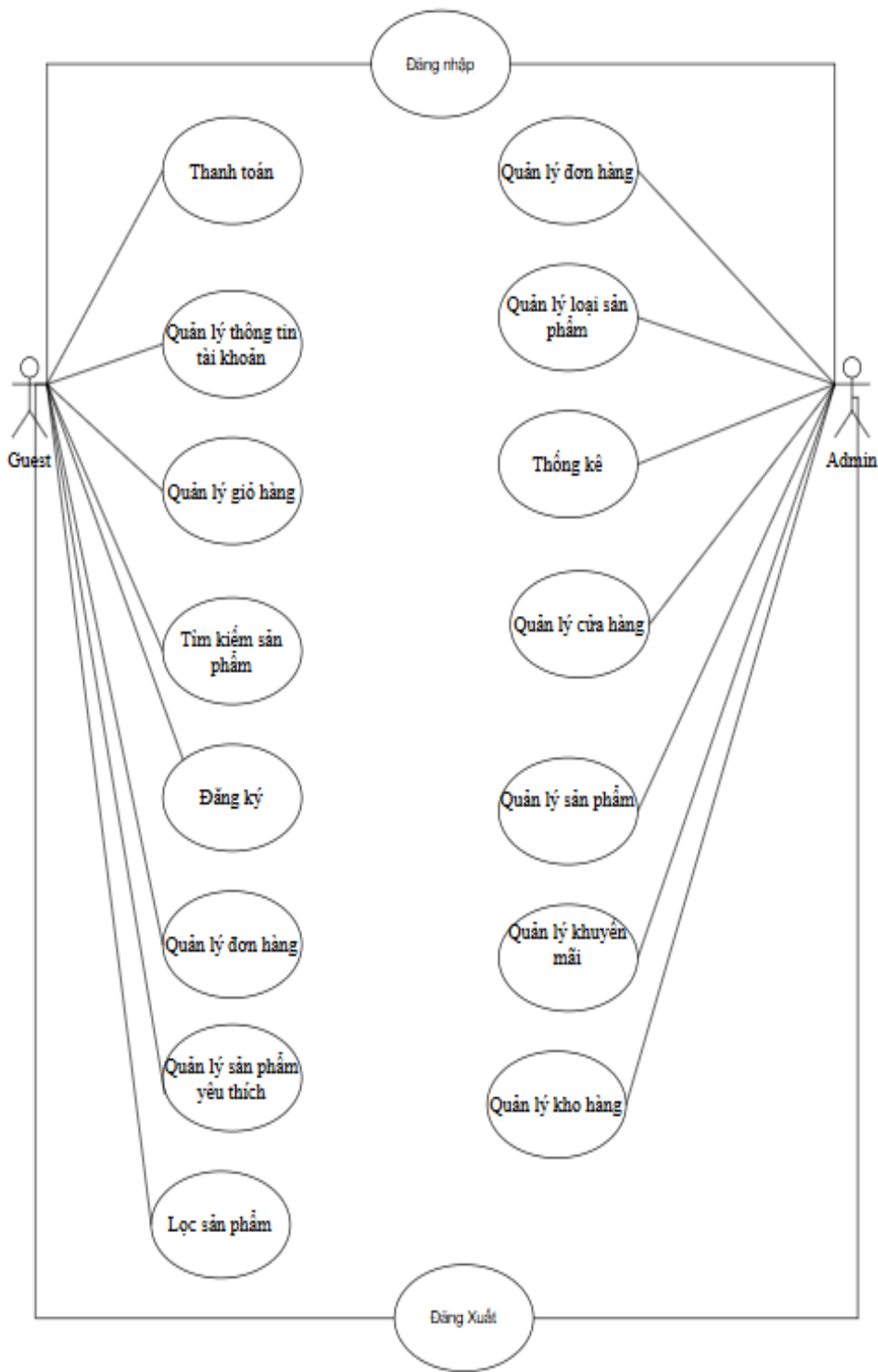
3.2.2 Đặc tả Actor

Tên Actor	Mô tả
Admin	Quản lý sản phẩm, danh mục sản phẩm, quản lý đơn hàng, quản lý khuyến mãi, thống kê
Guest	Mua hàng, quản lý giỏ hàng, tìm kiếm sản phẩm.

Bảng 3.2: Bảng mô tả Ký hiệu UML

3.2.3 Sơ đồ Use Case

a. Sơ đồ Use Case tổng quát

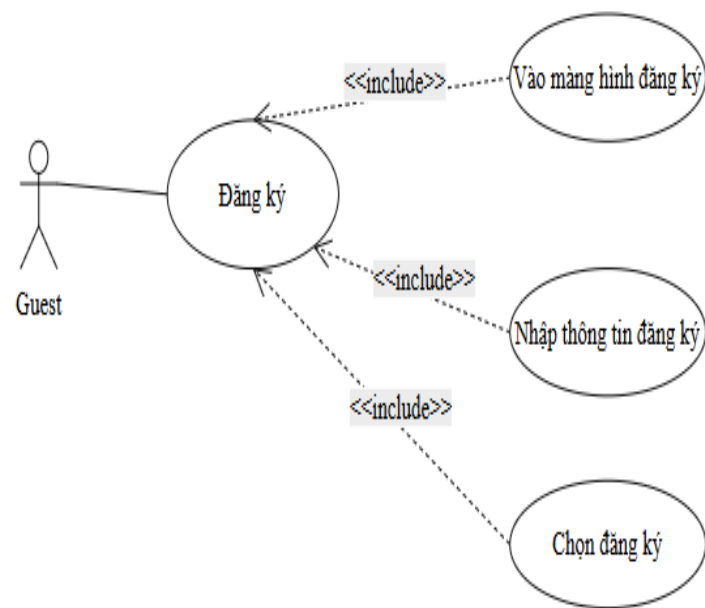


Sơ đồ 3.1: Sơ đồ usecase tổng quát

Biểu đồ Use Case tổng quát biểu diễn sơ đồ chức năng chính của hệ thống. Từ tập yêu cầu của hệ thống, biểu đồ Use case chỉ ra hệ thống cần thực hiện những điều gì để đáp ứng nhu cầu của người sử dụng hệ thống. Người quản trị (admin) thông qua đăng nhập để thực hiện tìm kiếm, quản lý, thống kê các sản phẩm, đơn hàng, khách hàng và chi tiết khác trên website. Khách hàng khi truy cập vào ứng dụng có thể thực hiện các chức năng: xem sản phẩm, tìm kiếm sản phẩm, đăng ký thành viên... và thông qua đăng nhập thành viên để đặt hàng, thay đổi thông tin cá nhân, xem thông tin đơn hàng. (Nguyễn Văn Ba, 2015; Nguyễn Văn Minh, 2021)

b. Sơ đồ Use Case chi tiết

Sơ đồ phân rã và kịch bản UC đăng ký



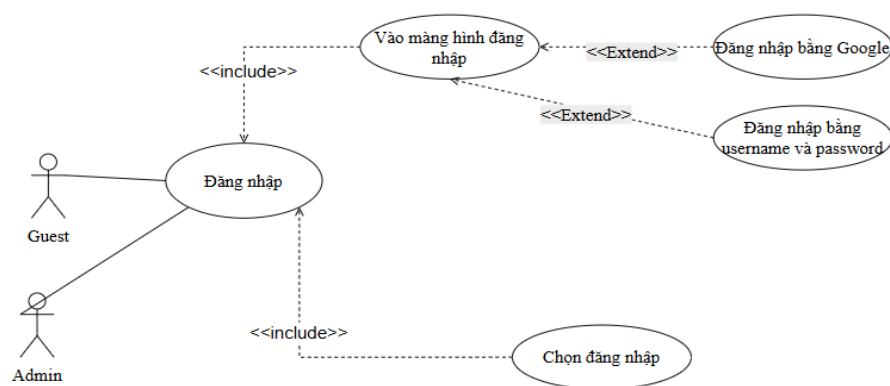
Sơ đồ 3.2: Sơ đồ phân rã chức năng đăng ký

Tên Use Case	Đăng ký
Mục tiêu	Cho phép khách (Guest) tạo tài khoản mới trong hệ thống
Tác nhân chính	Khách

Tiền điều kiện	Khách đã mở hệ thống
Hậu điều kiện	Khách đã tạo được tài khoản
<p>Luồng sự kiện chính:</p> <ul style="list-style-type: none"> - Khách hàng khởi chạy hệ thống. - Hiện màn hình Đăng nhập. - Khách nhấn nút đăng ký. - Hiện thị màn hình đăng ký. - Nhập thông tin đăng ký. - Hệ thống báo về đã đăng ký tài khoản thành công 	
<p>Luồng ngoại lệ:</p> <ul style="list-style-type: none"> - Khách nhập trùng username. Hệ thống trả về thông báo “Username đã tồn tại vui lòng đăng nhập” sau đó trở về bước - Mật khẩu và nhập lại mật khẩu không trùng nhau. Hệ thống thông báo “mật khẩu xác nhận không trùng khớp” sau đó quay lại bước 5. - User name có ký tự đặc biệt. Hệ thống trả về thông báo “Username có ký tự đặc biệt” sau đó trở về bước 5. - Mật khẩu không đủ ký tự. Hệ thống trả về thông báo “Mật khẩu không đủ 6 ký tự” sau đó trở về bước 5. 	

Bảng 3.3: Kịch bản đăng ký

c. Sơ đồ phân rã và kịch bản UC đăng nhập

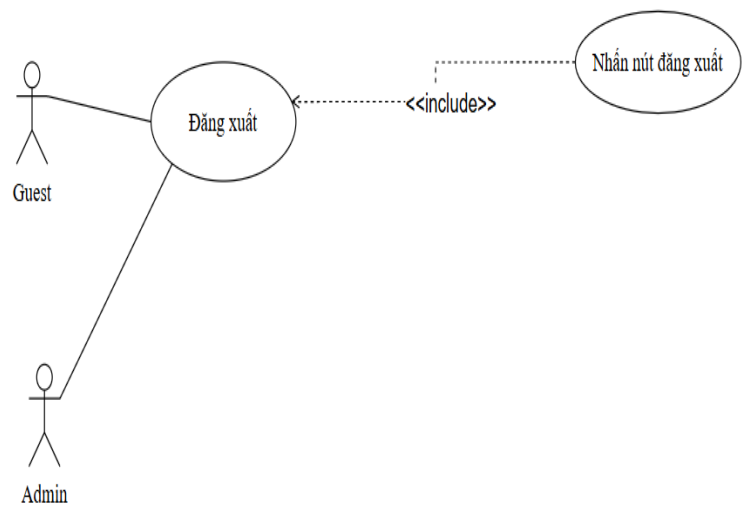


Sơ đồ 3.3: Sơ đồ phân rã UC đăng nhập

Tên Use Case	Đăng nhập
Mục tiêu	Cho phép Guest (Khách) hoặc Admin (Quản trị viên) xác thực danh tính để truy cập vào các tính năng của hệ thống.
Tác nhân chính	Khách, quản trị viên
Tiền điều kiện	Tác nhân đã mở hệ thống và có tài khoản
Hậu điều kiện	Tác nhân đã đăng nhập vào hệ thống chuyển đến trang chủ dành riêng cho vai trò của họ.
<p>Luồng sự kiện chính:</p> <ul style="list-style-type: none"> - Tác nhân khởi chạy hệ thống. - Hiện màn hình Đăng nhập. - Tác nhân nhập thông tin đăng nhập/ đăng nhập bằng Google. - Nhấn nút đăng nhập. - Hệ thống xác thực thông tin đăng nhập. - Hệ thống báo về đã đăng nhập. 	
<p>Luồng ngoại lệ:</p> <ul style="list-style-type: none"> - Tác nhân nhập sai thông tin đăng nhập. Hệ thống thông báo “Tên đăng nhập hoặc mật khẩu không đúng. Vui nhập lại”. Quay lại bước 3 - Tài khoản chưa được tạo. Hệ thống thông báo “Tài khoản chưa được tạo vui lòng đăng ký”. Quay lại bước 3. - Thiếu thông tin đăng nhập. Hệ thống thông báo “Vui lòng nhập đầy đủ tên đăng nhập/password”. - Hệ thống bị lỗi. Thông báo “ Hệ thống bị lỗi vui lòng quay lại sau.” 	

Bảng 3.4: Kịch bản đăng nhập

d. Sơ đồ phân rã và kịch bản UC đăng xuất



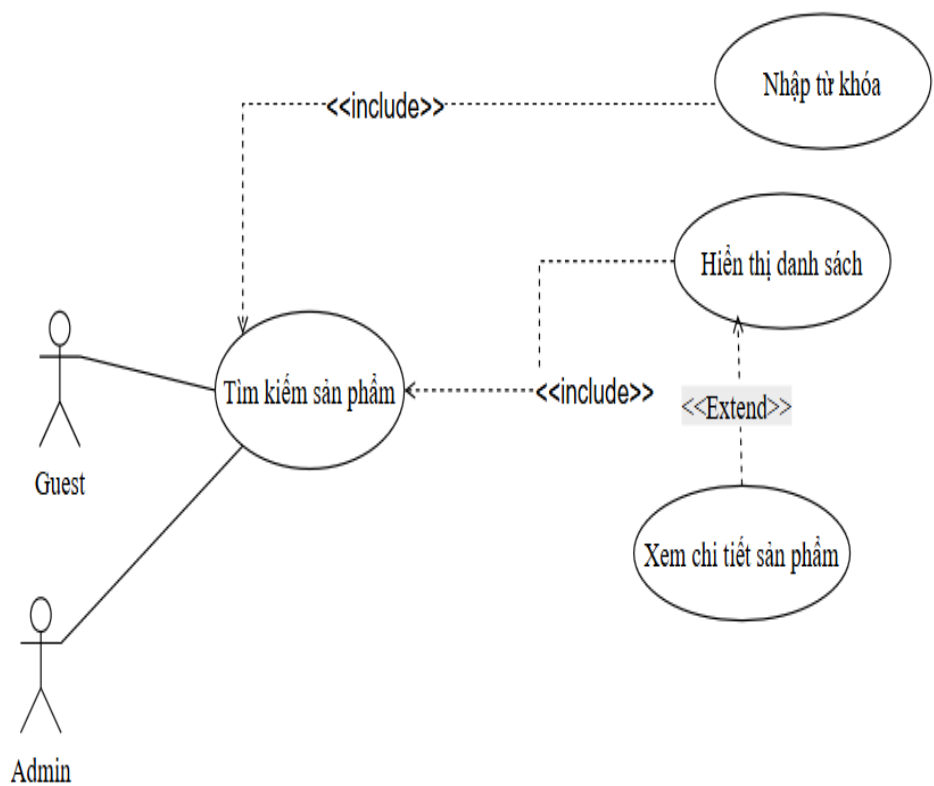
Sơ đồ 3.4: Sơ đồ phân rã UC đăng xuất

Tên Use Case	Đăng xuất
Mục tiêu	Cho phép Guest (Khách) hoặc Admin (Quản trị viên) thoát phiên làm việc với hệ thống.
Tác nhân chính	Khách, quản trị viên
Tiền điều kiện	Tác nhân đã mở hệ thống và đã đăng nhập.
Hậu điều kiện	Tác nhân đã thoát khỏi phiên làm việc với hệ thống và quay lại trang đăng nhập.
Luồng sự kiện chính:	
<div><ul style="list-style-type: none">- Tác nhân đã ở trong hệ thống. Ở trang profile- Nhấn nút đăng xuất.- Hệ thống xác thực hiện thông báo xác nhân đăng xuất.- Hệ thống báo về đã đăng xuất và trở về trang đăng nhập.</div>	

Luồng ngoại lệ: Nếu quá trình hủy bỏ phiên làm việc gặp lỗi kỹ thuật trên máy chủ (ví dụ: không thể xóa token), hệ thống không thể chuyển hướng hoặc vẫn giữ Tác nhân ở trạng thái đăng nhập. Hệ thống hiển thị thông báo: “Đăng xuất không thành công. Vui lòng thử lại.” (Và Tác nhân sẽ phải thử lại hoặc đóng trình duyệt)

Bảng 3.5: Kịch bản đăng xuất

e. Sơ đồ phân rã và kịch bản UC tìm kiếm sản phẩm

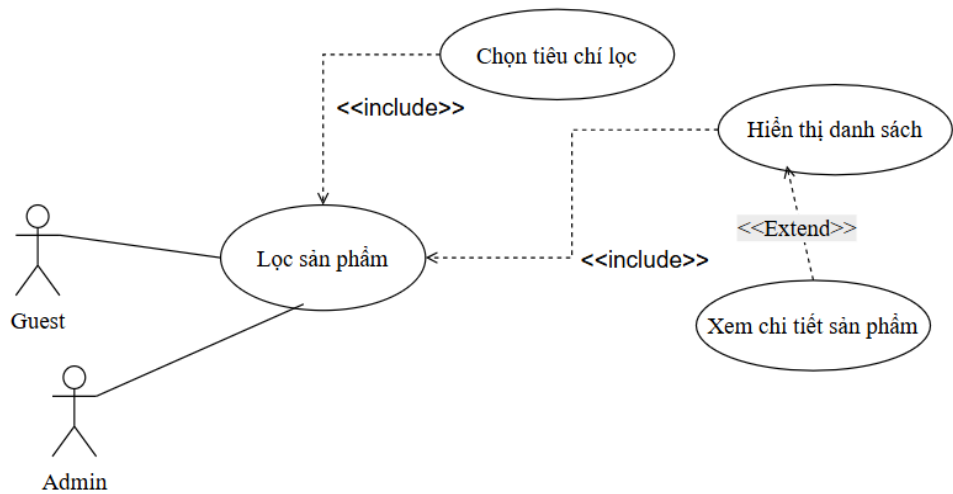


Sơ đồ 3.5: Sơ đồ phân rã UC tìm kiếm sản phẩm

Tên Use Case	Tìm kiếm sản phẩm
Mục tiêu	Cho phép Guest (Khách) hoặc Admin (Quản trị viên) tìm kiếm sản phẩm có trong hệ thống.
Tác nhân chính	Khách, quản trị viên
Tiền điều kiện	Tác nhân đã truy cập vào giao diện tìm kiếm
Hậu điều kiện	Hiển thị danh sách sản phẩm đã được tìm kiếm
Luồng được bao gồm	Nhập từ khóa , hiển thị danh sách sản phẩm
Luồng mở rộng	Xem chi tiết sản phẩm
Luồng sự kiện chính: <ul style="list-style-type: none"> - Tác nhân truy cập vào giao diện tìm kiếm - Nhập từ khóa tìm kiếm và nhấn nút tìm kiếm - Hệ thống xử lý từ khóa và truy vấn dữ liệu - Hệ thống hiện danh sách sản phẩm đã tìm kiếm. - Tác nhân nhấn vào sản phẩm để xem chi tiết sản phẩm. 	
Luồng ngoại lệ: <ul style="list-style-type: none"> - Không tìm thấy sản phẩm. Hệ thống thông báo “ không tìm thấy sản phẩm”. - Tác nhân nhập từ khóa không phù hợp . Hệ thống thông báo “Vui lòng nhập lại”. 	

Bảng 3.6: kịch bản UC tìm kiếm sản phẩm

d. Sơ đồ phân rã và kịch bản UC lọc sản phẩm



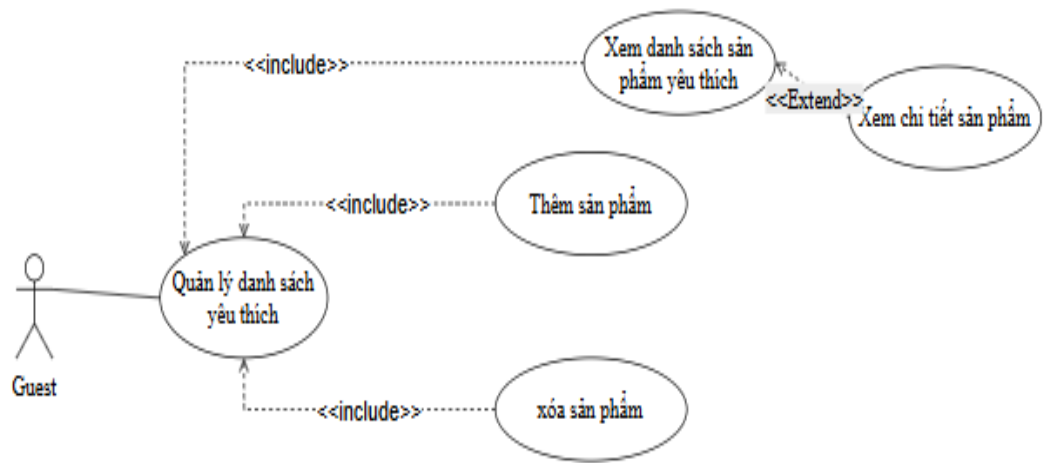
Sơ đồ 3.6: Sơ đồ phân rã UC lọc sản phẩm

Tên Use Case	Lọc sản phẩm
Mục tiêu	Cho phép Guest (Khách) hoặc Admin (Quản trị viên) tìm kiếm sản phẩm có trong hệ thống.
Tác nhân chính	Khách, quản trị viên
Tiền điều kiện	Tác nhân đã truy cập vào giao diện lọc
Hậu điều kiện	Hiển thị danh sách sản phẩm đã được lọc
Luồng được bao gồm	Chọn tiêu chí lọc, hiển thị danh sách đã lọc
Luồng mở rộng	Xem chi tiết sản phẩm
Luồng sự kiện chính: <ul style="list-style-type: none">- Tác nhân truy cập vào lọc sản phẩm.- Chọn tiêu chí lọc và nhấn lọc sản phẩm.- Hệ thống xử lý tiêu chí và truy vấn dữ liệu- Hệ thống hiện danh sách sản phẩm đã tìm kiếm.- Tác nhân nhấn vào sản phẩm để xem chi tiết sản phẩm	

Luồng ngoại lệ: Không tìm thấy sản phẩm. Hệ thống thông báo “ không tìm thấy sản phẩm phù hợp với tiêu chí lọc”.

Bảng 3.7: Kịch bản UC chi tiết sản phẩm

e. Sơ đồ phân rã và kịch bản UC quản lý danh sách yêu thích



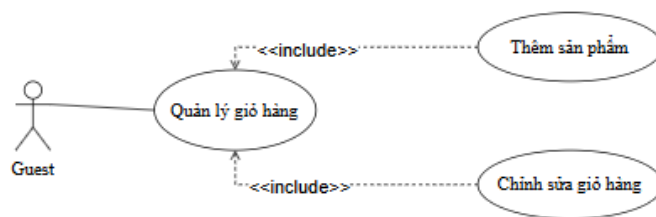
Sơ đồ 3.7: Sơ đồ phân rã UC quản lý danh sách yêu thích

Tên Use Case	Quản lý danh sách yêu thích
Mục tiêu	Cho phép Guest (Khách) xem, thêm và xóa sản phẩm ra khỏi danh sách các sản phẩm họ quan tâm.
Tác nhân chính	Khách
Tiền điều kiện	Khách đã đăng nhập vào hệ thống
Hậu điều kiện	Danh sách sản phẩm yêu thích đã được cập nhật
Luồng được bao gồm	Xem danh sách sản phẩm được yêu thích , thêm, xóa sản phẩm
Luồng mở rộng	Xem chi tiết sản phẩm
Luồng sự kiện chính: <ul style="list-style-type: none">- Tác nhân truy cập vào danh sách yêu thích (wishlist).- Hệ thống hiện danh sách sản phẩm yêu thích.	

<ul style="list-style-type: none"> - Hệ thống có thể thực hiện các hành động sau: <ul style="list-style-type: none"> + Thêm sản phẩm: Guest tìm kiếm một sản phẩm và nhấn nút "Thêm vào Yêu thích". Hệ thống kích hoạt Use Case "Thêm sản phẩm" và cập nhật danh sách. + Xóa sản phẩm: Guest nhấn nút "Xóa" bên cạnh một sản phẩm trong danh sách. Hệ thống kích hoạt Use Case "Xóa sản phẩm" và xóa sản phẩm khỏi danh sách. - Hệ thống có thể thông báo cập nhật thành công.
<p>Luồng ngoại lệ:</p> <ul style="list-style-type: none"> - Sản phẩm đã có trong danh sách (Áp dụng khi Thêm sản phẩm). Nếu Guest cố gắng thêm một sản phẩm đã có trong danh sách, hệ thống hiển thị thông báo: “Sản phẩm này đã có trong danh sách yêu thích của bạn. - Danh sách yêu thích trống (Áp dụng khi Xem danh sách). Nếu danh sách không có sản phẩm nào, hệ thống hiển thị thông báo: “Danh sách yêu thích của bạn hiện đang trống.”

Bảng 3.8: Kịch bản UC quản lý danh sách yêu thích

f. Sơ đồ phân rã và kịch bản UC quản lý giỏ hàng

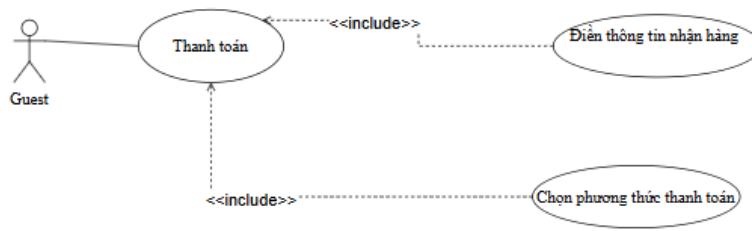


Sơ đồ 3.8: Sơ đồ phân rã UC quản lý giỏ hàng

Tên Use Case	Quản lý danh sách quản lý giỏ hàng
Mục tiêu	Cho phép Guest (Khách) xem, thêm và xóa sản phẩm ra khỏi danh sách sản phẩm trong giỏ hàng.
Tác nhân chính	Khách
Tiền điều kiện	Khách đã đăng nhập vào hệ thống và đang duyệt sản phẩm.
Hậu điều kiện	Giỏ hàng được cập nhật và sẵn sàng để thanh toán
Luồng được bao gồm	Thêm sản phẩm, chỉnh sửa giỏ hàng
<p>Luồng sự kiện chính:</p> <ul style="list-style-type: none"> - Tác nhân truy cập vào giỏ hàng. - Hệ thống hiện danh sách sản phẩm trong giỏ hàng. - Hệ thống có thể thực hiện các hành động sau: <ul style="list-style-type: none"> + Thêm sản phẩm: Guest tìm kiếm một sản phẩm và nhấn nút "Thêm vào giỏ hàng". Hệ thống kích hoạt Use Case "Thêm sản phẩm" và cập nhật danh sách. + Xóa sản phẩm: Guest nhấn nút "Xóa" bên cạnh một sản phẩm trong danh sách. Hệ thống kích hoạt Use Case "Xóa sản phẩm" và xóa sản phẩm khỏi danh sách. - Hệ thống có thể thông báo cập nhật thành công. 	
<p>Luồng ngoại lệ:</p> <ul style="list-style-type: none"> - Sản phẩm đã hết (Áp dụng khi Thêm sản phẩm). Nếu Guest cố gắng thêm một sản phẩm đã hết trong kho thì hệ thống hiển thị thông báo: “Sản phẩm này đã hết. - Vượt quá giới hạn tồn kho (Áp dụng khi Chỉnh sửa giỏ hàng). Nếu Guest nhập số lượng vượt quá số lượng tồn kho còn lại, hệ thống hiển thị thông báo: “Số lượng tồn kho chỉ còn [X] sản phẩm. Vui lòng giảm số lượng.” 	

Bảng 3.9: kịch bản UC quản lý giỏ hàng

g. Sơ đồ phân rã và kịch bản UC thanh toán



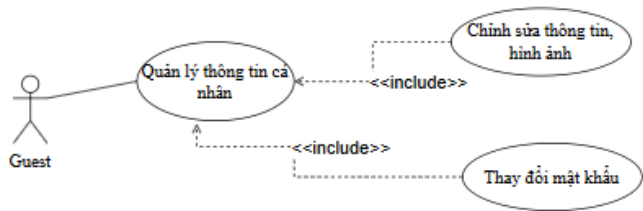
Sơ đồ 3.9: Sơ đồ phân rã UC thanh toán

Tên Use Case	Thanh toán
Mục tiêu	Cho phép Guest (Khách) hoàn tất quá trình mua hàng bằng cách cung cấp thông tin nhận hàng và lựa chọn phương thức thanh toán.
Tác nhân chính	Khách
Tiền điều kiện	Giỏ hàng đã có sản phẩm và Guest đã nhấn nút "Thanh toán".
Hậu điều kiện	Đơn hàng được tạo thành công và chờ xử lý.
Luồng được bao gồm	Điền thông tin, Chọn phương thức thanh toán
Luồng sự kiện chính: <ul style="list-style-type: none">- Hệ thống chuyển khách hàng đến trang thanh toán.- Khách hàng nhập thông tin nhận hàng.- Hệ thống kích hoạt Use Case "Chọn phương thức thanh toán".- Guest chọn một phương thức thanh toán (ví dụ: COD, Thẻ ngân hàng, Ví điện tử).- Guest nhấn nút "Đặt hàng" hoặc "Thanh toán ngay".- Hệ thống xác nhận thông tin, xử lý thanh toán (nếu là thanh toán trực tuyến) và tạo đơn hàng.	

<p>- Hệ thống hiển thị thông báo “Đặt hàng thành công”</p>
<p>Luồng ngoại lệ:</p> <p>- Thiếu/Sai thông tin nhận hàng (Xảy ra sau bước 2) . Nếu thông tin bị bỏ trống hoặc không đúng định dạng, hệ thống thông báo: “Vui lòng kiểm tra lại và điền đầy đủ thông tin nhận hàng.”</p> <p>- Thanh toán trực tuyến thất bại (Xảy ra sau bước 5). Nếu quá trình xử lý thẻ/ví điện tử thất bại, hệ thống hiển thị thông báo: “Thanh toán thất bại. Vui lòng kiểm tra lại thông tin thanh toán hoặc chọn phương thức khác.</p>

Bảng 3.10: Kịch bản UC thanh toán

h. Sơ đồ phân rã và kịch bản UC quản lý thông tin



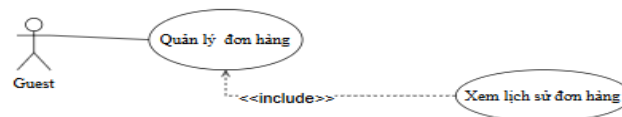
Sơ đồ 3.10: Sơ đồ phân rã UC quản lý thông tin cá nhân

Tên Use Case	Quản lý danh sách quản lý thông tin cá nhân
Mục tiêu	Cho phép Guest (Khách) xem, chỉnh sửa thông tin cá nhân và thay đổi mật khẩu
Tác nhân chính	Khách
Tiền điều kiện	Khách đã đăng nhập vào hệ thống và đang ở trang profile
Hậu điều kiện	Thông tin cá nhân, mật khẩu đã được cập nhật

Luồng được bao gồm	Chỉnh sửa thông tin, hình ảnh, thay đổi mật khẩu
<p>Luồng sự kiện chính:</p> <ul style="list-style-type: none"> - Guest truy cập trang thông tin cá nhân. - Guest chọn thực hiện một trong hai hành động: - Chỉnh sửa thông tin: Guest sửa đổi tên, email, số điện thoại, hoặc tải lên ảnh đại diện. Hệ thống kích hoạt "Chỉnh sửa thông tin, hình ảnh". - Thay đổi mật khẩu: Guest nhập mật khẩu hiện tại, mật khẩu mới và xác nhận mật khẩu mới. Hệ thống kích hoạt "Thay đổi mật khẩu". - Guest nhấn nút "Lưu/Cập nhật". - Hệ thống xác thực dữ liệu và cập nhật vào cơ sở dữ liệu. - Hệ thống hiển thị thông báo “Cập nhật thành công.” 	
<p>Luồng ngoại lệ:</p> <ul style="list-style-type: none"> - Mật khẩu hiện tại không đúng (Áp dụng cho Thay đổi mật khẩu). Hệ thống thông báo: “Mật khẩu hiện tại không chính xác.” - Định dạng dữ liệu không hợp lệ (Áp dụng cho Chỉnh sửa thông tin). Ví dụ: email sai định dạng, số điện thoại không đủ chữ số. Hệ thống thông báo lỗi tương ứng. 	

Bảng 3.11: Kịch bản UC quản lý thông tin cá nhân

i. Sơ đồ phân rã và kịch bản UC quản lý đơn hàng



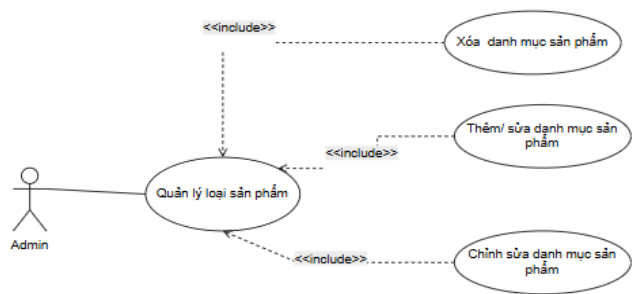
Sơ đồ 3.11: Sơ đồ phân rã UC quản lý đơn hàng

Tên Use Case	Quản lý danh sách quản lý đơn hàng
Mục tiêu	Cho phép Guest (Khách) xem lịch sử đơn hàng đã mua

Tác nhân chính	Khách
Tiền điều kiện	Khách đã đăng nhập và đã có ít nhất một đơn hàng.
Hậu điều kiện	Khách nắm được thông tin đơn hàng; đơn hàng có thể được thanh toán hoặc sản phẩm được đánh giá.
Luồng được bao gồm	Đánh giá sản phẩm đã mua, tiếp tục thanh toán
<p>Luồng sự kiện chính:</p> <ul style="list-style-type: none"> - Guest truy cập vào khu vực "Lịch sử đơn hàng". - Hệ thống hiển thị danh sách tất cả các đơn hàng của Guest với trạng thái hiện tại (Đang chờ xử lý, Đang giao, Đã giao, Đã hủy). - Guest có thể chọn một đơn hàng để xem chi tiết. - Dựa trên trạng thái đơn hàng, Guest có thể: - Đánh giá sản phẩm: Nếu trạng thái là "Đã giao", Guest có thể nhấn nút "Đánh giá". Hệ thống kích hoạt "Đánh giá sản phẩm đã mua". - Tiếp tục thanh toán: Nếu trạng thái là "Chờ thanh toán" (ví dụ: lỗi thanh toán trực tuyến trước đó), Guest nhấn nút "Tiếp tục thanh toán". Hệ thống kích hoạt "Tiếp tục thanh toán" để quay lại quy trình thanh toán. - Hệ thống xử lý hành động và cập nhật trạng thái/dữ liệu (đánh giá, thanh toán thành công). 	
<p>Luồng ngoại lệ:</p> <ul style="list-style-type: none"> - Đánh giá không hợp lệ (Áp dụng cho Đánh giá sản phẩm). Nếu Guest cố gắng gửi đánh giá trống hoặc không chọn sao. Hệ thống thông báo: “Vui lòng nhập nội dung đánh giá và/hoặc chọn số sao.” - Đơn hàng không đủ điều kiện. Nếu Guest cố gắng "Đánh giá" một đơn hàng chưa có trạng thái "Đã giao", hệ thống thông báo: “Chỉ có thể đánh giá sản phẩm đã được giao thành công.” 	

Bảng 3.12: Kịch bản UC quản lý đơn hàng

k. Sơ đồ phân rã và kịch bản UC quản lý danh mục



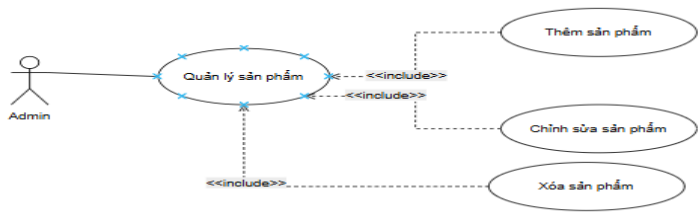
Sơ đồ 3.12: Sơ đồ phân rã quản lý danh mục

Tên Use Case	Quản lý danh sách quản lý danh mục
Mục tiêu	Cho phép admin xem, thêm và xóa danh mục sản phẩm ra khỏi danh sách danh mục sản phẩm
Tác nhân chính	Admin
Tiền điều kiện	Admin đã đăng nhập và truy cập trang quản lý danh mục sản phẩm.
Hậu điều kiện	Danh sách danh mục sản phẩm đã được cập nhật
Luồng được bao gồm	Thêm, xóa , sửa danh mục sản phẩm
Luồng sự kiện chính: <ul style="list-style-type: none">- Admin truy cập vào giao diện Quản lý danh mục sản phẩm.- Hệ thống hiển thị danh sách sản phẩm hiện tại.- Admin chọn thực hiện một trong các hành động sau:<ul style="list-style-type: none">+ Thêm danh mục sản phẩm: Admin nhấn nút "Thêm mới". Hệ thống kích hoạt "Thêm danh mục sản phẩm" (Điền thông tin chi tiết: tên, hình ảnh).+ Chỉnh sửa danh mục sản phẩm: Admin chọn một danh mục sản phẩm và nhấn "Chỉnh sửa". Hệ thống kích hoạt "Chỉnh sửa danh mục sản phẩm" (Admin cập nhật thông tin).+ Xóa danh mục sản phẩm: Admin chọn một hoặc nhiều sản phẩm và nhấn "Xóa". Hệ thống kích hoạt "Xóa danh mục sản phẩm" (Yêu cầu xác nhận).	

<ul style="list-style-type: none">- Hệ thống xử lý yêu cầu và cập nhật cơ sở dữ liệu.- Hệ thống hiển thị thông báo cập nhật thành công.
Luồng ngoại lệ: Thiếu thông tin bắt buộc (Áp dụng cho Thêm/Chỉnh sửa). Admin bỏ trống các trường bắt buộc (ví dụ: Tên danh mục sản phẩm).Hệ thống thông báo “Vui lòng điền đầy đủ các thông tin bắt buộc.”

Bảng 3.13: Kịch bản UC quản lý danh mục

1. Sơ đồ phân rã và kịch bản UC quản lý sản phẩm



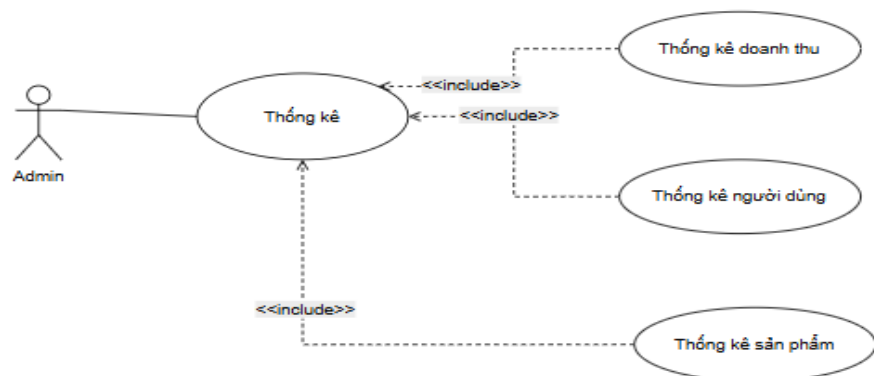
Sơ đồ 3.13: Sơ đồ phân rã UC quản lý sản phẩm

Tên Use Case	Quản lý danh sách quản lý sản phẩm
Mục tiêu	Cho phép admin xem, thêm và xóa sản phẩm ra khỏi danh sách sản phẩm
Tác nhân chính	Admin
Tiền điều kiện	Admin đã đăng nhập và truy cập trang quản lý sản phẩm.
Hậu điều kiện	Danh sách danh mục sản phẩm đã được cập nhật
Luồng được bao gồm	Thêm, xóa , sửa sản phẩm
Luồng sự kiện chính: <ul style="list-style-type: none">- Admin truy cập vào giao diện Quản lý sản phẩm.- Hệ thống hiển thị danh sách sản phẩm hiện tại.- Admin chọn thực hiện một trong các hành động sau:- Thêm danh mục sản phẩm: Admin nhấn nút "Thêm mới". Hệ thống kích hoạt "Thêm sản phẩm" (Điền thông tin chi tiết: tên, hình ảnh,...).	

<p>+ Chỉnh sửa danh mục sản phẩm: Admin chọn một danh mục sản phẩm và nhấn "Chỉnh sửa". Hệ thống kích hoạt "Chỉnh sản phẩm" (Admin cập nhật thông tin).</p> <p>+ Xóa sản phẩm: Admin chọn một hoặc nhiều sản phẩm và nhấn "Xóa". Hệ thống kích hoạt "Xóa sản phẩm" (Yêu cầu xác nhận).</p> <ul style="list-style-type: none"> - Hệ thống xử lý yêu cầu và cập nhật cơ sở dữ liệu. - Hệ thống hiển thị thông báo cập nhật thành công.
<p>Luồng ngoại lệ:</p> <ul style="list-style-type: none"> - Thiếu thông tin bắt buộc (Áp dụng cho Thêm/Chỉnh sửa). Admin bỏ trống các trường bắt buộc (ví dụ: Tên sản phẩm, danh mục,...). Hệ thống thông báo “Vui lòng điền đầy đủ các thông tin bắt buộc.” - Xóa sản phẩm thất bại (Áp dụng cho Xóa danh mục sản phẩm). Nếu danh mục sản phẩm đang liên kết với các đơn hàng chưa hoàn thành, hệ thống có thể ngăn chặn việc xóa. Hệ thống thông báo “Không thể xóa sản phẩm do đang có đơn hàng liên quan.”

Bảng 3.14: Kịch bản UC quản lý sản phẩm

m. Sơ đồ phân rã và kịch bản UC thống kê

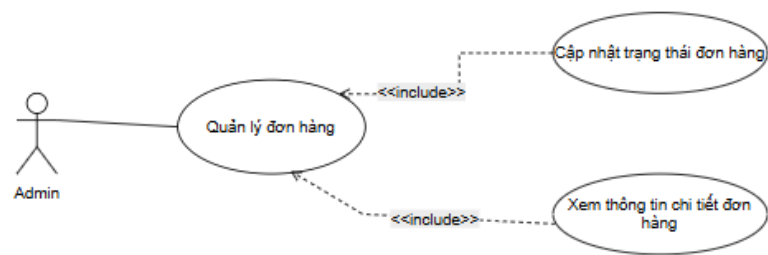


Sơ đồ 3.14: Sơ đồ phân rã UC thống kê

Tên Use Case	Quản lý danh sách quản lý giỏ hàng
Mục tiêu	Cho phép Admin (Quản trị viên) xem các báo cáo và số liệu thống kê về doanh thu, người dùng và sản phẩm.
Tác nhân chính	Admin
Tiền điều kiện	Admin đã đăng nhập và truy cập trang Thống kê.
Hậu điều kiện	Admin xem được các báo cáo theo yêu cầu.
Luồng được bao gồm	Thống kê doanh thu, thống kê người dùng, thống kê sản phẩm.
<p>Luồng sự kiện chính:</p> <ul style="list-style-type: none"> - Admin vào trang thống kê. - Hệ thống hiển thị các thống kê sau: <ul style="list-style-type: none"> + Thống kê doanh thu: (Use Case "Thống kê doanh thu"). Admin chọn kỳ báo cáo (ngày, tháng, năm). Hệ thống hiển thị biểu đồ, tổng doanh thu, lợi nhuận... + Thống kê người dùng: (Use Case "Thống kê người dùng"). Hệ thống hiển thị số lượng đăng ký mới, người dùng hoạt động... + Thống kê sản phẩm: (Use Case "Thống kê sản phẩm"). Hệ thống hiển thị sản phẩm bán chạy nhất, sản phẩm tồn kho cao nhất... 	
<p>Luồng ngoại lệ:</p> <ul style="list-style-type: none"> - Không có dữ liệu trong kỳ báo cáo. Nếu Admin chọn một kỳ báo cáo không có giao dịch/sự kiện nào. Hệ thống thông báo “Không tìm thấy dữ liệu trong kỳ báo cáo đã chọn.” - Lỗi truy vấn dữ liệu: Hệ thống gặp lỗi khi truy vấn cơ sở dữ liệu lớn. Hệ thống thông báo “Lỗi hệ thống khi tạo báo cáo. Vui lòng thử lại sau.” 	

Bảng 3.15: Kịch bản UC thống kê

n. Sơ đồ phân rã và kịch bản UC quản lý đơn hàng



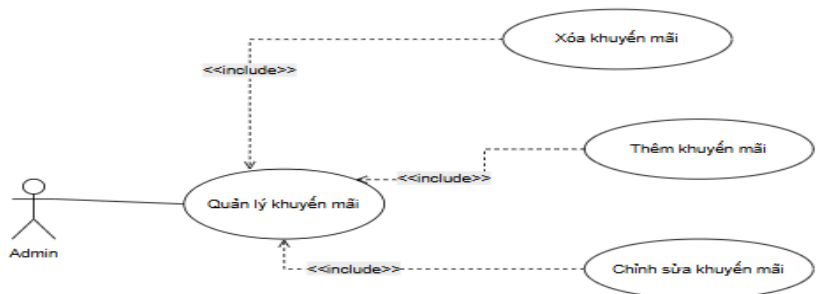
Sơ đồ 3.15: Sơ đồ phân rã UC quản lý đơn hàng

Tên Use Case	Quản lý danh sách quản lý đơn hàng
Mục tiêu	Cho phép Admin (Quản trị viên) xem chi tiết đơn hàng, và cập nhật trạng thái đơn hàng của khách hàng.
Tác nhân chính	Admin
Tiền điều kiện	Admin đã đăng nhập và truy cập trang quản lý đơn hàng.
Hậu điều kiện	Trạng thái đơn hàng được cập nhật chính xác và khách hàng được thông báo.
Luồng được bao gồm	Xem thông tin chi tiết đơn hàng, cập nhật trạng thái đơn hàng
Luồng sự kiện chính: <ul style="list-style-type: none">- Admin truy cập vào giao diện Quản lý đơn hàng.- Hệ thống hiển thị danh sách tất cả các đơn hàng.- Admin chọn một đơn hàng cụ thể.- Hệ thống kích hoạt "Xem thông tin chi tiết đơn hàng" (Hiển thị sản phẩm, thông tin khách hàng, lịch sử trạng thái).- Admin xác định trạng thái đơn hàng hiện tại (ví dụ: Chờ xác nhận, Đang đóng gói, Đang giao).	

<div><div>- Admin chọn "Cập nhật trạng thái đơn hàng" và chọn trạng thái mới (ví dụ: từ Chờ xác nhận sang Đang đóng gói).</div><div>- Hệ thống cập nhật trạng thái đơn hàng và (tùy chọn) gửi thông báo/email cho khách hàng.</div><div>- Hệ thống hiển thị thông báo “Cập nhật trạng thái thành công.”</div></div>
<div><div>Luồng ngoại lệ:</div><div><div>- Cập nhật trạng thái không hợp lệ. Nếu Admin cố gắng chuyển trạng thái theo một logic không cho phép (ví dụ: từ Đã hủy sang Đã giao).Hệ thống thông báo “Không thể chuyển sang trạng thái này.”</div><div>- Không tìm thấy đơn hàng. Nếu đơn hàng bị xóa hoặc không tồn tại.Hệ thống thông báo “Không tìm thấy đơn hàng này.”</div></div></div>

Bảng 3.16: Kịch bản UC quản lý đơn hàng

o. Sơ đồ phân rã và kịch bản UC quản lý khuyến mãi



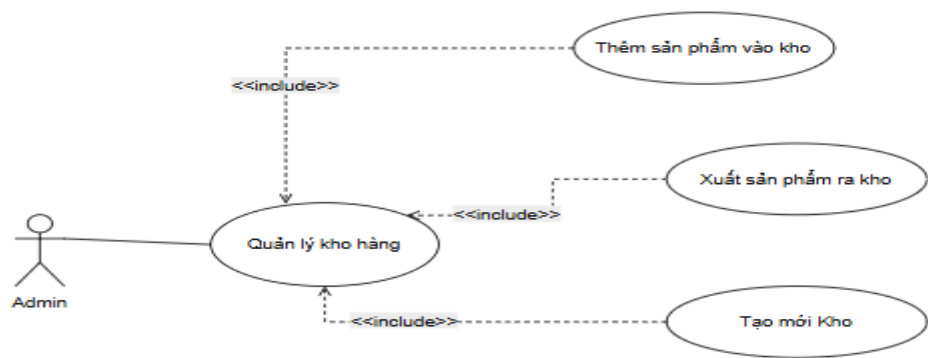
Sơ đồ 3.16: Sơ đồ phân rã UC quản lý khuyến mãi

Tên Use Case	Quản lý danh sách quản lý khuyến mãi
Mục tiêu	Cho phép Admin (Quản trị viên) tạo, chỉnh sửa và loại bỏ các chương trình khuyến mãi/mã giảm giá.
Tác nhân chính	Admin
Tiền điều kiện	Admin đã đăng nhập và truy cập trang quản lý khuyến mãi.

Hậu điều kiện	Danh sách khuyến mãi/giảm giá được cập nhật.
Luồng được bao gồm	Thêm, chỉnh sửa, xóa khuyến mãi
<p>Luồng sự kiện chính:</p> <ul style="list-style-type: none"> - Admin truy cập vào giao diện Quản lý khuyến mãi. - Hệ thống hiển thị danh sách các chương trình khuyến mãi hiện tại. - Admin chọn thực hiện một trong các hành động sau: <ul style="list-style-type: none"> + Thêm khuyến mãi: Admin nhấn "Thêm mới". Hệ thống kích hoạt "Thêm khuyến mãi" (Điền các thông tin: Mã, giá trị giảm, thời hạn, điều kiện áp dụng). + Chỉnh sửa khuyến mãi: Admin chọn một khuyến mãi và nhấn "Chỉnh sửa". Hệ thống kích hoạt "Chỉnh sửa khuyến mãi" (Admin cập nhật). + Xóa khuyến mãi: Admin chọn một khuyến mãi và nhấn "Xóa". Hệ thống kích hoạt "Xóa khuyến mãi" (Yêu cầu xác nhận). - Hệ thống xử lý yêu cầu và cập nhật cơ sở dữ liệu. - Hệ thống hiển thị thông báo cập nhật thành công. <p>Luồng ngoại lệ: Lỗi định dạng/logic khuyến mãi (Áp dụng cho Thêm/Chỉnh sửa) Ví dụ: Ngày kết thúc trước ngày bắt đầu, giá trị giảm giá không hợp lệ. Hệ thống thông báo “Thông tin khuyến mãi không hợp lệ. Vui lòng kiểm tra lại thời gian/giá trị.”</p>	

Bảng 3.17: Kịch bản quản lý khuyến mãi

p. Sơ đồ phân rã và kịch bản UC quản lý kho hàng



Sơ đồ 3.17: Sơ đồ phân rã UC quản lý kho hàng

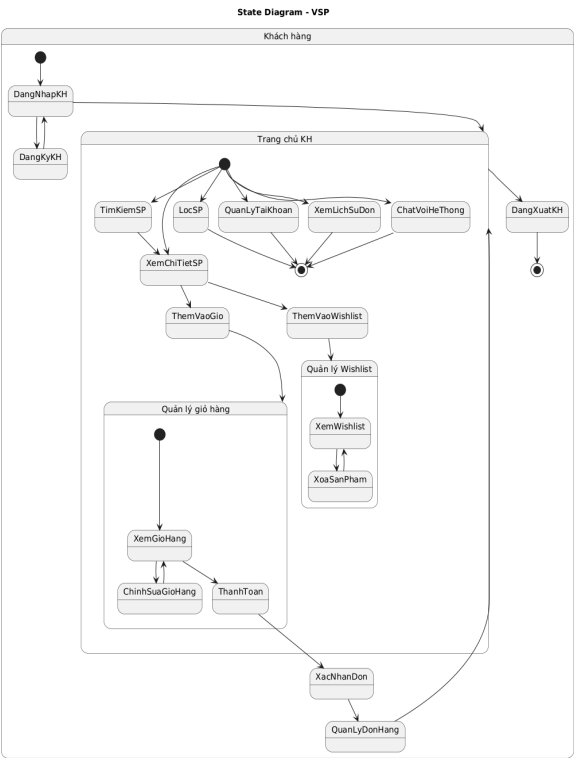
Tên Use Case	Quản lý danh sách quản lý kho hàng
Mục tiêu	Cho phép Admin (Quản trị viên) quản lý số lượng tồn kho, thực hiện nhập và xuất kho, và tạo kho mới.
Tác nhân chính	Admin
Tiền điều kiện	Admin đã đăng nhập và truy cập trang quản lý kho hàng.
Hậu điều kiện	Tồn kho sản phẩm được cập nhật chính xác.
Luồng được bao gồm	Thêm sản phẩm vào kho, xuất sản phẩm ra kho, tạo mới kho
Luồng sự kiện chính: <ul style="list-style-type: none">- Admin truy cập vào giao diện Quản lý kho hàng.- Hệ thống hiển thị danh sách các kho và tồn kho hiện tại của các sản phẩm.- Admin chọn thực hiện một trong các hành động sau:<ul style="list-style-type: none">+ Nhập kho: Admin chọn "Thêm sản phẩm vào kho" (Nhập số lượng, mã sản phẩm, kho đích). Tồn kho tăng.	

<div><div><div>+ Xuất kho: Admin chọn "Xuất sản phẩm ra kho" (Nhập số lượng, mã sản phẩm, kho nguồn). Tồn kho giảm.</div><div><div>+ Tạo kho mới: Admin chọn "Tạo mới kho" (Nhập tên kho, địa điểm).</div><div>- Hệ thống xử lý yêu cầu và cập nhật tồn kho/thông tin kho.</div><div>- Hệ thống hiển thị thông báo cập nhật thành công.</div></div></div></div>
<div><div>Luồng ngoại lệ:</div><div><div>- Số lượng xuất vượt tồn kho (Áp dụng cho Xuất kho). Nếu Admin nhập số lượng xuất lớn hơn số lượng tồn kho hiện có.Hệ thống thông báo “Số lượng xuất vượt quá số lượng tồn kho hiện tại.”</div><div>- Tên kho bị trùng (Áp dụng cho Tạo mới kho). Nếu tên kho Admin muốn tạo đã tồn tại.Hệ thống thông báo “Tên kho này đã tồn tại, vui lòng chọn tên khác.”</div></div></div>

Bảng 3.18: Kịch bản UC quản lý kho hàng

3.2.4 Sơ đồ trạng thái

a) Sơ đồ trạng thái của khách hàng



Sơ đồ 3.18: Sơ đồ trạng thái khách hàng

Quy trình hoạt động của khách hàng

Đăng nhập/Đăng ký:

- Khách hàng bắt đầu bằng việc Đăng Nhập KH vào hệ thống hoặc thực hiện Đăng Ký KH nếu chưa có tài khoản.
- Sau khi đăng nhập/đăng ký thành công, Khách hàng sẽ truy cập được Trang Chủ KH.

Trang Chủ KH (Customer Home Page): Sau khi đăng nhập, Khách hàng có thể thực hiện nhiều chức năng tìm kiếm, quản lý và tương tác:

- Tìm kiếm và Xem Sản phẩm:
- Tìm Kiếm SP hoặc Lọc SP để tìm sản phẩm mong muốn.
- Sau khi tìm được, Khách hàng có thể Xem Chi Tiết SP.

Quản lý Tài khoản & Đơn hàng:

- Quản Lý Tài Khoản (cập nhật thông tin cá nhân).
- Xem Lịch Sử Đơn (xem lại các đơn hàng đã đặt).
- Tương tác: Chat Với Hệ Thống để được hỗ trợ hoặc giải đáp thắc mắc.
- Tương tác với Sản phẩm:
 - Từ trạng thái Xem Chi Tiết SP, Khách hàng có thể lựa chọn:
 - Thêm Vào Giỏ (Add to Cart): Đưa sản phẩm vào quy trình mua hàng.
 - Thêm Vào Wishlist (Add to Wishlist): Đưa sản phẩm vào danh sách yêu thích.

Quản lý Giỏ hàng và Thanh toán: Nếu Khách hàng chọn Thêm Vào Giỏ, họ sẽ chuyển sang quy trình Quản lý giỏ hàng:

- Khách hàng có thể Xem Giỏ Hàng.
- Chỉnh Sửa Giỏ Hàng (thay đổi số lượng hoặc xóa sản phẩm).
- Tiến hành Thanh Toán để hoàn tất giao dịch.

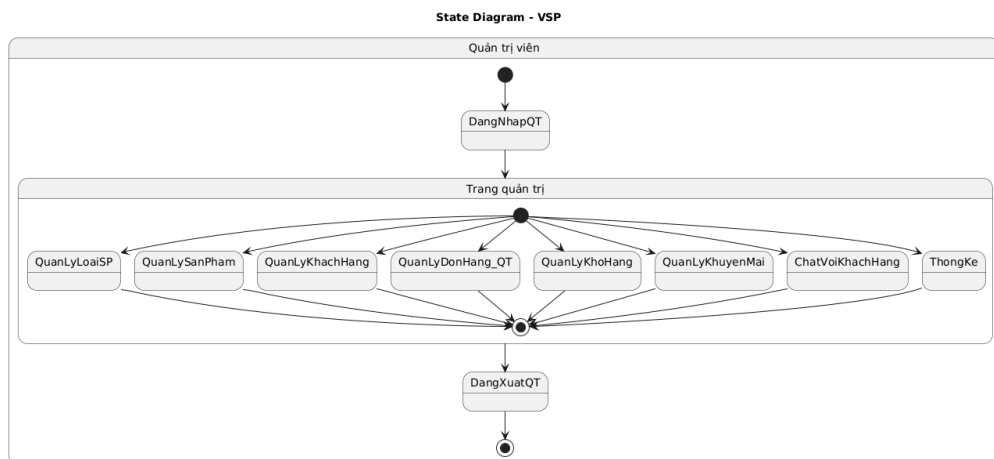
Quản lý Wishlist: Nếu Khách hàng chọn Thêm Vào Wishlist, họ sẽ chuyển sang quy trình Quản lý Wishlist:

- Khách hàng có thể Xem Wishlist.
- Xóa Sản Phẩm khỏi danh sách.

- Hoàn tất Đơn hàng: Sau khi Thanh Toán thành công, Khách hàng sẽ thực hiện: Xác Nhận Đơn (Xác nhận việc đặt hàng); Quản Lý Đơn Hàng (theo dõi trạng thái, v.v.).

Đăng xuất: Bất cứ lúc nào trong quá trình hoạt động (từ Trang Chủ KH hoặc sau khi Quản Lý Đơn Hàng), Khách hàng có thể chọn Đăng Xuất KH khỏi hệ thống để kết thúc phiên làm việc.

b) Sơ đồ trạng thái của quản trị viên



Sơ đồ 3.19: Sơ đồ trạng thái của quản trị viên

Quy trình hoạt động của quản trị viên

Đăng nhập QT (Admin Login):

- Quản Trị Viên (Admin) bắt đầu bằng việc đăng nhập vào hệ thống.
- Nếu đăng nhập thành công, Quản Trị Viên sẽ truy cập được Trang Quản Trị.

Trang Quản Trị (Admin Page):Sau khi đăng nhập thành công, Quản Trị Viên có thể thực hiện đồng thời nhiều chức năng quản lý quan trọng sau:

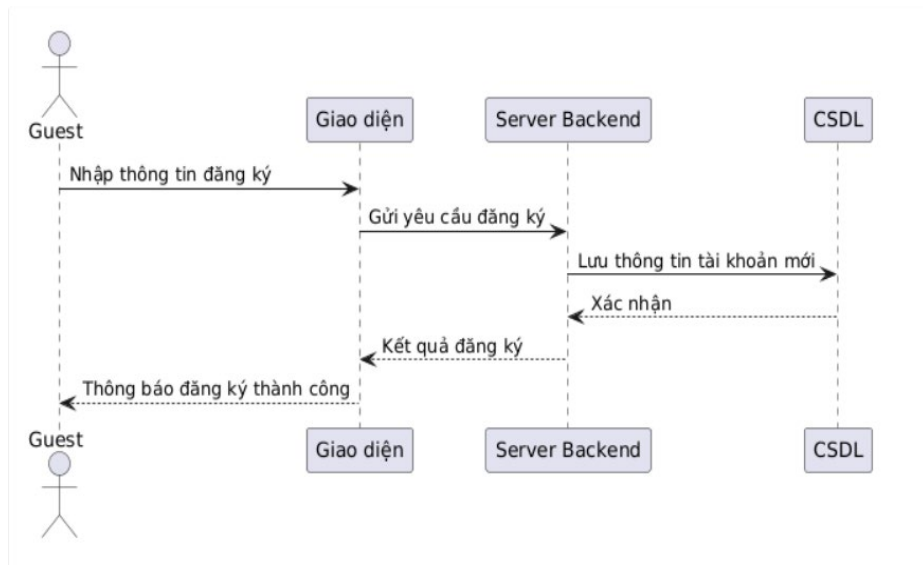
- Quản Lý Loại SP (Manage Product Categories): Cho phép Quản Trị Viên thực hiện các tác vụ liên quan đến việc sắp xếp và phân loại sản phẩm.
- Quản Lý Sản Phẩm (Manage Products): Cho phép Quản Trị Viên quản lý chi tiết sản phẩm, bao gồm thêm, sửa, hoặc xóa thông tin sản phẩm.
- Quản Lý Khách Hàng (Manage Customers): Cho phép Quản Trị Viên quản lý thông tin và tài khoản của tất cả khách hàng.
- Quản Lý Đơn Hàng QT (Manage Orders): Cho phép Quản Trị Viên xử lý, theo dõi, và cập nhật trạng thái các đơn hàng của khách hàng.

- Quản Lý Kho Hàng (Manage Inventory): Cho phép Quản Trị Viên quản lý và kiểm soát số lượng tồn kho của từng sản phẩm.
- Quản Lý Khuyến Mãi (Manage Promotions): Cho phép Quản Trị Viên thiết lập và quản lý các chương trình giảm giá, mã khuyến mãi.
- Chat Với Khách Hàng (Chat with Customers): Cho phép Quản Trị Viên giao tiếp và cung cấp hỗ trợ trực tiếp cho khách hàng.
- Thống Kê (Statistics): Cho phép Quản Trị Viên xem và phân tích các báo cáo, số liệu thống kê về hiệu suất hoạt động của hệ thống.

Đăng xuất QT (Admin Logout): Bất cứ lúc nào sau khi hoàn thành các tác vụ quản lý trên Trang Quản Trị, Quản Trị Viên có thể chọn Đăng Xuất QT khỏi hệ thống để kết thúc phiên làm việc.

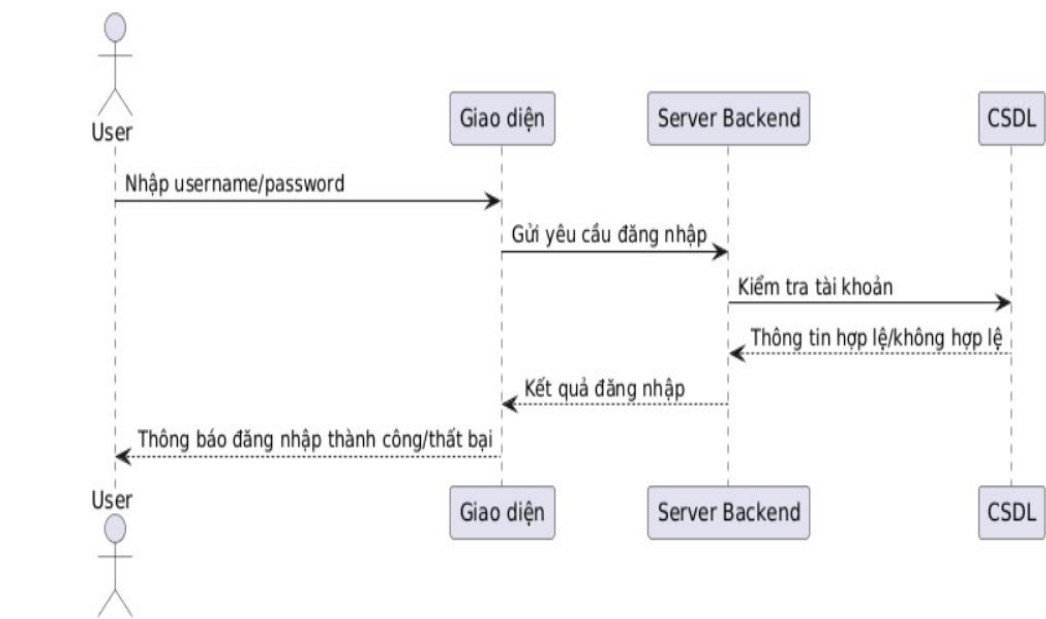
3.2.5 Sơ đồ tuần tự

a. Sơ đồ tuần tự đăng ký



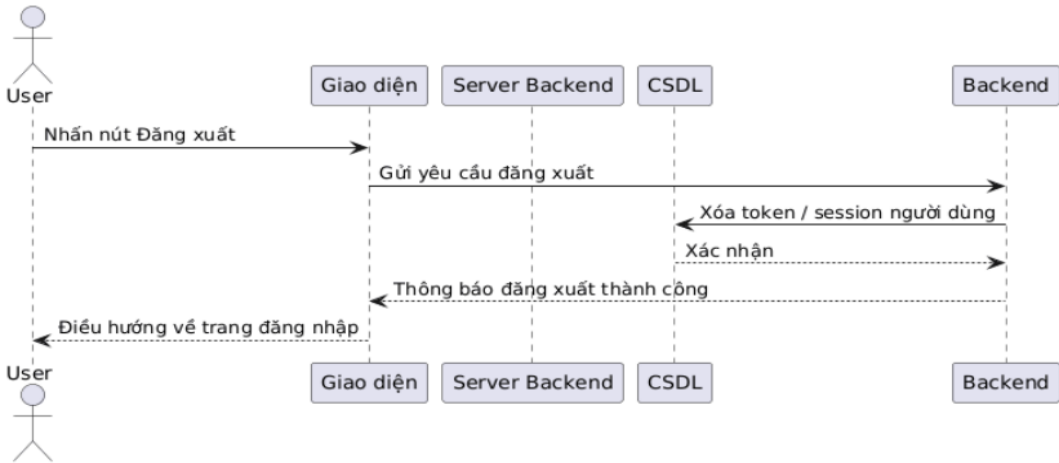
Sơ đồ 3.20: Sơ đồ tuần tự đăng ký

b. Sơ đồ tuần tự đăng nhập



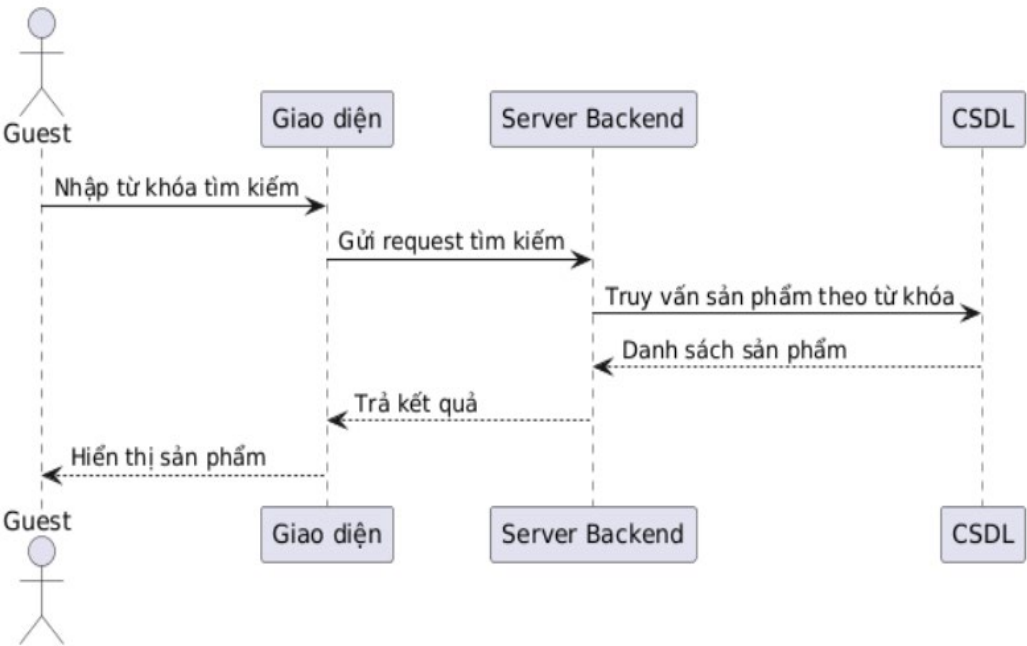
Sơ đồ 3.21: Sơ đồ tuần tự đăng nhập

c. Sơ đồ tuần tự đăng xuất



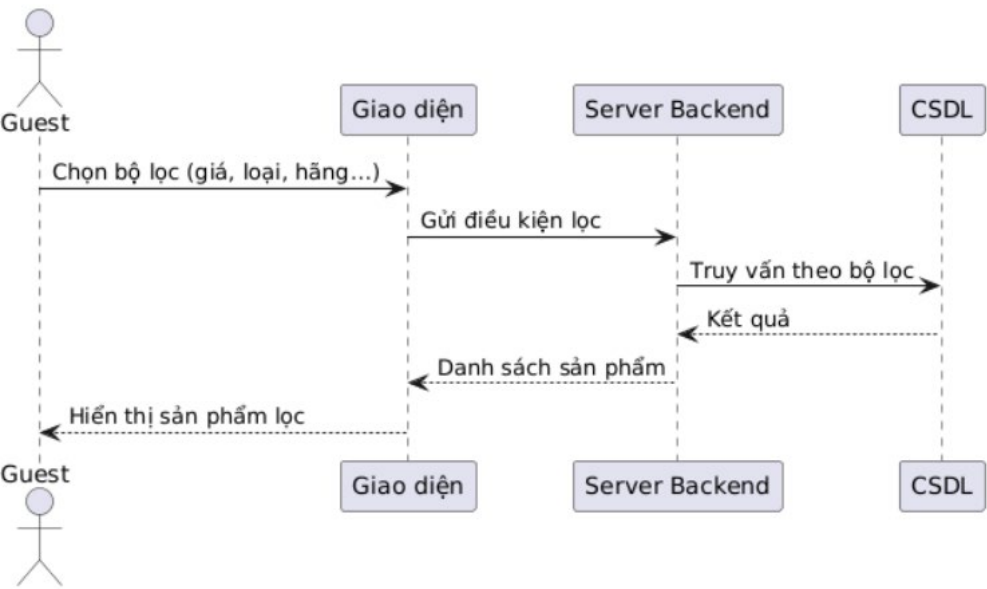
Sơ đồ 3.22: Sơ đồ tuần tự đăng xuất

d. Sơ đồ tuần tự tìm kiếm sản phẩm



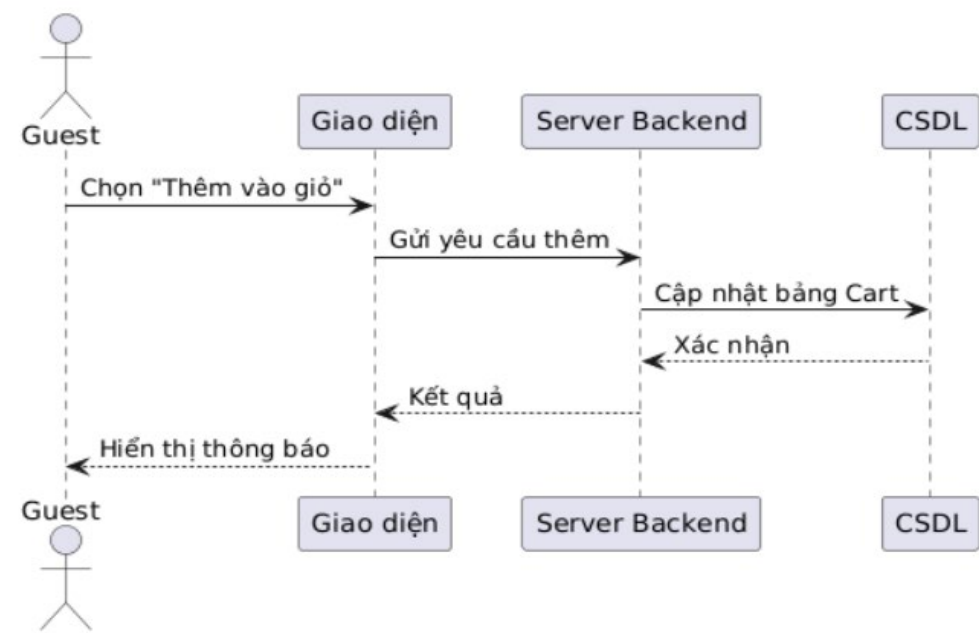
Sơ đồ 3.23: Sơ đồ tuần tự tìm kiếm sản phẩm

e. Sơ đồ tuần tự lọc sản phẩm



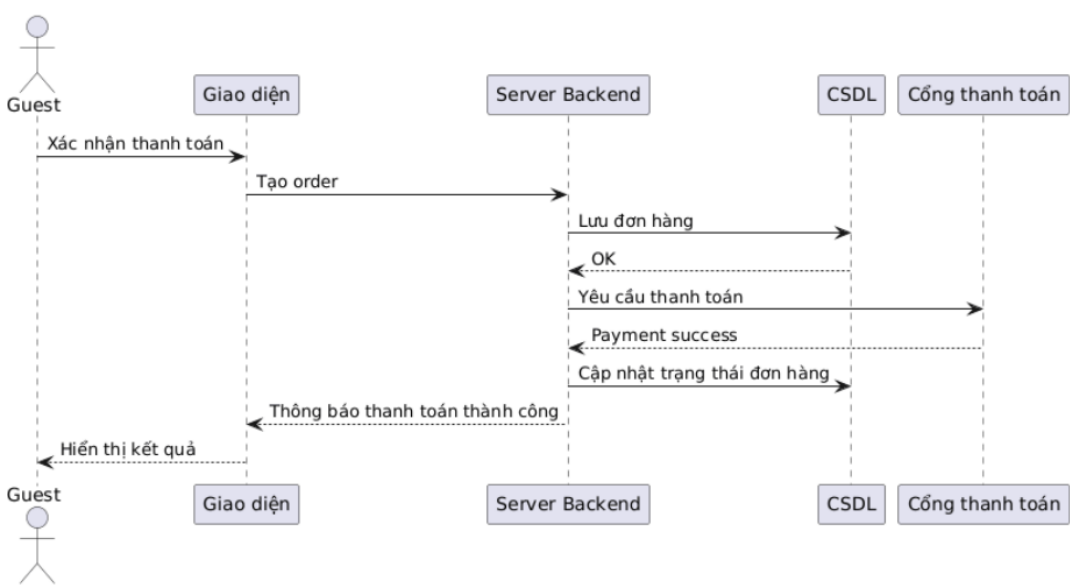
Sơ đồ 3.24: Sơ đồ tuần tự lọc sản phẩm

f. Sơ đồ tuần tự quản lý giỏ hàng



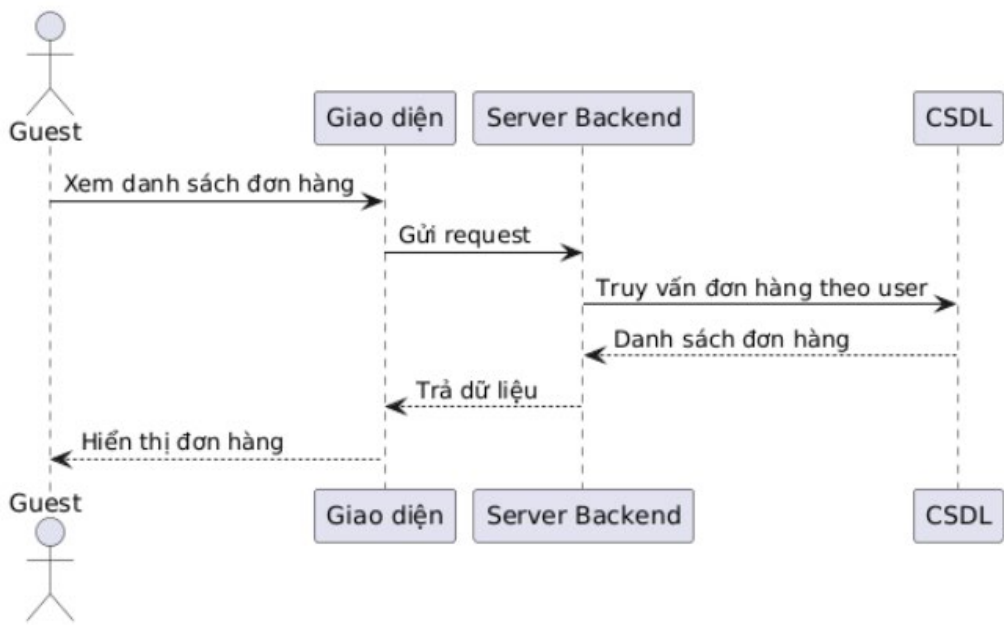
Sơ đồ 3.25: Sơ đồ tuần tự quản lý giỏ hàng của user

g. Sơ đồ tuần tự thanh toán



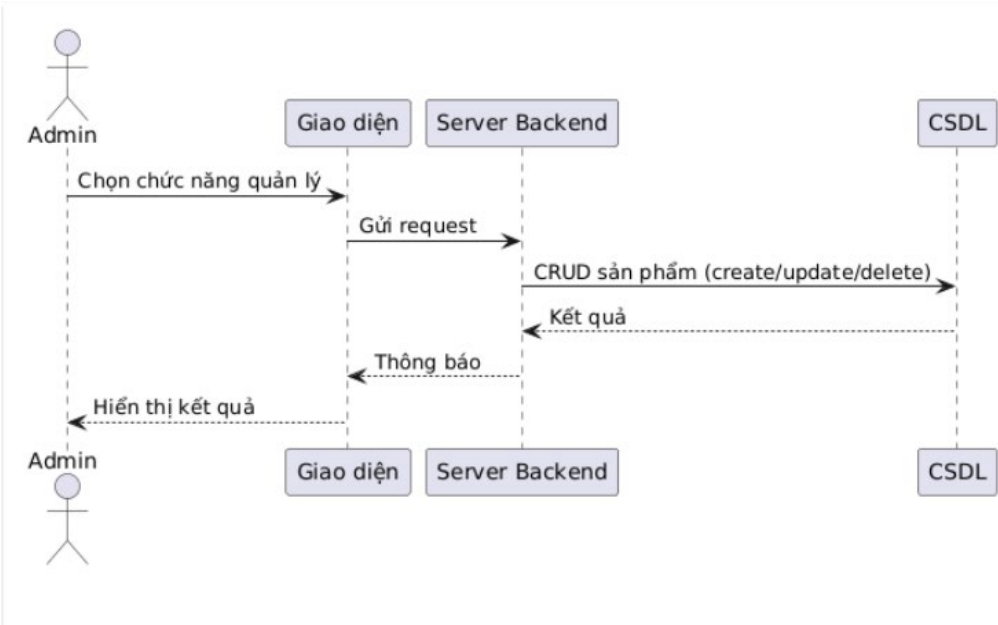
Sơ đồ 3.26: Sơ đồ tuần tự thanh toán

h. Sơ đồ tuần tự quản lý đơn hàng của khách hàng



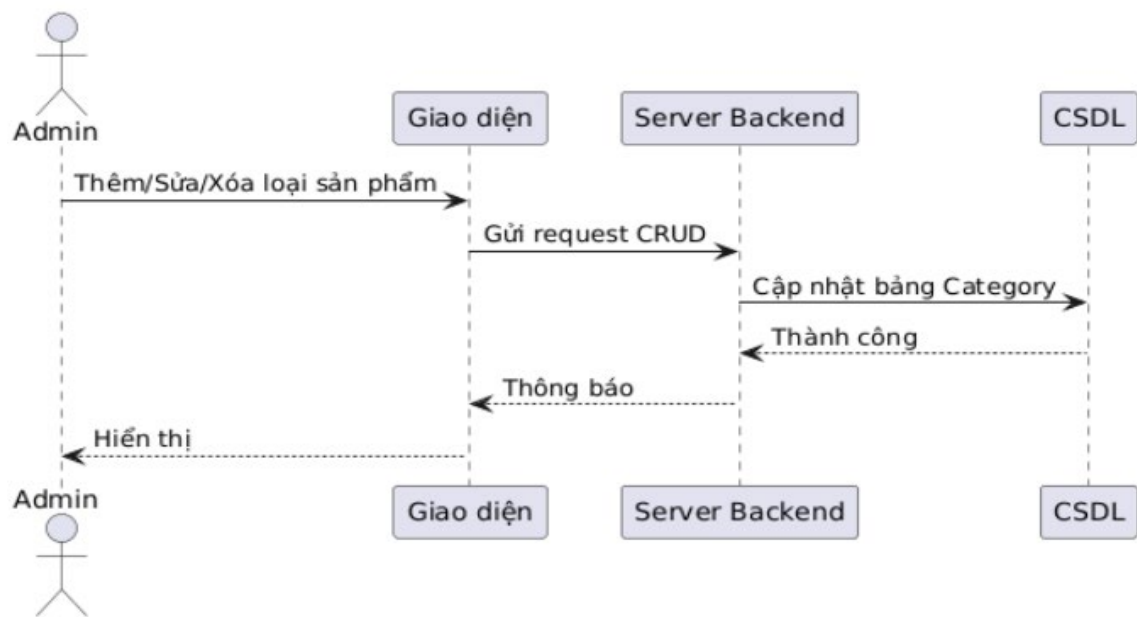
Sơ đồ 3.27: Sơ đồ quản lý đơn hàng của khách hàng

i. Sơ đồ tuần tự quản lý sản phẩm



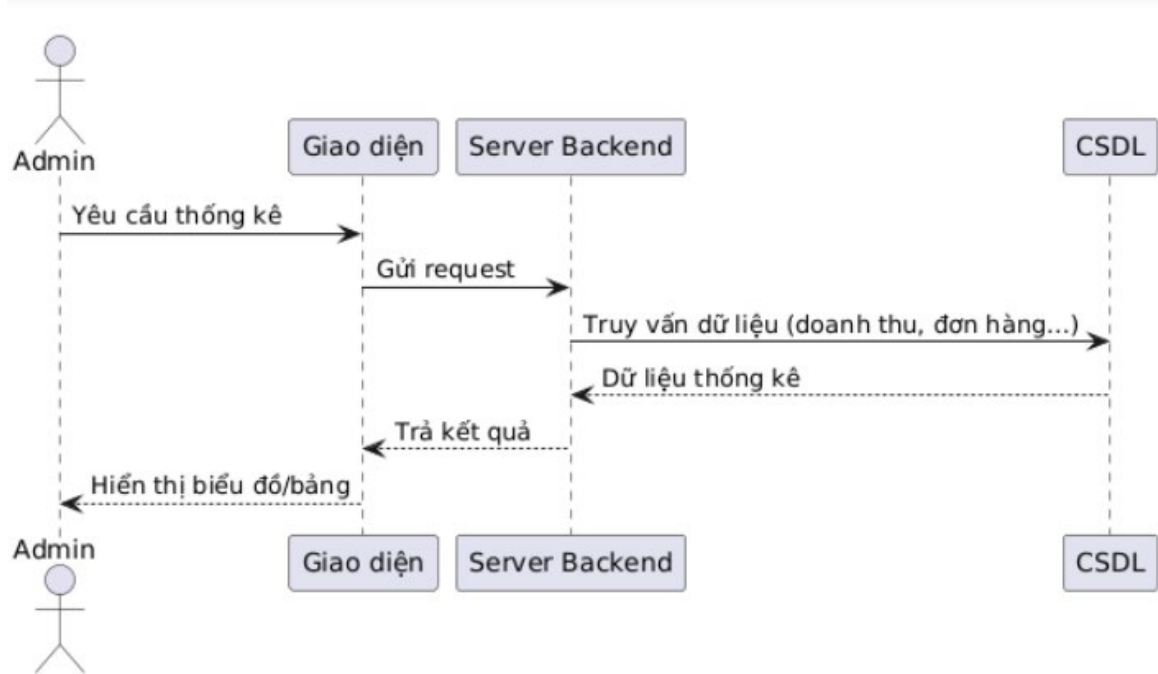
Sơ đồ 3.28: Sơ đồ quản lý sản phẩm của quản trị viên

j. Sơ đồ tuần tự quản lý danh mục sản phẩm



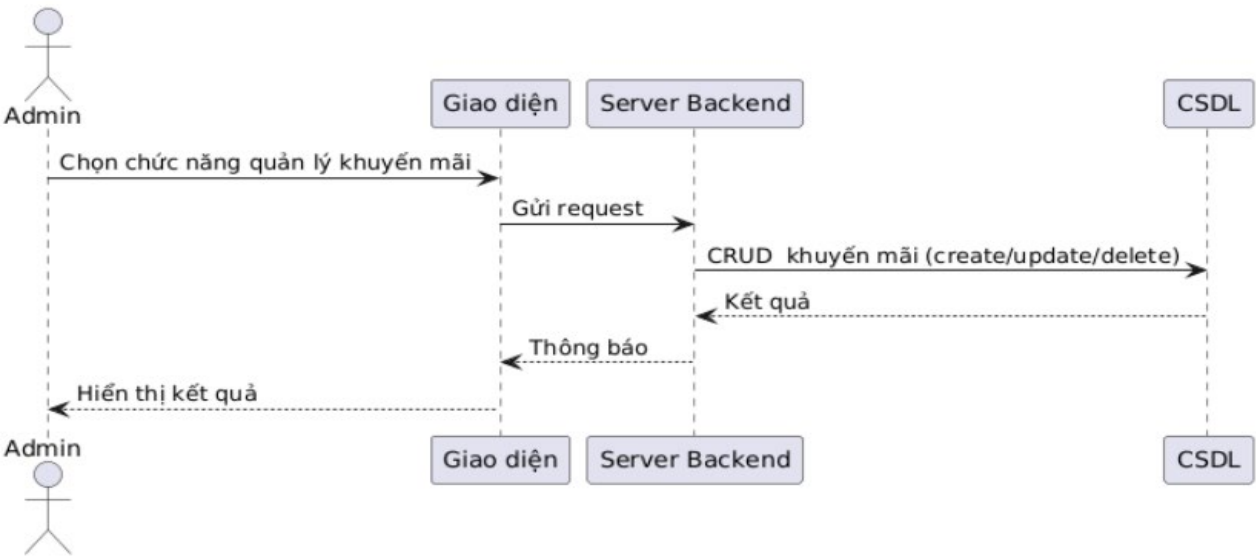
Sơ đồ 3.29: Sơ đồ quản lý danh mục sản phẩm

k. Sơ đồ tuần tự thống kê của quản trị viên



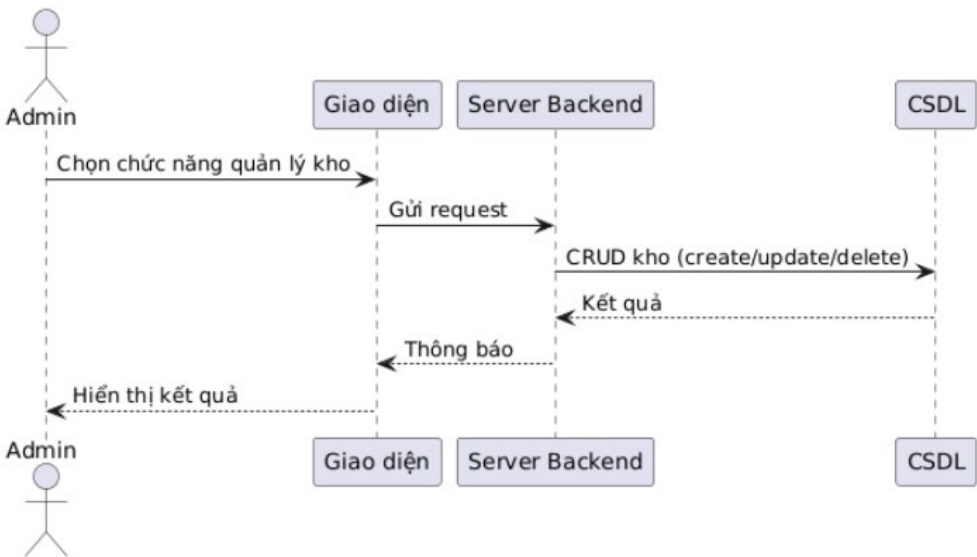
Sơ đồ 3.30: Sơ đồ tuần tự thống kê

l. Sơ đồ tuần tự quản lý khuyến mãi của admin



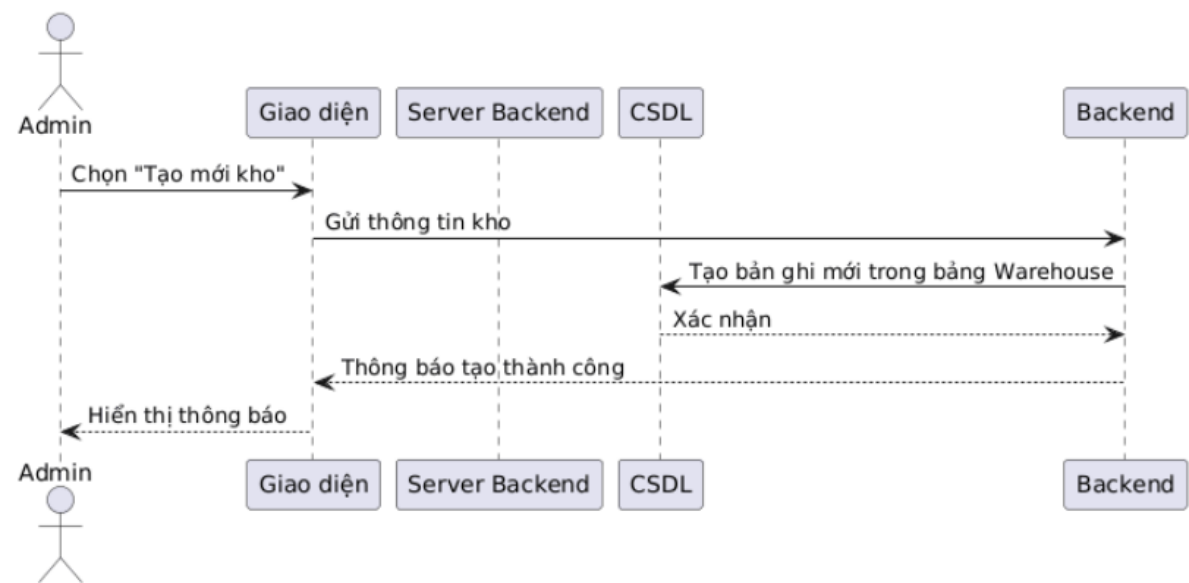
Sơ đồ 3.31: Sơ đồ tuần tự quản lý khuyến mãi

m. Sơ đồ tuần tự quản lý kho



Sơ đồ 3.32: Sơ đồ tuần tự quản lý kho

n. Sơ đồ tuần tự tạo mới kho



Sơ đồ 3.33: Sơ đồ tuần tự tạo mới kho

3.3 Thiết kế dữ liệu

Thiết kế dữ liệu đóng vai trò quan trọng trong việc xây dựng hệ thống thông tin, ảnh hưởng trực tiếp đến khả năng lưu trữ, xử lý và khai thác dữ liệu của hệ thống. Theo Elmasri & Navathe (2014), thiết kế cơ sở dữ liệu quan hệ cần đảm bảo các yếu tố tính toàn vẹn dữ liệu, tính nhất quán, khả năng mở rộng và hiệu quả truy vấn.

	<ul style="list-style-type: none"> - phone: String – Số điện thoại - googleId: String – ID đăng nhập Google (nếu có) - loginType: Enum – Loại đăng nhập (Local/Google) - createAt: DateTime – Thời gian tạo tài khoản - updateAt: DateTime – Thời gian cập nhật cuối
Phương thức	<ul style="list-style-type: none"> - login() – Xác thực đăng nhập - updateProfile() – Cập nhật thông tin cá nhân - changePassword() – Thay đổi mật khẩu
Lớp Role(Vai trò)	
Định nghĩa	Lớp quản lý các vai trò trong hệ thống (Admin, Customer, Manager...).
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - name: String – Tên vai trò - createAt: DateTime – Thời gian tạo - updateAt: DateTime – Thời gian cập nhật - deleteAt: DateTime – Thời gian xóa (soft delete)
Phương thức	<ul style="list-style-type: none"> - assignToUser(user: User) – Gán vai trò cho người dùng - removeFromUser(user: User) – Gỡ vai trò khỏi người dùng
Lớp UserRole(Người dùng -vai trò)	
Thuộc tính	<ul style="list-style-type: none"> - id: Integer – Khóa chính - userId: Long – Khóa ngoại đến User - roleId: Long – Khóa ngoại đến Role
Lớp UserAddress :Địa chỉ người dùng	
Định nghĩa	Lớp quản lý địa chỉ giao hàng của người dùng.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - userId: Long – Khóa ngoại đến User - receiverName: String – Tên người nhận - phone: String – Số điện thoại nhận hàng - addressDetail: String – Chi tiết địa chỉ - isDefault: Boolean – Đánh dấu địa chỉ mặc định

Phương thức	<ul style="list-style-type: none"> - <code>setAsDefault()</code> – Đặt làm địa chỉ mặc định - <code>validate()</code> – Kiểm tra tính hợp lệ của địa chỉ
Lớp Category (Danh mục)	
Định nghĩa	Lớp quản lý danh mục sản phẩm.
Thuộc tính	<ul style="list-style-type: none"> - <code>id</code>: Long – Khóa chính - <code>categoryName</code>: String – Tên danh mục - <code>picUrl</code>: String – URL ảnh đại diện danh mục - <code>createAt</code>: DateTime – Thời gian tạo - <code>updateAt</code>: DateTime – Thời gian cập nhật
Phương thức	<ul style="list-style-type: none"> - <code>getProducts()</code> – Lấy danh sách sản phẩm thuộc danh mục - <code>addProduct(product: Product)</code> – Thêm sản phẩm vào danh mục
Product (Sản phẩm)	
Định nghĩa	Lớp đại diện cho sản phẩm trong hệ thống.
Thuộc tính	<ul style="list-style-type: none"> - <code>id</code>: Long – Khóa chính - <code>categoryId</code>: Long – Khóa ngoại đến Category - <code>productName</code>: String – Tên sản phẩm - <code>model</code>: String – Model sản phẩm - <code>description</code>: String – Mô tả chi tiết - <code>standardCost</code>: BigDecimal – Giá gốc - <code>price</code>: BigDecimal – Giá bán - <code>productStatus</code>: Enum – Trạng thái sản phẩm (Active/Inactive/OutOfStock) - <code>picUrl</code>: String – URL ảnh sản phẩm - <code>createAt</code>: DateTime – Thời gian tạo - <code>updateAt</code>: DateTime – Thời gian cập nhật
Phương thức	<ul style="list-style-type: none"> - <code>calculateFinalPrice()</code> – Tính giá cuối cùng sau giảm giá - <code>updateStock(quantity: int)</code> – Cập nhật tồn kho - <code>isAvailable()</code> – Kiểm tra sản phẩm có sẵn hàng không
Lớp ProductSpecification (Thông số kỹ thuật)	

Định nghĩa	Lớp lưu trữ thông số kỹ thuật của sản phẩm.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - categoryId: Long – Khóa ngoại đến Category - productName: String – Tên sản phẩm - model: String – Model sản phẩm - description: String – Mô tả chi tiết - standardCost: BigDecimal – Giá gốc - price: BigDecimal – Giá bán - productStatus: Enum – Trạng thái sản phẩm (Active/Inactive/OutOfStock) - picUrl: String – URL ảnh sản phẩm - createdAt: DateTime – Thời gian tạo - updatedAt: DateTime – Thời gian cập nhật
Lớp ProductDiscount (Giảm giá sản phẩm)	
Định nghĩa	Lớp quản lý chương trình giảm giá cho từng sản phẩm.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - productId: Long – Khóa ngoại đến Product - discountAmount: BigDecimal – Số tiền giảm giá - startDate: DateTime – Ngày bắt đầu - endDate: DateTime – Ngày kết thúc - isActive: Boolean – Trạng thái kích hoạt
Phương thức	<ul style="list-style-type: none"> - isValid() – Kiểm tra giảm giá còn hiệu lực không - applyDiscount(price: BigDecimal) – Áp dụng giảm giá vào giá sản phẩm
Lớp Warehouse (Kho hàng)	
Định nghĩa	Lớp đại diện cho kho hàng trong hệ thống.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - userId: Long – Khóa ngoại đến User (quản lý kho) - warehouseName: String – Tên kho

	<ul style="list-style-type: none"> - description: String – Mô tả kho - location: String – Địa chỉ kho - createdAt: DateTime – Thời gian tạo - updatedAt: DateTime – Thời gian cập nhật
Phương thức	<ul style="list-style-type: none"> - getInventory() – Lấy danh sách tồn kho - addProduct(product: Product, quantity: int) – Thêm sản phẩm vào kho
Lớp Inventory (Tồn kho)	
Định nghĩa	Lớp quản lý số lượng sản phẩm trong kho.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - productId: Long – Khóa ngoại đến Product - warehouseId: Long – Khóa ngoại đến Warehouse - stock: Integer – Số lượng tồn kho - minStock: Integer – Mức tồn kho tối thiểu - updatedAt: DateTime – Thời gian cập nhật cuối
Phương thức	<ul style="list-style-type: none"> - updateStock(quantity: int) – Cập nhật số lượng tồn kho - checkStockLevel() – Kiểm tra mức tồn kho - isLowStock() – Kiểm tra tồn kho thấp hơn mức tối thiểu
Lớp InventoryLog (Nhật ký tồn kho)	
Định nghĩa	Lớp ghi log các thay đổi về tồn kho.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - productId: Long – Khóa ngoại đến Product - warehouseId: Long – Khóa ngoại đến Warehouse - changeType: Enum – Loại thay đổi (Nhập/Xuất/Điều chỉnh) - quantity: Integer – Số lượng thay đổi - note: String – Ghi chú - createdAt: DateTime – Thời gian tạo log - currentStock: Integer – Tồn kho hiện tại sau thay đổi
Phương thức	<ul style="list-style-type: none"> - logChange() – Ghi log thay đổi - getHistory() – Lấy lịch sử thay đổi

Lớp Order (Đơn hàng)	
Định nghĩa	Lớp đại diện cho đơn hàng của khách.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - userId: Long – Khóa ngoại đến User - orderDate: DateTime – Ngày đặt hàng - shippedDate: DateTime – Ngày giao hàng - note: String – Ghi chú đơn hàng - shipAddress: String – Địa chỉ giao hàng - shipFee: BigDecimal – Phí vận chuyển - paidDate: DateTime – Ngày thanh toán - orderStatus: Enum – Trạng thái đơn hàng - paymentType: Enum – Phương thức thanh toán - createAt: DateTime – Thời gian tạo - updateAt: DateTime – Thời gian cập nhật - addressId: Long – Khóa ngoại đến UserAddress
Phương thức	<ul style="list-style-type: none"> - calculateTotal() – Tính tổng tiền đơn hàng - updateStatus(status: OrderStatus) – Cập nhật trạng thái - cancelOrder() – Hủy đơn hàng - processPayment() – Xử lý thanh toán
Lớp OrderDetail (Chi tiết đơn hàng)	
Định nghĩa	Lớp lưu trữ thông tin từng sản phẩm trong đơn hàng.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - orderId: Long – Khóa ngoại đến Order - productId: Long – Khóa ngoại đến Product - quantity: Integer – Số lượng - unitPrice: BigDecimal – Đơn giá tại thời điểm đặt - discountPercentage: BigDecimal – Phần trăm giảm giá - discountAmount: BigDecimal – Số tiền giảm giá

Phương thức	<ul style="list-style-type: none"> - calculateSubtotal() – Tính thành tiền cho sản phẩm - applyDiscount() – Áp dụng giảm giá
Lớp Payment (Thanh toán)	
Định nghĩa	Lớp quản lý thông tin thanh toán của đơn hàng.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - orderId: Long – Khóa ngoại đến Order - paymentMethod: Enum – Phương thức thanh toán - amount: BigDecimal – Số tiền thanh toán - paymentStatus: Enum – Trạng thái thanh toán - transactionCode: String – Mã giao dịch - paidAt: DateTime – Thời gian thanh toán
Phương thức	<ul style="list-style-type: none"> - processPayment() – Xử lý thanh toán - verifyPayment() – Xác minh thanh toán - refund() – Hoàn tiền
Lớp Shipment (Vận chuyển)	
Định nghĩa	Lớp quản lý thông tin giao hàng.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - orderId: Long – Khóa ngoại đến Order - carrierName: String – Tên đơn vị vận chuyển - trackingNumber: String – Mã theo dõi - status: Enum – Trạng thái giao hàng - estimatedDelivery: Date – Ngày dự kiến giao - deliveredAt: DateTime – Thời gian giao thực tế
Phương thức	<ul style="list-style-type: none"> - updateStatus() – Cập nhật trạng thái giao hàng - trackShipment() – Theo dõi vận chuyển - calculateDeliveryDate() – Tính ngày giao dự kiến
Lớp Cart (Giỏ hàng)	
Định nghĩa	Lớp đại diện cho giỏ hàng của người dùng.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính

	<ul style="list-style-type: none"> - <code>userId</code>: Long – Khóa ngoại đến User - <code>createAt</code>: DateTime – Thời gian tạo - <code>updateAt</code>: DateTime – Thời gian cập nhật cuối
Phương thức	<ul style="list-style-type: none"> - <code>addItem(product: Product, quantity: int)</code> – Thêm sản phẩm vào giỏ - <code>removeItem(productId: Long)</code> – Xóa sản phẩm khỏi giỏ - <code>updateQuantity(productId: Long, quantity: int)</code> – Cập nhật số lượng - <code>calculateTotal()</code> – Tính tổng tiền giỏ hàng - <code>checkout()</code> – Chuyển giỏ hàng thành đơn hàng
Lớp CartDetail (Chi tiết giỏ hàng)	
Định nghĩa	Lớp lưu trữ thông tin từng sản phẩm trong giỏ hàng.
Thuộc tính	<ul style="list-style-type: none"> - <code>id</code>: Long – Khóa chính - <code>productId</code>: Long – Khóa ngoại đến Product - <code>cartId</code>: Long – Khóa ngoại đến Cart - <code>quantity</code>: Integer – Số lượng - <code>unitPrice</code>: BigDecimal – Đơn giá tại thời điểm thêm vào giỏ - <code>createAt</code>: DateTime – Thời gian thêm - <code>updateAt</code>: DateTime – Thời gian cập nhật cuối
Phương thức	<ul style="list-style-type: none"> - <code>calculateSubtotal()</code> – Tính thành tiền cho sản phẩm trong giỏ - <code>updatePrice()</code> – Cập nhật giá theo giá hiện tại của sản phẩm
Lớp Coupon (Phiếu giảm giá)	
Định nghĩa	Lớp quản lý mã giảm giá toàn hệ thống.
Thuộc tính	<ul style="list-style-type: none"> - <code>id</code>: Long – Khóa chính - <code>code</code>: String – Mã coupon - <code>discountType</code>: Enum – Loại giảm giá (Phần trăm/Cố định) - <code>discountValue</code>: BigDecimal – Giá trị giảm giá - <code>startDate</code>: DateTime – Ngày bắt đầu - <code>endDate</code>: DateTime – Ngày kết thúc - <code>usageLimit</code>: Integer – Giới hạn sử dụng - <code>usedCount</code>: Integer – Số lần đã sử dụng

	- isActive: Boolean – Trạng thái kích hoạt
Phương thức	<ul style="list-style-type: none"> - isValid() – Kiểm tra coupon còn hiệu lực không - applyToOrder(order: Order) – Áp dụng coupon vào đơn hàng - incrementUsage() – Tăng số lần sử dụng
Lớp UserCoupon (Người dùng - Coupon)	
Định nghĩa	Bảng trung gian quản lý coupon của người dùng.
Thuộc tính	<ul style="list-style-type: none"> - userId: Long – Khóa ngoại đến User - couponId: Long – Khóa ngoại đến Coupon - isUsed: Boolean – Đánh dấu đã sử dụng chưa
Phương thức	<ul style="list-style-type: none"> - markAsUsed() – Đánh dấu là đã sử dụng - isValidForUser(user: User) – Kiểm tra coupon có hợp lệ với người dùng không
Lớp UserInteraction (Tương tác người dùng)	
Định nghĩa	Lớp ghi nhận tương tác của người dùng với sản phẩm (xem, yêu thích, thêm vào giỏ, mua).
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - userId: Long – Khóa ngoại đến User - productId: Long – Khóa ngoại đến Product - interactionType: Enum – Loại tương tác - score: Integer – ĐI số tương tác (dùng cho gợi ý) - createdAt: DateTime – Thời gian tương tác
Phương thức	<ul style="list-style-type: none"> - recordInteraction() – Ghi nhận tương tác - calculateScore() – Tính điểm tương tác
Lớp ProductReview (Đánh giá sản phẩm)	
Định nghĩa	Lớp lưu trữ đánh giá và đánh giá sao của người dùng cho sản phẩm.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - userId: Long – Khóa ngoại đến User - productId: Long – Khóa ngoại đến Product - orderId: Long – Khóa ngoại đến Order (chỉ đánh giá sau khi mua)

	<ul style="list-style-type: none"> - rating: Integer – Số sao (1-5) - title: String – Tiêu đề đánh giá - content: String – Nội dung đánh giá - createdAt: DateTime – Thời gian tạo - updatedAt: DateTime – Thời gian cập nhật
Phương thức	<ul style="list-style-type: none"> - validateRating() – Kiểm tra rating hợp lệ - calculateAverageRating() – Tính điểm trung bình sản phẩm
Lớp Wishlist (Danh sách yêu thích)	
Định nghĩa	Lớp quản lý sản phẩm yêu thích của người dùng.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - userId: Long – Khóa ngoại đến User - productId: Long – Khóa ngoại đến Product - createdAt: DateTime – Thời gian thêm vào danh sách
Phương thức	<ul style="list-style-type: none"> - addToWishlist(product: Product) – Thêm sản phẩm vào danh sách yêu thích - removeFromWishlist(productId: Long) – Xóa sản phẩm khỏi danh sách
Lớp UserRecommendation (Gợi ý người dùng)	
Định nghĩa	Lớp lưu trữ kết quả gợi ý sản phẩm cho người dùng dựa trên thuật toán.
Thuộc tính	<ul style="list-style-type: none"> - id: Integer – Khóa chính - userId: Integer – Khóa ngoại đến User - productId: Integer – Khóa ngoại đến Product - score: Float – Điểm số gợi ý (độ phù hợp) - algorithm: String – Tên thuật toán gợi ý - createdAt: Timestamp – Thời gian tạo gợi ý
Phương thức	<ul style="list-style-type: none"> - generateRecommendations() – Tạo danh sách gợi ý - updateScore() – Cập nhật điểm số gợi ý
Lớp Notification (Thông báo)	
Định nghĩa	Lớp quản lý thông báo hệ thống gửi đến người dùng.

Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - userId: Long – Khóa ngoại đến User - title: String – Tiêu đề thông báo - message: String – Nội dung thông báo - isRead: Boolean – Đã đọc hay chưa - createdAt: DateTime – Thời gian tạo
Phương thức	<ul style="list-style-type: none"> - sendNotification() – Gửi thông báo - markAsRead() – Đánh dấu là đã đọc
Lớp ChatMessage (Tin nhắn chat)	
Định nghĩa	Lớp quản lý tin nhắn hỗ trợ khách hàng.
Thuộc tính	<ul style="list-style-type: none"> - id: Long – Khóa chính - userId: Long – Khóa ngoại đến User (người nhận) - senderId: Long – Người gửi (user hoặc admin) - message: String – Nội dung tin nhắn - messageType: Enum – Loại tin nhắn (Text/Image/File) - isRead: Boolean – Đã đọc hay chưa - createdAt: DateTime – Thời gian gửi - chatType: Enum – Loại chat (User-Admin/User-User) - adminId: Long – ID admin (nếu có)
Phương thức	<ul style="list-style-type: none"> - sendMessage() – Gửi tin nhắn - markAsRead() – Đánh dấu là đã đọc - getConversation() – Lấy lịch sử hội thoại

Bảng 3.19: Mô tả chi tiết thuộc tính và chi tiết

3.4 Thiết kế mô hình đề xuất sản phẩm

3.4.1 Giới thiệu bài toán

Trong các ứng dụng thương mại điện tử việc đề xuất sản phẩm có vai trò quan trọng nhằm tăng tỉ lệ mua hàng từ khách hàng, cá nhân hóa trải nghiệm khách hàng.

Để giải quyết bài toán này, em đã chọn mô hình NCF – Neural Collaborative Filtering.

3.4.2 Lý do chọn mô hình

Trong ứng dụng mua sắm đồ gia dụng, số lượng sản phẩm lớn và hành vi người dùng đa dạng. Để tăng khả năng cá nhân hóa trải nghiệm mua hàng, hệ thống cần đưa ra các gợi ý sản phẩm phù hợp với từng khách hàng dựa trên dữ liệu tương tác.

Mô hình được chọn vì:

- Học được quan hệ phi tuyến giữa User–Item.
- Kết hợp linh hoạt nhiều kiểu tương tác (purchase, rating, add_to_cart, wishlist, view).
- Cho kết quả chính xác hơn so với Matrix Factorization truyền thống.
- Dễ tích hợp vào API gợi ý để phục vụ mobile app.

3.4.3 Dữ liệu đầu vào của mô hình

Dữ liệu đầu vào là bảng user_interaction gồm các trường dữ liệu sau:

Thuộc tính	Ý nghĩa
Id	Khóa chính
UserId	Id người dùng
ProductId	Id sản phẩm
Interaction_type	Loại tương tác (view,wishlist,add_to_card,purchase
Score	Điểm số tương tác đã chuẩn hóa
Created_at	Thời gian lưu vào bảng

Bảng 3.20: Thuộc tính bảng tương tác người dùng

Dữ liệu đầu vào có 529408 dòng:

Thang điểm được chuẩn hóa

Tương tác	Điểm số
purchase	1
add_to_cart	2
rating	3
wishlist	4
view	5

Bảng 3.2: Bảng thang điểm chuẩn hóa

Để đảm bảo tính công bằng giữa các loại tương tác, mô hình chuẩn hóa điểm tương tác về miền $[0,1]$. Trường hợp người dùng mua sản phẩm thì điểm ưu tiên được thay thế theo lịch sử đơn hàng để tăng độ chính xác.

Ngoài các mẫu dương (positive samples), mô hình áp dụng negative sampling để sinh mẫu âm nhằm tạo dữ liệu cân bằng, giúp phân biệt rõ sản phẩm được quan tâm và không quan tâm.

3.4.4 Phân tích mô hình đề xuất NCF

a) Luồng hoạt động

Người dùng thao tác trên app (xem, mua, thêm giỏ hàng...)

Hệ thống lưu lại tương tác vào bảng user_interaction

Định kỳ (hoặc real-time), backend trích xuất dữ liệu và huấn luyện mô hình NCF

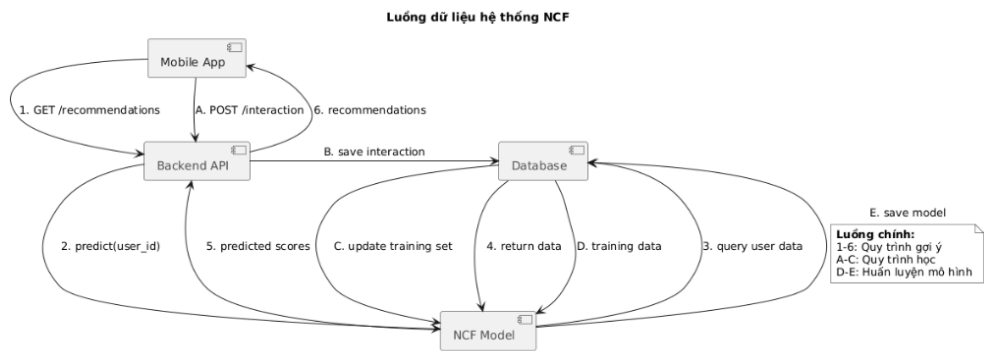
Mô hình học “embedding” người dùng và sản phẩm

NCF dự đoán mức độ phù hợp (preference score)

Backend trả về danh sách Top-N sản phẩm cho:

- Trang chủ (homepage)
- Sản phẩm liên quan (related items)
- Sản phẩm dành riêng (personalized recommendations)

a) Kiến trúc mô hình



Sơ đồ3.35: Sơ đồ luồng dữ liệu của mô hình

Thành phần	Vai trò
Mobile App	Gửi yêu cầu lấy sản phẩm gợi ý
Backend	Xử lý Api, load model, truy vấn database
MySQL	Lưu user interactions, user recomend
Mô hình NCF	Dự đoán điểm phù hợp

Bảng 3.22: Bảng mô tả luồng sự kiện mô hình

3.4.5 Cài đặt mô hình NCF

a. Nhóm hàm xử lý dữ liệu

- Hàm `get_db_connection()`: Dùng để kết nối với database
- Hàm `clean_and_load_real_data()`: Load bảng `user_interaction` từ DB và làm sạch dữ liệu.
- Hàm `prepare_database_data(df)`: Chuẩn hóa score, tính `final_score`, gom nhóm (người dùng và sản phẩm) lấy score cao nhất.
- Hàm `load_purchase_history_fallback()` Lấy lịch sử mua hàng khi thiếu dữ liệu.
- Hàm `generate_negative_samples_small(df, neg_per_pos)` Tạo negative samples để huấn luyện NCF.

b. Nhóm hàm xây dựng mô hình

- `SimpleNCF.__init__()`: Khởi tạo embedding cho người dùng và sản phẩm, khởi tạo các lớp MLP,

- `SimpleNCF.call()`: Nhận dữ liệu đầu vào `userId, productId`, Lấy embedding và nối embedding đưa qua MLP sau đó trả về dự đoán tương tác.

- `NCFRanking.__init__()`: Khởi tạo model ranking và khởi tạo loss `RankingLoss`

- `NCFRanking.compute_loss()`: Tính theo loss ranking và so sánh positive với negative score.

c. Nhóm hàm huấn luyện mô hình

- `Create_tf_dataset(df, batch_size)`: Chuyển `DataFrame` \rightarrow `TensorFlow Dataset`, `Shuffle`, `batch`, `cache`. Đây là hàm chuẩn bị dữ liệu đầu vào để train

- `Train_model(model, train_ds, val_ds, epochs)`: Chạy quá trình huấn luyện. Gọi `model.fit()` và Theo dõi loss và metri.

d. Nhóm hàm đánh giá mô hình

- `Calculate_precision_at_k`: Tính Precision@K = tỷ lệ item đúng trong top-K.

- `Calculate_recall_at_k`: Tính Recall@K = tỷ lệ item liên quan được tìm thấy trong top-K.

- `Calculate_ndcg_at_k(recommended_items, relevant_items, k)`: Tính NDCG@K : chất lượng sắp xếp gợi ý.

- `Calculate_map_at_k(all_recommendations, all_relevant_items, k)`: Tính MAP@K = trung bình precision có trọng số vị trí cho toàn bộ user.

- `Calculate_hit_rate_at_k(all_recommendations, all_relevant_items, k)`: HitRate@K = % user có ít nhất một item đúng trong top-K.

- `Calculate_coverage(recommended_items, all_items)`: Coverage = % số lượng item được hệ thống đề xuất.

- `Calculate_diversity(recommendations_by_user, item_embeddings, item_to_idx)`: Tính độ đa dạng recommendation dựa trên khoảng cách cosine giữa embedding item.

- `Calculate_novelty(recommendations_by_user, item_popularity)`: Novelty = mức độ mới mẻ, item càng ít phổ biến \rightarrow điểm càng cao.

- `Evaluate_recommendations_offline(test_df, recommendations_by_user, k_values)`: Đánh giá toàn diện mô hình gợi ý và trả về dict chứa toàn bộ metric.

- Evaluate_predictions_accuracy(y_true, y_pred):Đánh giá độ chính xác dự đoán rating:Đánh giá độ chính xác dự đoán rating qua các chỉ số MSE, RMSE,MAE,R2,Accuracy (sai số ≤ 0.1).

3.4.6 Kết luận

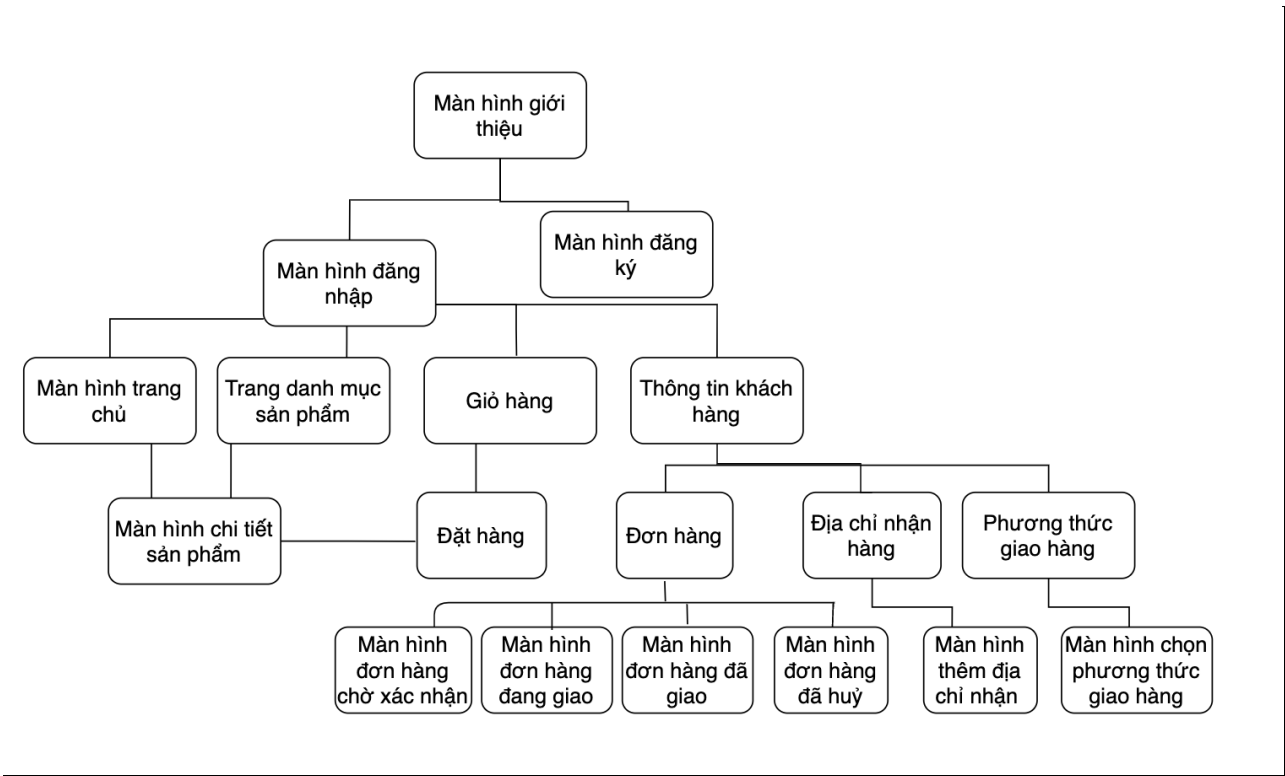
Trong đề tài “Xây dựng ứng dụng mua sắm đồ gia dụng”, mô hình NCF được sử dụng để:

- Học hành vi mua sắm thông qua bảng user_interaction (view, add_to_cart, wishlist, rating, purchase).
- Sử dụng embedding để biểu diễn mỗi khách hàng và sản phẩm trong hệ thống.
- Dự đoán mức độ quan tâm của người dùng đối với từng sản phẩm.
- Gợi ý Top sản phẩm phù hợp cho từng người dùng và lưu vào bảng user_recommendation của cơ sở dữ liệu.

Việc ứng dụng NCF giúp hệ thống gợi ý sản phẩm chính xác và mang tính cá nhân hóa cao hơn, từ đó cải thiện trải nghiệm người dùng và tăng khả năng mua hàng trong ứng dụng.(Nguyễn Văn Lộc, 2021; He et al., 2017)

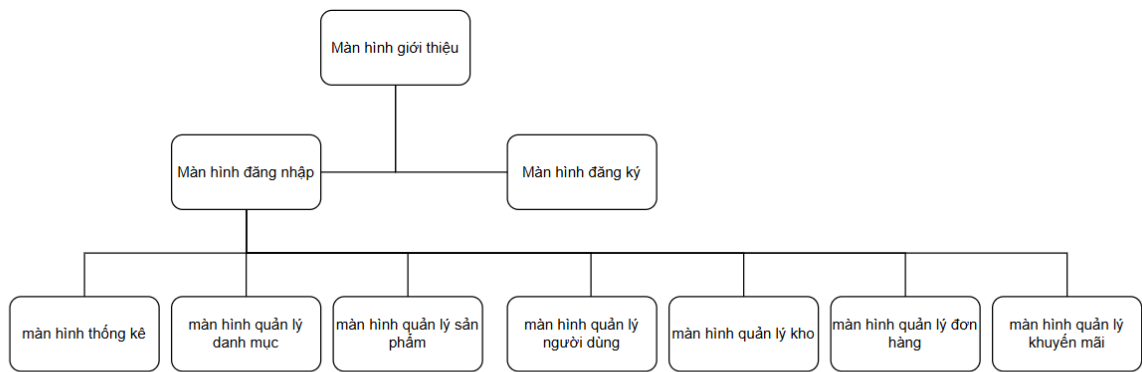
3.5 Thiết kế giao diện ứng dụng

3.5.1 Giao diện khách hàng



Sơ đồ 3.36: Sơ đồ liên kết giao diện khách hàng

3.5.2 Giao diện quản trị viên



Sơ đồ 3.37: Sơ đồ liên kết giao diện quản trị viên

3.6. Môi trường triển khai hệ thống

Hệ thống được triển khai và kiểm thử trên localhost với:

- Hệ điều hành: Microsoft Windows 11.
- Công cụ vẽ biểu đồ UML: draw.io.

- Ngôn ngữ sử dụng: kotlin, JavaScript.
- Framework sử dụng: ReactJs, Node.js + Express
- Hệ quản trị cơ sở dữ liệu: MySQL.
- IDE: Android Studio, Visual Studio Code.

3.7 Thiết kế giao diện hệ thống

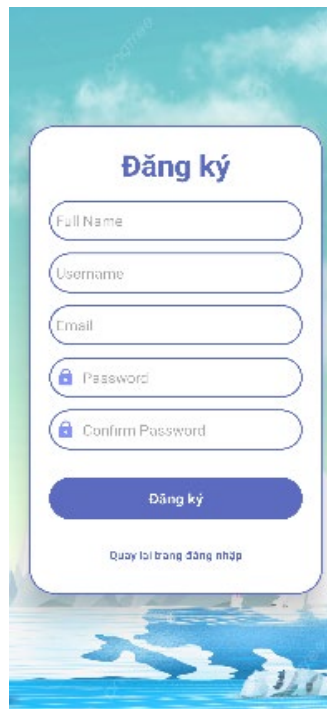
Danh sách các màn hình

STT	Tên màn hình	Ý nghĩa/Ghi chú
1	Trang đăng ký	Người dùng đăng ký tài khoản để đăng nhập vào ứng dụng.
2	Trang đăng nhập (khách hàng)	Người dùng tài khoản đã đăng nhập vào ứng dụng.
3	Trang chủ	Hiển thị tổng quan sản phẩm, banner, danh mục giúp người dùng điều hướng nhanh.
4	Trang tìm kiếm	Cho phép người dùng tìm kiếm sản phẩm theo từ khóa, bộ lọc, giá, danh mục...
5	Trang giỏ hàng	Hiển thị các sản phẩm người dùng đã thêm vào giỏ, cho phép cập nhật số lượng hoặc xóa sản phẩm.
6	Trang quản lý đơn hàng	Người dùng xem danh sách đơn hàng, trạng thái đơn, chi tiết đơn hàng.
7	Trang chat (khách hàng)	Người dùng nhắn tin trao đổi với quản lý viên/CSKH để hỗ trợ mua hàng.
8	Trang thông tin người dùng	Hiển thị và cho phép chỉnh sửa thông tin cá nhân.
9	Trang đổi mật khẩu	Người dùng đổi mật khẩu tài khoản của mình.
10	Trang yêu thích	Hiển thị danh sách sản phẩm người dùng đã đánh dấu yêu thích.
11	Trang Chi tiết sản phẩm	Hiển thị thông tin chi tiết từng sản phẩm: giá, mô tả, đánh giá, hình ảnh.

12	Trang đánh giá sản phẩm	Quản trị viên/nhân viên đăng nhập để truy cập hệ thống quản trị.
13	Trang đăng nhập (quản trị viên)	Quản trị viên/nhân viên đăng nhập để truy cập hệ thống quản trị.
14	Trang thống kê	Hiển thị biểu đồ, báo cáo doanh thu, số lượng đơn hàng, người dùng, kho...
15	Trang quản lý khách hàng	Quản trị viên theo dõi danh sách khách hàng, thông tin tài khoản, lịch sử mua hàng.
16	Trang quản lý danh mục sản phẩm	Cho phép thêm/sửa/xóa danh mục sản phẩm.
17	Trang quản lý đơn hàng	Quản lý viên trao đổi, hỗ trợ khách hàng thông qua hệ thống chat.
18	Trang chat (quản lý viên)	Quản lý viên trao đổi, hỗ trợ khách hàng thông qua hệ thống chat.
19	Trang quản lý kho	Quản trị viên xem số lượng tồn kho, điều chỉnh kho, theo dõi trạng thái sản phẩm.
20	Trang xuất kho	Thực hiện thao tác xuất hàng, cập nhật số lượng tồn kho.
21	Trang nhập kho	Thực hiện thêm số lượng hàng hóa vào kho, cập nhật dữ liệu tồn.
22	Trang lịch sử nhập xuất kho	Hiển thị lịch sử chi tiết các lần nhập/xuất kho.
23	Trang quản lý khuyến mãi	Tạo, chỉnh sửa, quản lý mã giảm giá và chương trình khuyến mãi.

Bảng 3.23: Danh sách màn hình của ứng dụng

3.7.1 Giao diện trang đăng ký

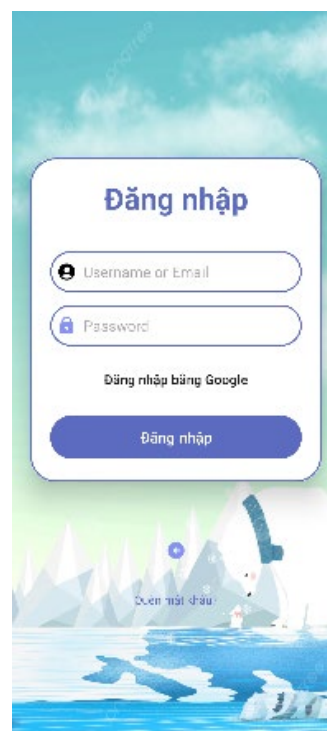


The registration page (Đăng ký) features a white rounded rectangle centered on a blue and white background with a polar bear and ice. The form includes the following fields and elements:

- Đăng ký** (Registration) title
- Full Name** input field
- Username** input field
- Email** input field
- Password** input field with a lock icon
- Confirm Password** input field with a lock icon
- Đăng ký** (Registration) button
- Quay lại trang đăng nhập** (Return to login page) link

Hình 3.1: Giao diện trang đăng ký

3.7.2 Giao diện trang đăng nhập

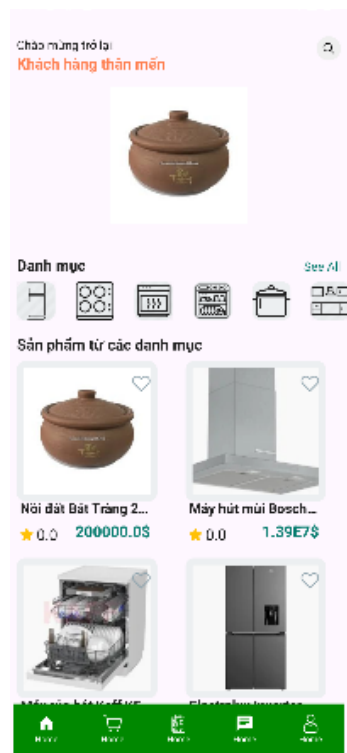


The login page (Đăng nhập) features a white rounded rectangle centered on a blue and white background with a polar bear and ice. The form includes the following fields and elements:

- Đăng nhập** (Login) title
- Username or Email** input field with a user icon
- Password** input field with a lock icon
- Đăng nhập bằng Google** (Login with Google) button
- Đăng nhập** (Login) button
- Quên mật khẩu** (Forgot password) link

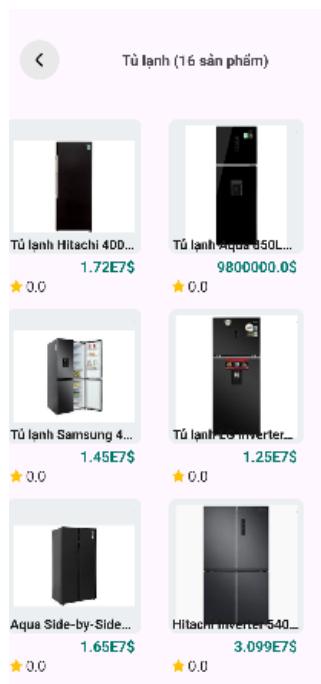
Hình 3.2: Giao diện đăng nhập

3.7.3 Giao diện trang chủ



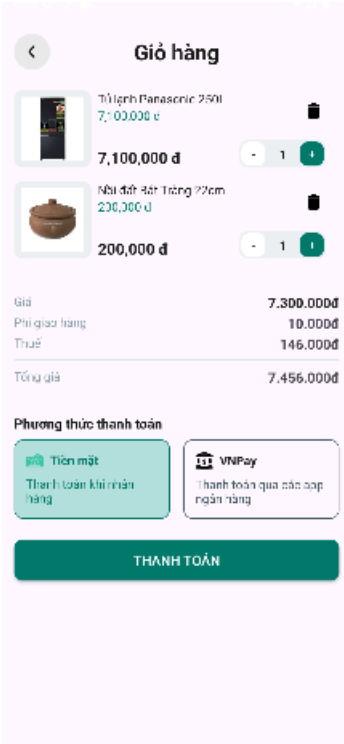
Hình 3.3:Giao diện trang chủ

3.7.4 Giao diện trang tìm kiếm



Hình 3.4: Giao diện trang chủ đề

3.7.5 Giao diện trang giỏ hàng



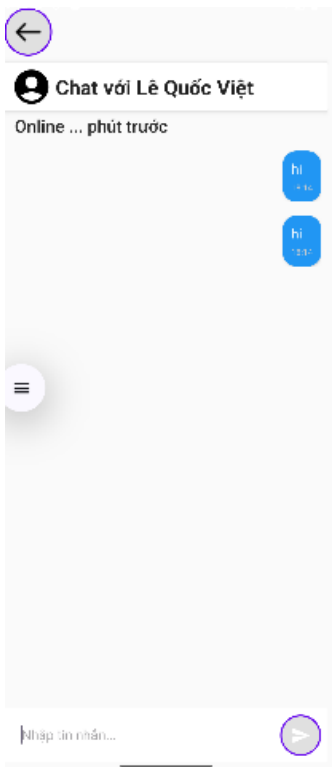
Hình 3.5: Giao diện trang giỏ hàng

3.7.6 Giao diện trang quản lý đơn hàng



Hình 3.6: Giao diện trang quản lý đơn hàng

3.7.7 Giao diện trang chat



Hình 3.7: Giao diện trang chat

3.7.8 Giao diện trang thông tin cá nhân



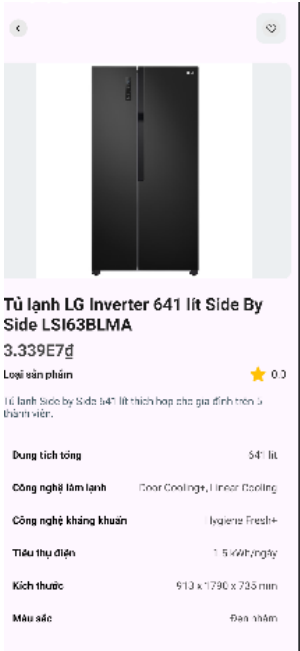
Hình 3.8 Giao diện trang thông tin cá nhân

3.7.9 Giao diện trang đổi mật khẩu



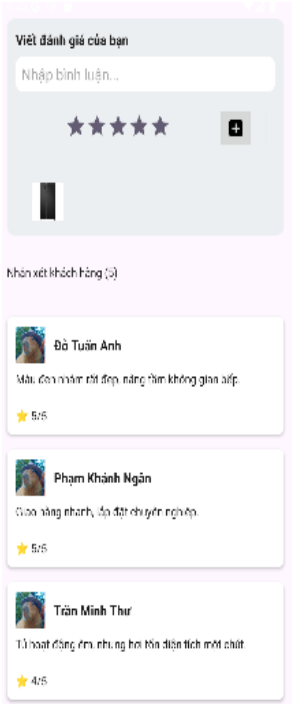
Hình 3.9 Giao diện trang đổi mật khẩu

3.7.10 Giao diện trang chi tiết sản phẩm



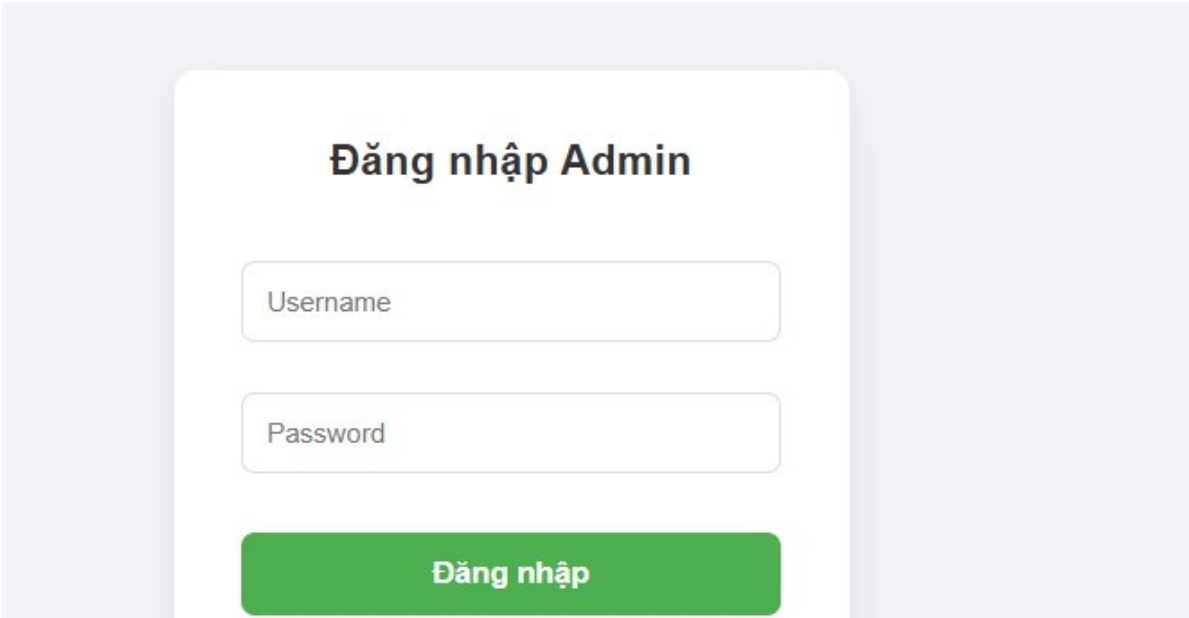
Hình 3.10:Giao diện trang chi tiết sản phẩm

3.7.11 Giao diện trang đánh giá sản phẩm



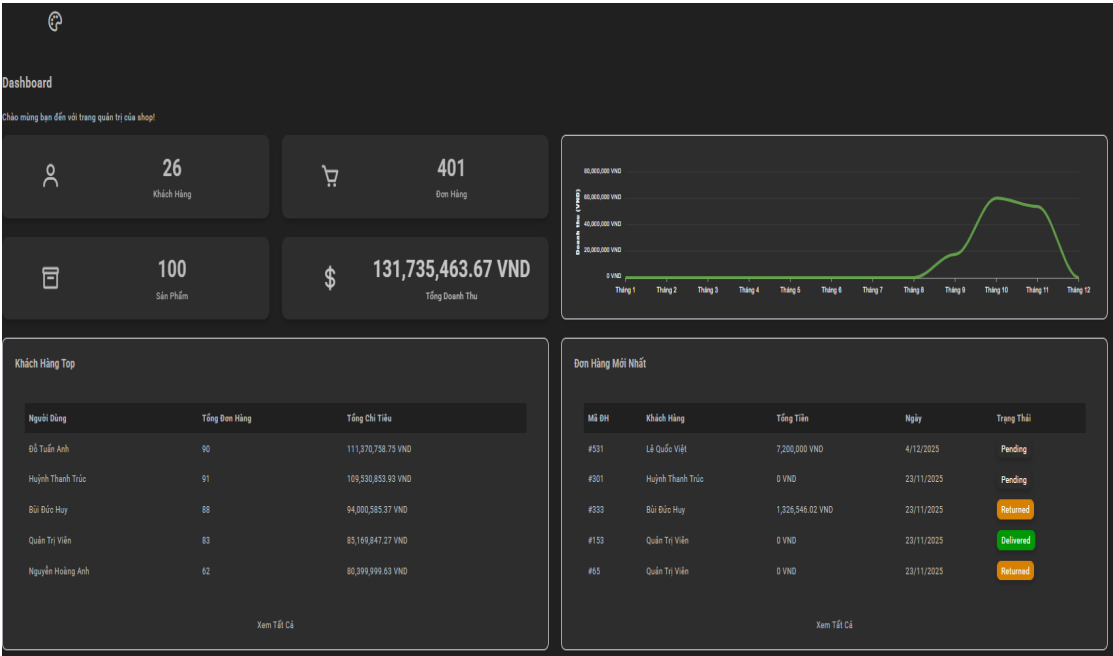
Hình 3.11: Giao diện trang đánh giá sản phẩm

3.7.12 Giao diện trang đăng nhập admin



Hình 3.12: Giao diện trang đăng nhập admin

3.7.13 Giao diện trang thống kê



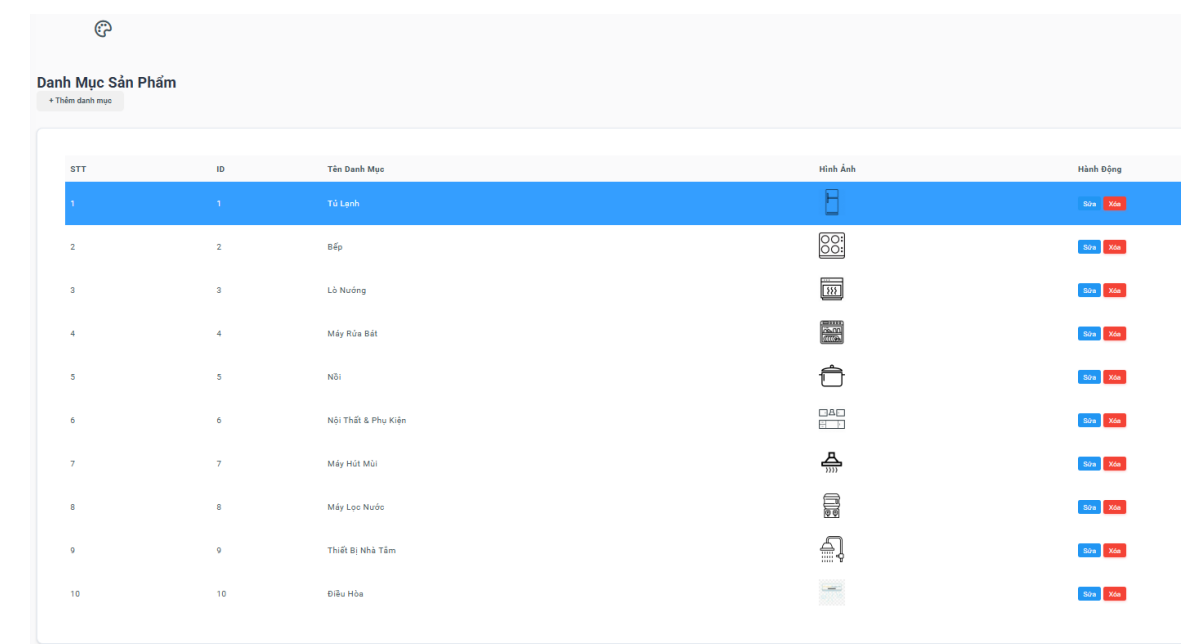
Hình 3.13: Giao diện trang thống kê

3.7.14 Giao diện trang quản lý khách hàng

Khách Hàng						
STT	Tên	Email	SĐT	Tổng Đơn Hàng	Tổng Chi Tiêu	Địa Chỉ
1	Lê Quốc Việt	Lequocviet3623@gmail.Com	0915000001	0	0 VND	Không Có Địa Chỉ
2	Nguyễn Hoàng Anh	Hoanganh@gmail.Com	0915000002	0	0 VND	Không Có Địa Chỉ
3	Trần Minh Thư	Minhthu@gmail.Com	0915000003	0	0 VND	Không Có Địa Chỉ
4	Phạm Khánh Ngân	Khanhngan@gmail.Com	0915000004	0	0 VND	Không Có Địa Chỉ
5	Đỗ Tuấn Anh	Khanhnga@gmail.Com	0915000005	0	0 VND	Không Có Địa Chỉ
6	Vũ Quang Huy	Quanghuy@gmail.Com	0915000006	0	0 VND	Không Có Địa Chỉ
7	Lê Mỹ Duyên	Myduyen@gmail.Com	0915000007	0	0 VND	Không Có Địa Chỉ
8	Huỳnh Thanh Trúc	Thanhtruc@gmail.Com	0915000008	0	0 VND	Không Có Địa Chỉ
9	Bùi Đức Huy	Duchuy@gmail.Com	0915000009	0	0 VND	Không Có Địa Chỉ
10	Quản Trị Viên	Admin@gmail.Com	0915000010	0	0 VND	Không Có Địa Chỉ

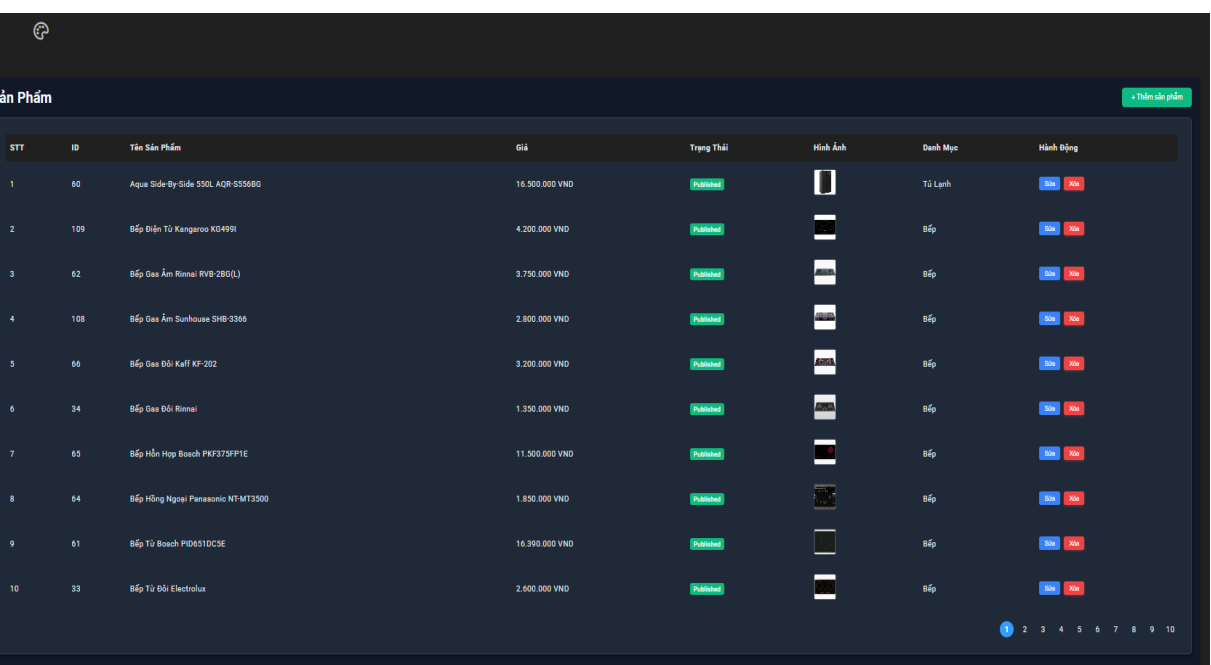
Hình 3.14: Giao diện trang quản lý khách hàng

3.7.15 Giao diện trang quản lý danh mục sản phẩm



Hình 3.15: Giao diện trang quản lý danh mục sản phẩm

3.7.16 Giao diện trang quản lý sản phẩm



Hình 3.16: Giao diện trang quản lý sản phẩm

3.7.17 Giao diện trang quản lý đơn hàng

Đơn Hàng

5. Lọc mới

STT	Mã Đơn	Khách Hàng	Tổng Tiền	Ngày Đặt	Trạng Thái	Phương Thức Thanh Toán	Hành Động
1	#531	Lê Quốc Việt	0 VND	4/12/2025	Chờ xử lý	COD	Xem
2	#301	Huỳnh Thanh Trúc	0 VND	23/11/2025	Chờ xử lý	CreditCard	Xem
3	#333	Bùi Đức Huy	0 VND	23/11/2025	Tải hàng	Momo	Xem
4	#153	Quần Tê Viên	0 VND	23/11/2025	Chờ giao	Momo	Xem
5	#39	Lê Quốc Việt	0 VND	23/11/2025	Chờ xử lý	VNPay	Xem
6	#65	Quần Tê Viên	0 VND	23/11/2025	Tải hàng	COD	Xem
7	#425	Phạm Khánh Ngân	0 VND	22/11/2025	Đang giao	Momo	Xem
8	#447	Bùi Đức Huy	0 VND	22/11/2025	Chờ giao	Momo	Xem
9	#337	Nguyễn Hoàng Anh	0 VND	22/11/2025	Chờ xử lý	PayPal	Xem
10	#351	Trần Minh Thư	0 VND	22/11/2025	Đang giao	COD	Xem

1234567891011121314151617181920212223242526272829303132333435363738394041

Hình 3.17: Giao diện trang quản lý đơn hàng

3.7.18 Giao diện trang chat

Người dùng

Lê Quốc Việt

7/12/2025

Việt Lê Quốc

7/12/2025

Việt Lê Quốc

7/12/2025

Chat với Việt Lê Quốc

1650000002@ex.hcmunre.edu.vn

← Quay lại

Việt Lê Quốc -08:01

hi

Việt Lê Quốc -08:01

hi

Việt Lê Quốc -08:08

test

Việt Lê Quốc -08:08

test

Lê Quốc Việt -08:57

Xin chào, shop hỗ trợ ạ

Lê Quốc Việt -01:02

Xin chào, shop hỗ trợ ạ

Lê Quốc Việt -01:16

hi

Việt Lê Quốc -08:29

hi

Bạn -08:14

xin chào bạn

Nhập tin nhắn...

Gửi

Hình 3.18: Giao diện trang chat

3.7.19 Giao diện trang Kho

Inventory

Tìm sản phẩm...

Tất cả kho

Làm mới

STT	ID	Sản Phẩm	Kho	Tồn	Ngưỡng	Trạng Thái	Hành Động
1	63	Tủ Lạnh LG Inverter 641 Lit Side By Side LSI63BLMA	Main Warehouse Updated	339	20	OK	<div>Nhập</div> <div>Xuất</div>
2	64	Tủ Lạnh LG Inverter 641 Lit Side By Side LSI63BLMA	Kho TP.HCM	119	20	OK	<div>Nhập</div> <div>Xuất</div>
3	90	Bếp Gas Đôi Rinnai	Kho TP.HCM	645	20	OK	<div>Nhập</div> <div>Xuất</div>
4	117	Kệ Nhà Tắm Inox 3 Tầng	Main Warehouse Updated	160	25	OK	<div>Nhập</div> <div>Xuất</div>
5	115	Vòi Sen Grohe Tầng Áp	Main Warehouse Updated	150	20	OK	<div>Nhập</div> <div>Xuất</div>
6	113	Máy Lọc Nước Kangaroo 9 Lõi	Main Warehouse Updated	180	25	OK	<div>Nhập</div> <div>Xuất</div>
7	111	Máy Lọc Nước Karofi 10 Lõi	Main Warehouse Updated	160	30	OK	<div>Nhập</div> <div>Xuất</div>
8	109	Máy Hút Mùi Faster 70cm	Main Warehouse Updated	140	25	OK	<div>Nhập</div> <div>Xuất</div>
9	107	Máy Hút Mùi Canzy 90cm	Main Warehouse Updated	150	20	OK	<div>Nhập</div> <div>Xuất</div>

Hình 3.19: Giao diện trang Kho

3.7.20 Giao diện trang xuất kho

Xuất kho

SẢN PHẨM

Aqua Side-by-Side 550L AQR-S556BG

Đã bán 100 sản phẩm, 90 dự.

Kho Side-by-Side 550L AQR-S556BG

KHO

Kho TP.HCM

SỐ LƯỢNG

1000

GHI CHÚ

Xuất kho qua lotus

Xuất kho

Hủy

Hình 3.20: Giao diện trang xuất kho

3.7.21 Giao diện trang Nhập kho

Nhập kho

SẢN PHẨM

Aqua Side-by-Side 550L AQR-S556BG

Đã tải 100 sản phẩm. Ví dụ: Aqua Side-by-Side 550L AQR-S556BG

KHO

Main Warehouse

SỐ LƯỢNG

1000

GHI CHÚ

Bổ xung thêm hàng

Nhập kho

Hủy

Hình3.21:Giao diện trang Nhập kho

3.7.22 Giao diện trang lịch sử kho

Inventory Logs

Nhật Ký Tồn Kho

Làm mới

STT	ID	Sản Phẩm	Kho	Loại	Số Lượng	Tồn Sau	Ghi Chú	Ngày
1	100	Aqua Side-By-Side 550L AQR-S556BG	Main Warehouse	NHẬP	1000	1000	Bổ Xung Thêm Hàng	07/12/2025, 19:03:24
2	99	Tủ Lạnh LG Inverter 641 Lit Side By Side LSI63BLMA	Main Warehouse Updated	XUẤT	10	339	Export For Store Display	04/12/2025, 19:35:37
3	98	Tủ Lạnh LG Inverter 641 Lit Side By Side LSI63BLMA	Main Warehouse Updated	NHẬP	100	349	Initial Import From Supplier	04/12/2025, 19:35:28
4	97	Tủ Lạnh LG Inverter 641 Lit Side By Side LSI63BLMA	Main Warehouse Updated	NHẬP	100	249	Initial Import	04/12/2025, 18:48:12
5	96	Bếp Gas Đôi Rinnai	Kho TP.HCM	NHẬP	500	645	Nhà Cung Cấp Gửi Bán	23/11/2025, 11:34:15
6	66	Tủ Lạnh LG Inverter 641 Lit Side By Side LSI63BLMA	Main Warehouse Updated	NHẬP	50	150	Test Note 01	22/11/2025, 01:17:36
7	67	Bếp Từ Đôi Kangaroo	Kho TP.HCM	XUẤT	20	130	Test Note 02	21/11/2025, 01:17:36
8	68	Lò Nướng Điện Sharp 38L	Main Warehouse Updated	XUẤT	15	145	Test Note 03	20/11/2025, 01:17:36
9	69	Máy Rửa Bát Bosch Series 4	Kho TP.HCM	NHẬP	70	200	Test Note 04	19/11/2025, 01:17:36

Hình 3.22: Giao diện trang lịch sử kho

3.7.22 Giao diện trang quản lý kho

Warehouse Management

Thêm kho mới

TÊN KHO

Kho Đà Nẵng

ĐỊA ĐIỂM

TP. Đà Nẵng

MÔ TẢ

Kho hàng chính Miền trung

Thêm

Hình 3.23: Giao diện trang quản lý kho

3.7.22 Giao diện trang quản lý khuyến mãi

Quản Lý Khuyến Mãi

Thêm khuyến mãi

STT	ID	Sản Phẩm	Giảm Giá	Ngày Bắt Đầu	Ngày Kết Thúc	Trạng Thái	Hành Động
1	3	Bếp Từ Đồi Kangaroo	1000000.0000	2025-12-01T17:00:00.000Z	2025-12-21T17:00:00.000Z	Hoạt Động	Xóa
2	1	Aqua Side-By-Side 550L AQR-S556BG	1000000.0000	2025-12-04T17:00:00.000Z	2025-12-18T17:00:00.000Z	Hoạt Động	Xóa

Hình 3.24: Giao diện trang quản lý kho

CHƯƠNG 4: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

4.1 Môi trường phát triển và môi trường triển khai

Hệ thống có thể chạy trên điện thoại Android (ứng dụng di động) và trên trình duyệt web (ứng dụng quản trị).

Công nghệ sử dụng:

Frontend Mobile: Kotlin (Android).

Frontend Web: ReactJS phiên bản 18.2.0.

Backend: Node.js kết hợp Express.

Cơ sở dữ liệu: MySQL phiên bản 8.0.63 trở lên.

Hệ thống hỗ trợ triển khai trên môi trường Local, Hosting hoặc Cloud (AWS, Vercel...).

4.2 Kết quả đạt được

Sau quá trình nghiên cứu, thiết kế và triển khai, đề tài “Phát triển ứng dụng mua sắm đồ gia dụng cho Android và hệ thống quản lý trên Web” đã đạt được các kết quả chính như.

Xây dựng thành công ứng dụng mua sắm đồ gia dụng trên Android, giao diện thân thiện, hiện đại, hỗ trợ các tính năng: duyệt sản phẩm, tìm kiếm, giỏ hàng, thanh toán, xem lịch sử đơn hàng và đánh giá sản phẩm.

Hệ thống được phát triển theo mô hình 3 lớp:

–Ứng dụng Mobile (Frontend): Sử dụng Kotlin để xây dựng giao diện và xử lý tương tác người dùng trên thiết bị Android.

–Website Quản trị (Frontend): Phát triển bằng ReactJS (v18.2.0) cho phép quản lý đơn hàng, sản phẩm, tồn kho, tài khoản người dùng.

–Backend: Xây dựng bằng Node.js + Express để xử lý nghiệp vụ, xác thực, quản lý dữ liệu và kết nối database.

–Database: MySQL dùng để lưu trữ sản phẩm, tài khoản, đơn hàng, danh mục...

Tích hợp các chức năng gợi ý sản phẩm và lọc thông minh, giúp người dùng dễ dàng tìm kiếm theo nhu cầu, thương hiệu, mức giá, hoặc đánh giá.

Áp dụng mô hình quản lý vai trò người dùng, bao gồm khách hàng, nhân viên và quản trị viên.

Mỗi vai trò có quyền truy cập chức năng riêng như mua hàng, kiểm soát kho, hoặc quản lý toàn hệ thống.

Tổ chức dữ liệu hiệu quả giữa các nền tảng:

MySQL

- Products: lưu thông tin sản phẩm đồ gia dụng.
- Categories: quản lý danh mục.
- Orders: quản lý đơn hàng và chi tiết đơn hàng.
- Users: thông tin tài khoản khách hàng và quản trị.
- Lưu hình ảnh sản phẩm hoặc gửi thông báo đẩy

Ứng dụng hoạt động ổn định trong môi trường thử nghiệm, tương thích với nhiều thiết bị Android và kết nối dữ liệu thời gian thực qua API.

4.3 Hạn chế

Trong quá trình thực hiện và triển khai, hệ thống vẫn còn tồn tại một số hạn chế như sau:

- Chưa phát triển ứng dụng cho iOS, khiến phạm vi người dùng trên thiết bị Apple còn hạn chế.
- Chưa tối ưu giao diện web và mobile cho mọi kích thước màn hình, một số màn hình nhỏ hoặc tablet có thể chưa hiển thị hoàn hảo.
- Tính năng gợi ý sản phẩm mới chỉ ở mức đơn giản, dựa trên phân loại và sản phẩm bán chạy, chưa ứng dụng machine learning nâng cao.
- Hệ thống bảo mật ở mức cơ bản, chưa có xác thực hai lớp (2FA), mã hóa nâng cao dữ liệu thanh toán hoặc sao lưu tự động.

4.4 Hướng phát triển

- Tăng cường cá nhân hóa trải nghiệm mua sắm
- Phân loại khách hàng theo mức độ hoạt động để đưa ra chương trình ưu đãi phù hợp.
- Tự động đề xuất combo sản phẩm gia dụng kèm theo.

- Tăng cường tương tác giữa khách hàng và cửa hàng
- Tích hợp tính năng chat trực tuyến hoặc chatbot tư vấn 24/7.
- Cho phép khách hàng gửi yêu cầu bảo hành, đổi trả trực tiếp trên ứng dụng.
- Đẩy thông báo tình trạng đơn hàng thời gian thực.
- Ứng dụng AI để tối ưu hoạt động bán hàng
- Dự đoán nhu cầu hàng hóa giúp quản lý kho tốt hơn.
- Chatbot AI trả lời tự động về sản phẩm, hướng dẫn sử dụng, tồn kho...
- Nâng cao tính tương tác và trải nghiệm người dùng
- Tích hợp hệ thống điểm thưởng, voucher và ưu đãi theo cấp độ thành viên.
- Thêm các mini-event và chương trình khuyến mãi theo mùa.
- Hỗ trợ xem nhanh sản phẩm dạng video review.

DANH MỤC TÀI LIỆU THAM KHẢO

Tiếng Việt:

- [1] Nguyễn Văn Ba. (2015). Phân tích và thiết kế hệ thống thông tin. Hà Nội, Việt Nam: Nhà xuất bản Đại học Quốc gia Hà Nội.
- [2] Nguyễn Văn Lộc. (2021). Hệ gợi ý dựa trên Deep Learning: Tổng quan và ứng dụng. Tạp chí Công nghệ Thông tin & Truyền thông, 12(4), 45–53.
- [3] Nguyễn Văn Minh. (2021). Phân tích thiết kế hệ thống thông tin hướng đối tượng. Hà Nội, Việt Nam: Nhà xuất bản Đại học Quốc gia Hà Nội.
- [4] Phạm Minh Tuấn. (2023). Lập trình và thực hành ReactJS. TP. Hồ Chí Minh, Việt Nam: Nhà xuất bản Khoa học và Kỹ thuật.
- [4] Phạm Trần Bích Thuận, & cộng sự. (2023). Ứng dụng Node.js và Redux trong phát triển hệ thống web/mobile. Tạp chí Khoa học và Công nghệ, 61, 1–9.
- [5] Pressman, R. S. (2016). Kỹ nghệ phần mềm – Cách tiếp cận thực hành. TP. Hồ Chí Minh, Việt Nam: Nhà xuất bản Khoa học và Kỹ thuật.
- [6] Sommerville, I. (2018). Kỹ nghệ phần mềm. Hà Nội, Việt Nam: Nhà xuất bản Thông tin và Truyền thông.
- [7] Võ Thanh Tùng. (2021). Phân tích và thiết kế hệ thống bán hàng trực tuyến (Luận văn thạc sĩ). Trường Đại học Kinh tế – Luật, Đại học Quốc gia TP. Hồ Chí Minh, Việt Nam.
- [8] Lê Quốc Bảo. (2022). Xây dựng ứng dụng mua sắm trên nền tảng di động (Luận văn thạc sĩ). Trường Đại học Công nghệ Thông tin, Đại học Quốc gia TP. Hồ Chí Minh, Việt Nam.
- [9] Lê Hoàng Long. (2023). Lập trình ứng dụng di động Android với Kotlin. TP. Hồ Chí Minh, Việt Nam: Nhà xuất bản Khoa học và Kỹ thuật.

Tiếng Anh:

- [1] Newman, S. (2021). Building mobile applications. Sebastopol, CA: O'Reilly Media.
- [2] He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. S. (2017). Neural collaborative filtering. Proceedings of the 26th International World Wide Web Conference, 173–182. <https://doi.org/10.1145/3038912.3052569>

- [3] Rendle, S. (2010). Factorization machines. Proceedings of the IEEE International Conference on Data Mining, 995–1000. <https://doi.org/10.1109/ICDM.2010.127>
- [4] Jošt, G., & Taneski, V. (2025). State-of-the-art cross-platform mobile application development frameworks: A comparative study of market and developer trends. Informatics, 12(2), 45. <https://doi.org/10.3390/informatics12020045>
- [5] Liu, Y. (2025). Demystifying React Native Android apps for static analysis. ACM Transactions on Software Engineering, 1–18.
- [6] Park, J. (2023). Recommendation systems based on neural collaborative filtering (Doctoral dissertation). Seoul National University, Seoul, South Korea.