

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP

NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

HỆ THỐNG WEBSITE QUẢN LÝ VÀ BÁN HÀNG TRỰC TUYẾN CHO CỬA HÀNG TIỆN LỢI

Người hướng dẫn: TS. NGUYỄN VĂN HÙNG

Sinh viên thực hiện: NGUYỄN VĂN CAO

Mã số sinh viên: 2151220004

Lớp: 21CT1

Đà Nẵng, 01/2026

TRƯỜNG ĐẠI HỌC KIẾN TRÚC ĐÀ NẴNG
KHOA CÔNG NGHỆ THÔNG TIN



ĐỒ ÁN TỐT NGHIỆP
NGÀNH: CÔNG NGHỆ THÔNG TIN

ĐỀ TÀI:

***HỆ THỐNG WEBSITE QUẢN LÝ VÀ
BÁN HÀNG TRỰC TUYẾN CHO CỬA
HÀNG TIỆN LỢI***

Người hướng dẫn: **TS. NGUYỄN VĂN HÙNG**

Sinh viên thực hiện: **NGUYỄN VĂN CAO**

Mã số sinh viên: **2151220004**

Lớp: **21CT1**

Đà Nẵng, 01/2026

TÓM TẮT

Tên đề tài: ... Hệ thống website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lợi...

Sinh viên thực hiện: Nguyễn Văn Cao

Mã số SV: 2151220004 Lớp: 21CT1

Đề tài xây dựng hệ thống website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lợi nhằm hỗ trợ quá trình kinh doanh, quản lý sản phẩm, đơn hàng, người dùng và thanh toán một cách hiệu quả. Hệ thống gồm hai phần: backend sử dụng Node.js để xử lý dữ liệu, xác thực người dùng, quản lý sản phẩm, đơn hàng, thanh toán qua App ngân hàng; và frontend sử dụng React để xây dựng giao diện thân thiện, dễ sử dụng cho khách hàng và quản trị viên. Dự án giúp tự động hóa quy trình bán hàng, giảm thiểu sai sót, nâng cao trải nghiệm khách hàng, đồng thời cung cấp các chức năng quản lý, thống kê doanh thu, sản phẩm, người dùng cho chủ cửa hàng. Hệ thống đảm bảo bảo mật thông tin, dễ dàng mở rộng và tích hợp các phương thức thanh toán hiện đại, phù hợp với xu hướng chuyển đổi số trong lĩnh vực bán lẻ hiện nay.

LỜI NÓI ĐẦU

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc ứng dụng các hệ thống quản lý và bán hàng trực tuyến đã trở thành xu hướng tất yếu, giúp các cửa hàng tiệm lợi nâng cao hiệu quả kinh doanh, tối ưu hóa quy trình quản lý, đồng thời đáp ứng tốt hơn nhu cầu ngày càng đa dạng của khách hàng. Việc xây dựng website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lợi không chỉ mang lại giải pháp hiện đại, tiện ích mà còn góp phần thúc đẩy quá trình chuyển đổi số trong lĩnh vực bán lẻ, tạo điều kiện cho các doanh nghiệp nhỏ tiếp cận công nghệ mới, nâng cao năng lực cạnh tranh trên thị trường.

Dự án này tập trung vào việc phát triển một hệ thống hoàn chỉnh, bao gồm các chức năng quản lý sản phẩm, đơn hàng, người dùng, thanh toán trực tuyến, cũng như cung cấp giao diện thân thiện cho cả khách hàng và quản trị viên. Quá trình thực hiện đề tài đã giúp em nâng cao kiến thức về lập trình web, hiểu rõ hơn về quy trình xây dựng và triển khai một hệ thống thực tế, đồng thời rèn luyện kỹ năng giải quyết vấn đề.

Trong suốt thời gian thực hiện, em đã nhận được sự hướng dẫn tận tình của thầy Nguyễn Văn Hưng, người đã giúp đỡ, động viên và truyền đạt những kiến thức quý báu, giúp em vượt qua những khó khăn, hoàn thành tốt nhiệm vụ được giao. Em xin gửi lời cảm ơn chân thành đến thầy, cũng như gia đình và bạn bè đã luôn ủng hộ, tạo điều kiện thuận lợi cho em trong quá trình nghiên cứu và triển khai đồ án tốt nghiệp.

Mặc dù đã cố gắng hết sức, nhưng do kiến thức và kinh nghiệm còn hạn chế, báo cáo không tránh khỏi những thiếu sót. Em rất mong nhận được sự góp ý của quý thầy cô và các bạn để hoàn thiện hơn trong tương lai. Em hy vọng rằng đề tài này sẽ mang lại những giá trị thiết thực, góp phần hỗ trợ các cửa hàng tiệm lợi trong công tác quản lý và kinh doanh.

CAM ĐOAN

Em xin cam đoan đồ án “Hệ thống website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lợi” là kết quả của quá trình học tập, nghiên cứu và thực hiện của bản thân dưới sự hướng dẫn của thầy Nguyễn Văn Hưng. Các số liệu, kết quả và nội dung trình bày trong báo cáo đều trung thực. Em hoàn toàn chịu trách nhiệm về tính trung thực của đồ án này.

Sinh viên thực hiện

Nguyễn Văn Cao

MỤC LỤC

TÓM TẮT	
LỜI NÓI ĐẦU.....	i
CAM ĐOAN.....	ii
MỤC LỤC	iii
DANH SÁCH CÁC BẢNG, HÌNH VẼ.....	vi
DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT	ix
MỞ ĐẦU.....	1
Chương 1: TỔNG QUAN VỀ CƠ SỞ LÝ THUYẾT VÀ ĐỀ TÀI	2
1.1. Tổng quan về cơ sở lý thuyết	2
1.1.1. Ngôn ngữ lập trình ReactJS và Nodejs	2
1.1.2. Cơ sở lý thuyết về MongoDB	3
1.1.3. Công cụ thiết kế (Draw.io, UML).....	5
1.1.4. RESTful API	6
1.2. Mô tả các chức năng chính của hệ thống.....	7
1.2.1. Phân tích các tính năng kỹ thuật nổi bật	7
1.2.2. Phát biểu bài toán	8
1.3. Danh sách người sử dụng hệ thống.....	9
1.3.1. Quản trị viên (Admin):.....	9
1.3.2. Khách hàng (User):	9
1.3.3. Khách truy cập (người dùng chưa đăng kí tài khoản):.....	9
1.4. Các yêu cầu của người dùng	10
Chương 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG	11
2.1. Phân tích yêu cầu hệ thống	11
2.1.1. Yêu cầu chức năng	11
2.1.2. Yêu cầu phi chức năng.....	11
2.1.3. Các ràng buộc và giả định.....	12
2.2. Mô hình hóa hệ thống.....	12
2.2.1. Sơ đồ ngữ cảnh.....	12
2.2.2. Sơ đồ luồng dữ liệu (DFD)	13
2.2.3. Sơ đồ các chức năng chính (BFD)	14
2.3. Thiết kế hệ thống.....	15
2.3.1. Thiết kế kiến trúc tổng thể.	15
2.3.2. Thiết kế cơ sở dữ liệu (ERD, bảng dữ liệu)	16

2.3.3. Nội dung thiết kế giao diện người dùng (UI/UX).....	23
2.3.4. Thiết kế các module chức năng (quản lý sản phẩm, đơn hàng, người dùng, thanh toán...).....	24
2.4. Sơ đồ hoạt động và các sơ đồ minh họa	25
2.4.1. Sơ đồ Use Case.	25
2.4.2. Sơ đồ hoạt động của các chức năng chính	30
2.4.3. Sơ đồ trình tự (Sequence Diagram)	31
2.4.4. Sơ đồ lớp (Class Diagram).....	40
2.5. Tổng kết chương	41
Chương 3: XÂY DỰNG VÀ TRIỂN KHAI HỆ THỐNG	42
3.1. Tổng quan về xây dựng và triển khai hệ thống.....	42
3.2. Môi trường và công cụ phát triển.....	42
3.2.1. Ngôn ngữ lập trình và Framework	42
3.2.2. Cơ sở dữ liệu	42
3.2.3. Công cụ hỗ trợ phát triển	42
3.2.4. Thư viện dùng để tích hợp	43
3.3. Quá trình xây dựng hệ thống.....	43
3.3.1. Khởi tạo dự án và cấu trúc thư mục	43
3.3.2. Thiết kế và xây dựng cơ sở dữ liệu	43
3.3.3. Phát triển Backend.....	47
3.3.4. Phát triển Frontend.....	48
3.3.5. Tích hợp các chức năng và dịch vụ.....	50
3.4. Xây dựng các chức năng chính	51
3.4.1. Đăng kí/Đăng nhập	51
3.4.2. Quản lý người dùng	52
3.4.3. Quản lý sản phẩm.....	53
3.4.4. Tìm kiếm sản phẩm, lựa chọn menu sản phẩm.....	55
3.4.5. Đặt hàng và quản lý đơn hàng.	56
3.4.6. Đánh giá sản phẩm.	58
3.4.7. Thanh toán.....	59
3.4.8. Thống kê doanh thu.....	60
3.5. Kiểm thử và đánh giá hệ thống.....	61
3.5.1. Mục tiêu kiểm thử	61
3.5.2. Quy trình và phương pháp kiểm thử	61
3.5.3. Kết quả kiểm thử.....	61

3.5.4. Đánh giá tổng hệ thống	61
3.6. Tổng kế chương.....	61
KẾT LUẬN	63
TÀI LIỆU THAM KHẢO.....	64

DANH SÁCH CÁC BẢNG, HÌNH VẼ

Bảng 1.1: Tính năng kỹ thuật của hệ thống “ Hệ thống website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lợi”	7
Bảng 2.1: Bảng User dữ liệu của hệ thống	18
Bảng 2.2: Bảng Product (Sản phẩm) dữ liệu hệ thống	18
Bảng 2.3: Bảng Order (Đơn hàng) dữ liệu hệ thống	19
Bảng 2.4: Bảng OrderDetail (Chi tiết đơn hàng) dữ liệu hệ thống	19
Bảng 2.5: Bảng Review (Đánh giá) dữ liệu hệ thống.....	19
Bảng 2.6: Bảng Cart (Giỏ hàng) dữ liệu hệ thống.....	20
Bảng 2.7: Bảng CartItem (Sản phẩm trong giỏ hàng)	20
Bảng 2.8: Bảng Payment (Thanh toán)	21
Bảng 2.9: Bảng Address (Địa chỉ).....	21
Bảng 2.10: Bảng Category (Danh mục sản phẩm)	21
Hình 1.1: Hình ảnh ReactJS	2
Hình 1.2: Hình ảnh Node js.	3
Hình 1.3: Hình ảnh của database MongoDB	3
Hình 1.4: Hình ảnh minh hoạ lưu trữ của MongoDB.....	4
Hình 1.5: Hình ảnh xử lý dữ liệu của MongoDB	4
Hình 1.6: Tính năng mở rộng của MongoDB	5
Hình 1.7: Hình ảnh giao diện Draw.io.....	5
Hình 1.8: Hình ảnh sơ đồ mẫu của UML Diagrams.....	6
Hình 1.9: Hình ảnh RESTful API.....	6
Hình 1.10: Sơ đồ kiến trúc hệ thống.....	8
Hình 2.1: Sơ đồ DFD ngữ cảnh	12
Hình 2.2: Sơ đồ luồng dữ liệu DFD	13
Hình 2.3: Hình ảnh sơ đồ các chức năng chính (BFD)	14
Hình 2.4: Hình ảnh thiết kế hệ thống tổng thể	15
Hình 2.5: Sơ đồ ERD hệ thống.	16
Hình 2.6: Hình sơ đồ dữ liệu dạng chuẩn thứ 3 (3NF).....	22
Hình 2.7: Sơ đồ Use Case tổng quát.....	25
Hình 2.8: Sơ đồ Use Case quản lý sản phẩm (Admin).....	26
Hình 2.9: Sơ đồ Use Case quản lý đơn hàng (Admin)	27
Hình 2.10: Sơ đồ Use Case quản lý người dùng (Admin)	28

Hình 2.11: Sơ đồ Use Case đăng kí/đăng nhập khách hàng (User).....	28
Hình 2.12: Sơ đồ Use Case chức năng mua hàng và theo dõi đơn hàng khách hàng (User)	29
Hình 2.13: Sơ đồ hoạt động các chức năng chính (Activity Diagram)	30
Hình 2.14: Sơ đồ trình tự quản lý thông kê Quản trị viên (Admin)	32
Hình 2.15: Sơ đồ trình tự quản lý sản phẩm Quản trị viên (Admin).....	33
Hình 2.16: Sơ đồ trình tự quản lý người dùng Quản trị viên (Admin)	34
Hình 2.17: Sơ đồ trình tự quản lý đơn hàng của Quản trị viên (Admin)	35
Hình 2.18: Sơ đồ trình tự đăng kí/đăng nhập của khách hàng (User)	36
Hình 2.19: Sơ đồ trình tự xem danh sách sản phẩm khách hàng (User)	36
Hình 2.20: Sơ đồ trình tự đặt hàng của khách hàng (User)	37
Hình 2.21: Sơ đồ trình tự chọn phương thức thanh toán khách hàng (User).	38
Hình 2.22: Sơ đồ trình tự quản lý thông tin các nhân khách hàng (User).....	39
Hình 2.23: Sơ đồ Lớp Class Diagram tổng quát	40
Hình 3.1: Bố cục giao diện Backend và Frontend.....	43
Hình 3.2: Các Collection chính của hệ thống trong CSDL MongoDB	44
Hình 3.3: Các thuộc tính của đối tượng người dùng (Users)	44
Hình 3.4: Các thuộc tính của đối tượng sản phẩm (Products).....	45
Hình 3.5: Các thuộc tính của đối tượng đơn hàng (Orders)	46
Hình 3.6: Bố cục của thư mục Models trong Backend.....	47
Hình 3.7: Các API RESTful cho các chức năng trong mục Controllers	47
Hình 3.8: Các routes tổ chức trong mục Backend.....	48
Hình 3.9: Các middlewares được tổ chức trong mục Backend	48
Hình 3.10: Bố cục các Components nhỏ theo các chức năng khác nhau	49
Hình 3.11: Bố cục các reducer và action trong Frontend	49
Hình 3.12: Sơ đồ giao tiếp của Axios trong hệ thống	50
Hình 3.13: Giao diện đăng kí.....	51
Hình 3.14: Giao diện đăng nhập.....	51
Hình 3.15: Giao diện quản lý danh sách người dùng	52
Hình 3.16: Giao diện chỉnh sửa người dùng.....	52
Hình 3.17: Giao diện quản lý danh sách sản phẩm	53
Hình 3.18: Giao diện thêm sản phẩm	54
Hình 3.19: Giao diện xoá sản phẩm.	55
Hình 3.20: Giao diện tìm kiếm sản phẩm.....	55
Hình 3.21: Giao diện menu lựa chọn loại sản phẩm.	55

Hình 3.22:	Giao diện giỏ hàng.....	56
Hình 3.23:	Giao diện xác nhận thông tin vận chuyển	56
Hình 3.24:	Giao diện danh sách đơn hàng.....	57
Hình 3.25:	Giao diện quản lý đơn hàng (Admin).....	57
Hình 3.26:	Giao diện quản lý/cập nhật đơn hàng.	58
Hình 3.27:	Giao diện đánh giá sản phẩm.....	58
Hình 3.28:	Giao diện hiển thị đánh giá người dùng	58
Hình 3.29:	Giao diện chọn phương thức thanh toán	59
Hình 3.30:	Giao diện thanh toán qua ngân hàng	59
Hình 3.31:	Giao diện đặt hàng/thanh toán thành công	60
Hình 3.32:	Giao diện thống kê doanh thu.....	60

DANH SÁCH CÁC KÝ HIỆU, CHỮ VIẾT TẮT

KÝ HIỆU:

- DFD (Data Flow Diagram): Sơ đồ luồng dữ liệu, dùng để mô tả luồng thông tin giữa các tác nhân và hệ thống.
- BFD (Business Function Diagram): Sơ đồ các chức năng chính, dùng để phân tích và phân cấp các chức năng nghiệp vụ của hệ thống.
- ERD (Entity-Relationship Diagram): Sơ đồ quan hệ thực thể, dùng để thiết kế cơ sở dữ liệu, mô tả mối quan hệ giữa các thực thể.
- 3NF (Third Normal Form): Chuẩn hóa cơ sở dữ liệu dạng thứ 3, loại bỏ dư thừa dữ liệu và đảm bảo tính nhất quán.
- UI/UX (User Interface/User Experience): Giao diện người dùng/Trải nghiệm người dùng, ký hiệu dùng để chỉ thiết kế giao diện và tương tác.
- → (Arrow (Mũi tên)): Ký hiệu chỉ hướng luồng dữ liệu hoặc quy trình trong sơ đồ DFD, Activity Diagram.
- [] (Bracket (Ngoặc vuông)): Ký hiệu dùng trong sơ đồ Use Case để chỉ điều kiện hoặc ghi chú (ví dụ: trạng thái đơn hàng).
- () (Parenthesis (Ngoặc đơn)): Ký hiệu dùng trong sơ đồ Sequence để chỉ tham số hoặc thông điệp (ví dụ: yêu cầu API).

CHỮ VIẾT TẮT:

- ReactJS (React JavaScript): Thư viện JavaScript dùng xây dựng giao diện frontend, lập lại trong phần công nghệ và triển khai.
- NodeJSNode JavaScriptMôi trường runtime JavaScript cho backend, thường dùng với Express để xử lý API.
- MongoDB (Mongo Database: Hệ quản trị cơ sở dữ liệu NoSQL, dùng lưu trữ dữ liệu sản phẩm, đơn hàng, người dùng.
- UML (Unified Modeling Language): Ngôn ngữ mô hình hóa thống nhất, dùng thiết kế sơ đồ Use Case, Sequence, Class Diagram.
- API (Application Programming Interface): Giao diện lập trình ứng dụng, đặc biệt RESTful API để kết nối frontend-backend.
- RESTful (Representational State Transfer): Kiến trúc API dựa trên HTTP, dùng trao đổi dữ liệu JSON giữa các thành phần hệ thống.
- COD (Cash On Delivery): Thanh toán khi nhận hàng, một phương thức thanh toán phổ biến trong module thanh toán.

- VS Code (Visual Studio Code): Môi trường phát triển mã nguồn, dùng lập trình Node.js và ReactJS.
- Git (Global Information Tracker): Hệ thống kiểm soát phiên bản mã nguồn, dùng quản lý dự án.
- Axios (Axios HTTP Client): Thư viện gửi yêu cầu HTTP từ frontend đến backend.
- Redux (Redux State Management): Thư viện quản lý trạng thái ứng dụng React, dùng đồng bộ dữ liệu giao diện.
- QR (Quick Response): Mã QR cho thanh toán ngân hàng, tích hợp trong module thanh toán trực tuyến.
- TS. (Tiến sĩ): Chữ viết tắt học vị của người hướng dẫn (TS. Nguyễn Văn Hưng).
- SV (Sinh viên): Chữ viết tắt chỉ sinh viên thực hiện (mã SV: 2151220004).

MỞ ĐẦU

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, việc ứng dụng các hệ thống quản lý và bán hàng trực tuyến đã trở thành xu hướng tất yếu, giúp các cửa hàng tiệm lợi nâng cao hiệu quả kinh doanh, tối ưu hóa quy trình quản lý và đáp ứng tốt hơn nhu cầu của khách hàng. Đề tài “Xây dựng website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lợi” được thực hiện nhằm mang lại giải pháp hiện đại, tiện ích, góp phần thúc đẩy quá trình chuyển đổi số trong lĩnh vực bán lẻ.

Mục đích của đề tài là xây dựng một hệ thống website hỗ trợ quản lý sản phẩm, đơn hàng, người dùng và thanh toán trực tuyến, giúp cửa hàng tiệm lợi tự động hóa quy trình kinh doanh, giảm thiểu sai sót và nâng cao trải nghiệm khách hàng. Mục tiêu hướng đến là phát triển một website thân thiện, dễ sử dụng cho cả khách hàng và quản trị viên, đảm bảo tính bảo mật, dễ dàng mở rộng và tích hợp phương thức thanh toán hiện đại như App ngân hàng.

Phạm vi nghiên cứu của đề tài tập trung vào các chức năng chính như quản lý sản phẩm, đơn hàng, người dùng, thanh toán trực tuyến, thống kê doanh thu và hỗ trợ quản trị viên trong công tác quản lý. Đối tượng nghiên cứu là các cửa hàng tiệm lợi có nhu cầu ứng dụng hệ thống quản lý và bán hàng trực tuyến, cũng như khách hàng sử dụng dịch vụ mua sắm online.

Phương pháp nghiên cứu được sử dụng bao gồm khảo sát thực tế hoạt động tại cửa hàng tiệm lợi, phân tích yêu cầu, thiết kế hệ thống, lập trình, kiểm thử và đánh giá sản phẩm. Quá trình thực hiện đề tài giúp người thực hiện nâng cao kiến thức về lập trình web, hiểu rõ hơn về quy trình xây dựng và triển khai một hệ thống thực tế.

Cấu trúc đồ án gồm các phần: tổng quan đề tài, cơ sở lý thuyết, phân tích và thiết kế hệ thống, triển khai và kiểm thử, đánh giá kết quả và hướng phát triển trong tương lai. Nội dung báo cáo được trình bày logic, khoa học nhằm làm rõ quá trình nghiên cứu, xây dựng và ứng dụng hệ thống vào thực tiễn.

Chương 1: TỔNG QUAN VỀ CƠ SỞ LÝ THUYẾT VÀ ĐỀ TÀI

1.1. Tổng quan về cơ sở lý thuyết

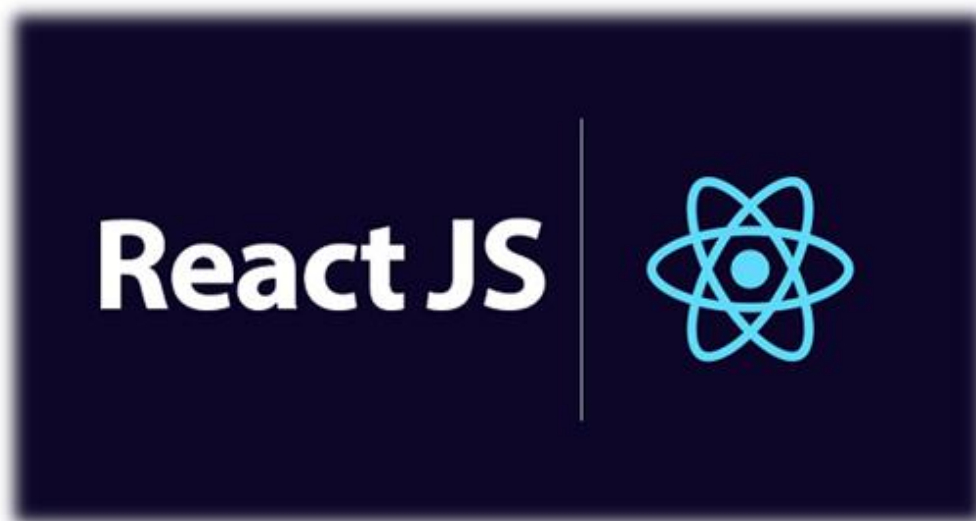
Trong thời đại công nghệ thông tin phát triển mạnh mẽ, các hệ thống quản lý và bán hàng trực tuyến ngày càng đóng vai trò quan trọng trong hoạt động kinh doanh của các cửa hàng tiệm lẻ. Việc ứng dụng website vào quản lý và bán hàng không chỉ giúp tối ưu hóa quy trình làm việc, giảm thiểu sai sót mà còn nâng cao trải nghiệm khách hàng, đáp ứng nhu cầu mua sắm mọi lúc, mọi nơi.

Các hệ thống này thường được xây dựng dựa trên nền tảng web, sử dụng các công nghệ hiện đại như Node.js, React, MongoDB, giúp đảm bảo tính linh hoạt, bảo mật và khả năng mở rộng. Việc nghiên cứu và phát triển website quản lý và bán hàng trực tuyến là một hướng đi phù hợp với xu thế chuyển đổi số hiện nay, góp phần nâng cao năng lực cạnh tranh cho các doanh nghiệp bán lẻ.

1.1.1. Ngôn ngữ lập trình ReactJS và Nodejs

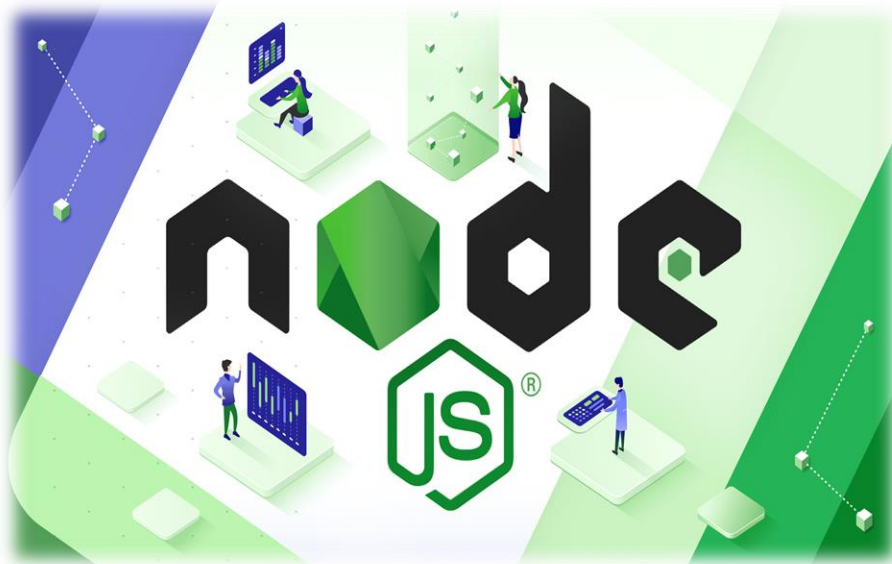
Đề tài " Hệ thống website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lẻ" sử dụng ReactJS và Node.js, hai công nghệ hiện đại giúp xây dựng các ứng dụng web động và mạnh mẽ:

- *ReactJS*: Là thư viện JavaScript dùng để xây dựng giao diện người dùng. ReactJS cho phép tạo ra các thành phần giao diện động, tái sử dụng được và mang lại trải nghiệm mượt mà cho người dùng. ReactJS sẽ được sử dụng cho phần giao diện khách hàng, nơi người dùng có thể xem sản phẩm, thêm vào giỏ hàng và thanh toán.



Hình 1.1: Hình ảnh ReactJS

- *Node.js*: Là môi trường chạy JavaScript trên server, hỗ trợ phát triển backend và xử lý các yêu cầu từ phía client. Node.js giúp xử lý các thao tác như quản lý đơn hàng, xử lý thanh toán và cập nhật kho hàng. Node.js kết hợp với MongoDB giúp quản lý dữ liệu nhanh chóng và ổn định. Bằng việc kết hợp ReactJS và Node.js, hệ thống sẽ có khả năng xử lý yêu cầu người dùng và hiển thị thông tin một cách nhanh chóng, nâng cao trải nghiệm người dùng.



Hình 1.2: Hình ảnh Node js.

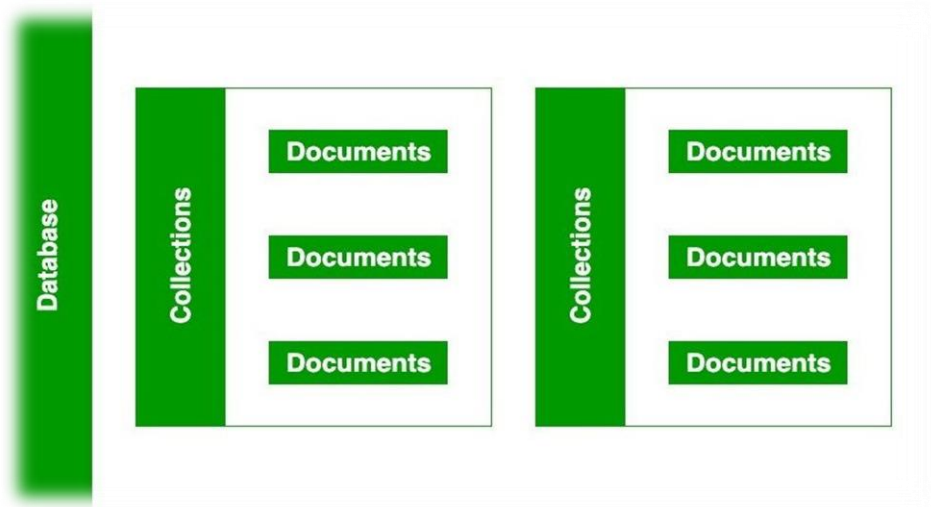
1.1.2. Cơ sở lý thuyết về MongoDB



Hình 1.3: Hình ảnh của database MongoDB

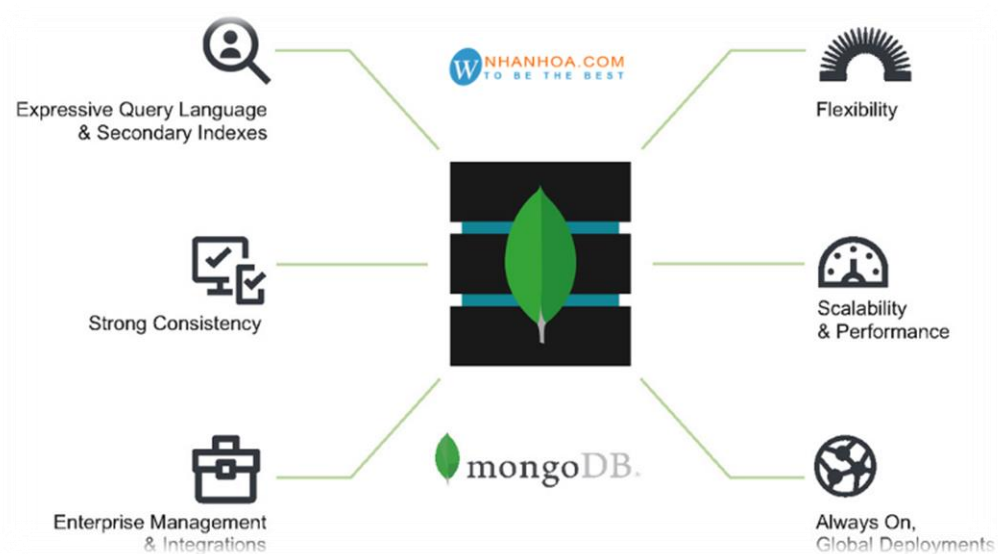
MongoDB là hệ quản trị cơ sở dữ liệu NoSQL mạnh mẽ, phù hợp với hệ thống quản lý dữ liệu phi cấu trúc và cho phép lưu trữ, truy xuất dữ liệu nhanh chóng. Các đặc điểm chính của MongoDB trong hệ thống này bao gồm:

- *Lưu trữ dạng document*: MongoDB lưu trữ dữ liệu dưới dạng JSON-like document, giúp tổ chức dữ liệu linh hoạt và mở rộng dễ dàng.



Hình 1.4: Hình ảnh minh họa lưu trữ của MongoDB

- *Khả năng xử lý dữ liệu lớn*: MongoDB được tối ưu hóa để xử lý một lượng lớn dữ liệu, phù hợp cho các trang web có lượng người dùng truy cập cao.



Hình 1.5: Hình ảnh xử lý dữ liệu của MongoDB

- *Tính năng mở rộng*: MongoDB cho phép mở rộng hệ thống theo chiều ngang (horizontal scaling), giúp duy trì hiệu suất hệ thống khi có nhiều người dùng. MongoDB sẽ lưu trữ thông tin sản phẩm, khách hàng và đơn hàng, đồng thời hỗ trợ truy vấn dữ liệu nhanh chóng cho các tác vụ hiển thị sản phẩm và quản lý hàng tồn.

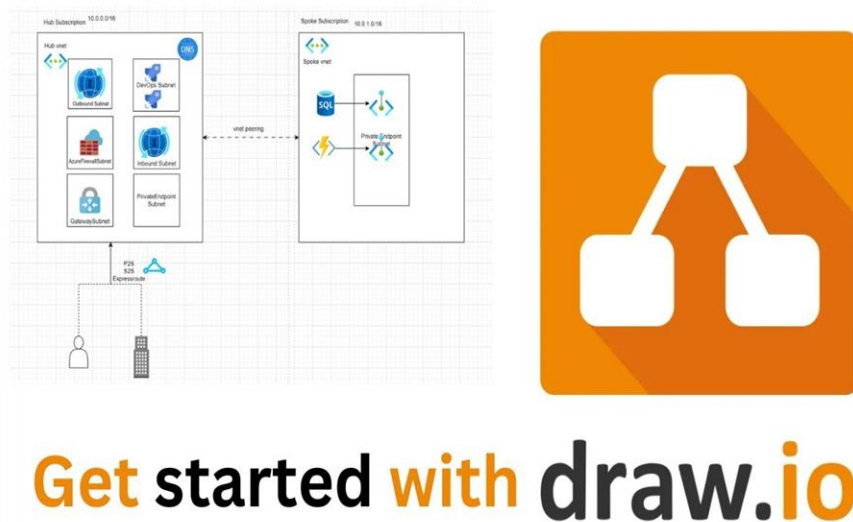


Hình 1.6: Tính năng mở rộng của MongoDB

1.1.3. Công cụ thiết kế (Draw.io, UML)

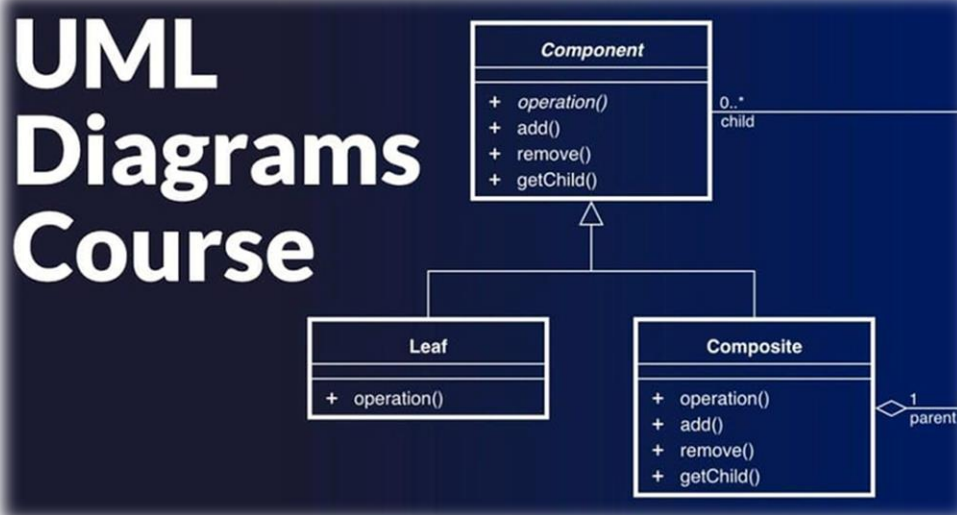
Việc sử dụng các công cụ thiết kế giúp tổ chức và trực quan hóa hệ thống một cách hiệu quả. Trong đề tài này:

- *Draw.io*: Công cụ thiết kế trực tuyến giúp xây dựng các sơ đồ luồng dữ liệu, biểu đồ quan hệ thực thể và các sơ đồ khác để mô tả mô hình hệ thống.



Hình 1.7: Hình ảnh giao diện Draw.io

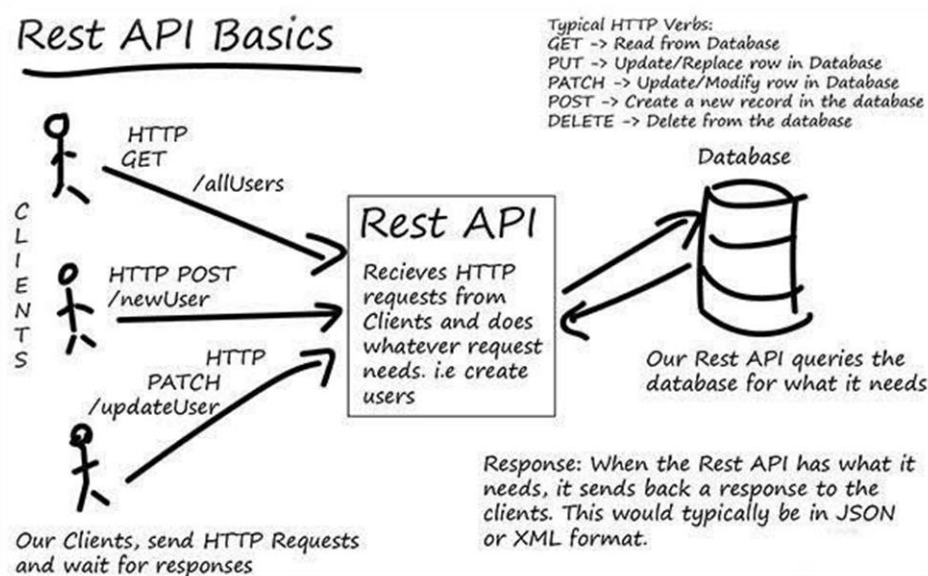
- *UML (Unified Modeling Language)*: Là ngôn ngữ mô hình hóa trực quan giúp thiết kế các biểu đồ lớp (class diagram), biểu đồ trình tự (sequence diagram) và biểu đồ use case. UML hỗ trợ trong việc phân tích và thiết kế hệ thống trước khi triển khai, giúp các thành phần hệ thống hoạt động mạch lạc và nhất quán. Các công cụ này sẽ được áp dụng trong suốt quá trình phát triển hệ thống, từ khâu thiết kế ban đầu đến khâu kiểm thử và triển khai, đảm bảo tính logic và chính xác của các thành phần trong hệ thống.



Hình 1.8: Hình ảnh sơ đồ mẫu của UML Diagrams

1.1.4. RESTful API

RESTful API (Representational State Transfer Application Programming Interface) là một kiểu kiến trúc phần mềm dùng để xây dựng các dịch vụ web, cho phép các hệ thống giao tiếp với nhau thông qua giao thức HTTP. REST tuân thủ các nguyên tắc như: sử dụng các phương thức HTTP (GET, POST, PUT, DELETE), tài nguyên được xác định bằng URL, dữ liệu trao đổi thường ở dạng JSON hoặc XML.



Hình 1.9: Hình ảnh RESTful API

Trong dự án website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lẻ, RESTful API đóng vai trò cầu nối giữa frontend (ReactJS) và backend (NodeJS). Frontend gửi các yêu cầu HTTP đến các endpoint của backend để thực hiện các chức năng như lấy

danh sách sản phẩm, tạo đơn hàng, đăng nhập, đăng ký, xử lý thanh toán... Backend nhận yêu cầu, xử lý dữ liệu và trả về kết quả cho frontend.

Việc sử dụng RESTful API giúp hệ thống dễ dàng mở rộng, bảo trì, tích hợp với các dịch vụ khác và đảm bảo tính nhất quán trong giao tiếp giữa các thành phần của ứng dụng.

1.2. Mô tả các chức năng chính của hệ thống

1.2.1. Phân tích các tính năng kỹ thuật nổi bật

“ Hệ thống website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lợi ” được xây dựng với nhiều tính năng kỹ thuật hiện đại, đáp ứng nhu cầu quản lý và kinh doanh hiệu quả.

Bảng 1.1: Tính năng kỹ thuật của hệ thống “ Hệ thống website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lợi ”

Tính năng	Mô tả chức năng	Công nghệ sử dụng
Quản lý sản phẩm	Thêm, sửa, xóa, tìm kiếm sản phẩm	Nodejs, MongoDB
Quản lý đơn hàng	Tạo, xử lý, theo dõi đơn hàng	Nodejs, MongoDB
Quản lý người dùng	Đăng kí, đăng nhập, phân quyền	JWT, Bcrypt

Mô tả:

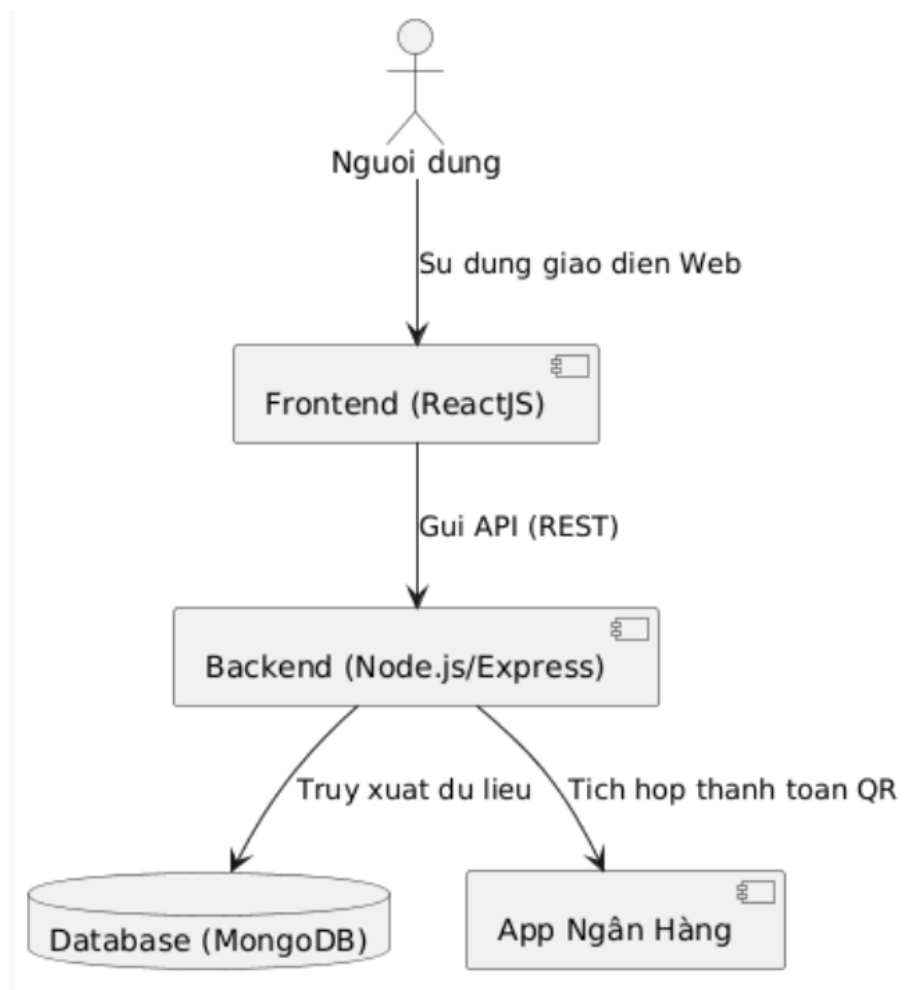
- *Quản lý sản phẩm:* Cho phép thêm mới, chỉnh sửa, xóa và tìm kiếm sản phẩm một cách nhanh chóng, giúp cửa hàng dễ dàng cập nhật thông tin hàng hóa.

- *Quản lý đơn hàng:* Hỗ trợ tạo đơn hàng, xử lý trạng thái đơn hàng, theo dõi quá trình giao nhận và cập nhật thông tin đơn hàng cho khách hàng.

- *Quản lý người dùng:* Phân quyền truy cập cho quản trị viên và khách hàng, đảm bảo bảo mật thông tin và thuận tiện trong quá trình sử dụng.

- *Thanh toán trực tuyến:* Tích hợp thanh toán qua ngân hàng, giúp khách hàng thực hiện giao dịch nhanh chóng, an toàn.

- *Thống kê doanh thu:* Cung cấp các báo cáo, biểu đồ thống kê doanh thu, sản phẩm bán chạy, hỗ trợ chủ cửa hàng trong việc phân tích và ra quyết định kinh doanh.



Hình 1.10: Sơ đồ kiến trúc hệ thống

Sơ đồ trên mô tả kiến trúc tổng thể của hệ thống cửa hàng tiện lợi tích hợp thanh toán QR. Người dùng tương tác với hệ thống thông qua giao diện web (Frontend – ReactJS). Frontend gửi các yêu cầu đến Backend (Node.js/Express) để xử lý nghiệp vụ, xác thực, truy xuất dữ liệu. Backend kết nối với cơ sở dữ liệu (MongoDB) để lưu trữ và truy xuất thông tin, đồng thời tích hợp với ứng dụng ngân hàng (App Ngân Hàng) để thực hiện các giao dịch thanh toán QR. Kiến trúc này giúp hệ thống hoạt động hiệu quả, đảm bảo tính bảo mật, dễ dàng.

1.2.2. Phát biểu bài toán

Bài toán đặt ra là xây dựng một “ Hệ thống website quản lý và bán hàng trực tuyến cho cửa hàng tiện lợi ”, giúp tự động hóa các quy trình kinh doanh, quản lý sản phẩm, đơn hàng, người dùng và thanh toán. Hệ thống cần đảm bảo tính bảo mật, dễ sử dụng, hỗ trợ quản trị viên trong việc quản lý cửa hàng và mang lại trải nghiệm mua sắm thuận tiện cho khách hàng.

1.3. Danh sách người sử dụng hệ thống

1.3.1. Quản trị viên (Admin):

- *Quản lý sản phẩm*: Thực hiện các chức năng thêm mới sản phẩm, chỉnh sửa thông tin sản phẩm, xóa sản phẩm không còn kinh doanh, cập nhật giá cả, hình ảnh và mô tả sản phẩm.

- *Quản lý đơn hàng*: Theo dõi danh sách các đơn hàng được đặt, xử lý đơn hàng (xác nhận, hủy, giao hàng), cập nhật trạng thái đơn hàng (chờ xử lý, đang giao, đã hoàn thành).

- *Quản lý người dùng*: Phân quyền truy cập cho các tài khoản, khóa hoặc mở tài khoản người dùng khi cần thiết, xem thông tin chi tiết của từng khách hàng để hỗ trợ hoặc xử lý các vấn đề phát sinh.

- *Thống kê và báo cáo*: Xem các báo cáo doanh thu theo ngày, tháng, năm; thống kê sản phẩm bán chạy, lượng khách hàng truy cập; xuất báo cáo phục vụ công tác quản lý và ra quyết định kinh doanh.

1.3.2. Khách hàng (User):

- *Đăng ký tài khoản*: Tạo tài khoản mới với thông tin cá nhân, xác thực email hoặc số điện thoại.

- *Đăng nhập hệ thống*: Truy cập vào hệ thống bằng tài khoản đã đăng ký để sử dụng các chức năng nâng cao.

- *Xem và tìm kiếm sản phẩm*: Duyệt danh mục sản phẩm, xem chi tiết sản phẩm, sử dụng chức năng tìm kiếm để tìm sản phẩm mong muốn nhanh chóng.

- *Thêm sản phẩm vào giỏ hàng*: Lựa chọn sản phẩm và thêm vào giỏ hàng để chuẩn bị đặt hàng.

- *Đặt hàng trực tuyến*: Thực hiện các bước đặt hàng, nhập thông tin giao nhận, xác nhận đơn hàng.

- *Thanh toán trực tuyến*: Lựa chọn phương thức thanh toán hiện đại như App ngân hàng để hoàn tất giao dịch một cách an toàn, tiện lợi.

- *Theo dõi trạng thái đơn hàng*: Xem tiến trình xử lý đơn hàng, nhận thông báo khi đơn hàng được xác nhận, giao hàng hoặc hoàn thành; xem lại lịch sử các đơn hàng đã mua.

1.3.3. Khách truy cập (người dùng chưa đăng kí tài khoản):

- *Xem thông tin sản phẩm*: Duyệt danh mục, xem chi tiết sản phẩm mà không cần đăng nhập.

- *Tìm kiếm sản phẩm*: Sử dụng chức năng tìm kiếm để tham khảo sản phẩm trước khi quyết định đăng ký tài khoản.

1.4. Các yêu cầu của người dùng

- *Giao diện thân thiện, dễ sử dụng:* Hệ thống cần có giao diện trực quan, bố cục hợp lý, dễ thao tác trên cả máy tính và thiết bị di động, giúp người dùng tiếp cận các chức năng nhanh chóng.

- *Tìm kiếm sản phẩm hiệu quả:* Chức năng tìm kiếm phải hoạt động nhanh, chính xác, hỗ trợ lọc theo tên, loại sản phẩm, giá cả, giúp khách hàng dễ dàng tìm được sản phẩm mong muốn.

- *Quản lý đơn hàng rõ ràng:* Thông tin đơn hàng cần được hiển thị đầy đủ, trạng thái đơn hàng được cập nhật liên tục, giúp cả khách hàng và quản trị viên theo dõi tiến trình xử lý đơn hàng dễ dàng.

- *Bảo mật thông tin:* Hệ thống phải đảm bảo an toàn dữ liệu cá nhân, thông tin giao dịch của khách hàng và quản trị viên, sử dụng các phương pháp mã hóa, xác thực hiện đại.

- *Thanh toán trực tuyến an toàn:* Tích hợp thanh toán ngân hàng uy tín đảm bảo giao dịch diễn ra nhanh chóng, bảo mật, hỗ trợ nhiều hình thức thanh toán.

- *Quản trị viên quản lý toàn diện:* Quản trị viên cần có quyền truy cập và quản lý toàn bộ hệ thống, từ sản phẩm, đơn hàng, người dùng đến các báo cáo thống kê, phục vụ công tác quản lý và phát triển kinh doanh.

Chương 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

2.1. Phân tích yêu cầu hệ thống

Phân tích yêu cầu hệ thống là bước quan trọng nhằm xác định các chức năng cần thiết, các tiêu chí phi chức năng và những ràng buộc, giả định trong quá trình xây dựng website quản lý và bán hàng trực tuyến cho cửa hàng tiệm lẻ. Việc xác định đúng và đủ các yêu cầu sẽ giúp hệ thống đáp ứng tốt nhu cầu thực tế.

2.1.1. Yêu cầu chức năng

- Hệ thống cho phép quản trị viên đăng nhập, quản lý thông tin sản phẩm, cập nhật giá cả, hình ảnh, mô tả sản phẩm.
- Quản trị viên có thể theo dõi, xử lý và cập nhật trạng thái các đơn hàng, từ khi khách hàng đặt hàng đến khi giao hàng hoàn tất.
- Chức năng quản lý người dùng cho phép phân quyền, khóa/mở tài khoản, xem thông tin khách hàng.
- Khách hàng có thể đăng ký, đăng nhập, xem và tìm kiếm sản phẩm, thêm sản phẩm vào giỏ hàng, đặt hàng trực tuyến và thanh toán qua phương thức hiện đại như App Ngân Hàng.
- Hệ thống cung cấp chức năng thống kê doanh thu, sản phẩm bán chạy, xuất báo cáo phục vụ công tác quản lý.
- Khách hàng có thể theo dõi trạng thái đơn hàng, xem tiến trình xử lý và xem lại lịch sử mua hàng.

2.1.2. Yêu cầu phi chức năng

- Giao diện website phải thân thiện, dễ sử dụng.
- Hệ thống đảm bảo bảo mật thông tin cá nhân và dữ liệu giao dịch của người dùng, sử dụng các phương pháp mã hóa và xác thực hiện đại.
- Thời gian phản hồi của hệ thống phải nhanh, đảm bảo trải nghiệm mượt mà cho người dùng khi thao tác.
- Hệ thống phải có khả năng mở rộng, dễ dàng bổ sung chức năng mới hoặc tích hợp với các dịch vụ bên ngoài.
- Dữ liệu phải được lưu trữ an toàn, có cơ chế sao lưu và phục hồi khi xảy ra sự cố.
- Website phải đảm bảo tính ổn định, hạn chế tối đa các lỗi phát sinh trong quá trình sử dụng.

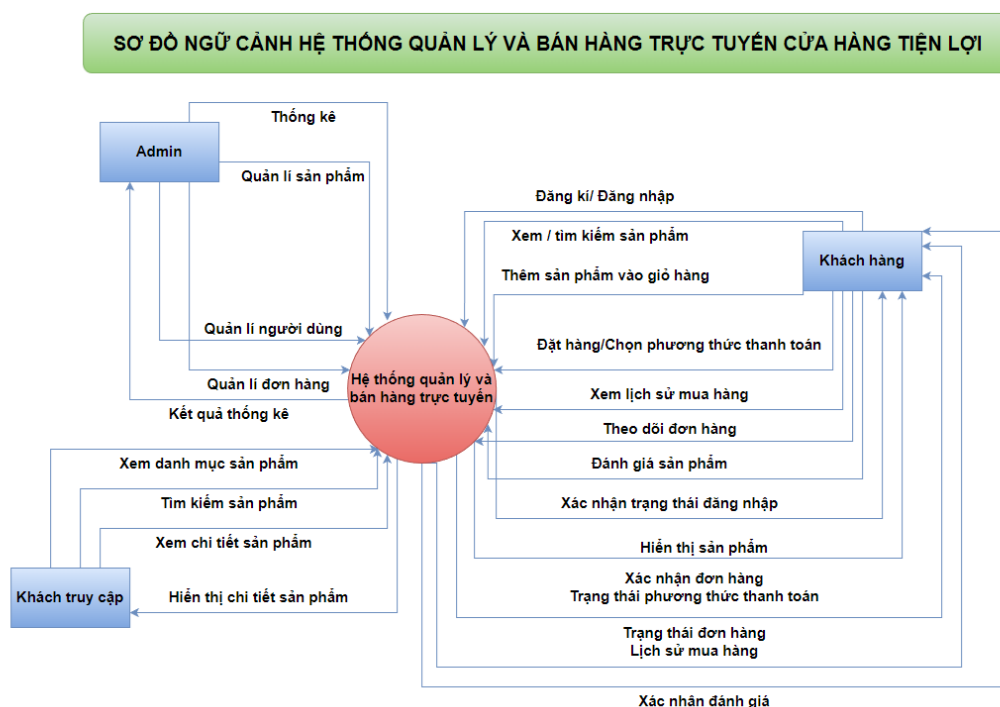
2.1.3. Các ràng buộc và giả định

- Hệ thống được xây dựng dựa trên nền tảng web, sử dụng các công nghệ như Node.js, ReactJS, MongoDB.
- Người dùng cần có kết nối Internet để truy cập và sử dụng các chức năng của hệ thống.
- Các giao dịch thanh toán trực tuyến được thực hiện thông qua app ngân hàng, giả định rằng khách hàng có tài khoản ngân hàng hỗ trợ dịch vụ này.
- Dữ liệu sản phẩm, đơn hàng, người dùng được lưu trữ tập trung trên máy chủ, đảm bảo tính nhất quán và bảo mật.

2.2. Mô hình hóa hệ thống

Trong quá trình phân tích và thiết kế hệ thống, việc mô hình hóa giúp xác định rõ các thành phần, chức năng và luồng dữ liệu giữa các tác nhân với hệ thống. Các sơ đồ như sơ đồ ngữ cảnh, sơ đồ luồng dữ liệu (DFD) và sơ đồ chức năng chính là công cụ trực quan hỗ trợ quá trình xây dựng hệ thống một cách logic, khoa học và phù hợp với yêu cầu thực tế của cửa hàng tiệm lợi.

2.2.1. Sơ đồ ngữ cảnh



Hình 2.1: Sơ đồ DFD ngữ cảnh

Mô tả sơ đồ hệ thống:

Sơ đồ ngữ cảnh dưới đây mô tả tổng quan các tác nhân bên ngoài và các luồng thông tin trao đổi giữa các tác nhân với hệ thống quản lý và bán hàng trực tuyến:

- Khách truy cập (người dùng chưa đăng ký tài khoản):

+ Có thể xem danh mục sản phẩm, tìm kiếm sản phẩm và xem chi tiết sản phẩm mà không cần đăng nhập.

+ Khi thực hiện các thao tác này, hệ thống sẽ trả về thông tin chi tiết sản phẩm tương ứng.

- Khách hàng (người dùng đã đăng ký):

+ Có thể đăng ký, đăng nhập vào hệ thống.

+ Sau khi đăng nhập, khách hàng có thể xem và tìm kiếm sản phẩm, thêm sản phẩm vào giỏ hàng, đặt hàng và chọn phương thức thanh toán.

+ Khách hàng cũng có thể xem lịch sử mua hàng, theo dõi trạng thái đơn hàng, đánh giá sản phẩm đã mua.

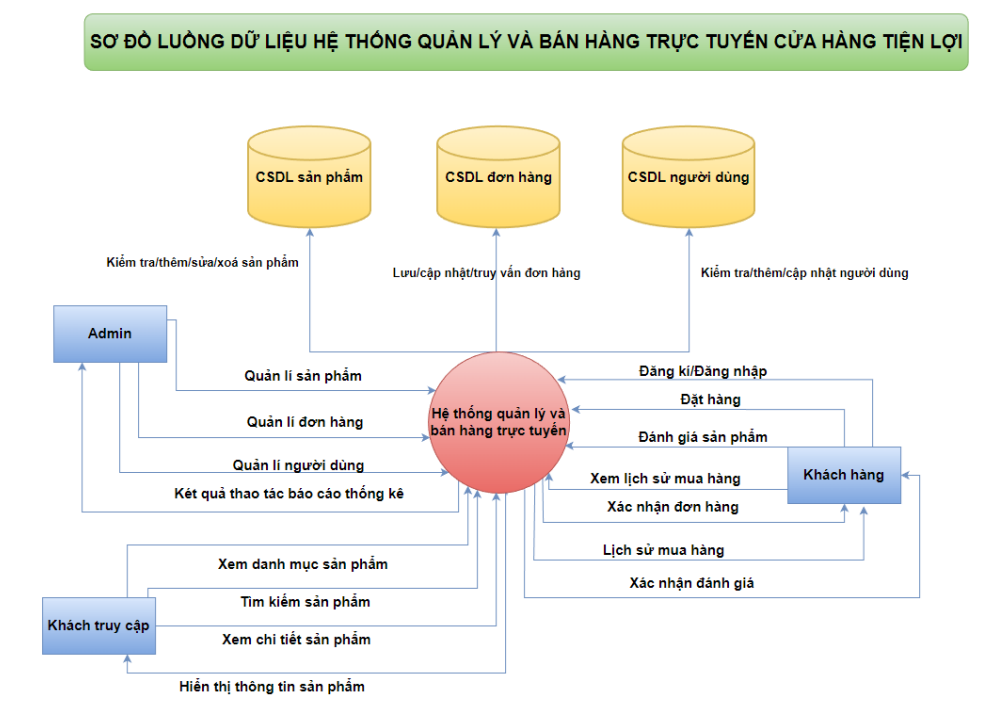
+ Hệ thống sẽ xác nhận trạng thái đăng nhập, hiển thị sản phẩm, xác nhận đơn hàng, trạng thái phương thức thanh toán, trạng thái đơn hàng, lịch sử mua hàng và xác nhận đánh giá cho khách hàng.

- Quản trị viên (Admin):

+ Có quyền quản lý sản phẩm, quản lý người dùng, quản lý đơn hàng và xem các báo cáo thống kê.

+ Hệ thống sẽ trả về kết quả các thao tác quản lý và các báo cáo thống kê cho quản trị viên.

2.2.2. Sơ đồ luồng dữ liệu (DFD)



Hình 2.2: Sơ đồ luồng dữ liệu DFD

Mô tả hệ thống sơ đồ luồng dữ liệu:

Admin (Quản trị viên): Thực hiện các chức năng quản lý sản phẩm, quản lý đơn hàng, quản lý người dùng và nhận kết quả báo cáo, thống kê từ hệ thống. Admin có thể kiểm tra, thêm, sửa, xóa sản phẩm; quản lý thông tin người dùng và cập nhật trạng thái đơn hàng.

Khách hàng: Có thể đăng ký, đăng nhập vào hệ thống, đặt hàng, đánh giá sản phẩm, xem lịch sử mua hàng. Hệ thống phản hồi lại các thông tin xác nhận đăng nhập, xác nhận đơn hàng, xác nhận đánh giá và cung cấp lịch sử mua hàng cho khách hàng.

Khách truy cập: Là người dùng chưa đăng ký tài khoản, có thể xem danh mục sản phẩm, tìm kiếm và xem chi tiết sản phẩm. Hệ thống trả về thông tin sản phẩm tương ứng với yêu cầu của khách truy cập.

Hệ thống kết nối với ba kho dữ liệu chính:

- **CSDL sản phẩm:** Lưu trữ thông tin về các sản phẩm, phục vụ cho việc quản lý và truy vấn sản phẩm.

- **CSDL đơn hàng:** Lưu trữ thông tin các đơn hàng, phục vụ cho việc đặt hàng, cập nhật trạng thái và truy vấn lịch sử mua hàng.

- **CSDL người dùng:** Lưu trữ thông tin người dùng, phục vụ cho việc đăng ký, đăng nhập, quản lý tài khoản.

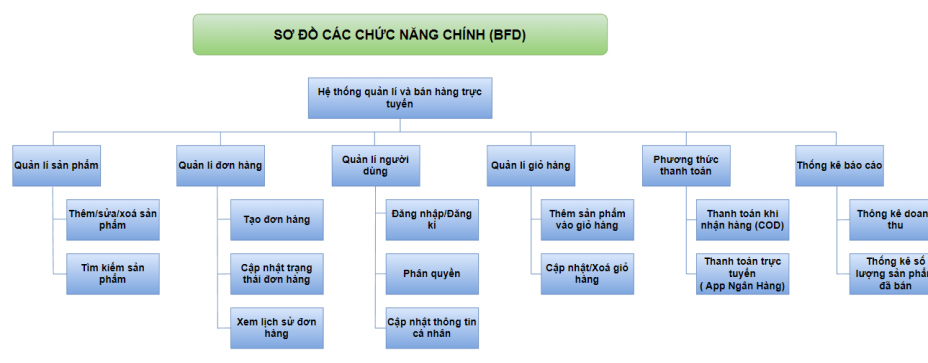
Các luồng dữ liệu giữa các tác nhân và hệ thống:

- Admin gửi các yêu cầu quản lý đến hệ thống và nhận lại kết quả thao tác, báo cáo.

- Khách hàng gửi các yêu cầu đăng ký, đăng nhập, đặt hàng, đánh giá sản phẩm, xem lịch sử mua hàng và nhận lại các xác nhận, trạng thái từ hệ thống.

- Khách truy cập gửi yêu cầu xem danh mục, tìm kiếm, xem chi tiết sản phẩm và nhận lại thông tin sản phẩm từ hệ thống.

2.2.3. Sơ đồ các chức năng chính (BFD)



Hình 2.3: Hình ảnh sơ đồ các chức năng chính (BFD)

Mô tả sơ đồ các chức năng chính (BFD) của hệ thống:

- Ở cấp cao nhất, hệ thống được chia thành các chức năng lớn: Quản lý sản phẩm, Quản lý đơn hàng, Quản lý người dùng, Quản lý giỏ hàng, Phương thức thanh toán và Thống kê báo cáo.

- Quản lý sản phẩm gồm các chức năng: thêm/sửa/xóa sản phẩm và tìm kiếm sản phẩm.

- Quản lý đơn hàng bao gồm: tạo đơn hàng, cập nhật trạng thái đơn hàng và xem lịch sử đơn hàng.

- Quản lý người dùng gồm: đăng nhập/dăng ký, phân quyền và cập nhật thông tin cá nhân.

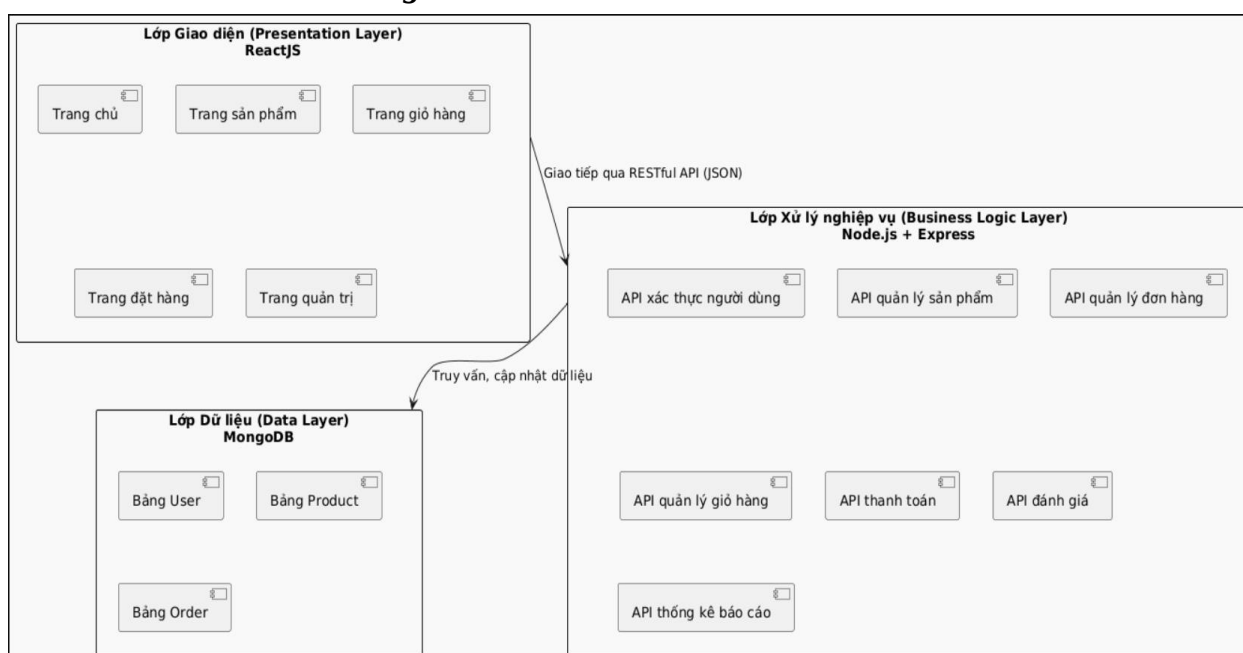
- Quản lý giỏ hàng gồm: thêm sản phẩm vào giỏ hàng và cập nhật/xóa sản phẩm trong giỏ.

- Phương thức thanh toán hỗ trợ hai hình thức: thanh toán khi nhận hàng (COD) và thanh toán trực tuyến (App Ngân Hàng).

- Thống kê báo cáo gồm: thống kê doanh thu và thống kê số lượng sản phẩm đã bán.

2.3. Thiết kế hệ thống

2.3.1. Thiết kế kiến trúc tổng thể.



Hình 2.4: Hình ảnh thiết kế hệ thống tổng thể

Hệ thống quản lý và bán hàng trực tuyến cho cửa hàng tiệm lẻ được xây dựng theo mô hình kiến trúc ba lớp (Three-tier Architecture), bao gồm:

Lớp giao diện người dùng (Frontend):

Được phát triển bằng ReactJS, đảm nhận vai trò hiển thị thông tin, nhận thao tác từ người dùng (khách truy cập, khách hàng, quản trị viên) và gửi các yêu cầu đến backend thông qua các API. Giao diện được thiết kế thân thiện, dễ sử dụng, hỗ trợ tốt trên cả máy tính và thiết bị di động.

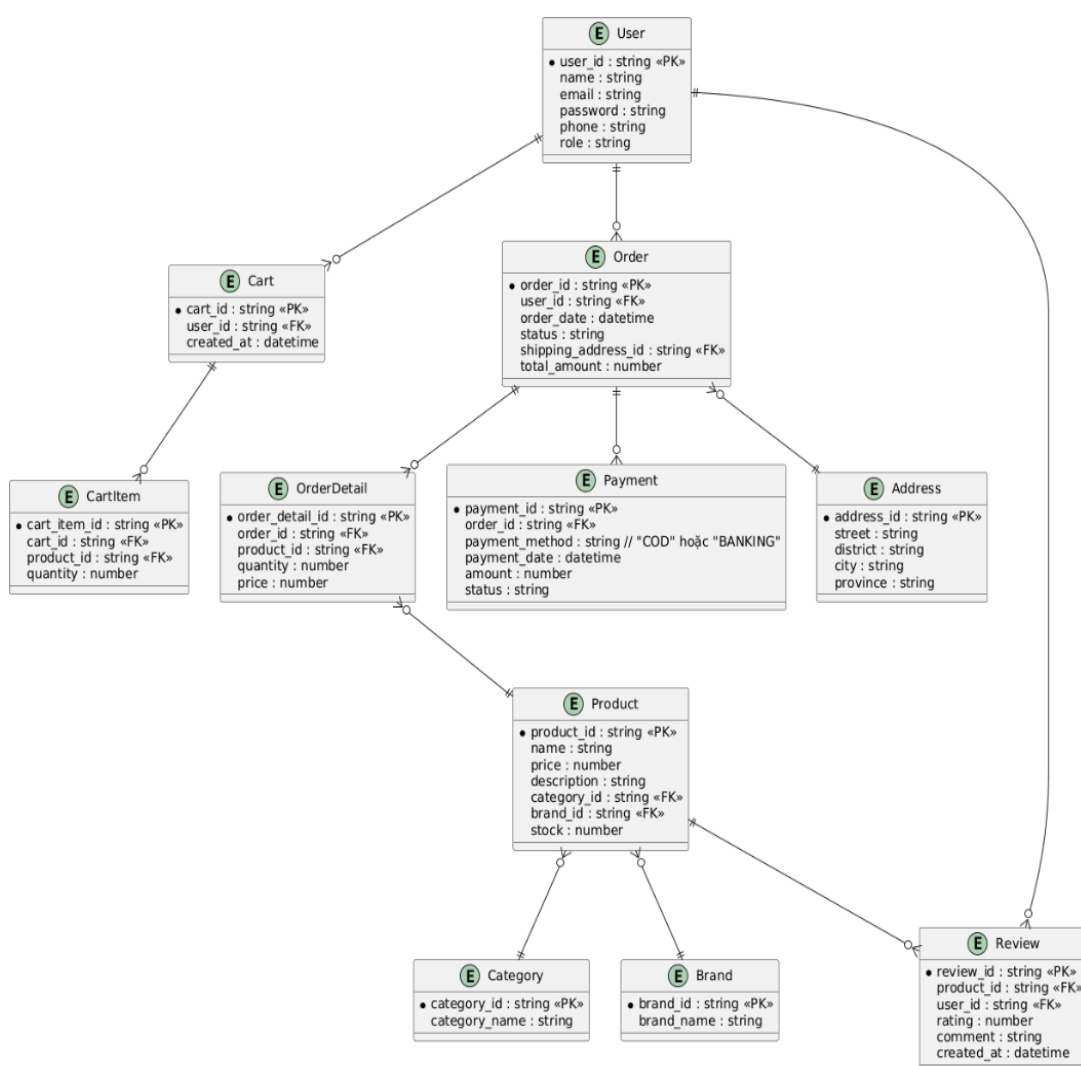
Lớp xử lý nghiệp vụ (Backend):

Sử dụng Node.js và Express để xây dựng các API RESTful, xử lý logic nghiệp vụ như xác thực người dùng, quản lý sản phẩm, đơn hàng, giỏ hàng, đánh giá, thanh toán và thống kê. Backend tiếp nhận yêu cầu từ frontend, xử lý dữ liệu và trả về kết quả phù hợp.

Lớp cơ sở dữ liệu (Database):

Sử dụng MongoDB để lưu trữ thông tin về người dùng, sản phẩm, đơn hàng, đánh giá,... Dữ liệu được tổ chức hợp lý, đảm bảo tính nhất quán, dễ truy vấn và mở rộng.

2.3.2. Thiết kế cơ sở dữ liệu (ERD, bảng dữ liệu)



Hình 2.5: Sơ đồ ERD hệ thống.

Mô tả sơ đồ ERD hệ thống quản lý và bán hàng trực tuyến:

- *User (Người dùng)*: Lưu thông tin tài khoản của khách hàng và quản trị viên, gồm các trường như user_id, name, email, password, phone, role. Một người dùng có thể có nhiều đơn hàng, giỏ hàng, đánh giá sản phẩm.

- *Order (Đơn hàng)*: Lưu thông tin các đơn hàng do người dùng đặt, gồm order_id, user_id, order_date, status, shipping_address, total_amount. Mỗi đơn hàng liên kết với một người dùng, một địa chỉ giao hàng, một thanh toán và nhiều chi tiết đơn hàng.

- *OrderDetail (Chi tiết đơn hàng)*: Lưu thông tin từng sản phẩm trong đơn hàng, gồm order_detail_id, order_id, product_id, quantity, price. Mỗi chi tiết đơn hàng liên kết với một đơn hàng và một sản phẩm.

- *Product (Sản phẩm)*: Lưu thông tin sản phẩm, gồm product_id, name, price, description, category_id, brand_id, stock. Sản phẩm liên kết với danh mục, thương hiệu và có thể có nhiều đánh giá, nhiều chi tiết đơn hàng, nhiều mục trong giỏ hàng.

- *Review (Đánh giá)*: Lưu các đánh giá sản phẩm của người dùng, gồm review_id, product_id, user_id, rating, comment, created_at. Mỗi đánh giá liên kết với một sản phẩm và một người dùng.

- *Cart (Giỏ hàng)*: Lưu thông tin giỏ hàng của người dùng, gồm cart_id, user_id, created_at. Mỗi giỏ hàng thuộc về một người dùng và có thể chứa nhiều sản phẩm.

- *CartItem (Sản phẩm trong giỏ hàng)*: Lưu các sản phẩm mà người dùng đã thêm vào giỏ hàng, gồm cart_item_id, cart_id, product_id, quantity.

- *Payment (Thanh toán)*: Lưu thông tin thanh toán cho đơn hàng, gồm: payment_id, order_id, payment_method (chỉ nhận giá trị "COD" hoặc "BANKING"), payment_date, amount, status.

- *Address (Địa chỉ)*: Lưu các địa chỉ giao hàng, gồm address_id, street, district, city, province. Địa chỉ liên kết với đơn hàng.

- *Category (Danh mục sản phẩm)*: Lưu các danh mục sản phẩm, gồm category_id, category_name. Mỗi sản phẩm thuộc một danh mục.

- *Brand (Thương hiệu)*: Lưu các thương hiệu sản phẩm, gồm brand_id, brand_name. Mỗi sản phẩm thuộc một thương hiệu.

Bảng 2.1: Bảng User dữ liệu của hệ thống

Tên trường	Kiểu dữ liệu	Mô tả
user_id	string	Khóa chính
name	string	Tên người dùng
email	string	Email
password	string	Mật khẩu
phone	string	Số điện thoại
role	string	Vai trò (admin/user)

Mô tả bảng User (Người dùng):

- Lưu trữ thông tin tài khoản của tất cả người dùng hệ thống, bao gồm khách hàng và quản trị viên.
- Các trường chính: user_id, name, email, password, phone, role.
- Dùng để xác thực đăng nhập, phân quyền và quản lý tài khoản.

Bảng 2.2: Bảng Product (Sản phẩm) dữ liệu hệ thống

Tên trường	Kiểu dữ liệu	Mô tả
product_id	string	Khóa chính
name	string	Tên sản phẩm
price	number	Giá
description	string	Mô tả
category_id	string	FK đến Category
brand_id	string	FK đến Brand
stock	number	Số lượng tồn

Mô tả bảng Product (Sản phẩm):

- Chứa thông tin chi tiết về từng sản phẩm được bán trên hệ thống.
- Các trường chính: product_id, name, price, description, image, category_id, brand_id, stock.
- Hỗ trợ quản lý, tìm kiếm, hiển thị sản phẩm cho khách hàng.

Bảng 2.3: Bảng Order (Đơn hàng) dữ liệu hệ thống

Tên trường	Kiểu dữ liệu	Mô tả
order_id	string	Khóa chính
user_id	string	FK đến User
order_date	datetime	Ngày đặt hàng
status	string	Trạng thái
shipping_address	string	Địa chỉ giao hàng
total_amount	number	Tổng tiền

Mô tả bảng Order (Đơn hàng):

- Lưu trữ các đơn hàng mà khách hàng đã đặt.
- Các trường chính: order_id, user_id, order_date, status, shipping_address, total_amount.
- Giúp theo dõi quá trình đặt hàng, xử lý và giao nhận.

Bảng 2.4: Bảng OrderDetail (Chi tiết đơn hàng) dữ liệu hệ thống

Tên trường	Kiểu dữ liệu	Mô tả
order_detail_id	string	Khóa chính
order_id	string	FK đến Order
product_id	string	FK đến Product
quantity	number	Số lượng
price	number	Giá

Mô tả bảng OrderDetail (Chi tiết đơn hàng):

- Lưu thông tin về từng sản phẩm trong mỗi đơn hàng.
- Các trường chính: order_detail_id, order_id, product_id, quantity, price.
- Hỗ trợ quản lý chi tiết từng mặt hàng trong đơn hàng.

Bảng 2.5: Bảng Review (Đánh giá) dữ liệu hệ thống

Tên trường	Kiểu dữ liệu	Mô tả
review_id	string	Khóa chính
product_id	string	FK đến Product
user_id	string	FK đến User
rating	number	Điểm đánh giá
comment	string	Nhận xét
created_at	datetime	Thời gian đánh giá

Mô tả bảng Review (Đánh giá):

- Lưu các đánh giá, nhận xét của khách hàng về sản phẩm.
- Các trường chính: review_id, product_id, user_id, rating, comment, created_at.
- Giúp khách hàng tham khảo ý kiến, quản trị viên kiểm soát chất lượng sản phẩm.

Bảng 2.6: Bảng Cart (Giỏ hàng) dữ liệu hệ thống

Tên trường	Kiểu dữ liệu	Mô tả
cart_id	string	Khóa chính
user_id	string	FK đến User
created_at	datetime	Thời gian tạo

Mô tả bảng Cart (Giỏ hàng):

- Lưu thông tin về giỏ hàng của từng người dùng.
- Các trường chính: cart_id, user_id, created_at.
- Hỗ trợ lưu trữ tạm thời các sản phẩm khách hàng dự định mua.

Bảng 2.7: Bảng CartItem (Sản phẩm trong giỏ hàng)

Tên trường	Kiểu dữ liệu	Mô tả
cart_item_id	string	Khóa chính
cart_id	string	FK đến Cart
product_id	string	FK đến Product
quantity	number	Số lượng

Mô tả bảng CartItem (Sản phẩm trong giỏ hàng):

- Lưu các sản phẩm mà người dùng đã thêm vào giỏ hàng.
- Các trường chính: cart_item_id, cart_id, product_id, quantity.
- Hỗ trợ thêm, xóa, cập nhật sản phẩm trong giỏ hàng.

Bảng 2.8: Bảng Payment (Thanh toán)

Tên trường	Kiểu dữ liệu	Mô tả
payment_id	string	Khóa chính
order_id	string	FK đến Order
payment_method	string	Phương thức thanh toán ("COD" hoặc "BANKING")
payment_date	datetime	Ngày thanh toán
amount	number	Số tiền
status	string	Trạng thái

Mô tả bảng Payment (Thanh toán):

- Lưu thông tin về các giao dịch thanh toán cho đơn hàng.
- Các trường chính: payment_id, order_id, payment_method, payment_date, amount, status.
- Quản lý các giao dịch tài chính của hệ thống.

Bảng 2.9: Bảng Address (Địa chỉ)

Tên trường	Kiểu dữ liệu	Mô tả
address_id	string	Khóa chính
street	string	Địa chỉ
district	string	Quận/Huyện
city	string	Thành phố
province	string	Tỉnh

Mô tả bảng Address (Địa chỉ):

- Lưu trữ các địa chỉ giao hàng của người dùng.
- Các trường chính: address_id, street, district, city, province.
- Hỗ trợ khách hàng lưu nhiều địa chỉ nhận hàng.

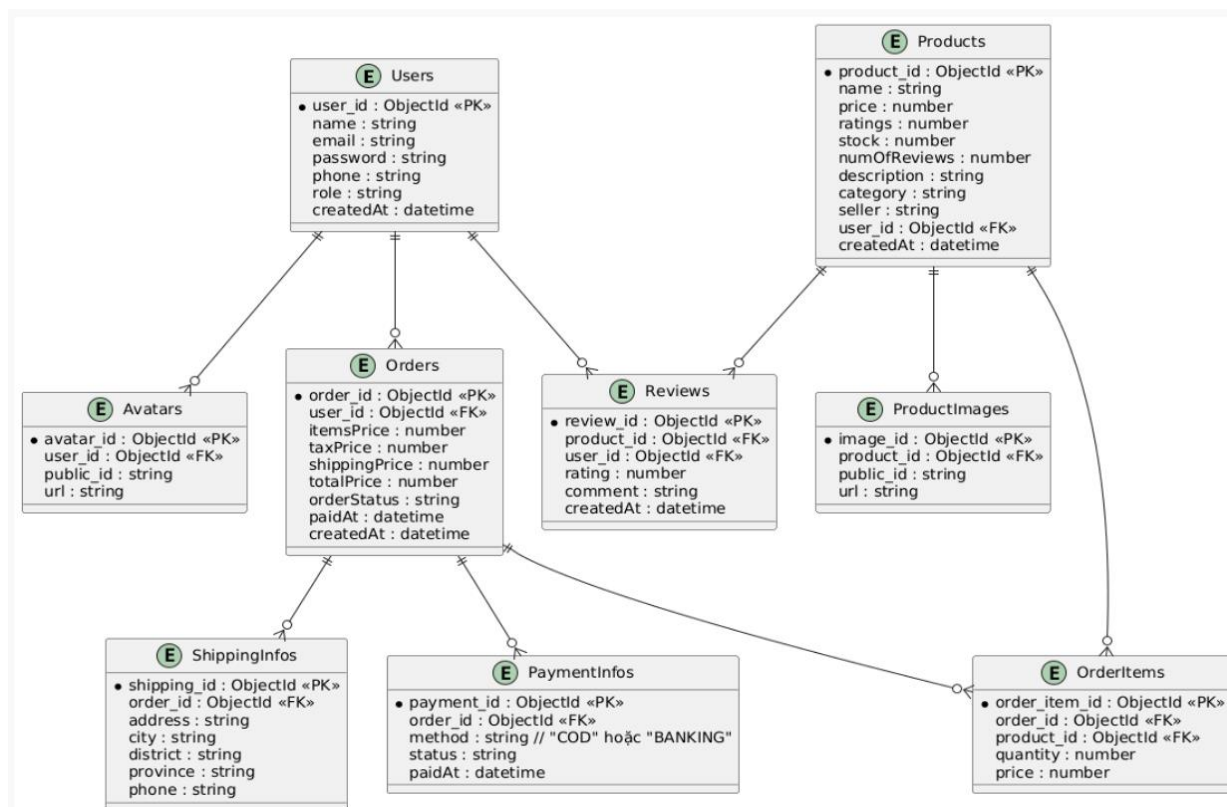
Bảng 2.10: Bảng Category (Danh mục sản phẩm)

Tên trường	Kiểu dữ liệu	Mô tả
category_id	string	Khóa chính
category_name	string	Tên danh mục

Mô tả bảng Category (Danh mục sản phẩm):

- Lưu thông tin về các danh mục sản phẩm.
- Các trường chính: category_id, category_name.
- Hỗ trợ phân loại, tìm kiếm sản phẩm theo nhóm.

Sau khi phân tích, các bảng dữ liệu đã được chuẩn hóa đến chuẩn thứ 3 (3NF) nhằm loại bỏ dư thừa, đảm bảo tính nhất quán và dễ mở rộng. Sơ đồ chuẩn hóa dữ liệu được thể hiện dưới đây:



Hình 2.6: Hình sơ đồ dữ liệu dạng chuẩn thứ 3 (3NF)

Mô tả sơ đồ dữ liệu dạng chuẩn thứ 3 (3NF):

- Users:

+ Lưu thông tin người dùng: user_id, name, email, password, phone, role, createdAt.

+ Mỗi người dùng có thể có nhiều đơn hàng (Orders), nhiều đánh giá (Reviews) và một hoặc nhiều avatar (Avatars).

- Avatars:

+ Lưu thông tin ảnh đại diện của người dùng: avatar_id, user_id, public_id, url.

+ Mỗi avatar liên kết với một người dùng.

- *Products:*

+ Lưu thông tin sản phẩm: product_id, name, price, ratings, stock, numOfReviews, description, category, seller, user_id (người tạo), createdAt.

+ Mỗi sản phẩm có thể có nhiều ảnh (ProductImages), nhiều đánh giá (Reviews) và nhiều mục trong đơn hàng (OrderItems).

- *ProductImages:*

+ Lưu thông tin ảnh sản phẩm: image_id, product_id, public_id, url.

+ Mỗi ảnh liên kết với một sản phẩm.

- *Reviews:*

+ Lưu đánh giá sản phẩm: review_id, product_id, user_id, rating, comment, createdAt.

+ Mỗi đánh giá liên kết với một sản phẩm và một người dùng.

- *Orders:*

+ Lưu thông tin đơn hàng: order_id, user_id, itemsPrice, taxPrice, shippingPrice, totalPrice, orderStatus, paidAt, createdAt.

+ Mỗi đơn hàng có thể có nhiều mục đơn hàng (OrderItems), một thông tin giao hàng (ShippingInfos), một thông tin thanh toán (PaymentInfos).

- *OrderItems:*

+ Lưu chi tiết từng sản phẩm trong đơn hàng: order_item_id, order_id, product_id, quantity, price.

+ Mỗi mục đơn hàng liên kết với một đơn hàng và một sản phẩm.

- *ShippingInfos:*

+ Lưu thông tin giao hàng: shipping_id, order_id, address, city, district, province, phone.

+ Mỗi thông tin giao hàng liên kết với một đơn hàng.

- *PaymentInfos:*

+ Lưu thông tin thanh toán: payment_id, order_id, method (COD hoặc BANKING), status, paidAt.

+ Mỗi thông tin thanh toán liên kết với một đơn hàng.

2.3.3. Nội dung thiết kế giao diện người dùng (UI/UX)

Giao diện người dùng của hệ thống được thiết kế với tiêu chí thân thiện, trực quan, dễ sử dụng cho cả khách hàng và quản trị viên. Màu sắc, bố cục và các thành phần giao diện được lựa chọn hài hòa, đảm bảo trải nghiệm nhất quán.

- *Trang chủ:* Hiển thị các sản phẩm nổi bật, danh mục sản phẩm, thanh tìm kiếm.

- *Trang danh mục và tìm kiếm sản phẩm*: Cho phép người dùng tìm kiếm sản phẩm theo tên thương hiệu.

- *Trang chi tiết sản phẩm*: Hiện thị thông tin chi tiết, hình ảnh, đánh giá và nút thêm vào giỏ hàng.

- *Trang giỏ hàng*: Hiện thị danh sách sản phẩm đã chọn, cho phép cập nhật số lượng, xóa sản phẩm, tính tổng tiền và chuyển sang đặt hàng.

- *Trang đặt hàng và thanh toán*: Thu thập thông tin giao hàng, lựa chọn phương thức thanh toán (COD, App Ngân Hàng), xác nhận đơn hàng.

- *Trang quản lý tài khoản*: Cho phép người dùng xem, cập nhật thông tin cá nhân, đổi mật khẩu, xem lịch sử mua hàng.

- *Trang quản trị*: Quản trị viên có thể quản lý sản phẩm, đơn hàng, người dùng, xem thống kê và báo cáo.

2.3.4. Thiết kế các module chức năng (quản lý sản phẩm, đơn hàng, người dùng, thanh toán...)

Hệ thống được chia thành các module chức năng chính, mỗi module đảm nhận một vai trò riêng biệt, đảm bảo tính rõ ràng.

- *Module quản lý sản phẩm*:

+ Cho phép quản trị viên thêm mới, chỉnh sửa, xóa và cập nhật thông tin sản phẩm (tên, giá, mô tả, hình ảnh, danh mục, thương hiệu, số lượng tồn kho).

+ Hỗ trợ tìm kiếm, lọc sản phẩm theo nhiều tiêu chí.

+ Hiện thị danh sách sản phẩm cho khách hàng và khách truy cập.

- *Module quản lý đơn hàng*:

+ Tiếp nhận đơn hàng từ khách hàng, lưu thông tin chi tiết từng đơn hàng.

+ Quản trị viên có thể cập nhật trạng thái đơn hàng (chờ xác nhận, đang giao, đã hoàn thành, hủy...).

+ Khách hàng có thể xem lịch sử đơn hàng, theo dõi trạng thái đơn hàng của mình.

- *Module quản lý người dùng*:

+ Hỗ trợ đăng ký, đăng nhập, xác thực tài khoản người dùng.

+ Quản trị viên có thể phân quyền, khóa/mở tài khoản, cập nhật thông tin người dùng.

+ Người dùng có thể cập nhật thông tin cá nhân, đổi mật khẩu.

- *Module quản lý giỏ hàng*:

+ Cho phép khách hàng thêm sản phẩm vào giỏ hàng, cập nhật số lượng, xóa sản phẩm khỏi giỏ.

+ Lưu trữ tạm thời các sản phẩm khách hàng dự định mua trước khi đặt hàng.

- Module thanh toán:

+ Hỗ trợ nhiều phương thức thanh toán: thanh toán khi nhận hàng (COD), thanh toán trực tuyến qua App Ngân Hàng.

+ Xác nhận trạng thái thanh toán, lưu thông tin giao dịch.

- Module quản lý đánh giá sản phẩm:

+ Cho phép khách hàng đánh giá, nhận xét sản phẩm đã mua.

+ Quản trị viên kiểm duyệt, quản lý các đánh giá để đảm bảo chất lượng nội dung.

- Module thống kê và báo cáo:

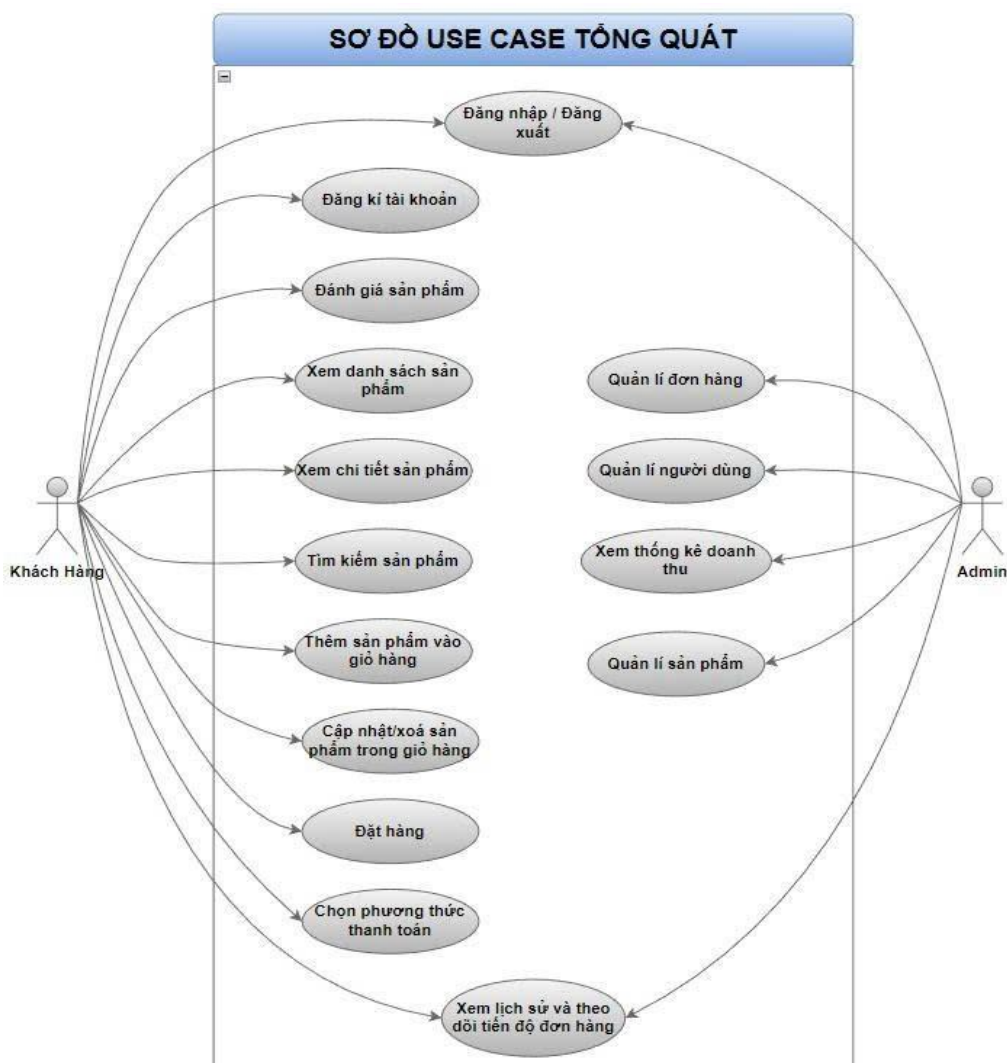
+ Thống kê doanh thu, số lượng đơn hàng, sản phẩm bán chạy.

+ Xuất báo cáo phục vụ công tác quản lý và ra quyết định kinh doanh.

2.4. Sơ đồ hoạt động và các sơ đồ minh họa

2.4.1. Sơ đồ Use Case.

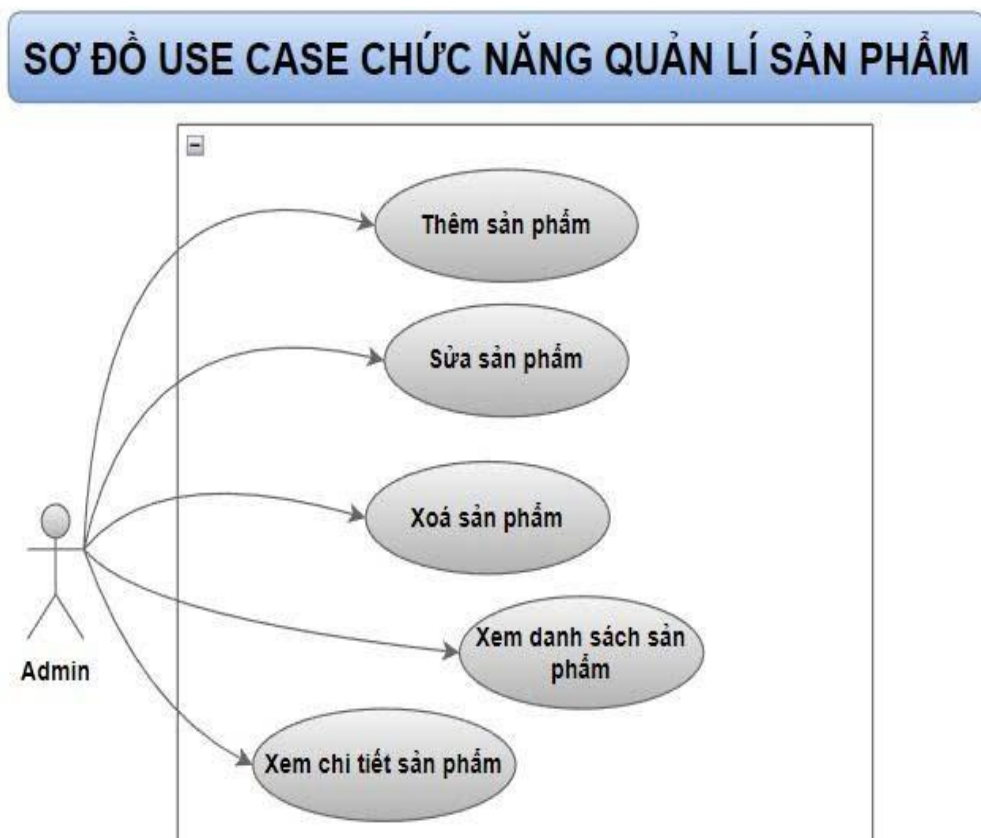
- Sơ đồ Use Case tổng quát:



Hình 2.7: Sơ đồ Use Case tổng quát

Mô tả sơ đồ Use Case tổng quát:

- + Khách hàng có thể thực hiện các chức năng:
 - + Đăng nhập/Đăng xuất, đăng ký tài khoản.
 - + Xem danh sách sản phẩm, xem chi tiết sản phẩm, tìm kiếm sản phẩm.
 - + Thêm sản phẩm vào giỏ hàng, cập nhật/xóa sản phẩm trong giỏ hàng.
 - + Đặt hàng, chọn phương thức thanh toán.
 - + Đánh giá sản phẩm đã mua.
 - + Xem lịch sử và theo dõi tiến độ đơn hàng.
 - + Quản trị viên (Admin) có thể thực hiện các chức năng:
 - + Quản lý sản phẩm (thêm, sửa, xóa, xem).
 - + Quản lý đơn hàng (xem, cập nhật trạng thái, xóa).
 - + Quản lý người dùng (xem, phân quyền, xóa).
 - + Xem thống kê doanh thu, số lượng đơn hàng, sản phẩm, người dùng.
- *Sơ đồ Use Case Quản trị viên (Admin):*
- + *Sơ đồ Use Case quản lí sản phẩm*



Hình 2.8: Sơ đồ Use Case quản lí sản phẩm (Admin)

Mô tả:

Admin là tác nhân chính, có thể thực hiện các chức năng:

Thêm sản phẩm mới vào hệ thống.

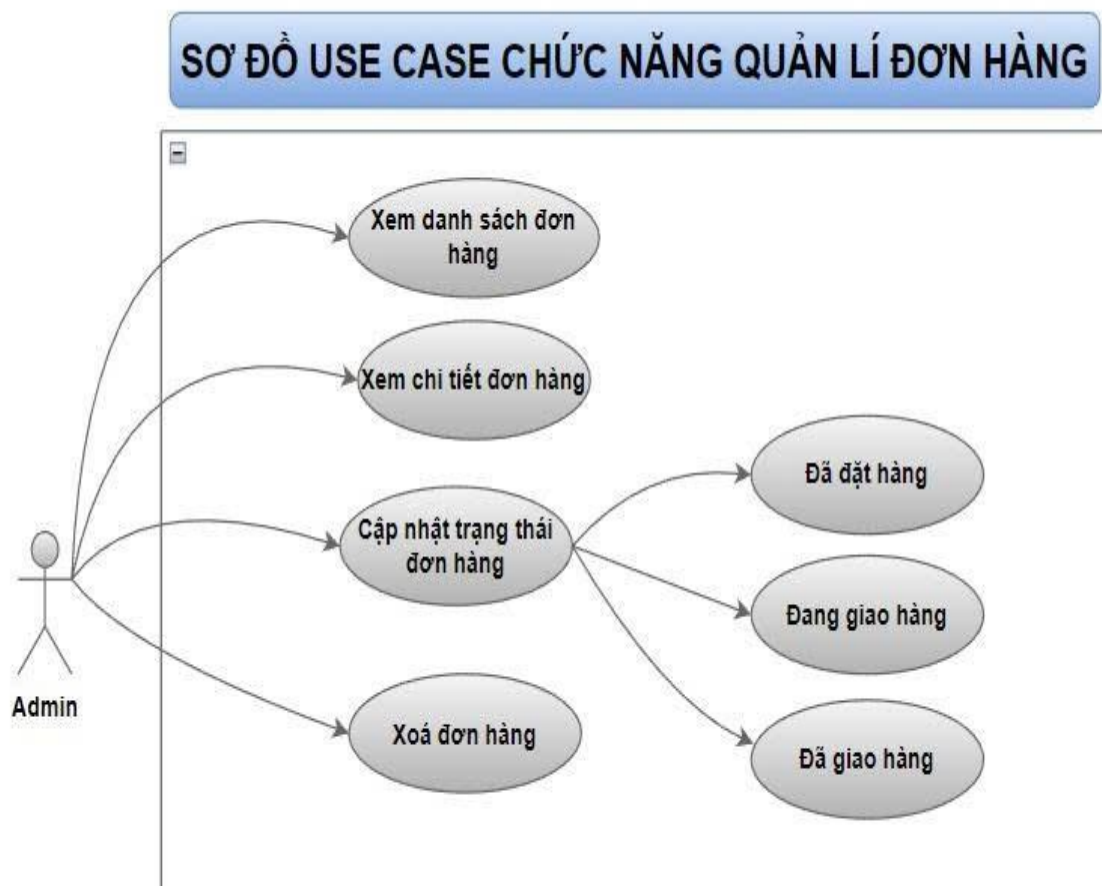
Sửa thông tin sản phẩm đã có.

Xóa sản phẩm khỏi hệ thống.

Xem danh sách tất cả sản phẩm hiện có.

Xem chi tiết thông tin của từng sản phẩm.

+ Sơ đồ Use Case quản lý đơn hàng



Hình 2.9: Sơ đồ Use Case quản lý đơn hàng (Admin)

Mô tả:

Admin là tác nhân chính, có thể thực hiện các chức năng:

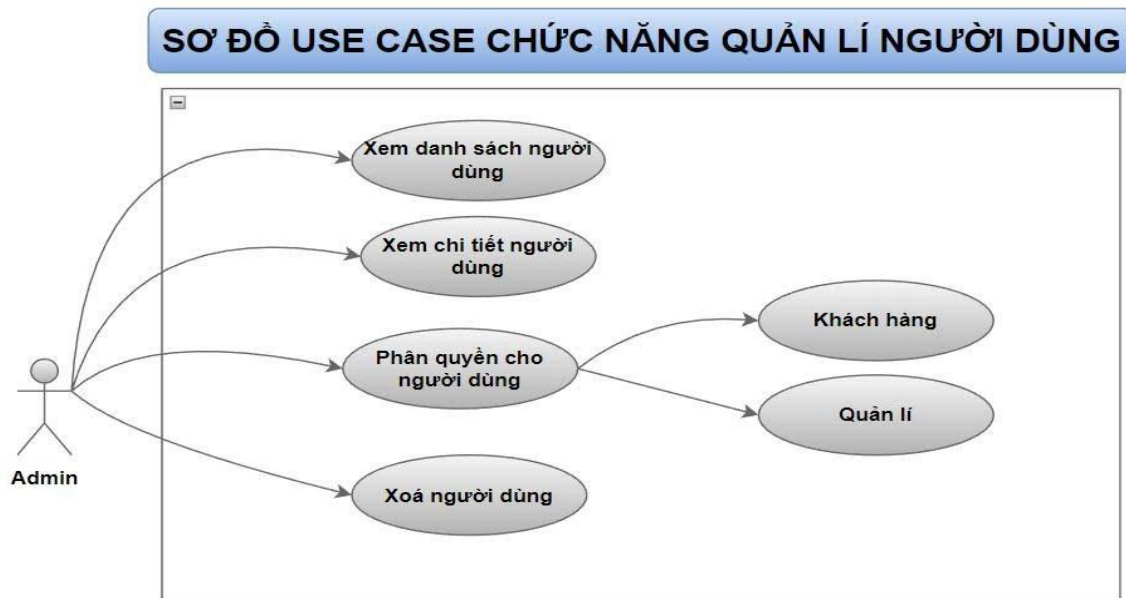
Xem danh sách tất cả các đơn hàng trong hệ thống.

Xem chi tiết thông tin của từng đơn hàng.

Cập nhật trạng thái đơn hàng, bao gồm các trạng thái: Đã đặt hàng, Đang giao hàng, Đã giao hàng.

Xóa đơn hàng khỏi hệ thống khi cần thiết.

+ Sơ đồ Use Case quản lý người dùng



Hình 2.10: Sơ đồ Use Case quản lý người dùng (Admin)

Mô tả:

Admin là tác nhân chính, có thể thực hiện các chức năng:

Xem danh sách tất cả người dùng trong hệ thống.

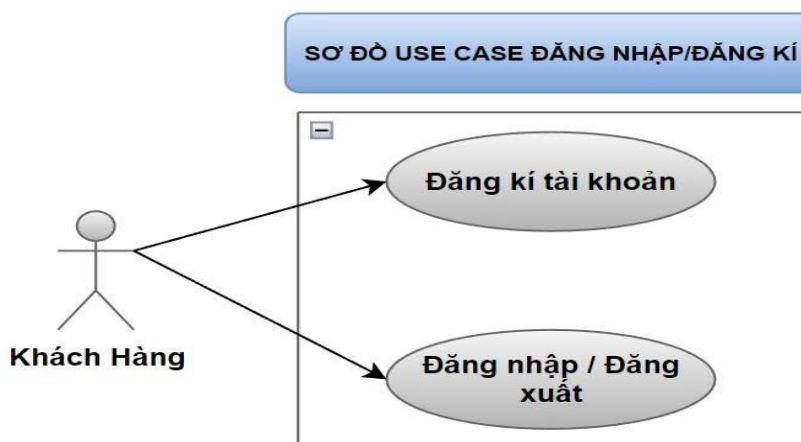
Xem chi tiết thông tin của từng người dùng.

Phân quyền cho người dùng, bao gồm phân biệt giữa khách hàng và quản lý (quản trị viên).

Xóa người dùng khỏi hệ thống khi cần thiết.

- Sơ đồ Use Case Khách hàng (User):

+ Sơ đồ Use đăng kí/đăng nhập



Hình 2.11: Sơ đồ Use Case đăng kí/đăng nhập khách hàng (User)

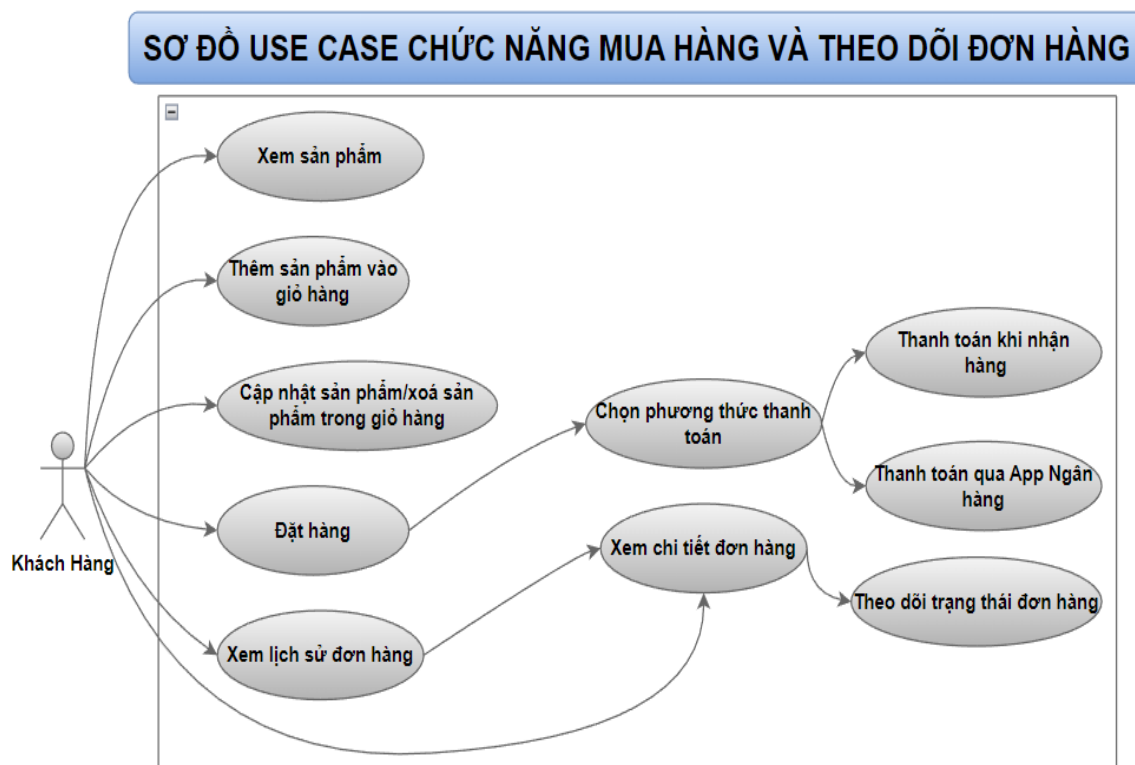
Mô tả sơ đồ Use Case đăng kí/đăng nhập:

Khách hàng là tác nhân chính, có thể thực hiện hai chức năng:

Đăng ký tài khoản: Cho phép khách hàng tạo mới một tài khoản trên hệ thống bằng cách cung cấp các thông tin cá nhân cần thiết.

Đăng nhập/Đăng xuất: Cho phép khách hàng truy cập vào hệ thống bằng tài khoản đã đăng ký và đăng xuất khi không sử dụng nữa.

+ Sơ đồ Use Case mua hàng và theo dõi đơn hàng



Hình 2.12: Sơ đồ Use Case chức năng mua hàng và theo dõi đơn hàng khách hàng (User)

Mô tả :

Khách hàng có thể thực hiện các chức năng:

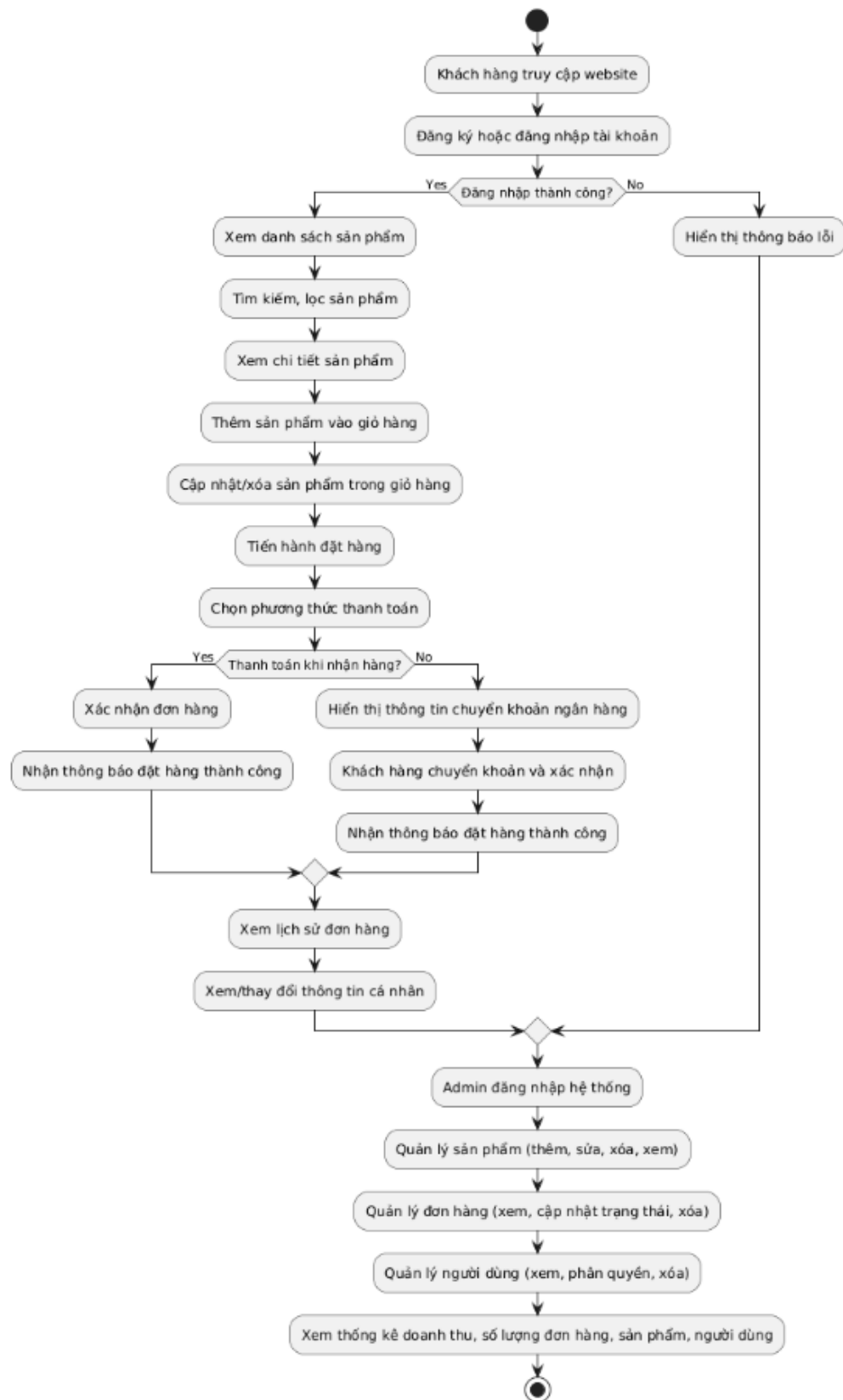
Xem sản phẩm, thêm sản phẩm vào giỏ hàng, cập nhật hoặc xóa sản phẩm trong giỏ hàng.

Đặt hàng và lựa chọn phương thức thanh toán (thanh toán khi nhận hàng hoặc thanh toán qua App Ngân Hàng).

Xem lịch sử đơn hàng, xem chi tiết từng đơn hàng và theo dõi trạng thái đơn hàng.

Quy trình mua hàng bắt đầu từ việc khách hàng xem sản phẩm, thêm vào giỏ hàng, tiến hành đặt hàng, chọn phương thức thanh toán phù hợp. Sau khi đặt hàng thành công, khách hàng có thể xem chi tiết đơn hàng và theo dõi trạng thái xử lý của từng đơn hàng trong lịch sử mua hàng.

2.4.2. Sơ đồ hoạt động của các chức năng chính



Hình 2.13: Sơ đồ hoạt động các chức năng chính (Activity Diagram)

Mô tả sơ đồ:

- *Khách hàng truy cập vào website*

+ Bắt đầu quy trình khi khách hàng truy cập vào website.

- *Đăng kí hoặc đăng nhập bằng tài khoản.*

+ Khách hàng thực hiện đăng ký hoặc đăng nhập.

+ Nếu đăng nhập thất bại, hệ thống hiển thị thông báo lỗi và kết thúc quy trình.

+ Nếu đăng nhập thành công, khách hàng tiếp tục sử dụng các chức năng của hệ thống.

- *Duyệt và chọn sản phẩm*

+ Khách hàng có thể xem danh sách sản phẩm, tìm kiếm, lọc sản phẩm, xem chi tiết sản phẩm.

+ Khách hàng thêm sản phẩm vào giỏ hàng, có thể cập nhật hoặc xóa sản phẩm trong giỏ.

- *Đặt hàng và chọn phương thức thanh toán*

+ Khách hàng tiến hành đặt hàng và chọn phương thức thanh toán.

+ Thanh toán khi nhận hàng (COD):

Khách hàng xác nhận đơn hàng.

Hệ thống ghi nhận đơn hàng và gửi thông báo đặt hàng thành công.

+ Chuyển khoản ngân hàng (Banking)

Hệ thống hiển thị thông tin chuyển khoản ngân hàng.

Khách hàng thực hiện chuyển khoản và xác nhận với hệ thống.

Hệ thống gửi thông báo đặt hàng thành công.

- *Sau khi đặt hàng thành công*

+ Khách hàng có thể xem lịch sử đơn hàng.

+ Khách hàng có thể xem hoặc thay đổi thông tin cá nhân.

- *Quản trị viên (Admin)*

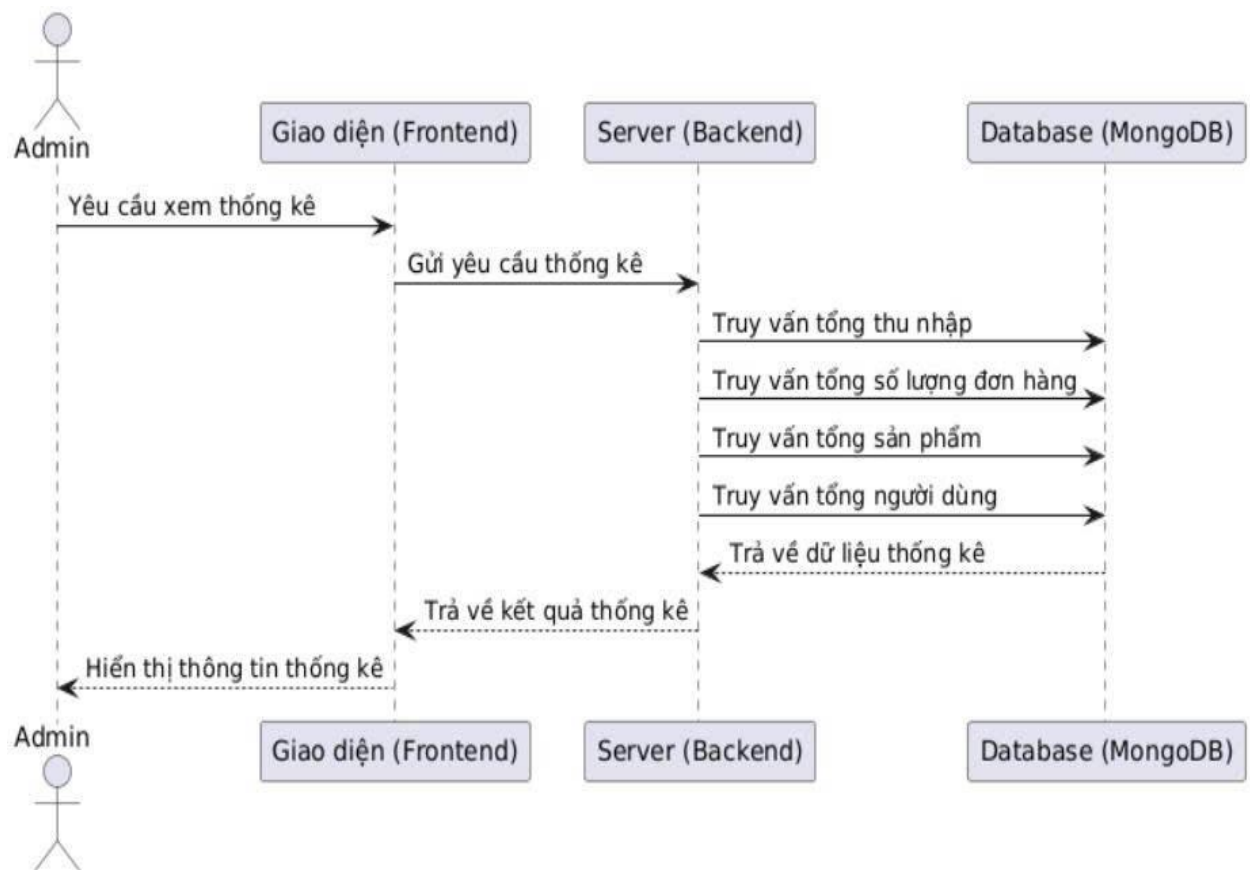
+ Admin đăng nhập hệ thống.

+ Thực hiện các chức năng quản lý: sản phẩm, đơn hàng, người dùng, xem thống kê doanh thu, số lượng đơn hàng, sản phẩm, người dùng.

2.4.3. Sơ đồ trình tự (Sequence Diagram)

- *Quản trị viên (Admin):*

+ Sơ đồ trình tự quản lý thống kê:

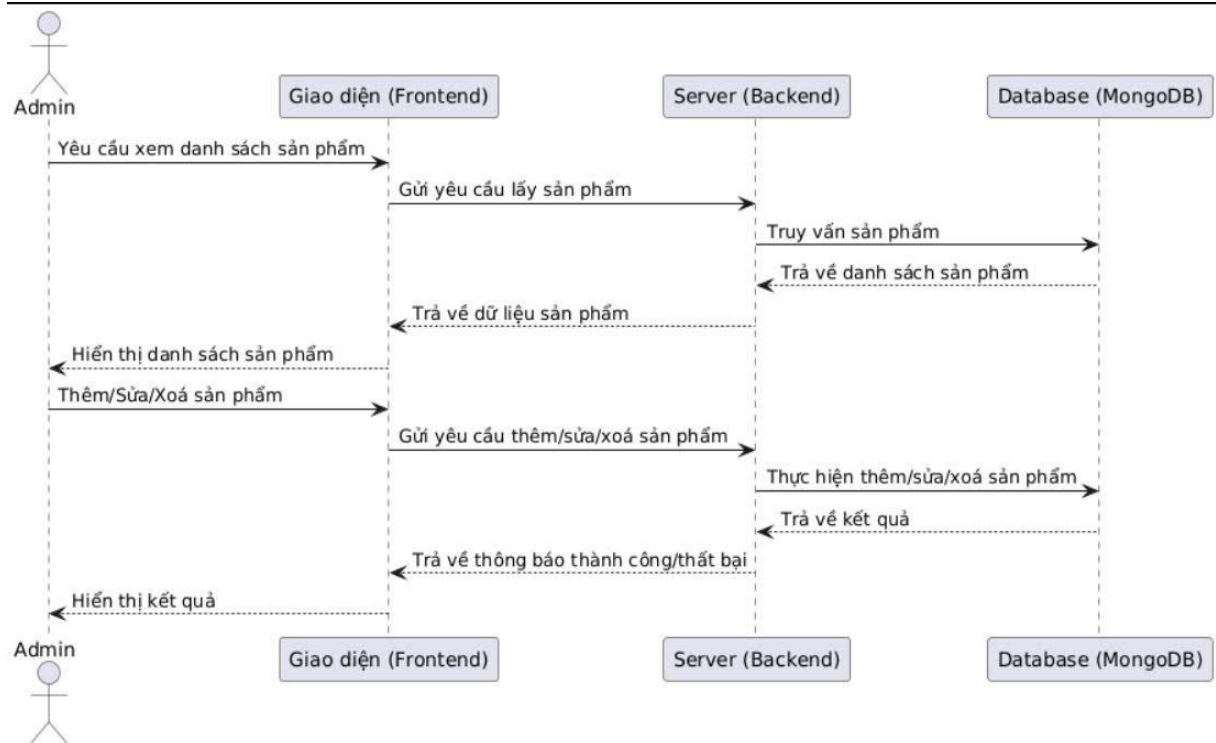


Hình 2.14: Sơ đồ trình tự quản lý thông kê Quản trị viên (Admin)

Mô tả quy trình sơ đồ trình tự quản lý thống kê:

- Admin thực hiện yêu cầu xem thông tin thống kê trên giao diện người dùng (Frontend).
- Giao diện (Frontend) tiếp nhận yêu cầu và gửi yêu cầu thống kê đến Server (Backend).
- Server (Backend) lần lượt truy vấn các thông tin cần thiết từ Database (MongoDB), bao gồm: tổng thu nhập, tổng số lượng đơn hàng, tổng số sản phẩm và tổng số người dùng.
- Database (MongoDB) trả về dữ liệu thống kê cho Server (Backend).
- Server (Backend) tổng hợp kết quả và trả về cho Giao diện (Frontend).
- Giao diện (Frontend) hiển thị thông tin thống kê cho Admin.

+ Sơ đồ trình tự quản lý sản phẩm:



Hình 2.15: Sơ đồ trình tự quản lý sản phẩm Quản trị viên (Admin)

Mô tả quy trình sơ đồ trình tự quản lý sản phẩm:

Admin gửi yêu cầu xem danh sách sản phẩm trên giao diện người dùng (Frontend).

Giao diện (Frontend) tiếp nhận yêu cầu và gửi yêu cầu lấy sản phẩm đến Server (Backend).

Server (Backend) truy vấn dữ liệu sản phẩm từ Database (MongoDB) và nhận về danh sách sản phẩm.

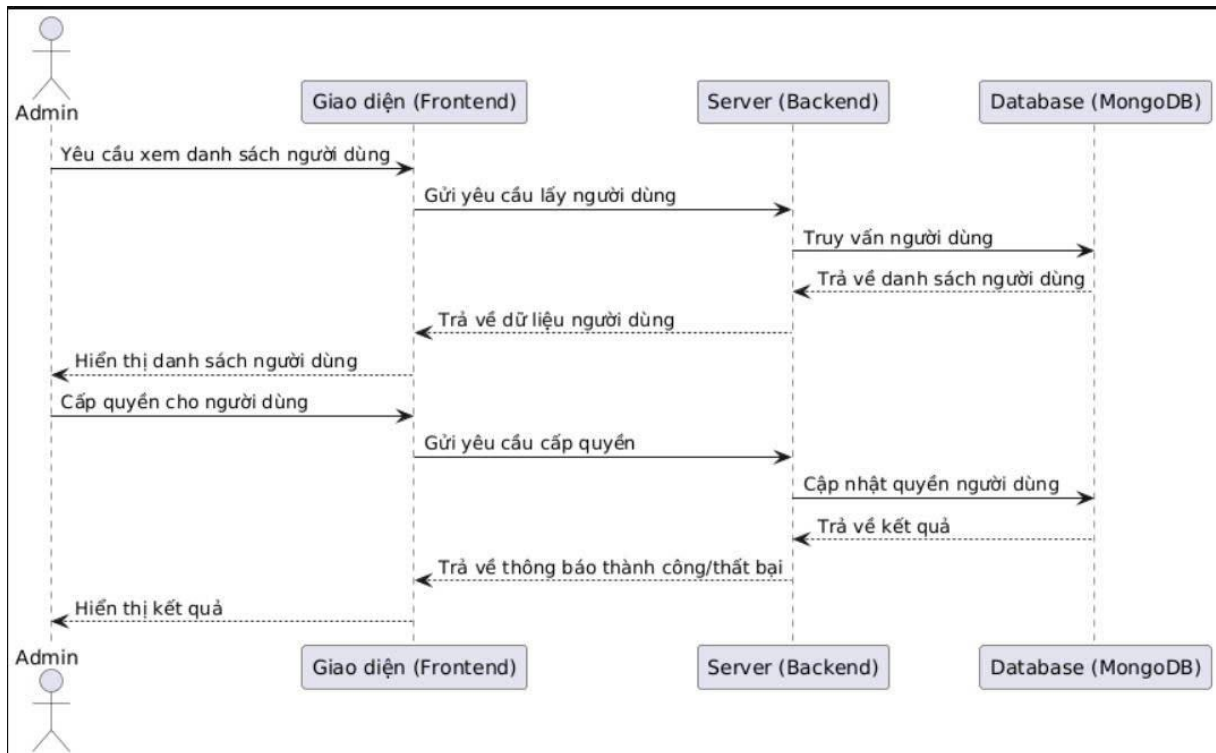
Server (Backend) trả về dữ liệu sản phẩm cho Giao diện (Frontend) để hiển thị cho admin.

Khi admin thực hiện thao tác thêm, sửa hoặc xóa sản phẩm, Giao diện (Frontend) gửi yêu cầu tương ứng đến Server (Backend).

Server (Backend) thực hiện thao tác thêm, sửa, xóa trên Database (MongoDB) và nhận kết quả trả về.

Server (Backend) trả về thông báo thành công hoặc thất bại cho Giao diện (Frontend), sau đó giao diện hiển thị kết quả cho admin.

+ Sơ đồ trình tự quản lý người dùng.

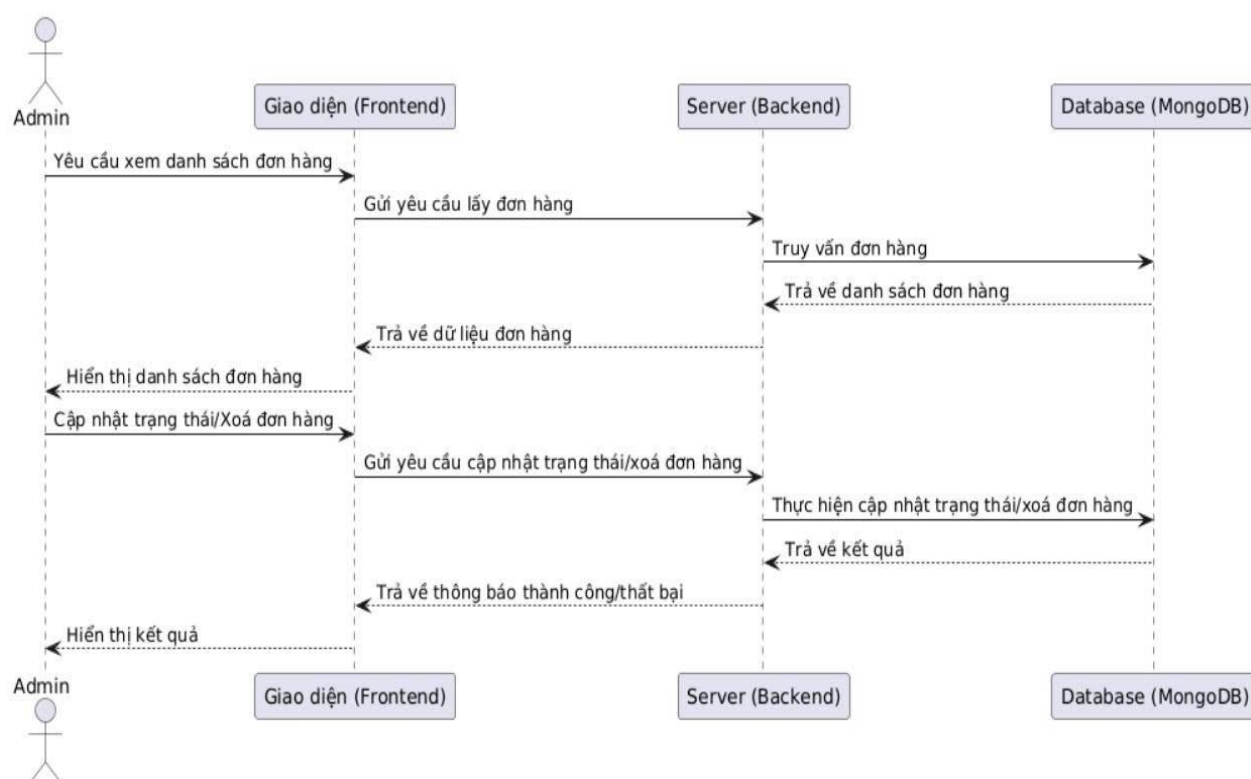


Hình 2.16: Sơ đồ trình tự quản lý người dùng Quản trị viên (Admin)

Mô tả quy trình sơ đồ trình tự quản lý người dùng:

- Admin gửi yêu cầu xem danh sách người dùng trên giao diện người dùng (Frontend).
- Giao diện (Frontend) tiếp nhận yêu cầu và gửi yêu cầu lấy danh sách người dùng đến Server (Backend).
- Server (Backend) truy vấn dữ liệu người dùng từ Database (MongoDB) và nhận về danh sách người dùng.
- Server (Backend) trả về dữ liệu người dùng cho Giao diện (Frontend) để hiển thị cho admin.
- Khi admin thực hiện thao tác cấp quyền cho người dùng, Giao diện (Frontend) gửi yêu cầu cập nhật quyền đến Server (Backend).
- Server (Backend) thực hiện cập nhật quyền người dùng trên Database (MongoDB) và nhận kết quả trả về.
- Server (Backend) trả về thông báo thành công hoặc thất bại cho Giao diện (Frontend), sau đó giao diện hiển thị kết quả cho admin.

+ Sơ đồ trình tự quản lý đơn hàng:



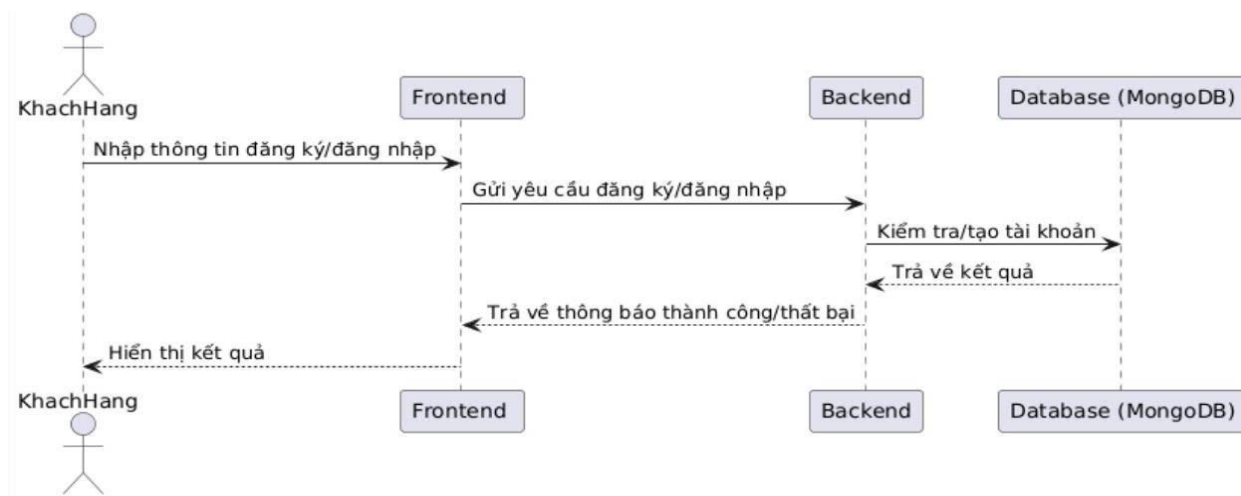
Hình 2.17: Sơ đồ trình tự quản lý đơn hàng của Quản trị viên (Admin)

Mô tả quy trình sơ đồ trình tự quản lý đơn hàng:

- Admin gửi yêu cầu xem danh sách đơn hàng trên giao diện người dùng (Frontend).
- Giao diện (Frontend) tiếp nhận yêu cầu và gửi yêu cầu lấy danh sách đơn hàng đến Server (Backend).
- Server (Backend) truy vấn dữ liệu đơn hàng từ Database (MongoDB) và nhận về danh sách đơn hàng.
- Server (Backend) trả về dữ liệu đơn hàng cho Giao diện (Frontend) để hiển thị cho admin.
- Khi admin thực hiện thao tác cập nhật trạng thái hoặc xóa đơn hàng, Giao diện (Frontend) gửi yêu cầu tương ứng đến Server (Backend).
- Server (Backend) thực hiện cập nhật trạng thái hoặc xóa đơn hàng trên Database (MongoDB) và nhận kết quả trả về.
- Server (Backend) trả về thông báo thành công hoặc thất bại cho Giao diện (Frontend), sau đó giao diện hiển thị kết quả cho admin.

- Khách hàng (User):

+ Sơ đồ trình tự đăng kí/đăng nhập:

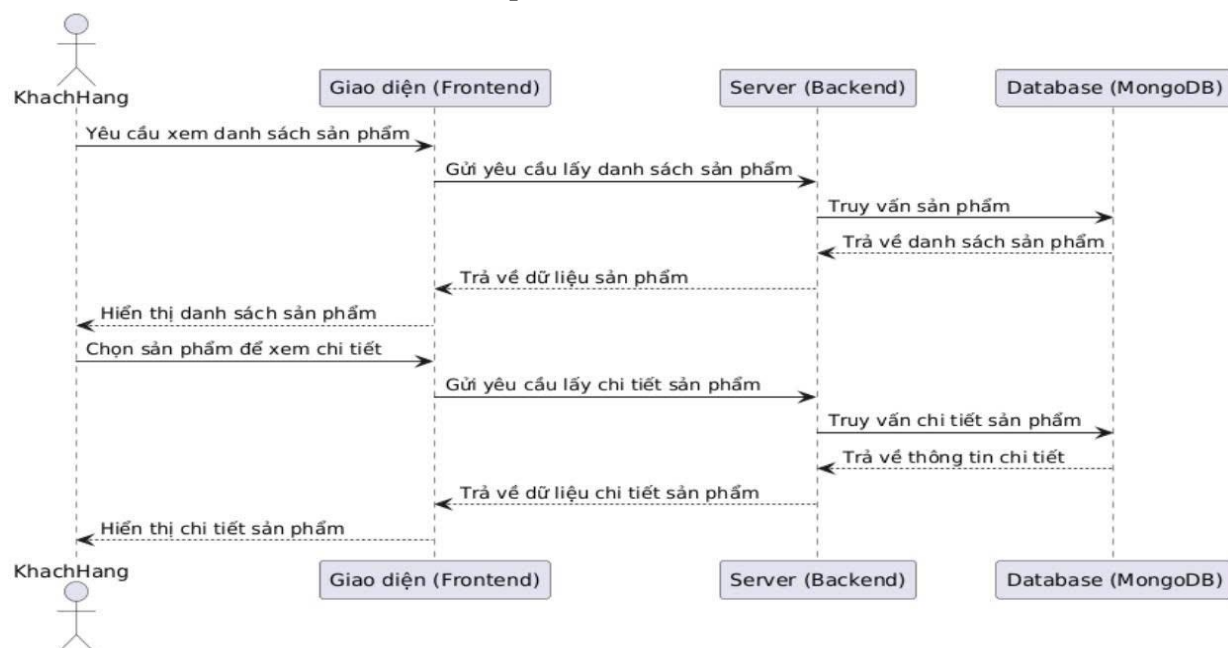


Hình 2.18: Sơ đồ trình tự đăng kí/đăng nhập của khách hàng (User)

Mô tả quy trình sơ đồ trình tự đăng kí/đăng nhập:

- Khách hàng nhập thông tin đăng ký hoặc đăng nhập trên giao diện người dùng (Frontend).
- Frontend tiếp nhận thông tin và gửi yêu cầu đăng ký/đăng nhập đến Backend.
- Backend kiểm tra hoặc tạo tài khoản trên Database (MongoDB).
- Database (MongoDB) trả về kết quả kiểm tra hoặc tạo tài khoản cho Backend.
- Backend trả về thông báo thành công hoặc thất bại cho Frontend.
- Frontend hiển thị kết quả cho khách hàng.

+ Sơ đồ trình tự xem danh sách sản phẩm:

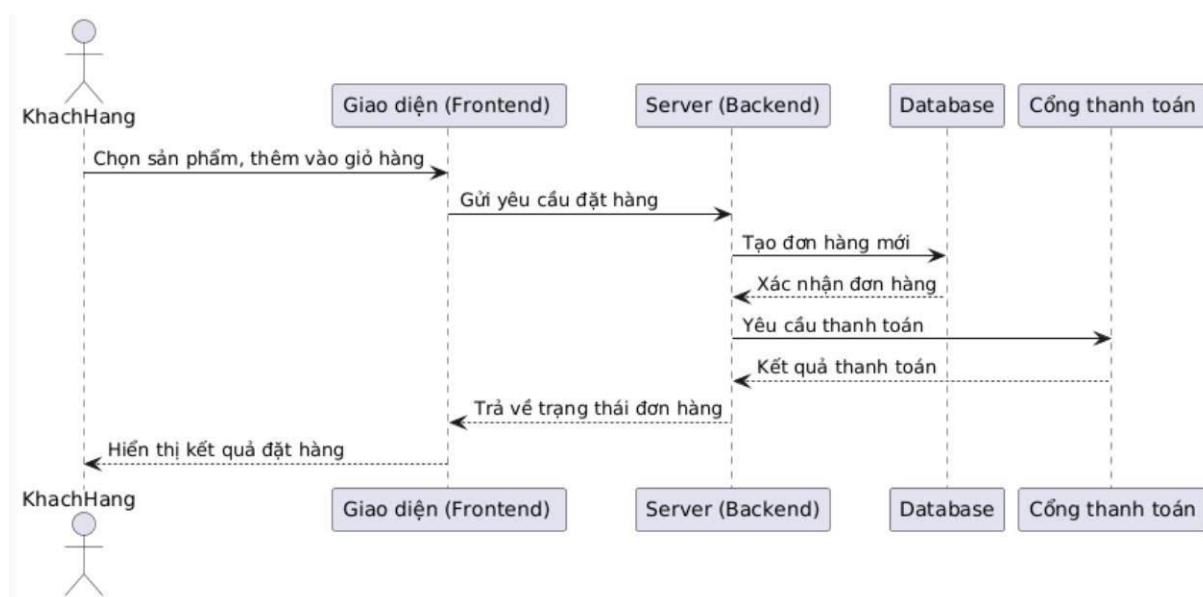


Hình 2.19: Sơ đồ trình tự xem danh sách sản phẩm khách hàng (User)

Mô tả quy trình sơ đồ trình tự xem danh sách sản phẩm:

- Khách hàng gửi yêu cầu xem danh sách sản phẩm trên giao diện người dùng (Frontend).
- Frontend gửi yêu cầu lấy danh sách sản phẩm đến Server (Backend).
- Server (Backend) truy vấn danh sách sản phẩm từ Database (MongoDB) và nhận về dữ liệu sản phẩm.
- Server (Backend) trả về dữ liệu sản phẩm cho Frontend để hiển thị cho khách hàng.
- Khách hàng chọn một sản phẩm để xem chi tiết, Frontend gửi yêu cầu lấy chi tiết sản phẩm đến Server (Backend).
- Server (Backend) truy vấn thông tin chi tiết sản phẩm từ Database (MongoDB) và nhận về dữ liệu chi tiết.
- Server (Backend) trả về dữ liệu chi tiết sản phẩm cho Frontend để hiển thị cho khách hàng.

+ *Sơ đồ trình tự đặt hàng:*



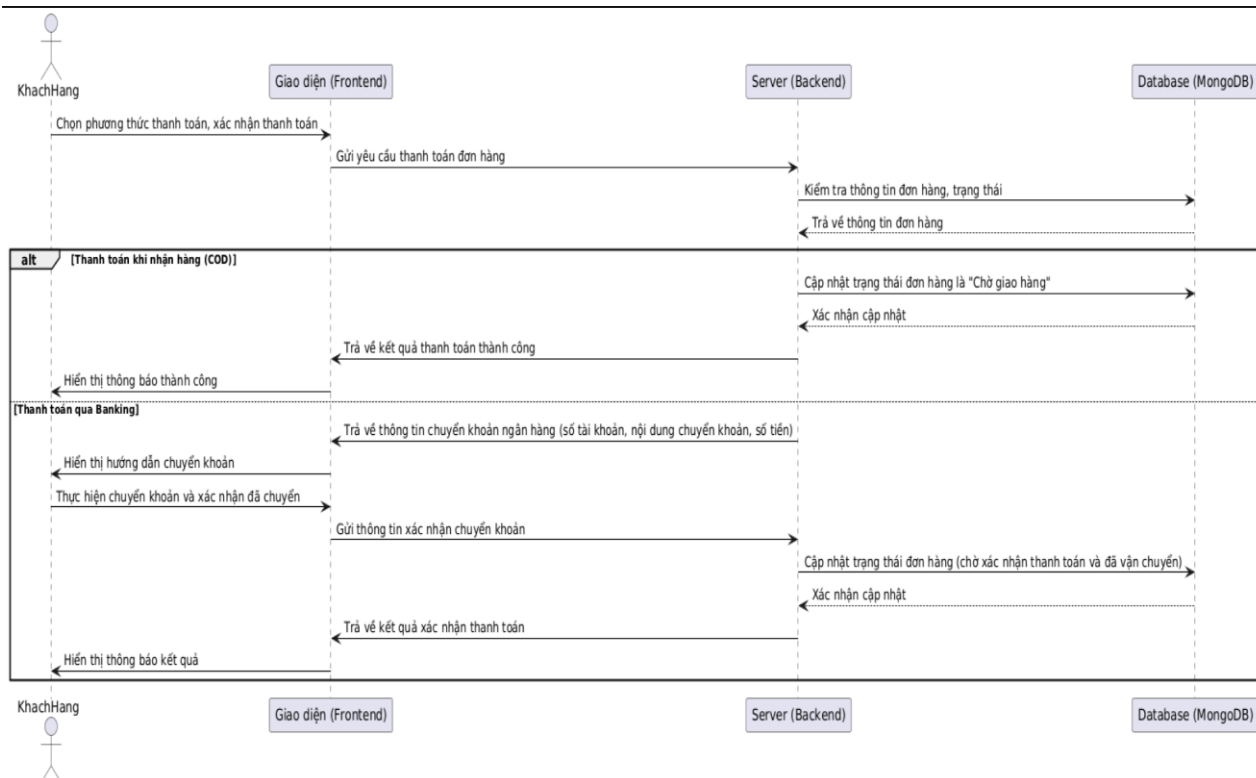
Hình 2.20: Sơ đồ trình tự đặt hàng của khách hàng (User)

Mô tả quy trình sơ đồ trình tự đặt hàng:

- Khách hàng chọn sản phẩm và thêm vào giỏ hàng trên giao diện người dùng (Frontend).
- Frontend gửi yêu cầu đặt hàng đến Server (Backend).
- Server (Backend) tạo đơn hàng mới trong Database và xác nhận đơn hàng.
- Server (Backend) gửi yêu cầu thanh toán đến Cổng thanh toán (App Ngân Hàng).

- Cổng thanh toán xử lý giao dịch và trả về kết quả thanh toán cho Server (Backend).
- Server (Backend) trả về trạng thái đơn hàng cho Frontend.
- Frontend hiển thị kết quả đặt hàng cho khách hàng.

+ Sơ đồ trình tự chọn phương thức thanh toán

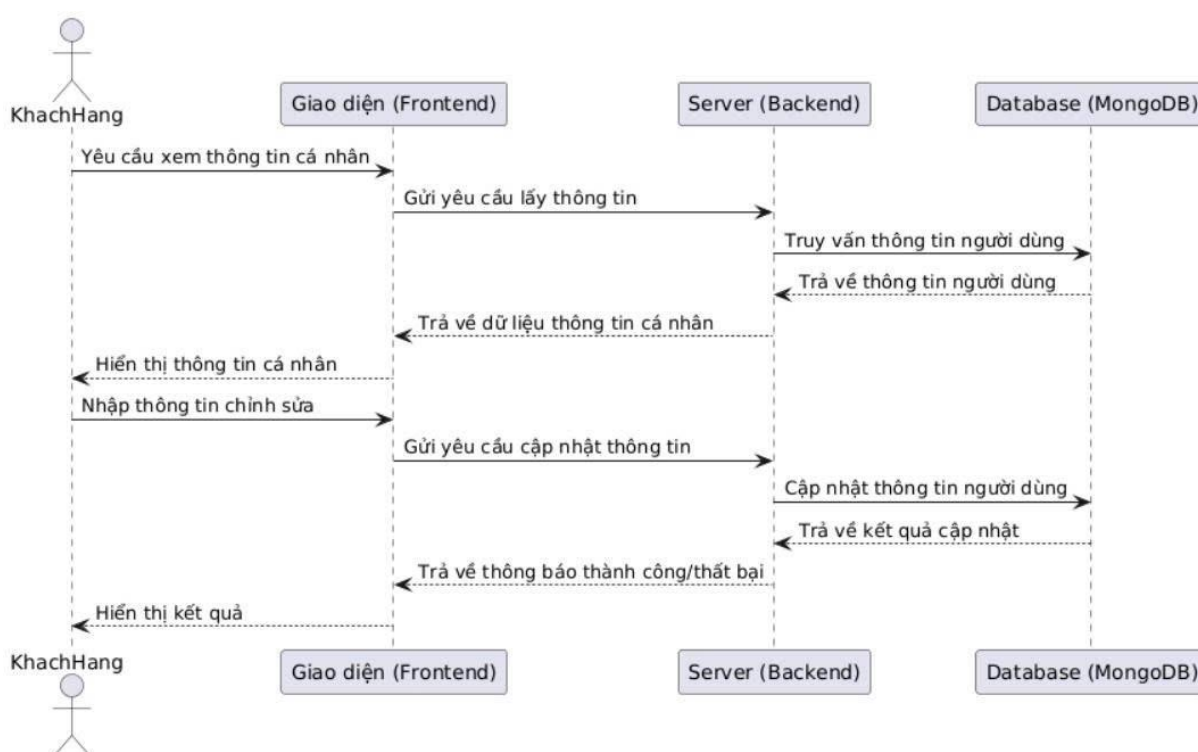


Hình 2.21: Sơ đồ trình tự chọn phương thức thanh toán khách hàng (User).

Mô tả quy trình sơ đồ trình tự chọn phương thức thanh toán:

- Bước 1: Khách hàng chọn phương thức thanh toán và xác nhận thanh toán trên giao diện người dùng (Frontend).
- Bước 2: Frontend gửi yêu cầu thanh toán đơn hàng đến Server (Backend).
- Bước 3: Backend kiểm tra thông tin đơn hàng, trạng thái và truy vấn dữ liệu từ Database (MongoDB), sau đó trả về thông tin đơn hàng cho Frontend.
- Bước 4:
 - + Nếu khách hàng chọn thanh toán khi nhận hàng (COD), Backend cập nhật trạng thái đơn hàng là "Chờ giao hàng" và xác nhận cập nhật. Frontend nhận kết quả và hiển thị thông báo thành công cho khách hàng.
 - + Nếu khách hàng chọn thanh toán qua ngân hàng, Backend gửi thông tin thanh toán đến QR banking. Frontend hiển thị giao diện chuyển khoản QR ngân hàng cho khách hàng thực hiện giao dịch và ghi nội dung chuyển khoản như yêu cầu. Sau khi giao dịch hoàn tất, khách hàng nhấn “Đã thanh toán xong và đợi duyệt đơn giao vận chuyển”.

+ Sơ đồ trình tự quản lý thông tin cá nhân:

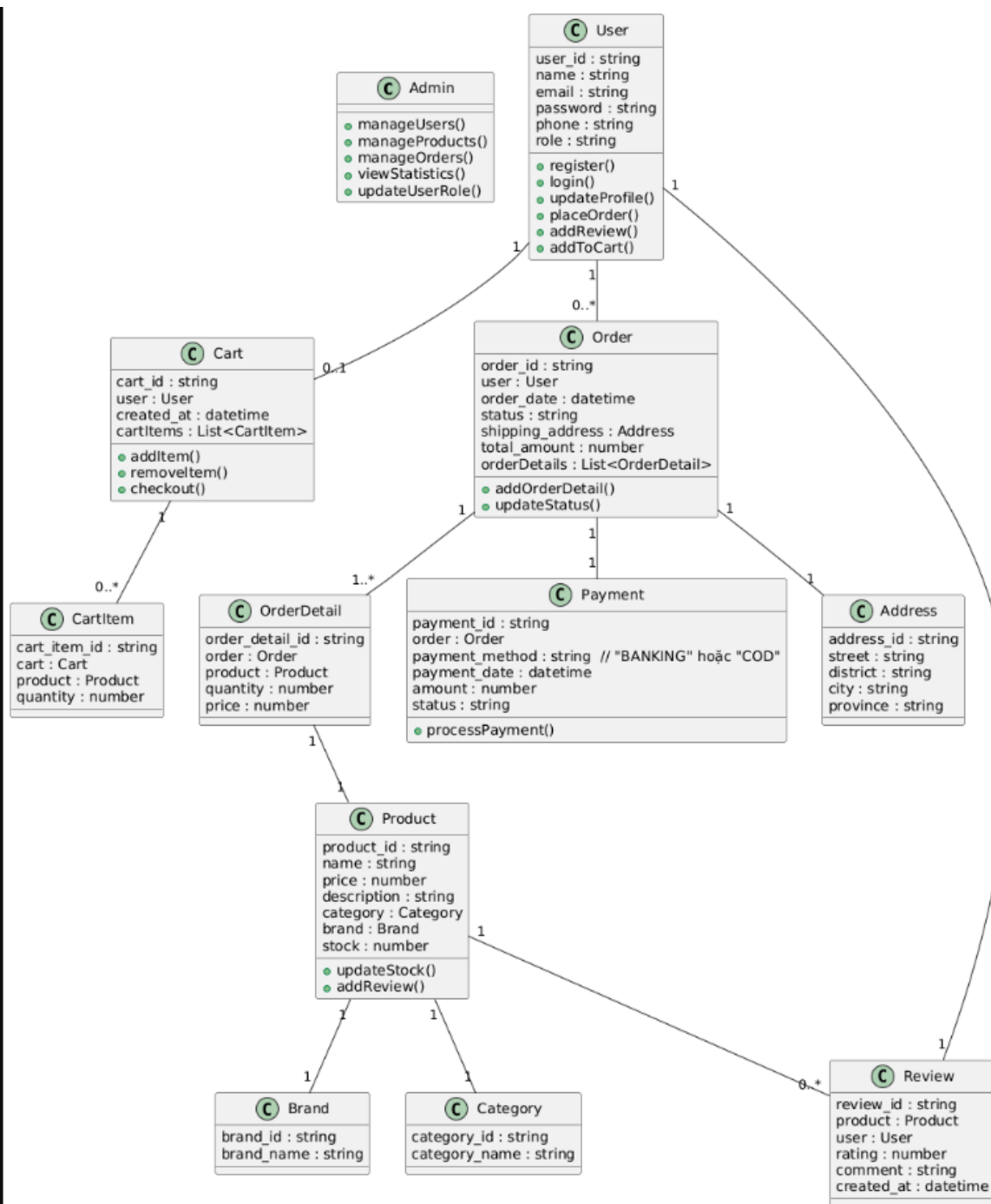


Hình 2.22: Sơ đồ trình tự quản lý thông tin các nhân khách hàng (User)

Mô tả quy trình sơ đồ trình tự quản lý thông tin cá nhân:

- Khách hàng gửi yêu cầu xem thông tin cá nhân trên giao diện người dùng (Frontend).
- Frontend gửi yêu cầu lấy thông tin cá nhân đến Server (Backend).
- Server (Backend) truy vấn thông tin người dùng từ Database (MongoDB) và nhận về dữ liệu cá nhân.
- Server (Backend) trả về dữ liệu thông tin cá nhân cho Frontend để hiển thị cho khách hàng.
- Khi khách hàng nhập thông tin chỉnh sửa, Frontend gửi yêu cầu cập nhật thông tin đến Server (Backend).
- Server (Backend) cập nhật thông tin người dùng trên Database (MongoDB) và nhận kết quả trả về.
- Server (Backend) trả về thông báo thành công hoặc thất bại cho Frontend, sau đó giao diện hiển thị kết quả cho khách hàng.

2.4.4. Sơ đồ lớp (Class Diagram)



Hình 2.23: Sơ đồ Lớp Class Diagram tổng quát

Mô tả sơ đồ Class Diagram tổng quát:

- *Lớp Admin*: Đại diện cho quản trị viên, có các phương thức quản lý người dùng, sản phẩm, đơn hàng, xem thống kê và cập nhật vai trò người dùng.

- *Lớp User*: Đại diện cho người dùng hệ thống, gồm các thuộc tính như user_id, name, email, password, phone, role và các phương thức như đăng ký, đăng nhập, cập nhật thông tin cá nhân, đặt hàng, đánh giá sản phẩm, thêm vào giỏ hàng.

- *Lớp Product*: Lưu trữ thông tin sản phẩm với các thuộc tính như `product_id`, `name`, `price`, `description`, `brand`, `category`, `stock` và các phương thức cập nhật tồn kho, thêm đánh giá.

- *Lớp Order*: Đại diện cho đơn hàng, liên kết với `User` và `Address`, gồm các thuộc tính `order_id`, `user`, `order_date`, `status`, `shipping_address`, `total_amount`, `orderDetails` và các phương thức thêm chi tiết đơn hàng, cập nhật trạng thái.

- *Lớp OrderDetail*: Lưu chi tiết từng sản phẩm trong đơn hàng, gồm `order_detail_id`, `order`, `product`, `quantity`, `price`.

- *Lớp Cart*: Đại diện cho giỏ hàng của người dùng, gồm `cart_id`, `user`, `created_at`, danh sách `CartItem` và các phương thức thêm, xóa sản phẩm, thanh toán.

- *Lớp CartItem*: Lưu thông tin từng sản phẩm trong giỏ hàng, gồm `cart_item_id`, `cart`, `product`, `quantity`.

- *Lớp Payment*: Lưu thông tin thanh toán cho đơn hàng, gồm `payment_id`, `order`, `payment_method` (chỉ nhận giá trị "BANKING" hoặc "COD"), `payment_date`, `amount`, `status` và phương thức xử lý thanh toán.

- *Lớp Address*: Lưu thông tin địa chỉ giao hàng, gồm `address_id`, `street`, `district`, `city`, `province`.

- *Lớp Review*: Lưu đánh giá sản phẩm của người dùng, gồm `review_id`, `product`, `user`, `rating`, `comment`, `created_at`.

- *Lớp Category*: Lưu thông tin danh mục sản phẩm, gồm `category_id`, `category_name`.

- *Lớp Brand*: Lưu thông tin thương hiệu sản phẩm, gồm `brand_id`, `brand_name`.

2.5. Tổng kết chương

Qua chương 2, em đã tiến hành phân tích và thiết kế hệ thống quản lý và bán hàng trực tuyến cho cửa hàng tiệm lẻ. Em đã xác định rõ các yêu cầu chức năng, phi chức năng, các ràng buộc và giả định của hệ thống.

Các mô hình hóa như sơ đồ ngữ cảnh, sơ đồ luồng dữ liệu, sơ đồ chức năng, sơ đồ ERD, sơ đồ hoạt động, sơ đồ trình tự và sơ đồ lớp đã giúp em hình dung rõ cấu trúc, luồng xử lý và mối quan hệ giữa các thành phần trong hệ thống. Việc chuẩn hóa cơ sở dữ liệu đến chuẩn thứ 3 giúp đảm bảo dữ liệu nhất quán, dễ bảo trì và mở rộng.

Những nội dung phân tích và thiết kế trong chương này là nền tảng quan trọng để em triển khai xây dựng hệ thống ở các chương tiếp theo. Đồng thời với cơ sở lý thuyết và mô hình thiết kế này sẽ giúp cho quá trình xây dựng hệ thống diễn ra thuận lợi hơn.

Chương 3: XÂY DỰNG VÀ TRIỂN KHAI HỆ THỐNG

3.1. Tổng quan về xây dựng và triển khai hệ thống

Sau khi hoàn thành các bước phân tích và thiết kế, em tiến hành xây dựng và triển khai hệ thống quản lý và bán hàng trực tuyến cho cửa hàng tiệm lẻ dựa trên các yêu cầu đã đề ra. Quá trình này bao gồm việc hiện thực hóa các chức năng chính, tích hợp các module, kiểm thử và đánh giá hệ thống trên môi trường thực tế.

Trong quá trình phát triển, em chú trọng đến việc đảm bảo tính ổn định, bảo mật và thân thiện với người dùng. Các công nghệ, công cụ và thư viện được lựa chọn phù hợp với mục tiêu đề tài, giúp hệ thống vận hành hiệu quả và dễ dàng mở rộng trong tương lai.

Việc triển khai hệ thống được thực hiện theo từng giai đoạn: xây dựng cơ sở dữ liệu, phát triển backend, xây dựng giao diện frontend, tích hợp các chức năng, kiểm thử tổng thể và hoàn thiện sản phẩm. Kết quả triển khai là một hệ thống hoàn chỉnh, đáp ứng tốt các nghiệp vụ quản lý, bán hàng và thanh toán trực tuyến cho cửa hàng tiệm lẻ.

3.2. Môi trường và công cụ phát triển.

3.2.1. Ngôn ngữ lập trình và Framework

- Phần backend của hệ thống được phát triển bằng Node.js kết hợp với Express. Node.js là nền tảng mạnh mẽ, hỗ trợ xây dựng các ứng dụng web hiệu năng cao, dễ mở rộng. Express giúp tổ chức các API RESTful một cách rõ ràng, thuận tiện cho việc phát triển và bảo trì.

- Phần frontend sử dụng ReactJS, một thư viện JavaScript phổ biến cho việc xây dựng giao diện người dùng động, thân thiện và dễ dàng quản lý trạng thái ứng dụng. ReactJS giúp tối ưu hóa trải nghiệm người dùng và tăng tốc độ phát triển giao diện.

3.2.2. Cơ sở dữ liệu

- MongoDB được lựa chọn làm hệ quản trị cơ sở dữ liệu. Đây là hệ quản trị cơ sở dữ liệu NoSQL, lưu trữ dữ liệu dưới dạng tài liệu (document), phù hợp với các ứng dụng web hiện đại nhờ khả năng mở rộng linh hoạt, truy vấn nhanh và dễ dàng tích hợp với Node.js.

3.2.3. Công cụ hỗ trợ phát triển

- Visual Studio Code (VS Code) là môi trường lập trình chính, cung cấp nhiều tiện ích mở rộng hỗ trợ lập trình JavaScript, Node.js, ReactJS, kiểm tra lỗi, tự động hoàn thành mã nguồn và quản lý dự án hiệu quả.

- Git được sử dụng để quản lý mã nguồn, giúp kiểm soát phiên bản, phối hợp làm việc nhóm và đảm bảo an toàn cho quá trình phát triển phần mềm.

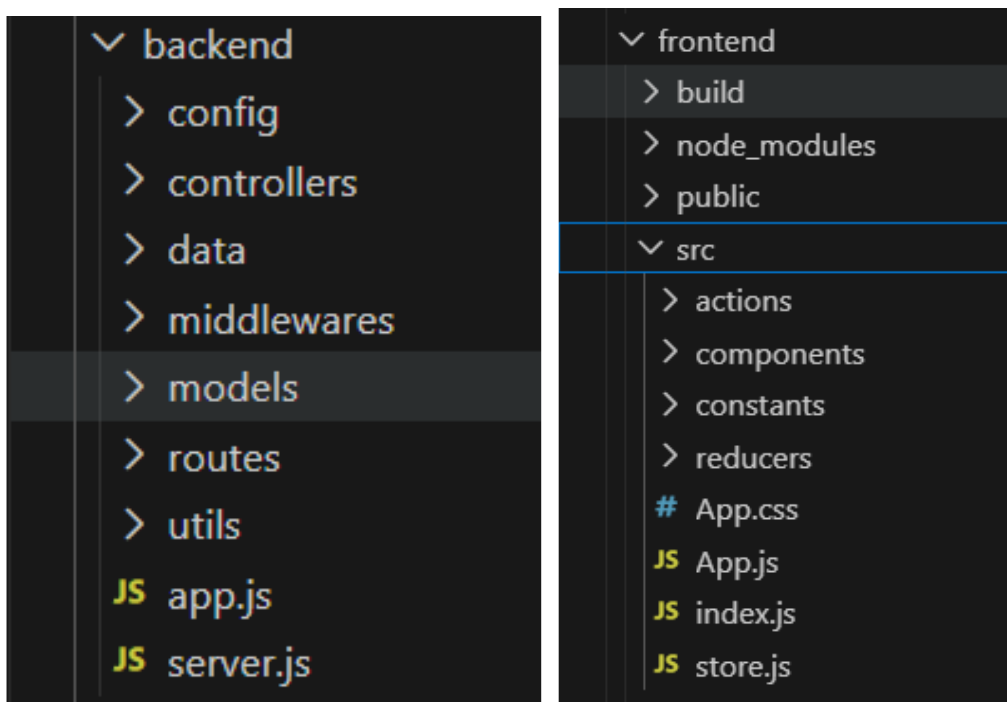
3.2.4. Thư viện dùng để tích hợp

- Axios được sử dụng để giao tiếp giữa frontend và backend thông qua các API.
- Redux hỗ trợ quản lý trạng thái ứng dụng trên frontend, giúp dữ liệu nhất quán giữa các thành phần giao diện.
- Chart.js được tích hợp để hiển thị các biểu đồ thống kê doanh thu, số lượng đơn hàng, sản phẩm bán chạy.
- Hệ thống tích hợp cổng thanh toán trực tuyến thông qua app Ngân Hàng, đáp ứng nhu cầu thanh toán hiện đại của khách hàng.

3.3. Quá trình xây dựng hệ thống

3.3.1. Khởi tạo dự án và cấu trúc thư mục

Thực hiện tạo hai thư mục chính là backend và frontend. Thư mục backend chứa mã nguồn *Node.js/Express*, các controller, model, route, middleware, cấu hình cơ sở dữ liệu và các tiện ích. Thư mục frontend chứa mã nguồn *ReactJS*, các component, actions, reducers, store, và các file cấu hình giao diện.



Hình 3.1: Bố cục giao diện Backend và Frontend

3.3.2. Thiết kế và xây dựng cơ sở dữ liệu

Hệ thống đã sử dụng MongoDB để lưu trữ dữ liệu. Các collection chính gồm: users, products, orders.

localhost:27017 > cuahangtienloi

[Open MongoDB shell](#) [Create collection](#) [Refresh](#)

Collection name	Properties	Storage size	Documents	Avg. document size	Indexes	Total index size
orders	-	36.86 kB	15	704.00 B	1	36.86 kB
products	-	77.82 kB	72	812.00 B	1	36.86 kB
users	-	36.86 kB	8	348.00 B	2	73.73 kB

Hình 3.2: Các Collection chính của hệ thống trong CSDL MongoDB

Chi tiết và các thuộc tính trong các Collection chính:

- User:

```
{
  "_id": ObjectId('641b330bd15ca136d82e2d66'),
  "role": "admin",
  "name": "admin Cao",
  "email": "admin@gmail.com",
  "password": "$2a$10$GTsmAtvMXCFwrEp1B04vZ.pngkmg8Rx0dE.y0km/mhmKpIJXXAQ5K",
  "avatar": Object,
  "createdAt": 2023-03-22T16:55:39.137+00:00,
  "__v": 0
}
```

Hình 3.3: Các thuộc tính của đối tượng người dùng (Users)

Mô tả thuộc tính của đối tượng người dùng trong hệ thống:

- + **_id**: Mã định danh duy nhất của người dùng.
- + **role**: Vai trò của người dùng trong hệ thống, ví dụ: "admin" (quản trị viên), dùng để phân quyền truy cập và thao tác.
- + **name**: Họ và tên đầy đủ của người dùng, phục vụ cho việc hiển thị và quản lý thông tin cá nhân.
- + **email**: Địa chỉ email của người dùng, dùng để đăng nhập và liên hệ.
- + **password**: Mật khẩu của người dùng đã được mã hóa (băm), đảm bảo an toàn bảo mật thông tin đăng nhập.
- + **avatar**: Thông tin về ảnh đại diện của người dùng, bao gồm:
 - public_id**: Mã định danh ảnh trên hệ thống lưu trữ (ví dụ Cloudinary).
 - url**: Đường dẫn truy cập trực tiếp đến ảnh đại diện.
 - createdAt**: Thời gian tài khoản được tạo, giúp theo dõi lịch sử đăng ký của người dùng.

- Products:

```
_id: ObjectId('691fc902ae75c146ac7b32ea')
price: 89500
ratings: 0
stock: 9
numOfReviews: 0
name: "cá xịn"
description: "TEST"
category: "Thịt-Trứng-Hải sản"
seller: "cao"
▶ images: Array (1)
  user: ObjectId('641b330bd15ca136d82e2d66')
  ▶ reviews: Array (empty)
  createdAt: 2025-11-21T02:05:54.378+00:00
  __v: 0
```

Hình 3.4: Các thuộc tính của đối tượng sản phẩm (Products)

Mô tả thuộc tính của đối tượng sản phẩm (Products):

- + *_id*: Mã định danh duy nhất của sản phẩm.
- + *price*: Giá bán của sản phẩm (đơn vị: VNĐ).
- + *ratings*: Điểm đánh giá trung bình của sản phẩm từ người dùng.
- + *stock*: Số lượng sản phẩm còn trong kho.
- + *numOfReviews*: Số lượng đánh giá mà sản phẩm đã nhận được.
- + *name*: Tên sản phẩm.
- + *description*: Mô tả chi tiết về sản phẩm.
- + *category*: Danh mục mà sản phẩm thuộc về (ví dụ: Thịt-Trứng-Hải sản).
- + *seller*: Tên người bán hoặc đơn vị cung cấp sản phẩm.
- + *images*: Danh sách các ảnh minh họa cho sản phẩm (mỗi ảnh là một đối tượng chứa thông tin về ảnh).
- + *user*: Mã định danh của người dùng đã tạo sản phẩm này.
- + *reviews*: Danh sách các đánh giá về sản phẩm (mỗi đánh giá là một đối tượng chứa thông tin chi tiết về nhận xét của người dùng).
- + *createdAt*: Thời gian sản phẩm được tạo trong hệ thống.

- Orders:

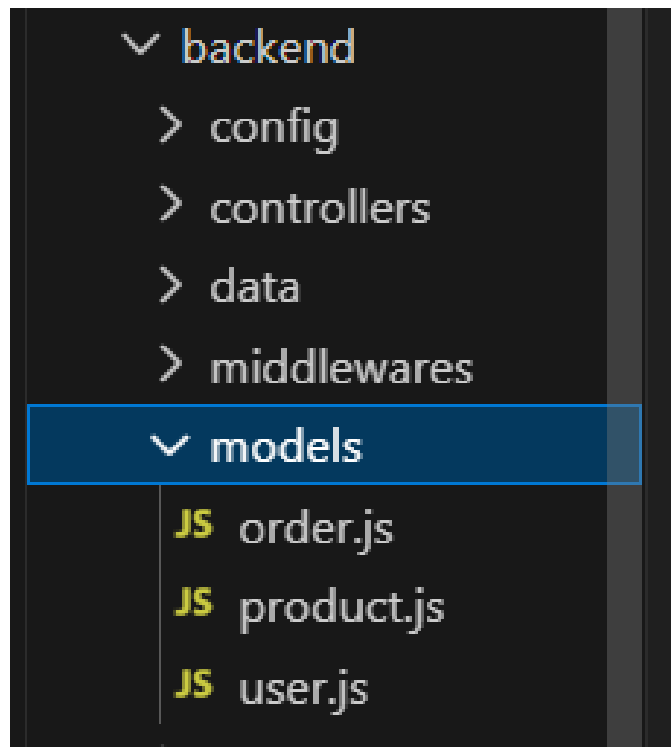
```
_id: ObjectId('6906e3d2a9a7fc1ef44f1412')
itemsPrice : 132310
taxPrice : 0
shippingPrice : 0
totalPrice : 132310
orderStatus : "Đã đặt hàng"
▶ orderItems : Array (1)
▶ shippingInfo : Object
▶ paymentInfo : Object
  paidAt : 2025-11-02T04:53:38.262+00:00
  user : ObjectId('68e6711c0651663434be78b9')
  createdAt : 2025-11-02T04:53:38.266+00:00
__v : 0
```

Hình 3.5: Các thuộc tính của đối tượng đơn hàng (Orders)

Mô tả thuộc tính của đối tượng đơn hàng (Orders):

- + *_id*: Mã định danh duy nhất của đơn hàng.
- + *itemsPrice*: Tổng giá trị các sản phẩm trong đơn hàng (chưa bao gồm thuế và phí vận chuyển).
- + *taxPrice*: Số tiền thuế áp dụng cho đơn hàng.
- + *shippingPrice*: Phí vận chuyển của đơn hàng.
- + *totalPrice*: Tổng số tiền phải thanh toán cho đơn hàng (đã bao gồm giá sản phẩm, thuế và phí vận chuyển).
- + *orderStatus*: Trạng thái hiện tại của đơn hàng (ví dụ: "Đã đặt hàng").
- + *orderItems*: Danh sách các sản phẩm trong đơn hàng (mỗi phần tử là một đối tượng chứa thông tin chi tiết về sản phẩm).
- + *shippingInfo*: Thông tin về địa chỉ giao hàng (là một đối tượng chứa các trường như địa chỉ, số điện thoại, tên người nhận, v.v.).
- + *paymentInfo*: Thông tin về thanh toán của đơn hàng (là một đối tượng chứa các trường như trạng thái thanh toán, mã giao dịch, v.v.).
- + *paidAt*: Thời gian đơn hàng được thanh toán.
- + *user*: Mã định danh của người dùng đã đặt đơn hàng.
- + *createdAt*: Thời gian đơn hàng được tạo trong hệ thống.

Các schema được định nghĩa trong thư mục *backend/models* như *user.js*, *product.js*, *order.js*, v.v. Dữ liệu mẫu được lưu trong các file JSON để phục vụ kiểm thử.

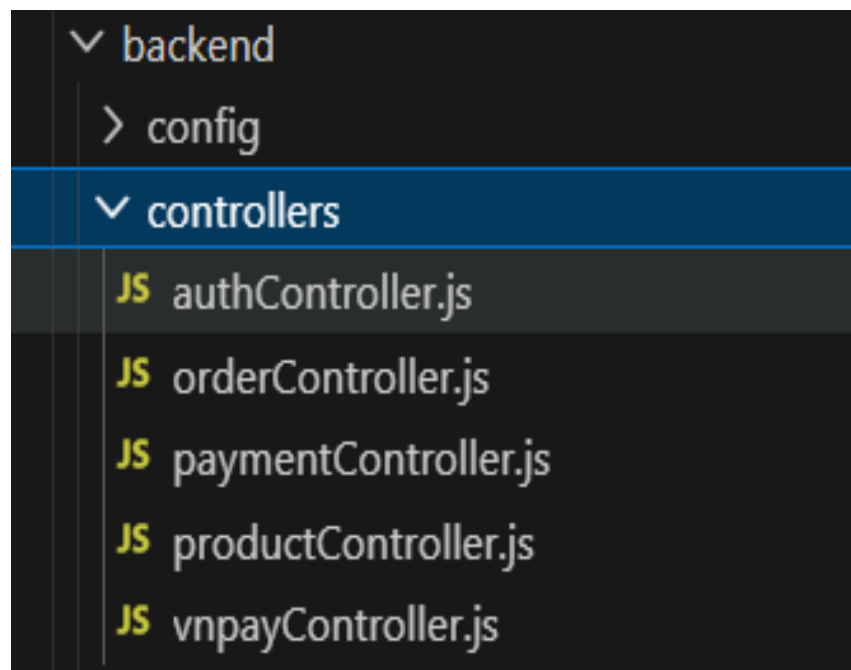


Hình 3.6: Bố cục của thư mục Models trong Backend

3.3.3. Phát triển BackEnd

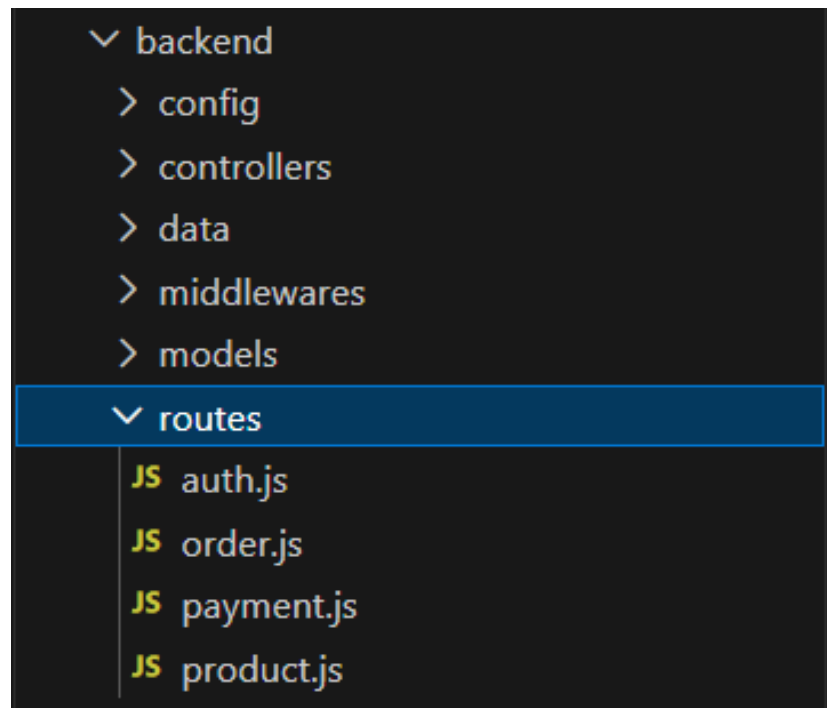
Tiếp theo là xây dựng các API RESTful cho các chức năng: đăng ký, đăng nhập, quản lý người dùng, quản lý sản phẩm, quản lý đơn hàng, đánh giá sản phẩm, thanh toán (tích hợp App Banking Thanh toán khi nhận hàng (COD)).

- Các *controller* xử lý logic trong *backend/controllers*:



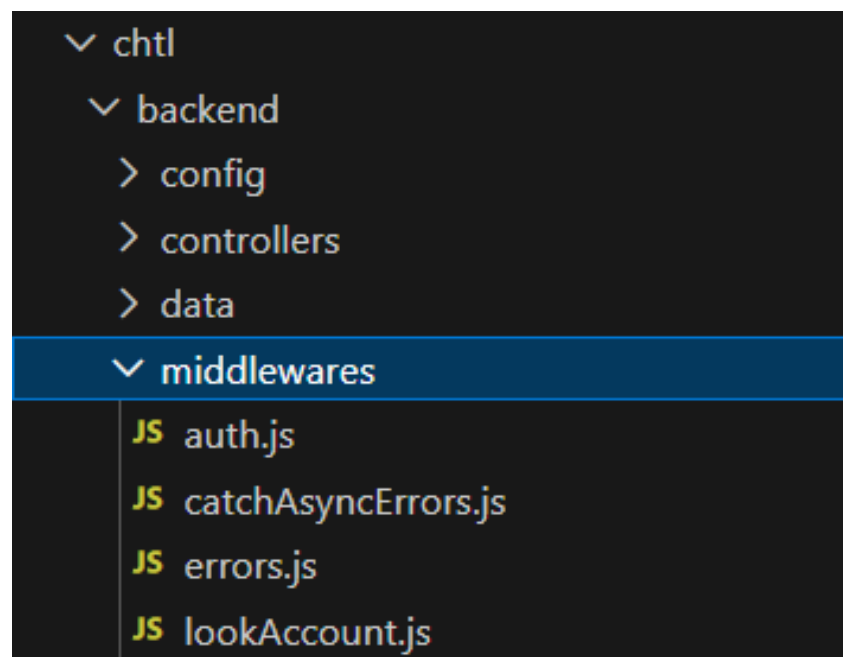
Hình 3.7: Các API RESTful cho các chức năng trong mục Controllers

- Các *route* được tổ chức trong thư mục *backend/routes*:



Hình 3.8: Các routes tổ chức trong mục Backend

- Các *middleware* như xác thực, phân quyền, xử lý lỗi được đặt trong *backend/middlewares*.

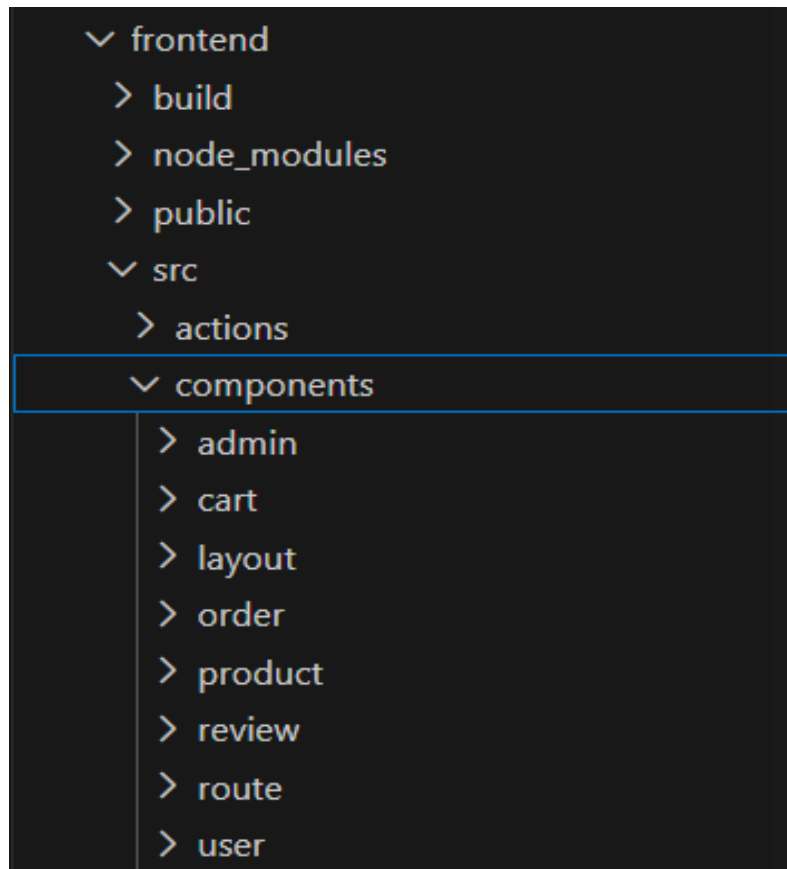


Hình 3.9: Các middlewares được tổ chức trong mục Backend

3.3.4. Phát triển Frontend

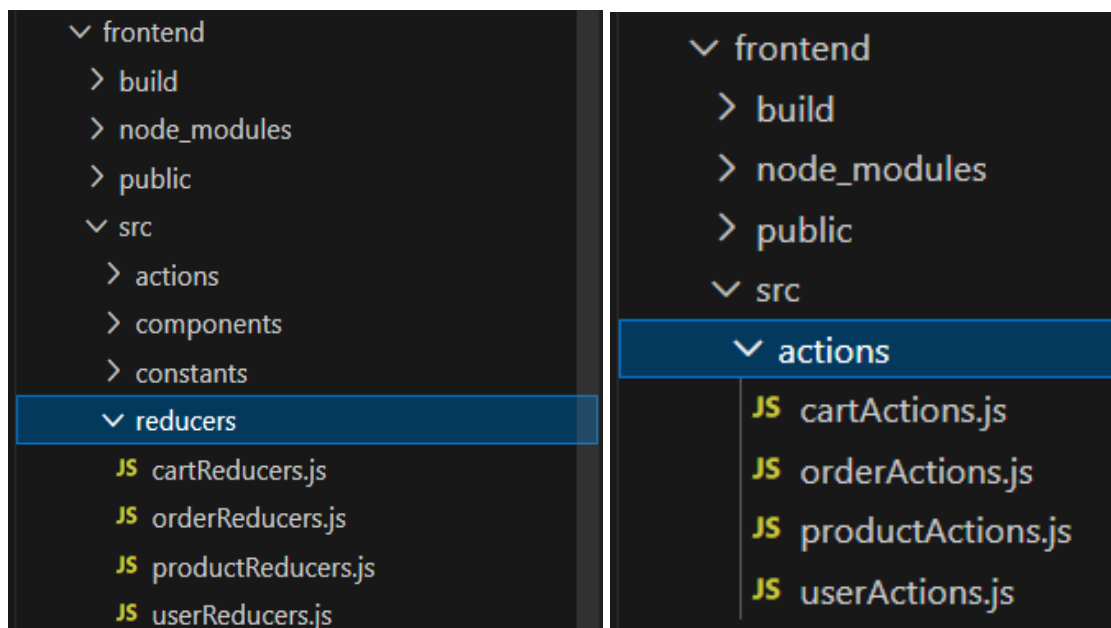
Giao diện người dùng được xây dựng bằng ReactJS, chia thành các component nhỏ theo chức năng: quản lý sản phẩm, đơn hàng, người dùng, giỏ hàng, thanh toán,

đánh giá, thống kê. Các component được tổ chức trong các thư mục như *components/admin*, *components/cart*, *components/layout*, v.v.



Hình 3.10: Bố cục các Components nhỏ theo các chức năng khác nhau

Quản lý trạng thái sử dụng Redux, các action và reducer được đặt trong các thư mục riêng biệt.



Hình 3.11: Bố cục các reducer và action trong Frontend

3.3.5. Tích hợp các chức năng và dịch vụ

Hệ thống sử dụng Axios để giao tiếp giữa frontend và backend. Các chức năng như tìm kiếm, lọc sản phẩm, phân trang, đánh giá sản phẩm, quản lý trạng thái đơn hàng, thống kê doanh thu đều được hiện thực hóa rõ ràng.

Thanh toán được tích hợp qua App Banking và thanh toán khi nhận hàng (COD), đảm bảo đa dạng phương thức thanh toán cho khách hàng.

- Giao tiếp giữa Backend với Frontend

- + Được thực hiện qua thư viện Axios ở phía frontend (thường trong các file actions hoặc services của React).

- + Các request này sẽ gọi đến các API RESTful do backend xây dựng (các controller trong backend)

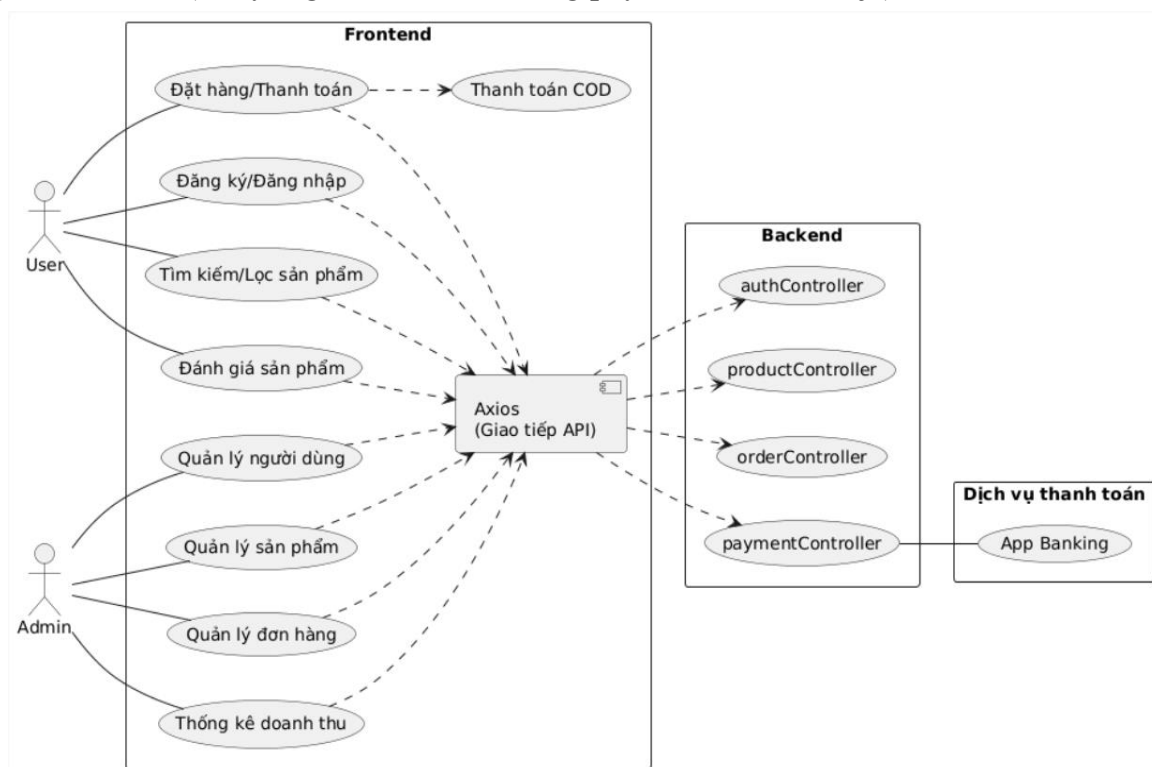
- Các chức năng nghiệp vụ khác

- + Tìm kiếm, lọc sản phẩm, phân trang, đánh giá sản phẩm, quản lý trạng thái đơn hàng, thống kê doanh thu:

- + Được hiện thực hóa ở cả frontend (giao diện, gọi API) và backend (xử lý logic trong các controller như productController.js, orderController.js).

- + Thanh toán qua App Banking và COD:

- + Được tích hợp ở cả frontend (giao diện chọn phương thức thanh toán, gửi thông tin) và backend (xử lý logic thanh toán trong paymentController.js).

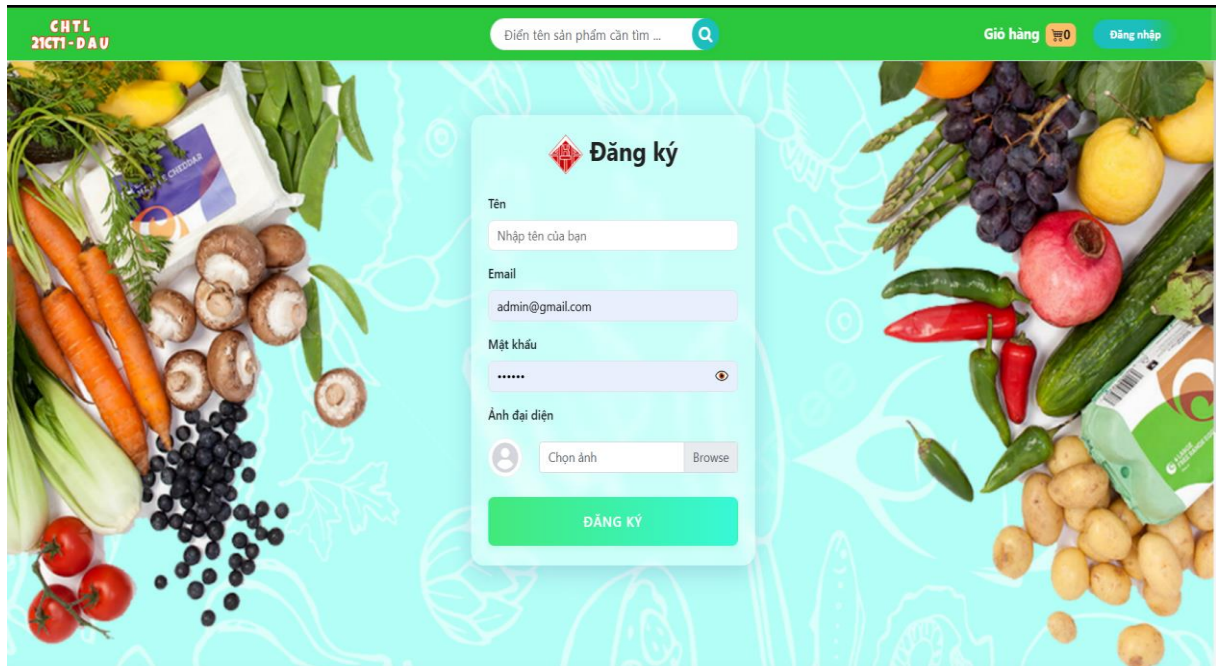


Hình 3.12: Sơ đồ giao tiếp của Axios trong hệ thống

3.4. Xây dựng các chức năng chính

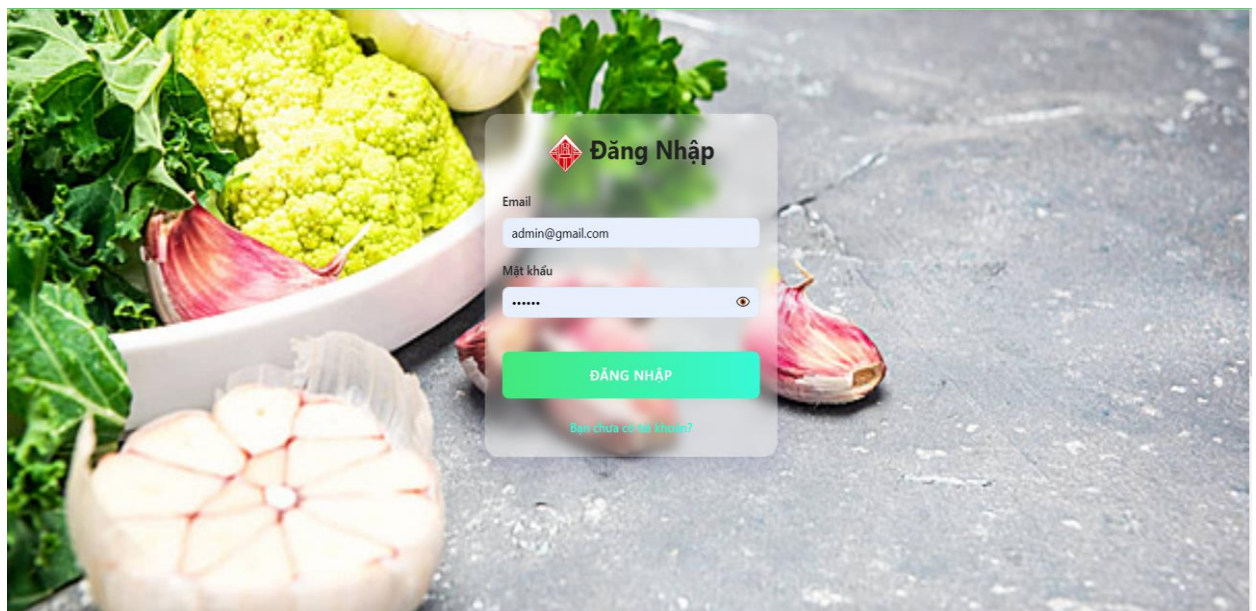
3.4.1. Đăng kí/Đăng nhập

- Đăng kí:



Hình 3.13: Giao diện đăng kí

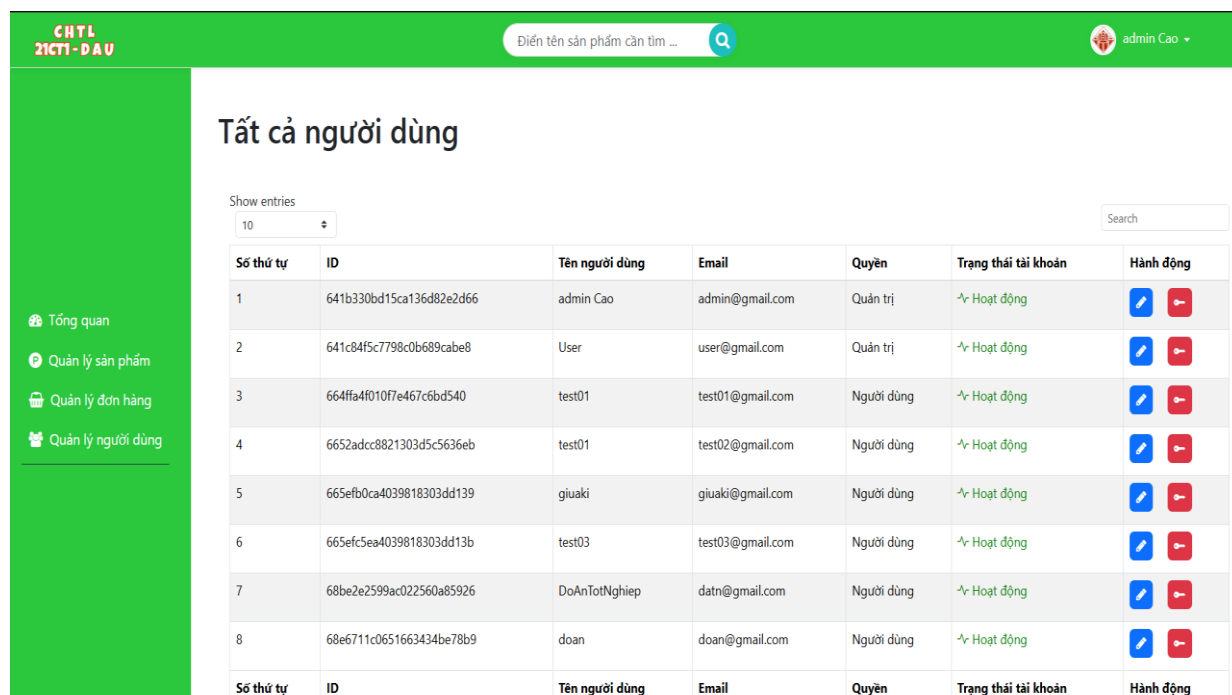
- Đăng nhập:



Hình 3.14: Giao diện đăng nhập

3.4.2. Quản lý người dùng

- Danh sách người dùng:

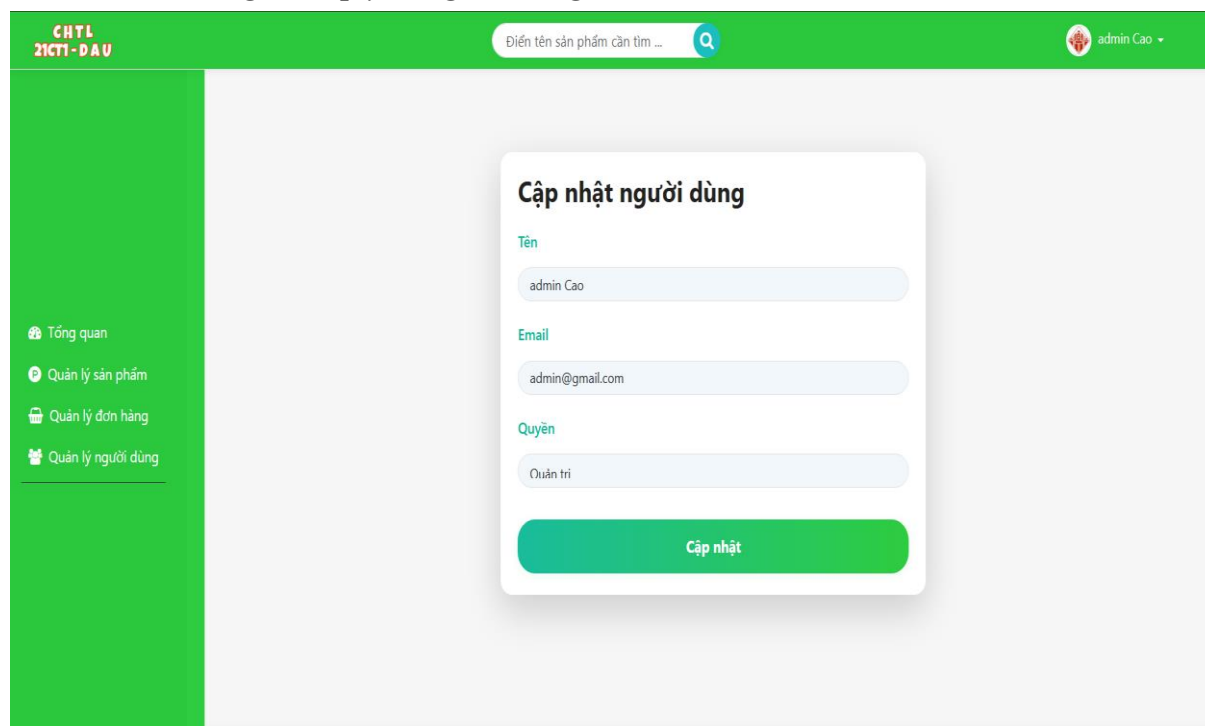


The screenshot shows a web application interface for managing users. On the left is a green sidebar with navigation links: Tổng quan, Quản lý sản phẩm, Quản lý đơn hàng, and Quản lý người dùng (highlighted). The main content area has a green header with the logo 'CHTL 21CT1 - ĐÀU', a search bar, and a user profile 'admin Cao'. Below the header, the title 'Tất cả người dùng' is displayed. A 'Show entries' dropdown is set to '10'. A table lists 8 users with columns: Số thứ tự, ID, Tên người dùng, Email, Quyền, Trạng thái tài khoản, and Hành động. Each user row has edit and delete icons. At the bottom, there is a summary row with the same column headers.

Số thứ tự	ID	Tên người dùng	Email	Quyền	Trạng thái tài khoản	Hành động
1	641b330bd15ca136d82e2d66	admin Cao	admin@gmail.com	Quản trị	Hoạt động	[Edit] [Delete]
2	641c84f5c7798c0b689cabe8	User	user@gmail.com	Quản trị	Hoạt động	[Edit] [Delete]
3	664ffa4f010f7e467c6bd540	test01	test01@gmail.com	Người dùng	Hoạt động	[Edit] [Delete]
4	6652adcc8821303d5c5636eb	test01	test02@gmail.com	Người dùng	Hoạt động	[Edit] [Delete]
5	665efb0ca4039818303dd139	giuaki	giuaki@gmail.com	Người dùng	Hoạt động	[Edit] [Delete]
6	665efc5ea4039818303dd13b	test03	test03@gmail.com	Người dùng	Hoạt động	[Edit] [Delete]
7	68be2e2599ac022560a85926	DoAnTotNghiep	datn@gmail.com	Người dùng	Hoạt động	[Edit] [Delete]
8	68e6711c0651663434be78b9	doan	doan@gmail.com	Người dùng	Hoạt động	[Edit] [Delete]
Số thứ tự	ID	Tên người dùng	Email	Quyền	Trạng thái tài khoản	Hành động

Hình 3.15: Giao diện quản lý danh sách người dùng

- Chỉnh sửa thông tin - quyền người dùng:



The screenshot shows the 'Cập nhật người dùng' (Update user) form. It is a white modal box centered on a light gray background. The form contains three input fields: 'Tên' (Name) with the value 'admin Cao', 'Email' with the value 'admin@gmail.com', and 'Quyền' (Role) with the value 'Quản trị'. Below these fields is a green 'Cập nhật' (Update) button. The left sidebar and top header are identical to the previous screenshot.

Hình 3.16: Giao diện chỉnh sửa người dùng

3.4.3. Quản lý sản phẩm

- Danh sách sản phẩm:

Tổng quan

Quản lý sản phẩm

Quản lý đơn hàng

Quản lý người dùng

Tất cả sản phẩm

Thêm sản phẩm mới

Show entries10

Search

STT	ID	Tên sản phẩm	Giá	Số lượng	Giờ tạo	Ngày tạo	Hành động
1	664fe7f0010f7e467c6bd4d9	Bánh gạo táo biến bạch tuộc nướng One One gói 104g	23.000 VNĐ	242	08:05:52	24/5/2024	 
2	664fe909010f7e467c6bd4db	Bánh gạo vị bò nướng tiêu một nắng Richy Jinju Gold gói 134.4g	26.500 VNĐ	234	08:10:33	24/5/2024	 
3	664fe984010f7e467c6bd4dd	Kẹo socola nhân đậu phộng M&M's gói 90g	45.000 VNĐ	430	08:12:36	24/5/2024	 
4	664fea0e010f7e467c6bd4df	Bánh chocopie Orion hộp 396g (12 cái)	54.500 VNĐ	45	08:14:54	24/5/2024	 
5	664fea96010f7e467c6bd4e1	Kẹo cà phê Coffeeshot Classic KOPIKO gói 135g	15.500 VNĐ	654	08:17:10	24/5/2024	 
6	664feb21010f7e467c6bd4e3	Kẹo hương dâu kem Alpenliebe gói 115.5g	14.500 VNĐ	123	08:19:29	24/5/2024	 
7	664feb70010f7e467c6bd4e5	Kẹo nhai Mentos cầu vồng hương trái cây (3 viên) gói 120g	16.500 VNĐ	43	08:20:48	24/5/2024	 
8	664febff010f7e467c6bd4e7	Kẹo mềm hương trái cây Alpenliebe 2Chew gói 84g	14.000 VNĐ	536	08:23:11	24/5/2024	 

Hình 3.17: Giao diện quản lý danh sách sản phẩm

- Thêm/sửa/xoá sản phẩm:

Thêm sản phẩm mới

Tên sản phẩm

Giá

0

Mô tả

Danh mục

Rau-Củ-Trái cây

Số lượng tồn

0

Nhà sản xuất

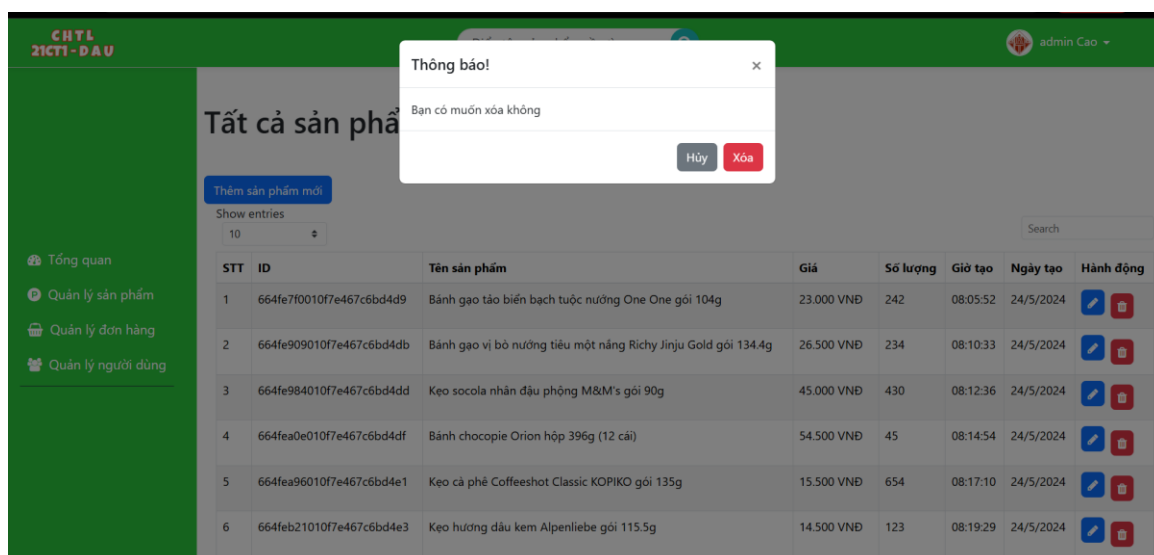
Hình sản phẩm

Chọn tệp

Không có tệp nào được chọn

Thêm mới

Hình 3.18: Giao diện thêm sản phẩm



Hình 3.19: Giao diện xóa sản phẩm.

3.4.4. Tìm kiếm sản phẩm, lựa chọn menu sản phẩm.

- Giao diện tìm kiếm sản phẩm:



Hình 3.20: Giao diện tìm kiếm sản phẩm

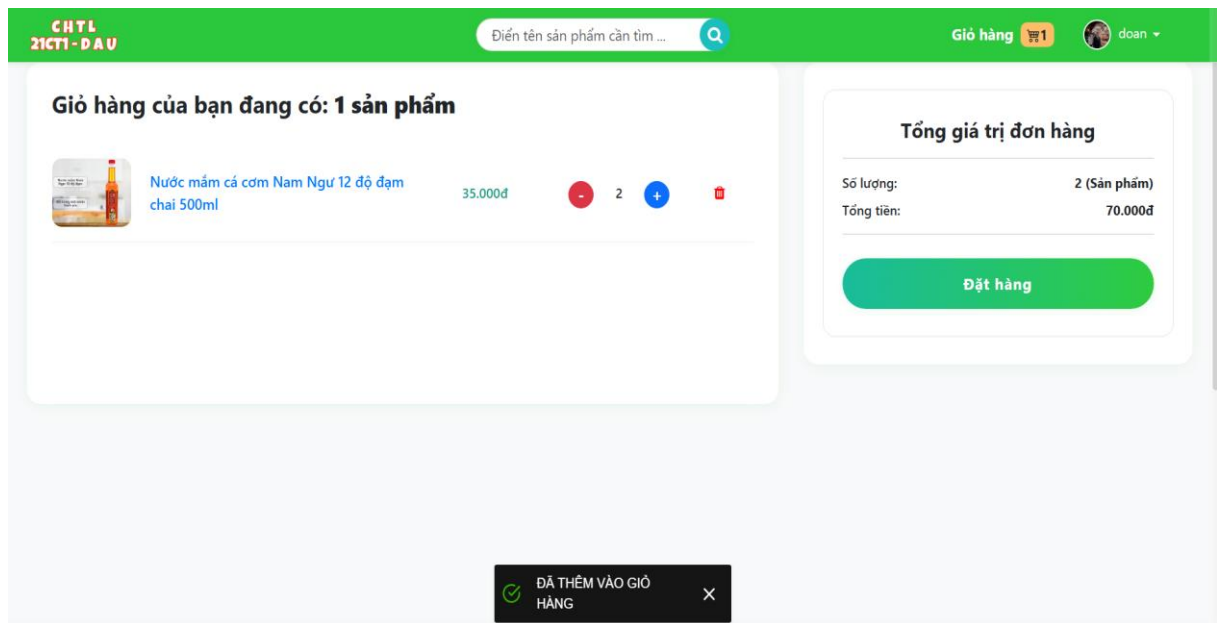
- Giao diện lựa chọn mặt hàng sản phẩm trên thanh menu:



Hình 3.21: Giao diện menu lựa chọn loại sản phẩm.

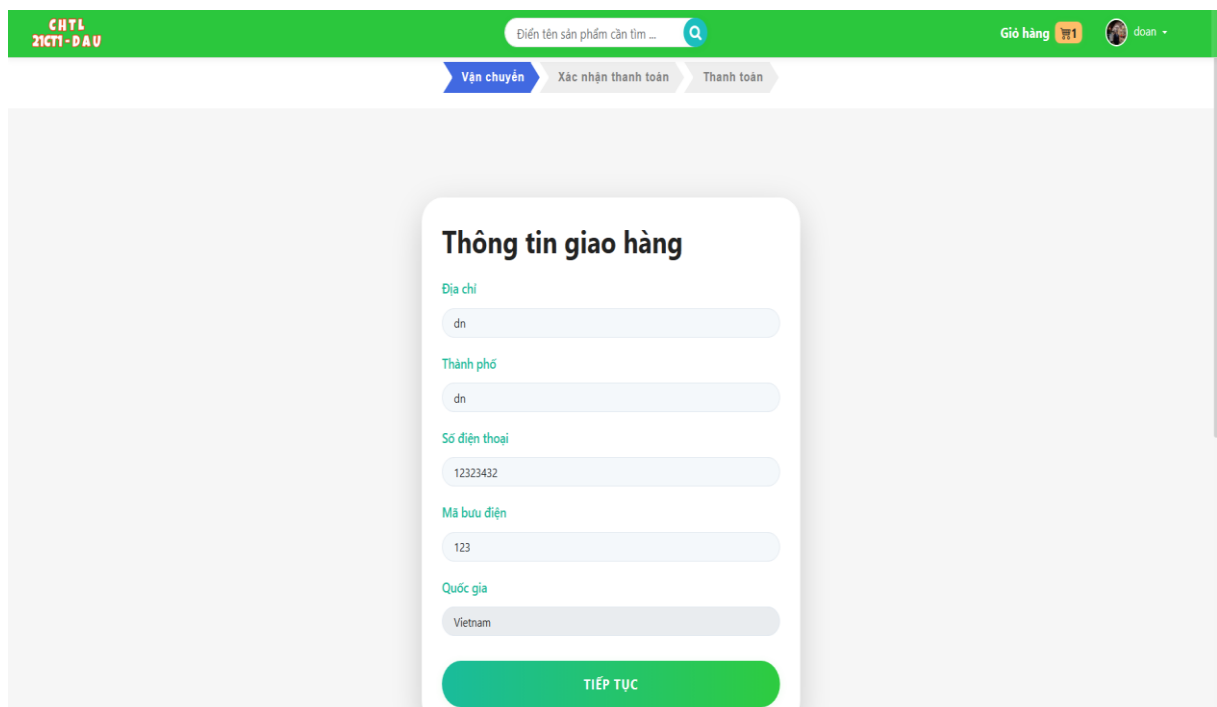
3.4.5. Đặt hàng và quản lý đơn hàng.

- Giao diện giỏ hàng:



Hình 3.22: Giao diện giỏ hàng.

- Giao diện xác nhận đơn hàng:



Hình 3.23: Giao diện xác nhận thông tin vận chuyển

- Giao diện danh sách đơn hàng của người dùng:

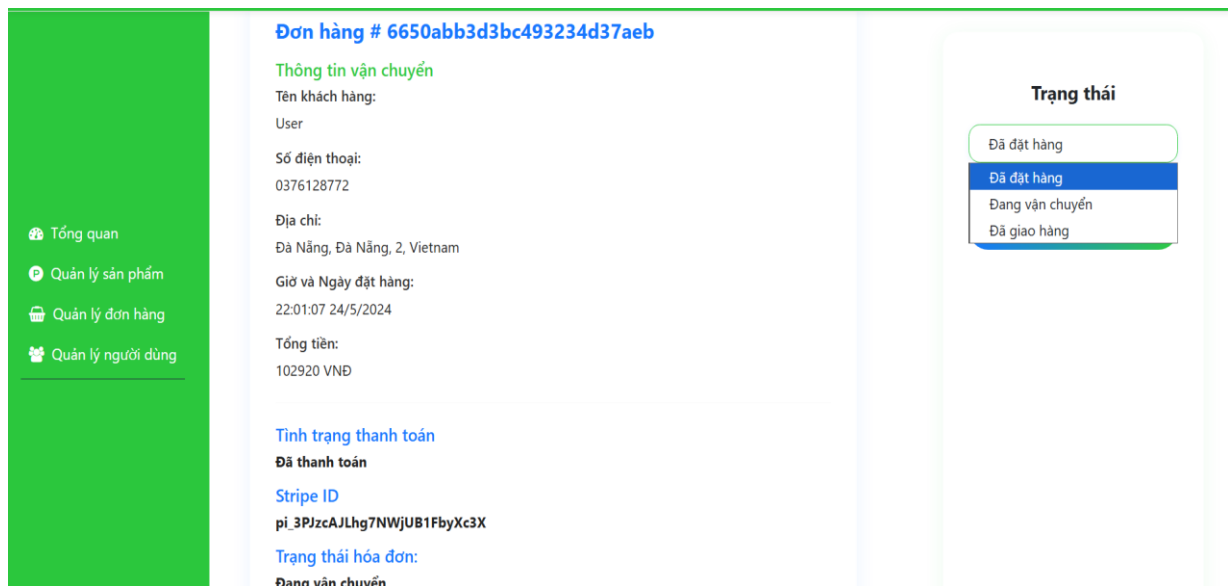
CHTL 21CT1 - Đ A U Điền tên sản phẩm cần tìm ... Q Giỏ hàng 1 doan ▼							
Đơn hàng của tôi							
STT	ID	Số lượng	Tổng tiền	Trạng thái	Giờ tạo	Ngày tạo	Hành động
1	6906d5e31b4dfe31187ef943	2	271.690đ	Đã đặt hàng	10:54:11	2/11/2025	
2	6906d8713831791f7c9ca0f5	1	117.665đ	Đã đặt hàng	11:05:05	2/11/2025	
3	6906da5bc8f0a52edc086c6f	1	75.145đ	Đã đặt hàng	11:13:15	2/11/2025	
4	6906dbdf89ce84372cf54036	1	26.665đ	Đã đặt hàng	11:19:43	2/11/2025	
5	6906e3d2a9a7fc1ef44f1412	1	132.310đ	Đã đặt hàng	11:53:38	2/11/2025	
6	692984e597b3354468e0cdd3	1	57.470đ	Đã đặt hàng	18:17:57	28/11/2025	
7	6929850197b3354468e0cde5	1	69.590đ	Đã đặt hàng	18:18:25	28/11/2025	
8	69298ca72768d229d8317e53	1	119.180đ	Đã đặt hàng	18:51:03	28/11/2025	
9	69299837e293804b543d8f07	2	165.640đ	Đang vận chuyển	19:40:23	28/11/2025	

Hình 3.24: Giao diện danh sách đơn hàng

- Giao diện quản lý đơn hàng của Admin:

CHTL 21CT1 - Đ A U Điền tên sản phẩm cần tìm ... Q admin Cao ▼								
Tất cả đơn hàng								
Show entries 10 ▼		Search						
Số thứ tự	ID	Số mặt hàng	Tổng tiền	Giờ tạo	Ngày tạo	Trạng thái	Hành động	
1	6650abb3d3bc493234d37aeb	1	102920 VNĐ	22:01:07	24/5/2024	Đang vận chuyển		
2	6652b28b8821303d5c5636ec	1	227250 VNĐ	10:54:51	26/5/2024	Đã đặt hàng		
3	665c03520b2a003ab482a1a9	1	249975 VNĐ	12:29:54	2/6/2024	Đã đặt hàng		
4	665efe80a4039818303dd13c	1	588325 VNĐ	18:46:08	4/6/2024	Đã giao hàng		
5	66614db371722036a048b81b	2	207555 VNĐ	12:48:35	6/6/2024	Đã giao hàng		
6	6730972e9893c11cbcbf239f	1	100900 VNĐ	18:21:18	10/11/2024	Đã đặt hàng		
7	67309a8a9893c11cbcbf23a3	1	111000 VNĐ	18:35:38	10/11/2024	Đã đặt hàng		

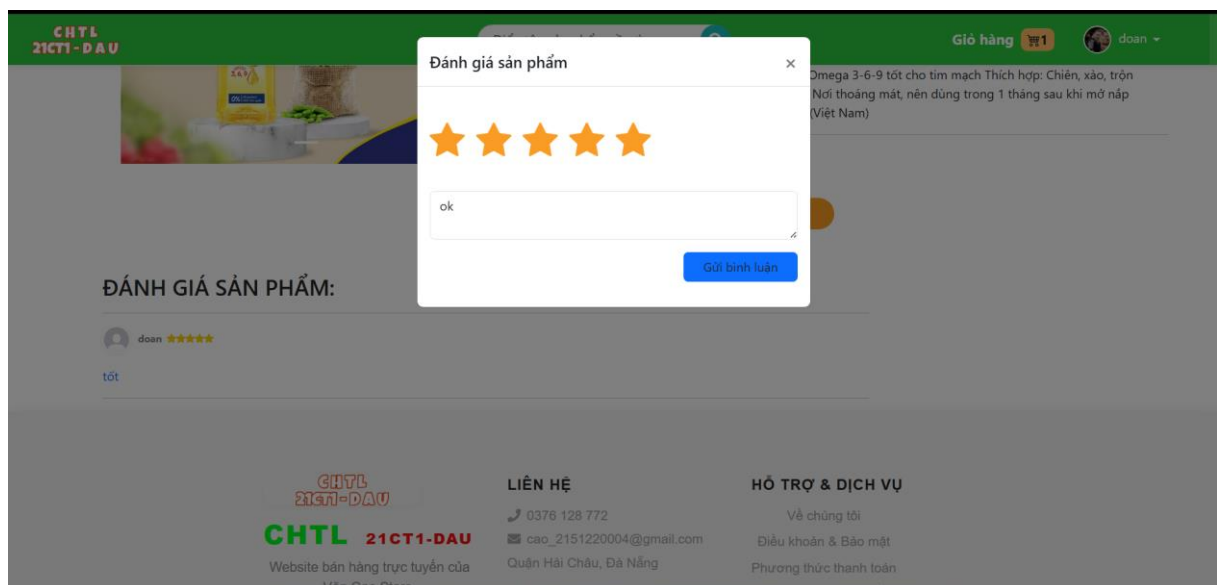
Hình 3.25: Giao diện quản lý đơn hàng (Admin)



Hình 3.26: Giao diện quản lý/cập nhật đơn hàng.

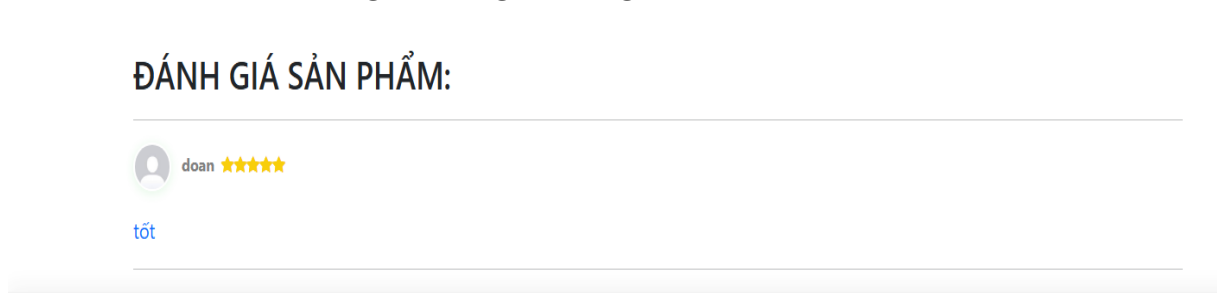
3.4.6. Đánh giá sản phẩm.

- Giao diện đánh giá sản phẩm:



Hình 3.27: Giao diện đánh giá sản phẩm

- Giao diện hiển thị đánh giá của người dùng:



Hình 3.28: Giao diện hiển thị đánh giá người dùng

3.4.7. Thanh toán.

- Giao diện chọn phương thức thanh toán:

The screenshot shows a payment selection screen with three tabs: 'Vận chuyển', 'Xác nhận thanh toán', and 'Thanh toán'. The 'Xác nhận thanh toán' tab is active. It contains a form with the following sections:

- Thông tin vận chuyển:** Tên khách hàng: doan, Số điện thoại: 12323432, Địa chỉ: đn, đn, 123, Vietnam.
- Chọn phương thức thanh toán:** Two radio buttons: 'Thanh toán khi nhận hàng(COD)' (selected) and 'Thanh toán qua Banking App'.
- Các mặt hàng trong giỏ hàng của bạn:** A list showing 'Nước mắm cá cơm Nam Ngư 12 độ đậm chai 500ml' with a quantity of 2 and a price of 35.000đ, totaling 70.000đ.
- Tổng giá trị đơn hàng:** A summary box on the right showing: Thành tiền: 70.000đ, Phí vận chuyển: 10.000đ, Thuế: 700đ, and a total of 80.700đ. A green 'Tiếp tục' button is at the bottom.

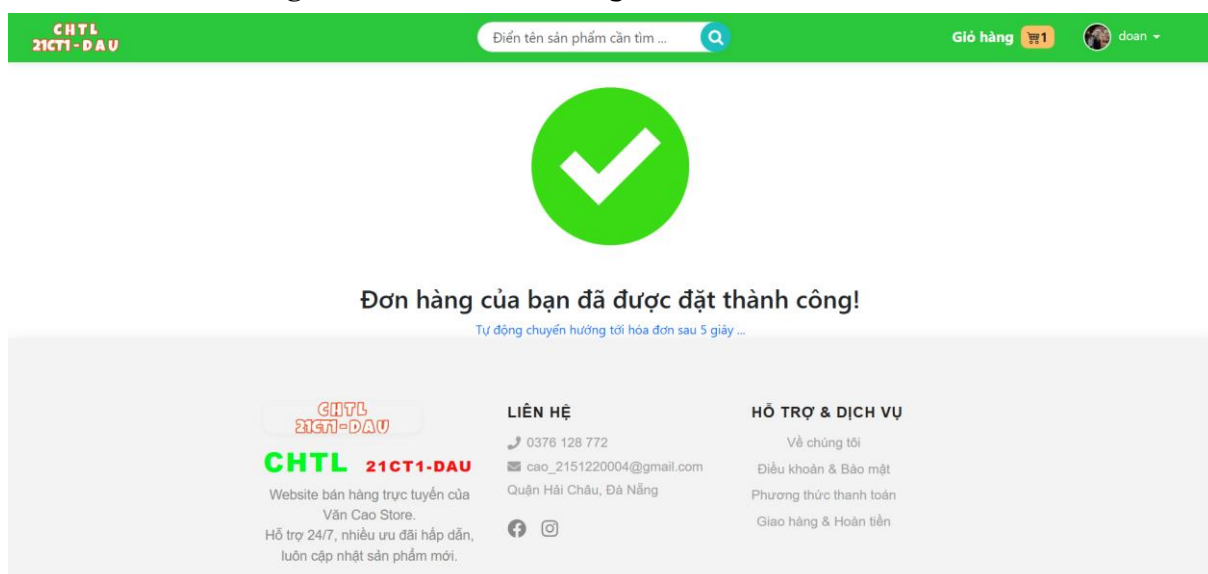
Hình 3.29: Giao diện chọn phương thức thanh toán

- Giao diện thanh toán trực tuyến qua Banking:

The screenshot shows the online banking payment interface. At the top, there's a green header with 'CHTL 21CT1 - ĐÀU', a search bar, and a shopping cart icon showing 1 item. Below the header are three tabs: 'Vận chuyển', 'Xác nhận thanh toán', and 'Thanh toán'. The 'Thanh toán' tab is active, displaying a QR code for payment. The QR code is labeled with 'BIDV' and 'VIETOR' logos. Below the QR code, the customer's name 'NGUYỄN VĂN CAO' and phone number '562342324' are shown. The total amount 'Số tiền: 80.700đ' is displayed. Below the QR code, the order number 'Nội dung chuyển khoản: ORDER1764340298765' is shown. At the bottom, there's a message 'Sau khi thanh toán thành công, vui lòng nhấn nút bên dưới để xác nhận đặt hàng.' and a blue button labeled 'Tôi đã thanh toán xong'.

Hình 3.30: Giao diện thanh toán qua ngân hàng

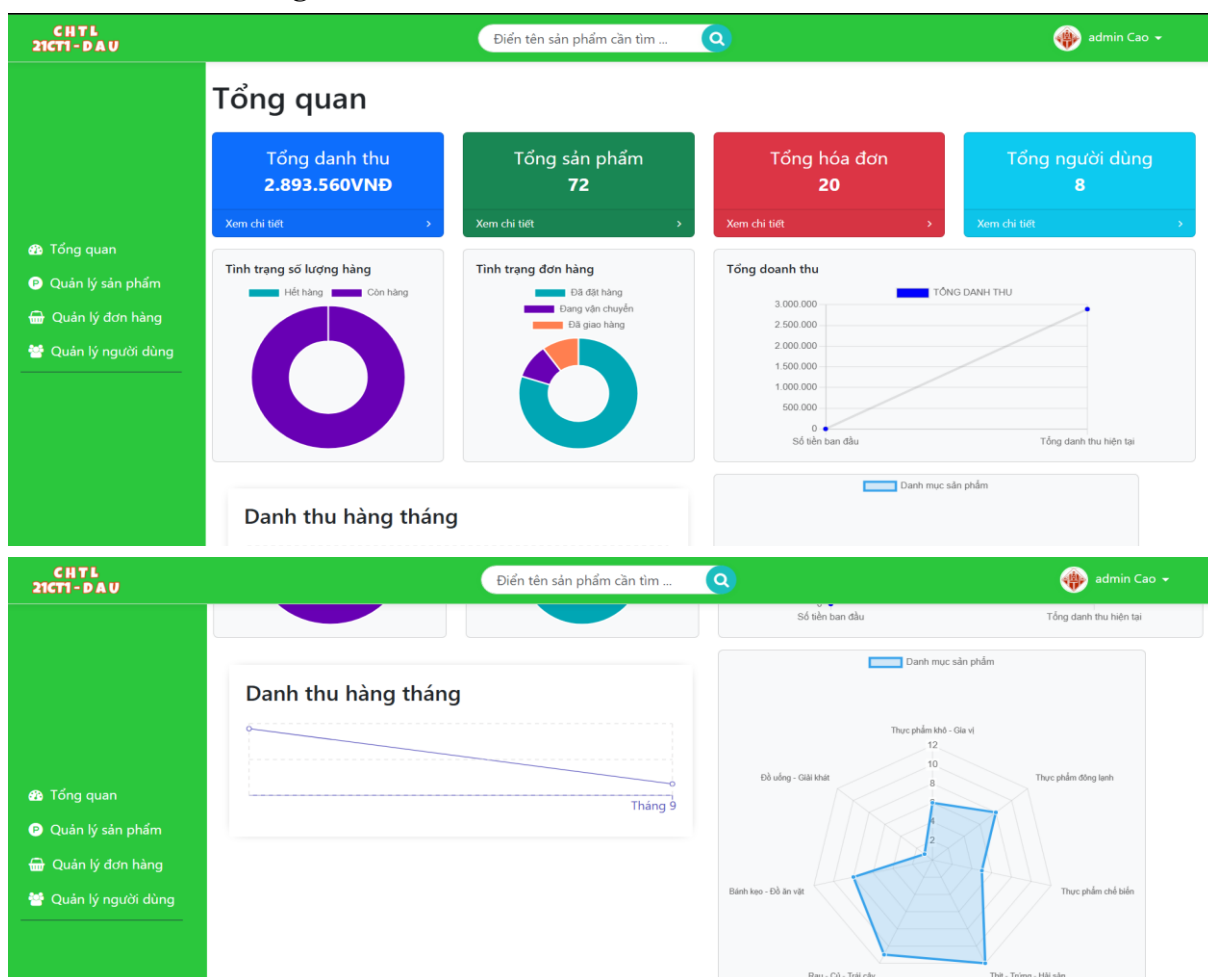
- Giao diện Đặt hàng/thanh toán thành công:



Hình 3.31: Giao diện đặt hàng/thanh toán thành công

3.4.8. Thống kê doanh thu

- Giao diện các thống kê và biểu đồ doanh thu:



Hình 3.32: Giao diện thống kê doanh thu

3.5. Kiểm thử và đánh giá hệ thống

3.5.1. Mục tiêu kiểm thử

Mục tiêu của quá trình kiểm thử là đảm bảo hệ thống đáp ứng đầy đủ các yêu cầu chức năng, phi chức năng, đồng thời phát hiện và khắc phục các lỗi tiềm ẩn trước khi triển khai thực tế. Việc kiểm thử giúp đánh giá mức độ ổn định, hiệu năng và khả năng bảo mật của hệ thống.

3.5.2. Quy trình và phương pháp kiểm thử

Quy trình kiểm thử được thực hiện theo các bước sau:

Kiểm thử chức năng (Functional Testing): Thực hiện kiểm tra các chức năng chính như đăng ký, đăng nhập, đặt hàng, thanh toán, quản lý sản phẩm, quản lý đơn hàng, v.v. Đảm bảo các chức năng hoạt động đúng theo yêu cầu đề ra.

Kiểm thử giao diện người dùng (UI Testing): Đánh giá tính thân thiện, trực quan của giao diện, khả năng hiển thị trên nhiều thiết bị và trình duyệt khác nhau.

Kiểm thử hiệu năng (Performance Testing): Đo lường tốc độ phản hồi của hệ thống, khả năng xử lý đồng thời nhiều yêu cầu từ người dùng.

Kiểm thử bảo mật (Security Testing): Kiểm tra các lỗ hổng bảo mật, xác thực và phân quyền người dùng.

3.5.3. Kết quả kiểm thử

Sau quá trình kiểm thử, hệ thống đạt được các kết quả như sau:

- Các chức năng chính đều hoạt động ổn định, đáp ứng đúng yêu cầu nghiệp vụ.
- Giao diện người dùng hiển thị tốt trên cả máy tính và thiết bị di động, đảm bảo trải nghiệm người dùng.
- Hệ thống có tốc độ phản hồi nhanh, xử lý tốt với số lượng người dùng truy cập đồng thời ở mức trung bình.
- Không phát hiện lỗ hổng bảo mật nghiêm trọng trong quá trình kiểm thử.

3.5.4. Đánh giá tổng hệ thống

Qua quá trình kiểm thử, có thể đánh giá hệ thống đã đáp ứng đầy đủ các yêu cầu đặt ra về chức năng, hiệu năng và bảo mật. Tuy nhiên, để đảm bảo hệ thống vận hành ổn định trong môi trường thực tế với số lượng người dùng lớn, cần tiếp tục theo dõi, kiểm thử và tối ưu hóa khi triển khai thực tế.

3.6. Tổng kết chương.

Trong chương này, hệ thống đã được trình bày chi tiết về kiến trúc, các chức năng chính, quy trình xây dựng cũng như các công nghệ được sử dụng. Bên cạnh đó, quá trình kiểm thử và đánh giá hệ thống cũng đã được thực hiện nhằm đảm bảo hệ thống đáp ứng đầy đủ các yêu cầu đề ra về chức năng, hiệu năng và bảo mật. Kết quả kiểm thử cho thấy

hệ thống hoạt động ổn định, giao diện thân thiện và có khả năng mở rộng trong tương lai. Tuy nhiên, để hệ thống vận hành hiệu quả hơn khi triển khai thực tế, cần tiếp tục theo dõi, kiểm thử và tối ưu hóa trong các giai đoạn tiếp theo.

KẾT LUẬN

Kết luận:

Dự án đã hoàn thiện một hệ thống thương mại điện tử với các chức năng chính như quản lý sản phẩm, đơn hàng, người dùng và tích hợp thanh toán qua App Ngân Hàng hoặc thanh toán khi nhận hàng (COD). Hệ thống đáp ứng tốt nhu cầu mua sắm trực tuyến, mang lại trải nghiệm thuận tiện cho người dùng và hỗ trợ quản lý hiệu quả cho quản trị viên.

Hạn chế:

Mặc dù đã đáp ứng được các chức năng cơ bản, dự án vẫn còn một số hạn chế như: chưa tối ưu hiệu năng cho lượng truy cập lớn, bảo mật còn đơn giản, thiếu các cơ chế kiểm thử tự động để đảm bảo chất lượng phần mềm và thanh toán qua VNpay.

Hướng phát triển:

Sắp tới hoặc trong tương lai, dự án có thể phát triển thêm các tính năng nâng cao như: tích hợp nhiều cổng thanh toán khác nhau, xây dựng hệ thống gợi ý sản phẩm thông minh, phát triển ứng dụng di động, bổ sung chức năng chat hỗ trợ khách hàng, tăng cường bảo mật, tối ưu hiệu năng và triển khai các công cụ kiểm thử tự động cũng như quy trình CI/CD để nâng cao chất lượng và khả năng mở rộng của hệ thống.

TÀI LIỆU THAM KHẢO

1. Nguyễn Thanh Bình, “Giáo trình Phát triển ứng dụng Web với Node.js và React,” NXB Bách Khoa Hà Nội, 2021.
<https://www.nxbkhn.vn/>
2. Nguyễn Văn Hiếu, “Lập trình Web với JavaScript, HTML5 và CSS3,” NXB Thông tin và Truyền thông, 2020.
<https://www.nxbtttt.vn/>
3. Nguyễn Văn Tuấn, “An toàn và bảo mật hệ thống thông tin,” NXB Khoa học và Kỹ thuật, 2019.
<https://www.nxbkhkt.com.vn/>
4. MDN Web Docs, “Express.js Guide,” Mozilla, 2025.
https://developer.mozilla.org/en-US/docs/Learn/Server-side/Express_Nodejs
5. ReactJS, “Getting Started – React,” Meta, 2025.
<https://react.dev/learn>
6. OWASP Foundation, “OWASP Top Ten Web Application Security Risks,” 2025.
<https://owasp.org/www-project-top-ten/>