

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN

BÙI HẢI HIỆP

XÂY DỰNG TRANG WEB BÁN ĐỒ NỘI THẤT CHO CỬA
HÀNG FURNI VỚI ANGULAR VÀ NODE.JS

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

HƯNG YÊN - 2024

BÙI HẢI HIỆP

XÂY DỰNG TRANG WEB BÁN ĐỒ NỘI THẤT
CHO CỬA HÀNG FURNI VỚI ANGULAR NODE.JS

2024

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN

BÙI HẢI HIỆP

XÂY DỰNG TRANG WEB BÁN ĐỒ NỘI THẤT CHO CỬA
HÀNG FURNI VỚI ANGULAR VÀ NODE.JS

NGÀNH: CÔNG NGHỆ THÔNG TIN
CHUYÊN NGÀNH: KỸ THUẬT PHẦN MỀM

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

NGƯỜI HƯỚNG DẪN

(Ký và ghi rõ họ tên)

CHỦ TỊCH HỘI ĐỒNG

(Ký và ghi rõ họ tên)

HƯNG YÊN - 2024

NHẬN XÉT

Nhận xét của giảng viên hướng dẫn:

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

GIẢNG VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

Nhận xét của giảng viên phản biện 1:

.....

.....

.....

.....

.....

.....

GIẢNG VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

Nhận xét của giảng viên phản biện 2:

.....

.....

.....

.....

.....

.....

GIẢNG VIÊN HƯỚNG DẪN

(Ký và ghi rõ họ tên)

LỜI CAM ĐOAN

Tôi tên là Bùi Hải Hiệp xin cam đoan đồ án tốt nghiệp “Xây dựng trang web bán đồ nội thất cho cửa hàng Furni với angular và node.js” là công trình nghiên cứu của bản thân. Những phần sử dụng tài liệu tham khảo trong đồ án đã được nêu rõ trong phần tài liệu tham khảo. Các số liệu, kết quả trình bày trong đồ án là hoàn toàn trung thực, nếu sai em xin chịu hoàn toàn trách nhiệm và chịu mọi kỷ luật của bộ môn và nhà trường đề ra.

Hưng Yên, ngày ... tháng ... năm.....

Sinh viên

MỤC LỤC

DANH MỤC CÁC THUẬT NGỮ	7
DANH MỤC HÌNH VẼ, ĐỒ THỊ.....	8
DANH MỤC CÁC BẢNG BIỂU	10
CHƯƠNG 1: MỞ ĐẦU	12
1.1. Lý do chọn đề án.....	12
1.2. Mục tiêu của đề án	13
1.2.1. Mục tiêu tổng quát.....	13
1.2.2. Mục tiêu cụ thể.....	13
1.3. Giới hạn và phạm vi của đề án	13
1.3.1. Đối tượng nghiên cứu.....	13
1.3.2. Phạm vi nghiên cứu.....	14
1.4. Nội dung thực hiện.....	14
1.5. Phương pháp tiếp cận	15
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT.....	17
2.1. Quy trình phát triển phần mềm.....	17
2.2. Phương pháp phân tích thiết kế hướng đối tượng.....	19
2.3. Công nghệ áp dụng.....	20
CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG.....	26
3.1. Phân tích yêu cầu bài toán	26
3.1.1. Khảo sát bài toán	26
3.1.2. Phân tích yêu cầu bài toán	26
3.2. Đặc tả yêu cầu phần mềm.....	28
3.2.1. Các yêu cầu chức năng.....	28

3.2.2 Biểu đồ lớp thực thể	34
3.2.3. Các yêu cầu phi chức năng.....	34
3.3. Thiết kế hệ thống.....	35
3.3.1. Thiết kế kiến trúc.....	35
3.3.2. Thiết kế cơ sở dữ liệu	36
3.3.3. Thiết kế lớp đối tượng	40
3.3.4. Thiết kế giao diện.....	50
CHƯƠNG 4: TRIỂN KHAI WEBSITE.....	55
4.1. Triển khai các chức năng cho ứng dụng.....	55
4.2. Xây dựng các chức năng	60
4.2.1. Các chức năng phân hệ người dùng	60
4.2.2. Các chức năng phân hệ quản trị.....	62
4.3. Kiểm thử và triển khai ứng dụng	62
4.3.1. Kiểm thử chức năng đăng nhập.....	62
4.3.2. Kiểm thử chức năng quản lý sản phẩm	64
KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	67
TÀI LIỆU THAM KHẢO	68

DANH MỤC CÁC THUẬT NGỮ

Từ viết tắt	Từ đầy đủ	Giải thích
API	Application Programming Interface	Giao diện lập trình ứng dụng
CSDL	Cơ sở dữ liệu	Cơ sở dữ liệu
HTML	Hypertext Markup Language	Ngôn ngữ Đánh dấu siêu văn bản
HTTP	HyperText Transfer Protocol	Giao thức truyền tải siêu văn bản
HTTPS	Hypertext Transfer Protocol Secure	Một phần mở rộng của <u>Hypertext Transfer Protocol</u> (HTTP).
JSON	JavaScript Object Notation	Một kiểu định dạng dữ liệu tuân theo một quy luật nhất định
UML	Unifited Modeling Language	Ngôn ngữ mô hình hóa mục đích chung
EJS	Embedded JavaScript	Công cụ template cho phép tạo ra HTML động bằng cách nhúng JavaScript vào bên trong các file <code>.ejs`</code>

DANH MỤC HÌNH VẼ, ĐỒ THỊ

Hình 3. 1: Biểu đồ use case tổng quát phân hệ admin.....	29
Hình 3. 2: Hình usecase quản lý loại sản phẩm.....	30
Hình 3. 3: Hình usecase quản lý sản phẩm	30
Hình 3. 4: Hình usecase quản lý đơn chưa xử lý.....	30
Hình 3. 5: Hình usecase quản lý đơn đang giao	31
Hình 3. 6: Hình usecase quản lý giao thành công	31
Hình 3. 7: Hình usecase quản lý đơn hàng đã hủy	31
Hình 3. 8: Hình usecase tổng quát trang người dùng	33
Hình 3. 9: Hình usecase phân rã xem sản phẩm.....	33
Hình 3. 10: Hình usecase phân rã đơn hàng.....	34
Hình 3. 11: Biểu đồ lớp thực thể	34
Hình 3. 12: Biểu đồ VOPC của use case Đăng Nhập.....	40
Hình 3. 13: Biểu đồ VOPC của thêm sản phẩm.....	41
Hình 3. 14: Biểu đồ VOPC của sửa sản phẩm	41
Hình 3. 15: Biểu đồ VOPC của xóa sản phẩm	42
Hình 3. 16: Biểu đồ VOPC của use thêm loại sản phẩm.....	42
Hình 3. 17: Biểu đồ VOPC của use case sửa loại sản phẩm.....	43
Hình 3. 18: Biểu đồ VOPC của use case xóa loại sản phẩm.....	43
Hình 3. 19: Biểu đồ VOPC của use case đặt hàng	44
Hình 3. 20: Biểu đồ tuần tự của use case sửa sản phẩm	45
Hình 3. 21: Biểu đồ tuần tự của xóa sản phẩm.....	45
Hình 3. 22: Biểu đồ tuần tự của use thêm loại sản phẩm.....	46
Hình 3. 23: Biểu đồ tuần tự của use case sửa loại sản phẩm	46

Hình 3. 24: Biểu đồ tuần tự của use case xóa loại sản phẩm	47
Hình 3. 25: Biểu đồ lớp thiết kế của thêm sản phẩm.....	47
Hình 3. 26: Biểu đồ lớp thiết kế của use case sửa sản phẩm	48
Hình 3. 27: Biểu đồ lớp thiết kế của xóa sản phẩm.....	48
Hình 3. 28: Biểu đồ lớp thiết kế của use thêm loại sản phẩm.....	49
Hình 3. 29: Biểu đồ lớp thiết kế của use case sửa loại sản phẩm.....	49
Hình 3. 30: Biểu đồ lớp thiết kế của use case xóa loại sản phẩm	50
Hình 3. 31: Bảng quản lý danh mục	50
Hình 3. 32: Quản lý sản phẩm.....	51
Hình 3. 33: Xem chi tiết hóa đơn	51
Hình 3. 34: Quản lý thống kê	52
Hình 3. 35: Trang chủ	52
Hình 3. 36: Trang sản phẩm	53
Hình 3. 37: Trang chi tiết sản phẩm.....	53
Hình 3. 38: Trang thanh toán.....	54

DANH MỤC CÁC BẢNG BIỂU

Bảng 3. 1: Yêu cầu khách hàng	26
Bảng 3. 2: Yêu cầu chức năng.....	26
Bảng 3. 3: Yêu cầu phi chức năng.....	27
Bảng 3. 4: Các yêu cầu chức năng.....	28
Bảng 3.5: Các yêu cầu chức năng.....	32
Bảng 3. 6: Cấu trúc bảng categories	36
Bảng 3. 7: Cấu trúc bảng detail_product.....	36
Bảng 3. 8: Cấu trúc bảng account.....	37
Bảng 3.9: Cấu trúc bảng favorite.....	37
Bảng 3.10: Cấu trúc bảng comment.....	37
Bảng 3. 11: Cấu trúc bảng blog	38
Bảng 3. 12: Cấu trúc bảng news	38
Bảng 3. 13: Cấu trúc bảng wishlist	38
Bảng 3. 14: Cấu trúc bảng slide.....	38
Bảng 3. 15: Cấu trúc bảng gift.....	39
Bảng 3. 16: Cấu trúc bảng rule.....	39
Bảng 3. 17: Cấu trúc bảng bill.....	39
Bảng 3. 18: Cấu trúc bảng bill_detail	39
Bảng 3. 19: Cấu trúc bảng provider.....	40
Bảng 3. 20: Biểu đồ tuần tự của thêm sản phẩm	44
Bảng 4. 1: API phân hệ quản trị	55
Bảng 4. 2: API phân hệ người dùng	58

Bảng 4. 3: Chức năng nghiệp vụ phân hệ người dùng.....	60
Bảng 4. 4: Chức năng phân hệ quản trị.....	62
Bảng 4. 5: Chức năng phân hệ quản trị.....	63
Bảng 4. 6: Kiểm thử chức năng thêm sản phẩm.....	64
Bảng 4. 7: Kiểm thử chức năng cập nhật sản phẩm	65
Bảng 4. 8: Kiểm thử chức năng xóa sản phẩm.....	66
Bảng 4. 9: Kiểm thử chức năng tìm kiếm	66

CHƯƠNG 1: MỞ ĐẦU

1.1. Lý do chọn đề án

Trong thời đại công nghệ số hiện nay, việc phát triển trang web bán hàng trực tuyến là một điều cần thiết để nâng cao hiệu quả kinh doanh của các cửa hàng bán lẻ. Tính cấp thiết của đề tài này được phản ánh qua sự phát triển mạnh mẽ của thương mại điện tử trong những năm gần đây.

Một trong những lợi ích đáng kể của việc phát triển trang web bán hàng trực tuyến là tạo ra một kênh bán hàng hiệu quả và tiết kiệm chi phí. Thay vì phải thuê một cửa hàng đắt đỏ và mất nhiều thời gian để trang trí và quản lý, các doanh nghiệp bán lẻ có thể sử dụng một trang web bán hàng trực tuyến để tiếp cận một đối tượng khách hàng rộng lớn hơn, bao gồm cả khách hàng ở những nơi khác xa.

Ngoài ra, một trang web bán hàng trực tuyến còn giúp nâng cao tính năng đa dạng của sản phẩm. Khách hàng có thể dễ dàng tìm kiếm và so sánh giá cả của các sản phẩm khác nhau trên một trang web bán hàng trực tuyến. Hơn nữa, các doanh nghiệp bán lẻ có thể quản lý số lượng lớn sản phẩm trên trang web của họ, tạo ra nhiều loại sản phẩm khác nhau và cập nhật chúng một cách dễ dàng, do đó giúp tăng doanh số và lợi nhuận của cửa hàng.

Tuy nhiên, việc phát triển một trang web bán hàng trực tuyến cũng đặt ra một số thách thức và khó khăn. Ví dụ, cần phải đầu tư một số tiền lớn vào việc phát triển và quản lý trang web. Các doanh nghiệp bán lẻ cần phải đảm bảo rằng trang web của họ được thiết kế đẹp mắt, dễ sử dụng và đáp ứng tốt với các thiết bị di động. Ngoài ra, các doanh nghiệp bán lẻ cần phải cân nhắc và đảm bảo tính an toàn trong việc xử lý thông tin cá nhân của khách hàng trên trang web của họ.

Tóm lại, việc xây dựng một trang web bán hàng trực tuyến cho mỗi cửa hàng là rất cấp thiết và quan trọng trong việc nâng cao hiệu quả kinh doanh, đáp ứng nhu cầu của khách hàng và cạnh tranh trong thị trường hiện nay. Do đó, em quyết định xây dựng trang web bán đồ nội thất cho cửa hàng Furni, với hi vọng trang web sẽ giúp cửa hàng tiết kiệm chi phí, tăng khả năng tiếp cận đến khách hàng mới và cũng tăng cường quan hệ với khách hàng thông qua một trải nghiệm mua sắm thuận tiện, nhanh chóng và trung thực.

1.2. Mục tiêu của đề án

1.2.1. Mục tiêu tổng quát

Tạo ra một trang web bán hàng trực tuyến hoàn chỉnh, bao gồm các tính năng như tìm kiếm sản phẩm, tra cứu thông tin, đặt hàng và thanh toán, từ đó giúp tăng cường hiệu quả kinh doanh, cạnh tranh trong thị trường và đáp ứng nhu cầu mua sắm của khách hàng một cách thuận tiện và nhanh chóng.

1.2.2. Mục tiêu cụ thể

Thiết kế giao diện trang web dễ sử dụng, trực quan và tương thích trên nhiều thiết bị khác nhau, giúp khách hàng dễ dàng tìm kiếm và mua sản phẩm.

Cung cấp các tính năng tìm kiếm sản phẩm, lọc sản phẩm theo các tiêu chí khác nhau (giá, tính năng...), giúp khách hàng tìm kiếm sản phẩm nhanh chóng và chính xác.

Cho phép khách hàng xem chi tiết thông tin về sản phẩm, bao gồm hình ảnh, mô tả, đánh giá, giá cả và thông tin liên quan khác, giúp khách hàng đưa ra quyết định mua sắm chính xác.

Cung cấp tính năng đặt hàng và thanh toán trực tuyến, giúp khách hàng dễ dàng và nhanh chóng hoàn thành quá trình mua sắm.

Cung cấp tính năng quản lý đơn hàng, giúp cửa hàng thời trang dễ dàng quản lý, xử lý và giao hàng cho khách hàng.

Tạo ra một trải nghiệm mua sắm trực tuyến thuận tiện, nhanh chóng và trung thực, giúp tăng tính trung thực và lòng tin của khách hàng đối với cửa hàng.

Tăng khả năng tiếp cận đến khách hàng mới, giúp cửa hàng mở rộng thị trường và tăng doanh thu kinh doanh.

1.3. Giới hạn và phạm vi của đề án

1.3.1. Đối tượng nghiên cứu

Cửa hàng Furni và khách hàng có nhu cầu mua sắm các sản phẩm nội thất của cửa hàng này. Nghiên cứu sẽ tập trung vào việc xây dựng và phát triển trang web bán hàng trực tuyến để giải quyết các vấn đề bất cập, hạn chế của hệ thống bán hàng trực tiếp của cửa hàng, đồng thời cung cấp cho khách hàng một phương thức mua sắm tiện lợi, đáp ứng nhu cầu của họ và tăng cường hiệu quả kinh doanh của cửa hàng.

1.3.2. Phạm vi nghiên cứu

Phạm vi không gian của đề tài là tại cửa hàng bán đồ nội thất, nơi mà nghiên cứu sẽ tập trung vào việc xây dựng và phát triển trang web bán hàng trực tuyến.

Phạm vi thời gian của đề tài bao gồm số liệu thứ cấp thu thập trong khoảng thời gian từ 01/03/2024 đến 01/05/2024, bao gồm dữ liệu về lượng sản phẩm bán ra, số lượng khách hàng, doanh thu và các thông tin khác liên quan đến hoạt động kinh doanh của cửa hàng. Số liệu sơ cấp cũng sẽ được thu thập từ khảo sát, phỏng vấn khách hàng để hiểu rõ hơn về nhu cầu của họ và các vấn đề mà họ gặp phải trong quá trình mua sắm.

Ý nghĩa khoa học của đề tài là đóng góp vào việc nghiên cứu, phát triển các phương thức bán hàng trực tuyến hiệu quả, giúp cải thiện hoạt động kinh doanh của cửa hàng và đáp ứng nhu cầu của khách hàng. Đồng thời, đề tài cũng mang tính thực tiễn cao khi giúp cửa hàng cải thiện phương thức kinh doanh, thu hút thêm khách hàng và tăng doanh số bán hàng.

1.4. Nội dung thực hiện

Trong phạm vi của đề án này em sẽ trình bày nội dung thực hiện theo 5 chương cụ thể như sau:

- Chương 1 là trình bày tổng quan về đề tài. Chương này nêu ra lý do chọn đề tài, mục tiêu của đề tài, giới hạn và phạm vi của đề tài, nội dung thực hiện, phương pháp tiếp cận.
- Chương 2 là trình bày về cơ sở lý thuyết. Chương này trình bày những kiến thức cơ sở về quy trình phát triển phần mềm, phương pháp phân tích thiết kế hướng đối tượng, hệ quản trị cơ sở dữ liệu, thiết kế giao diện website, lập trình phía front-end, lập trình phía back-end.
- Chương 3 là phần trình bày những kiến thức cơ sở về phân tích và thiết kế hệ thống. Chương này sẽ đặc tả các yêu cầu của phần mềm và thiết kế hệ thống. Tại phần đặc tả yêu cầu phần mềm sẽ đặc tả các yêu cầu chức năng, xác định các thực thể và các yêu cầu phi chức năng của website. Tại phần thiết kế hệ thống sẽ thiết kế lớp đối tượng, thiết kế cơ sở dữ liệu và thiết kế giao diện cho website.

- Chương 4 là phần triển khai website. Chương này sẽ trình bày các kết quả đã đạt được khi triển khai các chức năng cho phân hệ người dùng, chức năng cho phân hệ quản trị nội dung và hướng dẫn cho khách hàng cài đặt, sử dụng.
- Chương 5 là kết luận và hướng phát triển. Chương này sẽ tổng kết lại các kết quả đã đạt được đồng thời chỉ ra những hạn chế và xác định hướng phát triển tiếp theo của website.

1.5. Phương pháp tiếp cận

a) Đối với lý thuyết

- Tìm hiểu về lập trình website theo mô hình API: Nghiên cứu về các khái niệm cơ bản của API, cách tạo và quản lý API, và cách sử dụng API để giao tiếp giữa các ứng dụng khác nhau.
- Tìm hiểu về Angular và cách xây dựng giao diện bằng Angular: Nắm vững các khái niệm cơ bản về Angular, cấu trúc của một ứng dụng Angular, và cách tạo các thành phần giao diện người dùng (UI) bằng Angular.
- Tìm hiểu kỹ thuật lập trình, cách thức hoạt động và phương pháp hướng đối tượng trong JavaScript và TypeScript: Hiểu rõ về cú pháp và các tính năng của JavaScript và TypeScript, cách thức hoạt động của chúng, và áp dụng các nguyên tắc lập trình hướng đối tượng trong quá trình phát triển.
- Tìm hiểu cách thức hoạt động của mô hình Client-Server: Nghiên cứu về kiến trúc Client-Server, cách thức giao tiếp giữa Client và Server, và các giao thức truyền thông thường dùng.
- Tìm hiểu cách lưu trữ và quản lý dữ liệu trong hệ quản trị cơ sở dữ liệu MySQL: Nắm vững các khái niệm cơ bản về MySQL, cách tạo và quản lý cơ sở dữ liệu, cách viết các truy vấn SQL để thao tác với dữ liệu.

b) Đối với lập trình

- Thiết kế theo mô hình API sử dụng Node.js và Express.js: Sử dụng Node.js và Express.js để xây dựng các API RESTful, phục vụ cho các yêu cầu của ứng dụng.

- Ngôn ngữ lập trình JavaScript và TypeScript: Sử dụng JavaScript và TypeScript để phát triển các thành phần của ứng dụng, từ phía máy chủ đến phía khách hàng.
- Hệ quản trị cơ sở dữ liệu MySQL: Sử dụng MySQL để lưu trữ và quản lý dữ liệu của ứng dụng.
- Xây dựng trang web đúng theo mô tả của hệ thống dựa trên các yêu cầu thực tế: Dựa vào các yêu cầu của hệ thống và mô tả chi tiết, phát triển một trang web bán hàng trực tuyến cho cửa hàng nội thất Furni, đảm bảo tính tiện dụng, hiệu quả và đáp ứng nhu cầu của khách hàng.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Quy trình phát triển phần mềm

Quy trình phát triển phần mềm là một chuỗi các hoạt động của nhà phân tích (Analyst), nhà thiết kế (Designer), người phát triển (Developer) và người dùng (User) để phát triển và thực hiện một hệ thống thông tin. Những hoạt động này được thực hiện trong nhiều giai đoạn khác nhau.

Các giai đoạn của chu trình phát triển phần mềm:

- Phân tích yêu cầu (Analysis)

Sau khi đã xem xét về tính khả thi của hệ thống cũng như tạo lập một bức tranh sơ bộ của dự án, chúng ta bước sang giai đoạn thường được coi là quan trọng nhất trong các công việc lập trình: hiểu hệ thống cần xây dựng. Người thực hiện công việc này là nhà phân tích.

Quá trình phân tích nhìn chung là hệ quả của việc trả lời câu hỏi "Hệ thống cần phải làm gì?". Quá trình phân tích bao gồm việc nghiên cứu chi tiết hệ thống doanh nghiệp hiện thời, tìm cho ra nguyên lý hoạt động của nó và những vị trí có thể được nâng cao, cải thiện. Bên cạnh đó là việc nghiên cứu xem xét các chức năng mà hệ thống cần cung cấp và các mối quan hệ của chúng, bên trong cũng như với phía ngoài hệ thống. Trong toàn bộ giai đoạn này, nhà phân tích và người dùng cần cộng tác mật thiết với nhau để xác định các yêu cầu đối với hệ thống, tức là các tính năng mới cần phải được đưa vào hệ thống.

Những mục tiêu cụ thể của giai đoạn phân tích là:

- Xác định hệ thống cần phải làm gì.
- Nghiên cứu thấu đáo tất cả các chức năng cần cung cấp và những yếu tố liên quan
- Xây dựng một mô hình nêu bật bản chất vấn đề từ một hướng nhìn có thực (trong đời sống thực).
- Trao định nghĩa vấn đề cho chuyên gia lĩnh vực để nhận sự đánh giá, góp ý.
- Kết quả của giai đoạn phân tích là bản Đặc Tả Yêu Cầu (Requirements Specifications).

- Thiết kế hệ thống (Design of the System)

Sau giai đoạn phân tích, khi các yêu cầu cụ thể đối với hệ thống đã được xác định, giai đoạn tiếp theo là thiết kế cho các yêu cầu mới. Công tác thiết kế xoay quanh câu hỏi chính: Hệ thống làm cách nào để thỏa mãn các yêu cầu đã được nêu trong Đặc Tả Yêu Cầu?

Một số các công việc thường được thực hiện trong giai đoạn thiết kế:

- Nhận biết form nhập liệu tùy theo các thành phần dữ liệu cần nhập.
- Nhận biết báo cáo và những output mà hệ thống mới phải sản sinh.
- Thiết kế forms (vẽ trên giấy hay máy tính, sử dụng công cụ thiết kế).
- Nhận biết các thành phần dữ liệu và bảng để tạo cơ sở dữ liệu
- Ước tính các thủ tục giải thích quá trình xử lý từ input đến output.

Kết quả giai đoạn thiết kế là Đặc tả thiết kế (Design Specifications). Bản Đặc Tả Thiết kế chi tiết sẽ được chuyển sang cho các lập trình viên để thực hiện giai đoạn xây dựng phần mềm.

- Xây dựng phần mềm (Software Construction)

Đây là giai đoạn viết lệnh (code) thực sự, tạo hệ thống. Từng người viết code thực hiện những yêu cầu đã được nhà thiết kế định sẵn. Cũng chính người viết code chịu trách nhiệm viết tài liệu liên quan đến chương trình, giải thích thủ tục (procedure) mà anh ta tạo nên được viết như thế nào và lý do cho việc này.

Để đảm bảo chương trình được viết nên phải thỏa mãn mọi yêu cầu có ghi trước trong bản Đặc Tả Thiết Kế Chi Tiết, người viết code cũng đồng thời phải tiến hành thử nghiệm phần chương trình của mình. Phần thử nghiệm trong giai đoạn này có thể được chia thành hai bước chính:

- Thử nghiệm đơn vị:

Người viết code chạy thử các phần chương trình của mình với dữ liệu giả (test/dummy data). Việc này được thực hiện theo một kế hoạch thử, cũng do chính người viết code soạn ra. Mục đích chính trong giai đoạn thử này là xem chương trình có cho ra những kết quả mong đợi. Giai đoạn thử nghiệm đơn vị nhiều khi được gọi là "Thử hộp trắng" (White Box Testing)

- Thử nghiệm đơn vị độc lập:

Công việc này do một thành viên khác trong nhóm đảm trách. Cần chọn người không có liên quan trực tiếp đến việc viết code của đơn vị chương trình cần thử nghiệm để đảm bảo tính “độc lập”. Công việc thử đợt này cũng được thực hiện dựa trên kế hoạch thử do người viết code soạn lên.

- Thử nghiệm hệ thống (System Testing)

Sau khi các thủ tục đã được thử nghiệm riêng, cần phải thử nghiệm toàn bộ hệ thống. Mọi thủ tục được tích hợp và chạy thử, kiểm tra xem mọi chi tiết ghi trong Đặc Tả Yêu Cầu và những mong chờ của người dùng có được thoả mãn. Dữ liệu thử cần được chọn lọc đặc biệt, kết quả cần được phân tích để phát hiện mọi lệch lạc so với mong chờ.

- Thực hiện, triển khai (System Implementation)

Trong giai đoạn này, hệ thống vừa phát triển sẽ được triển khai sao cho phía người dùng. Trước khi để người dùng thật sự bắt tay vào sử dụng hệ thống, nhóm các nhà phát triển cần tạo các file dữ liệu cần thiết cũng như huấn luyện cho người dùng, để đảm bảo hệ thống được sử dụng hữu hiệu nhất.

- Bảo trì, nâng cấp (System Maintenance)

Tùy theo các biến đổi trong môi trường sử dụng, hệ thống có thể trở nên lỗi thời hay cần phải được sửa đổi nâng cấp để sử dụng có hiệu quả. Hoạt động bảo trì hệ thống có thể rất khác biệt tùy theo mức độ sửa đổi và nâng cấp cần thiết.

2.2. Phương pháp phân tích thiết kế hướng đối tượng

Trong kỹ nghệ phần mềm để sản xuất được một sản phẩm phần mềm người ta chia quá trình phát triển sản phẩm ra nhiều giai đoạn như thu thập và phân tích yêu cầu, phân tích và thiết kế hệ thống, phát triển (coding), kiểm thử, triển khai và bảo trì. Trong đó, giai đoạn phân tích, thiết kế bao giờ cũng là giai đoạn khó khăn và phức tạp nhất. Giai đoạn này giúp chúng ta hiểu rõ yêu cầu đặt ra, xác định giải pháp, mô tả chi tiết giải pháp. Nó trả lời 2 câu hỏi What (phần mềm này làm cái gì?) và How (làm nó như thế nào?).

Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng

ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD).

2.3. Công nghệ áp dụng

❖ Angular:

Angular là một javascript framework do google phát triển để xây dựng các Single Page Application (SPA) bằng JavaScript, HTML và TypeScript. Angular cung cấp các tính năng tích hợp cho animation, http service và có các tính năng như auto-complete, navigation, toolbar, menus, ... Code được viết bằng TypeScript, biên dịch thành JavaScript và hiển thị tương tự trong trình duyệt.

Kiến thức cơ bản cần thiết để học Angular:

- HTML
- CSS
- JavaScript
- TypeScript
- Document Object Model (DOM)

Các phiên bản của Angular:

❖ AngularJS:

- Phiên bản đầu tiên của Angular, được bắt đầu từ năm 2009 và ra mắt vào ngày 20/10/2010 bởi lập trình viên Misko Hevery tại Google như là một dự án “viết cho vui”.
- AngularJS được viết theo mô hình MVC (Model-View-Controller):
 - Model: Thành phần trung tâm, quản lý dữ liệu và hành vi của ứng dụng.
 - View: Được tạo ra dựa trên thông tin của Model.
 - Controller: Đóng vai trò trung gian giữa Model và View, xử lý logic.

❖ Angular 2:

- Ra đời vào tháng 3 năm 2015, thay thế AngularJS với các khái niệm mới nhằm đơn giản hóa và tối ưu quá trình phát triển.
- Thay đổi Controllers và \$scope bằng components và directives.

- Components = directives + template, tạo nên view của ứng dụng và xử lý logic trên view.
- Hoàn toàn được viết bằng TypeScript.
- Nhanh hơn AngularJS, hỗ trợ đa nền tảng và đa trình duyệt, cấu trúc code đơn giản và dễ sử dụng hơn.

❖ Angular 4:

- Ra mắt vào tháng 3/2017, là một phiên bản nâng cấp từ Angular 2 với việc giảm thiểu code, giảm kích thước tệp đóng gói xuống 60%, đẩy nhanh quá trình phát triển ứng dụng.

❖ Angular 5:

- Phát hành vào ngày 1 tháng 11 năm 2017, với mục tiêu thay đổi về tốc độ và kích thước, nhanh hơn và nhỏ hơn Angular 4.
- Tính năng mới:
 - Sử dụng HTTPClient thay vì HTTP, nhanh, an toàn và hiệu quả hơn.
 - RxJs 5.5 mặc định.
 - Multiple export aliases: Một component có thể được xuất bằng nhiều bí danh để giảm bớt quá trình di chuyển.
 - Internationalized Pipes for Number, Date, and Currency: Các pipe mới để tiêu chuẩn hóa tốt hơn.
 - Tối ưu hóa build production với build optimizer, cải thiện tốc độ biên dịch với TypeScript transforms và sử dụng lệnh “ng serve –aot”.

❖ Angular 6:

- Cập nhật CLI với các lệnh mới như ng-update và ng-add.
- Angular Element: Cho phép triển khai các component của Angular dưới dạng component web.
- Multiple Validators cho phép nhiều Validators trên form builder.
- Tree-shakeable providers loại bỏ mã code chết.
- Sử dụng RxJS 6 với syntax thay đổi.

❖ Angular 7:

- Phát hành vào ngày 18 tháng 10 năm 2018.
- Tính năng mới:
 - ScrollingModule: Để scroll load dữ liệu.
 - Drag and Drop: Thêm tính năng kéo và thả.
 - Cập nhật RxJS 6.3.

❖ Angular 8:

- Ra mắt ngày 28 tháng 5 năm 2019 với cải tiến CLI workflow, Dynamic imports for lazy routes, v.v.

❖ Angular 9:

- Ra mắt vào ngày 6 tháng 2 năm 2020, di chuyển tất cả các ứng dụng để sử dụng trình biên dịch Ivy và thời gian chạy theo mặc định.
- Cập nhật để hoạt động với TypeScript 3.6 và 3.7.

❖ Node.js:

Node.js là một môi trường thời gian chạy nguồn mở được xây dựng trên thời gian chạy JavaScript, chủ yếu sử dụng động cơ V8 của Google Chrome. Nó cho phép thực hiện JavaScript ở phía máy chủ, trao quyền cho việc phát triển các ứng dụng web có khả năng mở rộng và hiệu suất cao. Node.js sử dụng mô hình I/O không đồng bộ để xử lý đồng thời nhiều kết nối, tăng hiệu suất ứng dụng.

Node.js tích hợp với NPM (Trình quản lý gói nút), một công cụ quản lý gói nguồn mở giúp đơn giản hóa việc cài đặt và quản lý các thư viện và công cụ phát triển. Với một cộng đồng lớn và sôi động, Node.js đã trở thành một trong những môi trường phát triển phía máy chủ phổ biến nhất.

❖ EJS:

EJS là viết tắt của các mẫu JavaScript nhúng. Đó là một ngôn ngữ khuôn mẫu được sử dụng để phát triển web phía máy chủ. EJS cho phép bạn viết mã HTML với các biểu thức JavaScript nhúng được đánh giá trên máy chủ trước khi HTML cuối cùng được gửi đến máy khách (trình duyệt web của người dùng).

Dưới đây là một số đặc điểm chính của EJS:

- Cú pháp đơn giản: EJS sử dụng cú pháp đơn giản và dễ học để nhúng mã JavaScript trong các mẫu HTML. Bạn có thể sử dụng các dấu phân cách như `<% %>` để viết mã JavaScript và `<%= %>` để xuất kết quả của biểu thức JavaScript trực tiếp vào HTML.

- Tách các mối quan tâm: EJS giúp tách logic trình bày (HTML) khỏi logic ứng dụng (JavaScript). Điều này làm cho mã của bạn có thể duy trì hơn và dễ hiểu hơn.

- Nội dung động: EJS cho phép bạn tạo nội dung động dựa trên dữ liệu được truyền từ máy chủ sang mẫu. Điều này làm cho các ứng dụng web của bạn tương tác hơn và thân thiện với người dùng.

- Các tính năng tạo khuôn mẫu: EJ cung cấp các tính năng khuôn mẫu khác nhau như vòng lặp, điều kiện và bao gồm, giúp việc viết các mẫu có thể tái sử dụng và có thể duy trì dễ dàng hơn.

❖ Kết hợp EJS và Node.js:

EJS (các mẫu JavaScript nhúng) kết hợp với Node.js là một cách hiệu quả để xây dựng một ứng dụng web là đầu ra. EJS cung cấp các phương thức đơn giản và hiệu quả để tạo nội dung HTML động dựa trên dữ liệu từ phụ trợ Node.js, trong khi Node.js xử lý logic và xử lý dữ liệu trên máy chủ.

Lợi thế:

Các mẫu đơn giản và hiệu quả: EJS cung cấp cú pháp dễ dàng để nhúng biểu thức JavaScript trong các mô hình HTML, giúp tạo nội dung động mà không cần mã phức tạp.

Phân chia rõ ràng: Phân biệt rõ ràng logic được trình bày (HTML và EJ) với logic ứng dụng (javascript và node.js), tăng cường bảo trì và dễ đọc mã.

Linh hoạt và có thể mở rộng: Node.js xử lý nhiều yêu cầu hiệu quả nhờ tính đồng nhất và bản chất không chặn, EJ tạo ra một nội dung động nhẹ nhàng, giúp áp dụng tải trọng cao và mở rộng hiệu quả

Phát triển toàn diện: EJS và Node.js cho phép phát triển toàn diện, xây dựng cả phía trước và phụ trợ của ứng dụng web với bộ công nghệ JavaScript.

❖ Kết hợp Angular và Node.js:

Sự kết hợp giữa Angular và Node.js thường được sử dụng để xây dựng các ứng

dụng web đầy đủ. Angular phụ trách phía máy khách, xây dựng các giao diện người dùng linh hoạt và tương tác, trong khi Node.js xử lý phía máy chủ và tương tác với cơ sở dữ liệu.

Các khung như Express.js thường được sử dụng cùng với Node.js để xây dựng phía máy chủ của ứng dụng. Sự kết hợp này mang lại lợi ích về hiệu suất, dễ bảo trì và phát triển nhanh chóng. Angular và Node.js tạo thành một cặp lý tưởng để xây dựng các ứng dụng web hiện đại và mạnh mẽ.

❖ MySQL:

MySQL là một hệ thống quản lý cơ sở dữ liệu quan hệ nguồn mở (RDBMS) được phát triển và hỗ trợ bởi Oracle Corporation. Được thiết kế cho sự ổn định, hiệu suất cao và dễ sử dụng, MySQL là một trong những RDBMS phổ biến nhất trên toàn thế giới. Nó thường được sử dụng để lưu trữ và quản lý dữ liệu trong các ứng dụng web, hệ thống doanh nghiệp và các dự án phần mềm khác nhau.

Dưới đây là một số tính năng chính của MySQL:

Nguồn mở: MySQL là một dự án nguồn mở, có nghĩa là mã nguồn của nó được công khai và miễn phí sử dụng. Điều này thúc đẩy sự phát triển và đóng góp từ một cộng đồng các nhà phát triển.

Ngôn ngữ SQL: MySQL sử dụng SQL (ngôn ngữ truy vấn có cấu trúc) làm ngôn ngữ truy vấn chính để tương tác với cơ sở dữ liệu. SQL là ngôn ngữ tiêu chuẩn để quản lý cơ sở dữ liệu quan hệ.

Hỗ trợ đa nền tảng: MySQL có thể chạy trên các hệ điều hành khác nhau, bao gồm Linux, Windows, MacOS, v.v. Điều này cung cấp sự linh hoạt để triển khai trên các môi trường khác nhau.

Khả năng mở rộng: MySQL hỗ trợ các công cụ lưu trữ khác nhau và cung cấp khả năng mở rộng, cho phép các ứng dụng mở rộng từ nhỏ đến lớn mà không làm giảm hiệu suất.

Hiệu suất cao: Cơ sở dữ liệu MySQL được thiết kế cho hiệu suất cao, sử dụng các cơ chế lưu trữ thông minh, lập chỉ mục hiệu quả và các tối ưu hóa khác.

Cộng đồng và hỗ trợ: MySQL có một cộng đồng người dùng lớn và tích cực cung cấp hỗ trợ thông qua các diễn đàn, tài liệu trực tuyến và các tài nguyên khác.

Bảo mật: MySQL cung cấp các tính năng bảo mật mạnh mẽ, bao gồm quản lý người dùng, kiểm soát truy cập, tùy chọn mã hóa và các biện pháp để bảo vệ dữ liệu nhạy cảm.

Sao lưu và phục hồi: MySQL hỗ trợ các công cụ sao lưu và phục hồi, đảm bảo bảo vệ dữ liệu chống lại mất mát.

MySQL là một giải pháp đáng tin cậy và được áp dụng rộng rãi để quản lý cơ sở dữ liệu, được sử dụng trong các bối cảnh và loại dự án phần mềm khác nhau trên toàn thế giới.

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

3.1. Phân tích yêu cầu bài toán

3.1.1. Khảo sát bài toán

Bảng 3. 1: Yêu cầu khách hàng

STT	Tên yêu cầu
1	Đăng nhập vào hệ thống quản trị
2	Hiển thị thông tin loại sản phẩm lên thanh menu của website
3	Hiển thị nội dung của website như sản phẩm mới nhất...
4	Hiển thị thông tin chi tiết về sản phẩm
5	Hiển thị thông tin kết quả tìm kiếm
6	Quản lý giỏ hàng khi khách hàng thêm sản phẩm vào giỏ hàng
7	Chức năng đặt hàng
8	Chắc năng tìm kiếm thông tin theo yêu cầu của khách hàng
9	Hiển thị thông tin chi tiết sản phẩm khi chọn xem một sản phẩm
10	Hiển thị trang thông tin giới thiệu về cửa hàng
11	Quản lý dữ liệu khách hàng

3.1.2. Phân tích yêu cầu bài toán

Bảng 3. 2: Yêu cầu chức năng

STT	Yêu cầu chức năng nghiệp vụ	Yêu cầu chức năng hệ thống
1	Hiển thị danh sách sản phẩm	Tạo và quản lý cơ sở dữ liệu sản phẩm
2	Tìm kiếm sản phẩm	Cung cấp chức năng tìm kiếm theo tên
3	Xem chi tiết sản phẩm	Hiển thị thông tin chi tiết về sản phẩm, bao gồm hình ảnh, mô tả, đánh giá và đánh giá của người dùng

STT	Yêu cầu chức năng nghiệp vụ	Yêu cầu chức năng hệ thống
4	Thêm sản phẩm vào giỏ hàng	Lưu trữ thông tin sản phẩm trong giỏ hàng của người dùng
5	Đặt hàng	Xử lý đơn đặt hàng, bao gồm thông tin giao hàng, phương thức thanh toán và xác nhận đơn hàng
6	Quản lý tài khoản	Cho phép người dùng đăng nhập, đăng ký, quản lý thông tin cá nhân và địa chỉ giao hàng
7	Thanh toán	Cung cấp các phương thức thanh toán an toàn và xử lý thanh toán thành công hoặc thất bại
8	Hỗ trợ chăm sóc khách hàng	Cung cấp kênh liên hệ trực tuyến để khách hàng có thể gửi câu hỏi, yêu cầu hỗ trợ và nhận được phản hồi từ nhân viên chăm sóc khách hàng

Bảng 3. 3: Yêu cầu phi chức năng

STT	Yêu cầu phi chức năng	Mô tả
1	Giao diện	Giao diện hệ thống phải dễ sử dụng, trực quan, thân thiện với mọi người dùng.
2	Tốc độ xử lý	Hệ thống phải xử lý nhanh chóng và chính xác.
3	Bảo mật dữ liệu	Tính bảo mật và độ an toàn cao.

3.2. Đặc tả yêu cầu phần mềm

3.2.1. Các yêu cầu chức năng

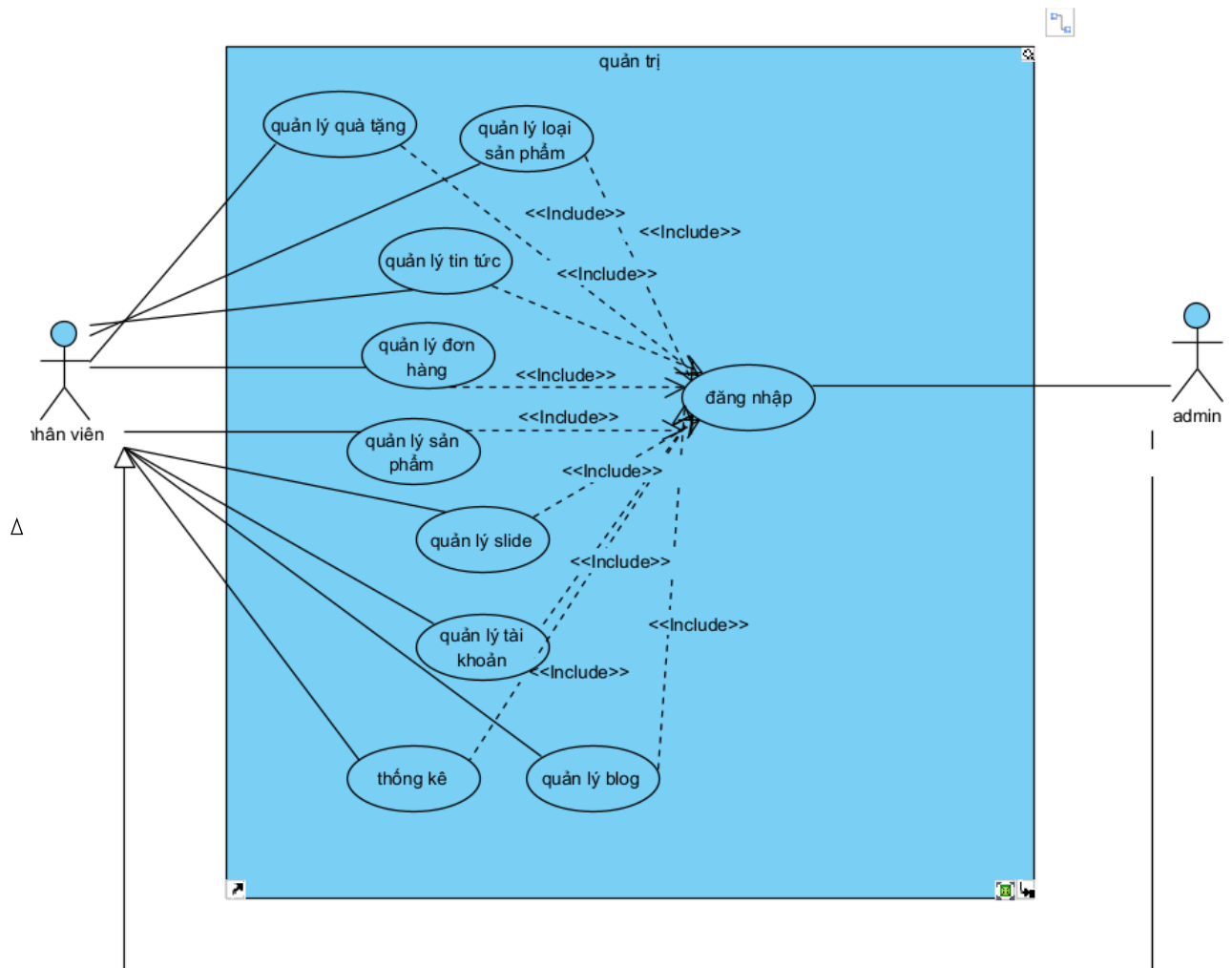
a) Chức năng của phân hệ quản trị nội dung

Bảng 3. 4: Các yêu cầu chức năng

STT	Chức năng	Mô tả
1.	Quản lý sản phẩm	Cho phép người quản trị có thể thêm, sửa, xóa, xem, tìm kiếm sản phẩm.
2.	Quản lý danh mục sản phẩm	Cho phép người quản trị có thể thêm, sửa, xóa, xem, tìm kiếm danh mục sản phẩm.
3.	Quản lý tài khoản khách hàng	Cho phép người quản trị quản lý thông tin tài khoản khách hàng trên trang web
4.	Quản lý đơn hàng	Cho phép người quản trị theo dõi số lượng đơn hàng, chi tiết đơn hàng.
5.	Quản lý đơn hàng chưa xử lý	Cho phép người quản trị xác nhận, xem chi tiết, hủy đơn hàng chưa xử lý.
6.	Quản lý đơn hàng đang giao	Cho phép người quản trị xem chi tiết, hủy đơn hàng đang giao.
7.	Quản lý đơn hàng thành công	Cho phép người quản trị xem chi tiết đơn hàng đã giao thành công.
8.	Quản lý đơn hàng hủy	Cho phép người quản trị xem chi tiết, đặt lại đơn hàng.
9.	Quản lý slide	Cho phép người quản trị chỉnh sửa lại ảnh hiển thị bên trang người dùng.
10.	Quản lý nhà cung cấp	Cho phép người quản trị quản lý thông tin của các nhà cung cấp
11.	Quản lý thông tin liên hệ	Cho phép người quản trị cập nhật thông tin cửa hàng.
12.	Quản lý tài khoản	Cho phép người quản trị quản lý thông tin tài khoản như đổi mật khẩu

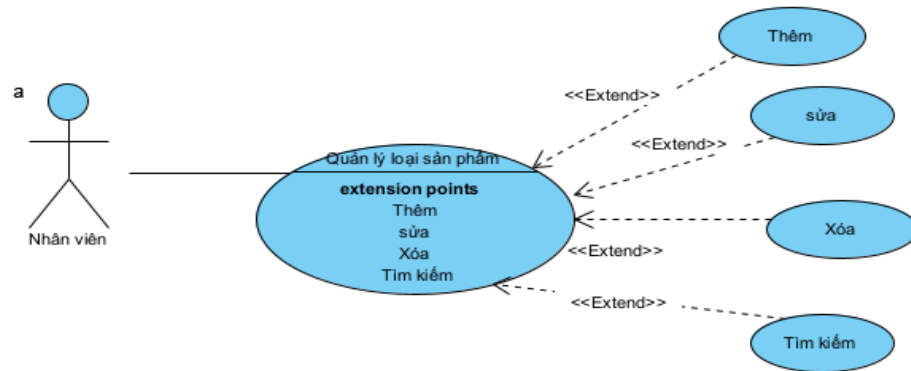
STT	Chức năng	Mô tả
13.	Thống kê	Cho phép người quản trị xem các báo cáo thống kê về đơn hàng, khách hàng, doanh thu, sản phẩm và các thông tin khác liên quan đến hoạt động của trang web.
14.	Đăng nhập	Cho phép người quản trị có thể đăng nhập vào tài khoản của mình trên website.
15.	Đăng xuất	Cho phép người quản trị có thể đăng xuất khỏi tài khoản của mình trên website.

❖ Biểu đồ Use case tổng quát của trang quản trị

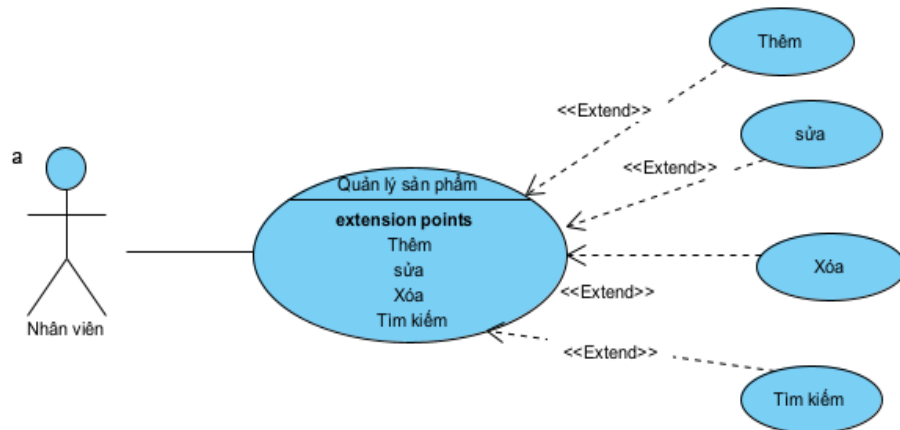


Hình 3. 1: Biểu đồ use case tổng quát phân hệ admin

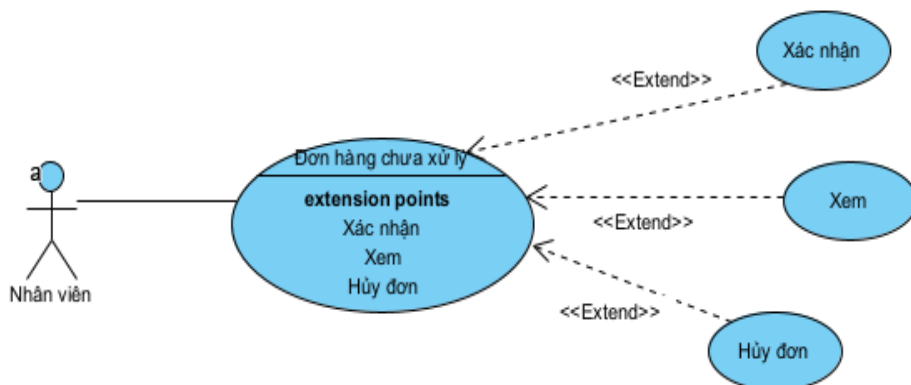
❖ Một số biểu đồ Use case phân rã của trang quản trị:



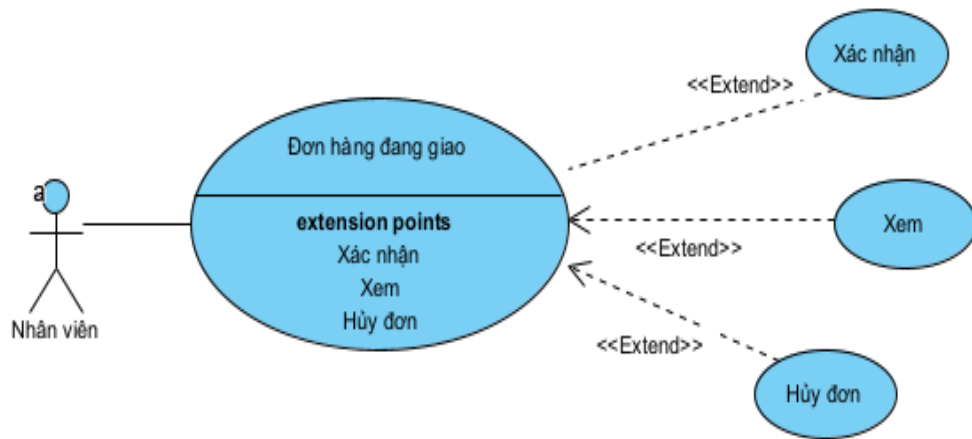
Hình 3. 2: Hình usecase quản lý loại sản phẩm



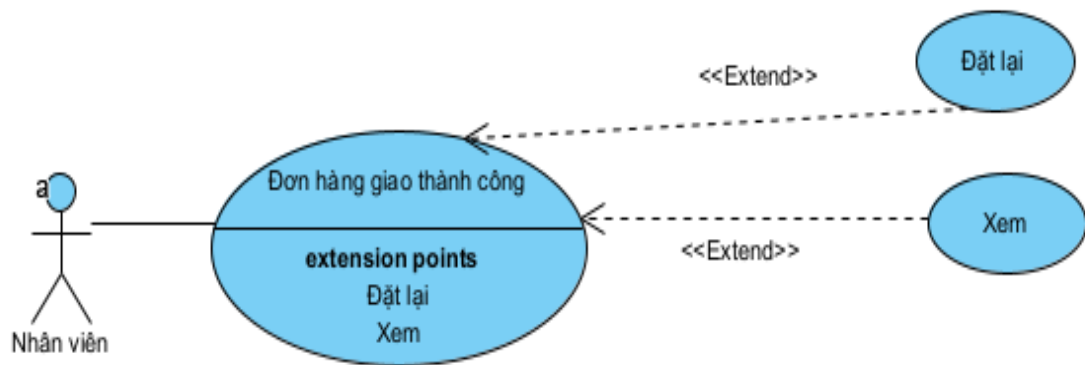
Hình 3. 3: Hình usecase quản lý sản phẩm



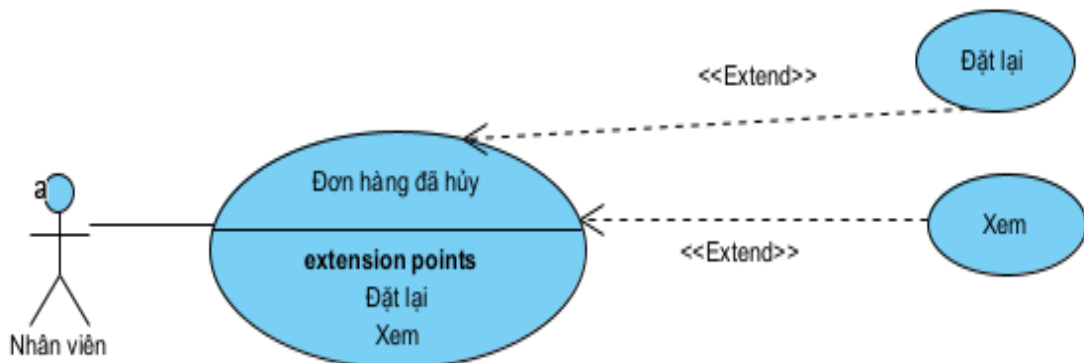
Hình 3. 4: Hình usecase quản lý đơn chưa xử lý



Hình 3. 5: Hình usecase quản lý đơn đang giao



Hình 3. 6: Hình usecase quản lý giao thành công

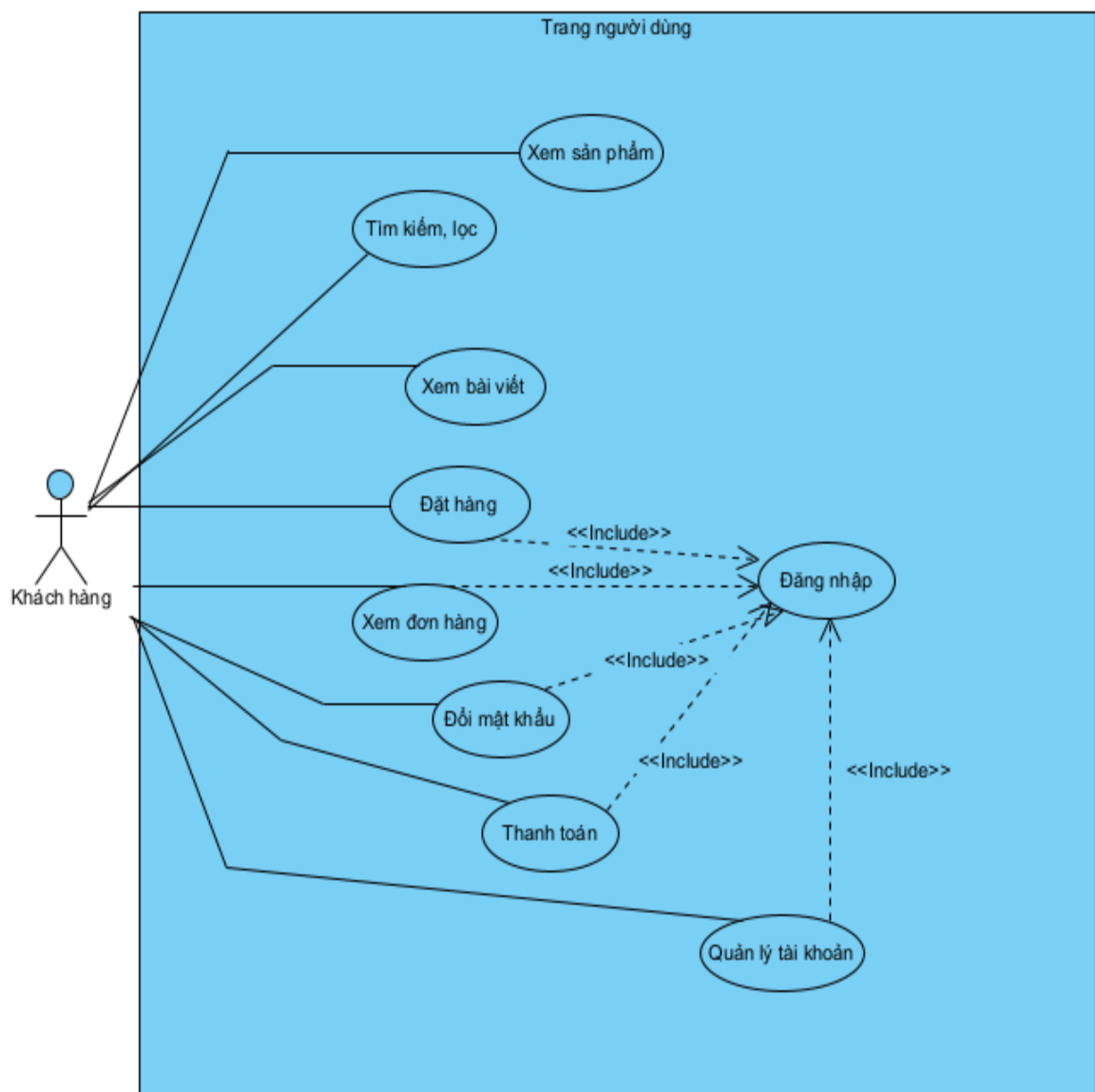


Hình 3. 7: Hình usecase quản lý đơn hàng đã hủy

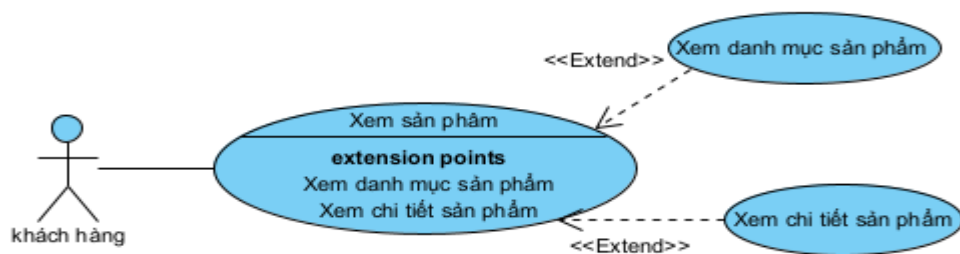
b) Chức năng của phân hệ người dùng

Bảng 3.5: Các yêu cầu chức năng

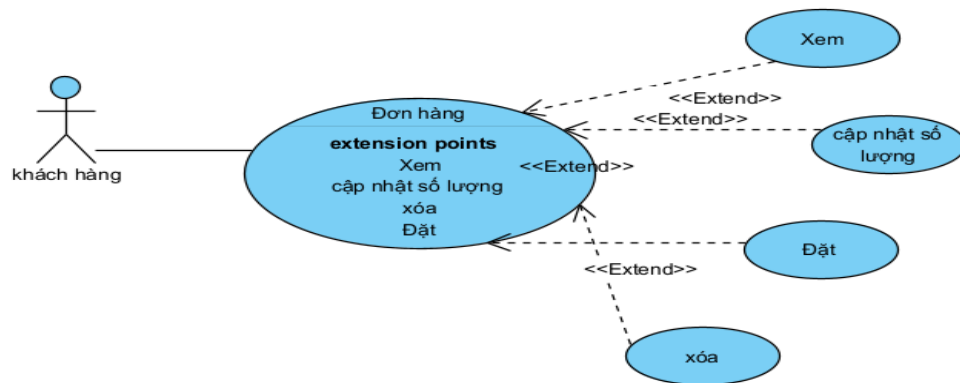
STT	Chức năng	Mô tả
16.	Xem sản phẩm	Người dùng có thể xem danh sách sản phẩm hoặc xem thông tin chi tiết của một sản phẩm cụ thể trên trang chủ.
17.	Tìm kiếm, lọc tin	Người dùng có thể tìm kiếm và lọc sản phẩm theo các tiêu chí như danh mục, thương hiệu, giá tiền ...
18.	Thêm sản phẩm vào giỏ hàng	Người dùng sau khi đăng nhập có thể thêm sản phẩm vào giỏ hàng.
19.	Giỏ hàng	Người dùng được phép thao tác các chức năng như tăng giảm số lượng, xóa sản phẩm, áp voucher.
20.	Đặt hàng	Người dùng có thể đặt hàng với những sản phẩm được thêm vào giỏ hàng.
21.	Đơn hàng	Người dùng có thể theo dõi đơn hàng của mình.
22.	Đổi mật khẩu	Người dùng có thể đổi mật khẩu tài khoản của mình.
23.	Đăng ký	Người dùng có thể đăng ký tài khoản để sử dụng các chức năng trên website.
24.	Đăng nhập	Người dùng có thể đăng nhập vào tài khoản của mình trên website.



Hình 3. 8: Hình usecase tổng quát trang người dùng

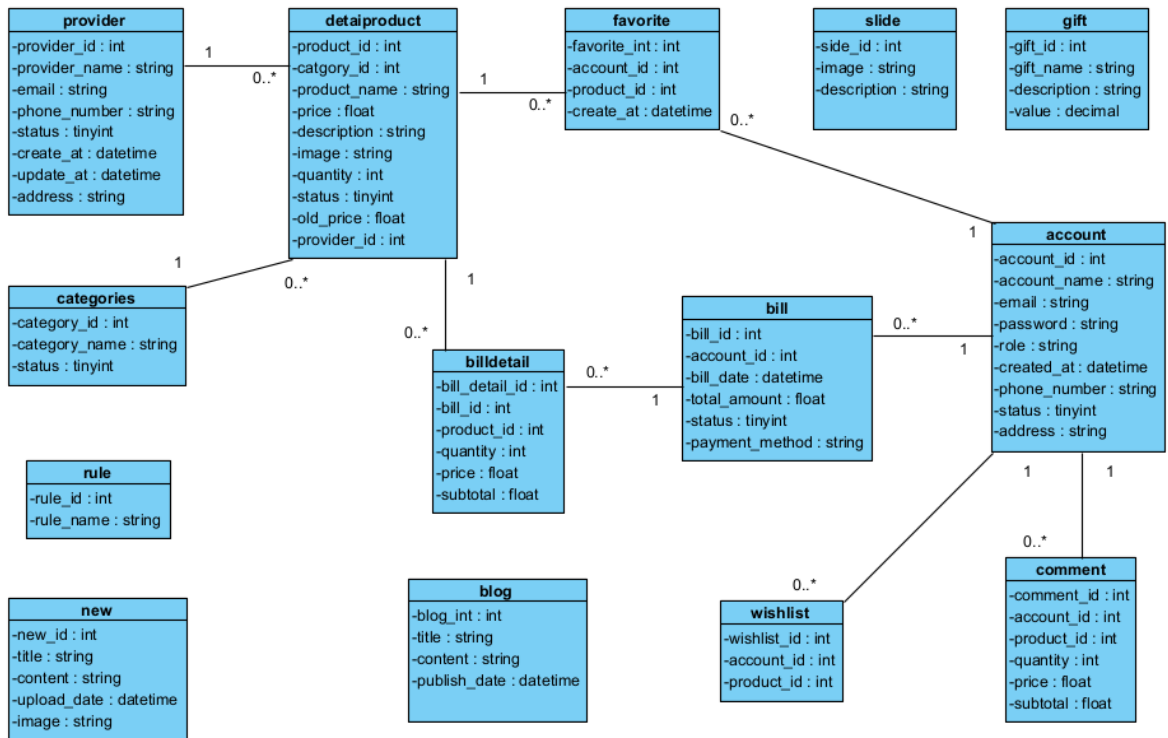


Hình 3. 9: Hình usecase phân rã xem sản phẩm



Hình 3. 10: Hình usecase phân rã đơn hàng

3.2.2 Biểu đồ lớp thực thể



Hình 3. 11: Biểu đồ lớp thực thể

3.2.3. Các yêu cầu phi chức năng

Tính bảo mật truy cập: Bảo mật truy cập là mức độ mà hệ thống được bảo vệ chống lại các lỗi cố ý và xâm nhập từ các nguồn bên trong và bên ngoài.

Tính khả năng truy cập: Khả năng truy cập là mức độ mà hệ thống phần mềm có thể được sử dụng bởi những người có phạm vi khả năng rộng nhất để đạt được mục tiêu trong một bối cảnh nhất định.

Tính sẵn có hay tính khả dụng: là thuộc tính của phần mềm có sẵn và sẵn sàng thực hiện nhiệm vụ của bạn khi bạn cần sử dụng.

Tính bảo mật: bảo mật là mức độ mà hệ thống phần mềm bảo vệ dữ liệu nhạy cảm và chỉ cho phép truy cập dữ liệu được ủy quyền.

Tính hiệu quả: Hiệu quả là mức độ mà hệ thống phần mềm xử lý dung lượng và thời gian đáp ứng.

Tính toán vẹn: là mức độ mà dữ liệu được duy trì bởi hệ thống phần mềm là chính xác, xác thực của dữ liệu.

Độ tin cậy: Độ tin cậy là mức độ mà hệ thống phần mềm thực hiện nhất quán các chức năng được chỉ định mà không gặp sự cố khi người dùng sử dụng.

Tính an toàn: là mức độ mà một hệ thống phần mềm ngăn chặn tác hại đối với con người hoặc thiệt hại cho môi trường trong bối cảnh sử dụng dự định.

Tính khả dụng: là sự dễ dàng mà người dùng có thể tìm hiểu, vận hành chuẩn bị đầu vào và giải thích đầu ra thông qua tương tác với một hệ thống phần mềm.

Bảo mật: Các thông tin về tài khoản, khách hàng, hóa đơn được bảo mật chặt chẽ không để mất và rò rỉ thông tin.

Hiệu suất: Thời gian phản hồi nhanh; Giao diện trang web: Đẹp, dễ nhìn, dễ sử dụng. Khả năng phục hồi: Phục hồi được dữ liệu khi mất mát hoặc bị xóa nhầm.

3.3. Thiết kế hệ thống

3.3.1. Thiết kế kiến trúc

- Mô hình API

Web API hoạt động:

1. Đầu tiên là xây dựng URL API để bên thứ ba có thể gửi request dữ liệu đến máy chủ cung cấp nội dung, dịch vụ thông qua giao thức HTTP hoặc HTTPS.
2. Tại web server cung cấp nội dung, các ứng dụng nguồn sẽ thực hiện kiểm tra xác thực nếu có và tìm đến tài nguyên thích hợp để tạo nội dung trả về kết quả.
3. Server trả về kết quả theo định dạng JSON hoặc XML thông qua giao thức HTTP/HTTPS.

4. Tại nơi yêu cầu ban đầu là ứng dụng web hoặc ứng dụng di động, dữ liệu JSON/XML sẽ được parse để lấy data. Sau khi có được data thì thực hiện tiếp các hoạt động như lưu dữ liệu xuống Cơ sở dữ liệu, hiển thị dữ liệu...

3.3.2. Thiết kế cơ sở dữ liệu

- Mô tả cấu trúc bảng

Bảng 3. 6: Cấu trúc bảng categories

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Category_id	Int	Primary key, auto-increment	Mã danh mục sản phẩm
2	Category_name	Varchar	Not Null	Tên danh mục
3	Status	Tinyint	-	Trạng thái của danh mục sản phẩm

Bảng 3. 7: Cấu trúc bảng detail_product

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Product_id	Int	Primary key, auto-increment	Mã sản phẩm
2	Category_id	Int	Foreign key	Mã danh mục sản phẩm
3	Product_name	Varchar	Not null	Tên sản phẩm
4	Price	Float	Not null	Giá sản phẩm
5	Description	Text	-	Mô tả
6	Old_price	Float	-	Giá cũ
7	Quantity	Int	Not Null	Số lượng
8	Status	Tinyint	-	Trạng thái
9	Provider_id	int	Foreign key	Mã nhà cung cấp

Bảng 3. 8: Cấu trúc bảng account

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Account_id	Int	Primary key, auto-increment	Mã tài khoản
2	Account_name	Varchar	Not Null	Tên người dùng
3	Email	Varchar	Not Null, unique index	Email
4	Password	Varchar	Not Null	Mật khẩu
5	Role	Varchar	-	Vai trò
6	Created_at	Timestamp	-	Ngày tạo
7	Phone_number	Varchar	-	Số điện thoại
8	Status	Tinyint	-	Trạng thái
9	Address	Varchar	-	Địa chỉ

Bảng 3.9: Cấu trúc bảng favorite

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Favorite_id	Int	Primary key, auto-increment	Mã yêu thích
2	Account_id	Int	Foreign_key	Mã tài khoản
3	Product_id	Int	Foreign_key	Mã sản phẩm
5	created_at	Timestamp	-	Ngày tạo

Bảng 3.10: Cấu trúc bảng comment

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Comment_id	Int	Primary key, auto-increment	Mã bình luận
2	Account_id	Int	Foreign_key	Mã tài khoản
3	Product_id	Int	Foreign_key	Mã sản phẩm
4	Comment_date	Timestamp	-	Ngày bình luận

Bảng 3. 11: Cấu trúc bảng blog

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Blog_id	Int	Primary key, auto-increment	Mã blog
2	Title	Varchar	Varchar	Tiêu đề
3	Content	Text	-	Nội dung
4	Publish_date	Timestamp	-	Ngày đăng

Bảng 3. 12: Cấu trúc bảng news

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	New_id	Int	Primary key, auto-increment	Mã tin tức
2	Title	Varchar	Varchar	Tiêu đề
3	Content	Text	-	Nội dung
4	Upload_date	Timestamp	-	Ngày đăng
5	Image	Varchar	-	Hình ảnh

Bảng 3. 13: Cấu trúc bảng wishlist

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Wishlist_id	Int	Primary key, auto-increment	Mã danh sách liên quan
2	Account_id	Int	-	Mã tài khoản
3	Product_id	Int	-	Mã sản phẩm

Bảng 3. 14: Cấu trúc bảng slide

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Slide_id	Int	Primary key, auto-increment	Mã slide
2	Image	String	-	Hình ảnh
3	Description	String	-	Mô tả

Bảng 3. 15: Cấu trúc bảng gift

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Gift_id	Int	Primary key, auto-increment	Mã quà tặng
2	Gift_name	Varchar	Not null	Tên quà tặng
3	Description	Text	-	Mô tả
4	Value	decimal	Not null	Giá trị

Bảng 3. 16: Cấu trúc bảng rule

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Rule_id	Int	Primary key, auto-increment	Mã quy tắc
2	Rule_name	Varchar	Not null	Tên quy tắc

Bảng 3. 17: Cấu trúc bảng bill

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Bill_id	Int	Primary key, auto-increment	Mã đơn hàng
2	Account_id	Int	Foreign_key	Mã khách hàng
3	Bill_date	Timestamp	-	Ngày tạo hóa đơn
4	Total_amount	Float	-	Tổng tiền
5	Status	Tinyint	-	Trạng thái
6	Payment_method	Varchar	-	Phương thức thanh toán

Bảng 3. 18: Cấu trúc bảng bill_detail

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Bill_detail_id	Int	Primary key, auto-increment	Mã chi tiết đơn hàng

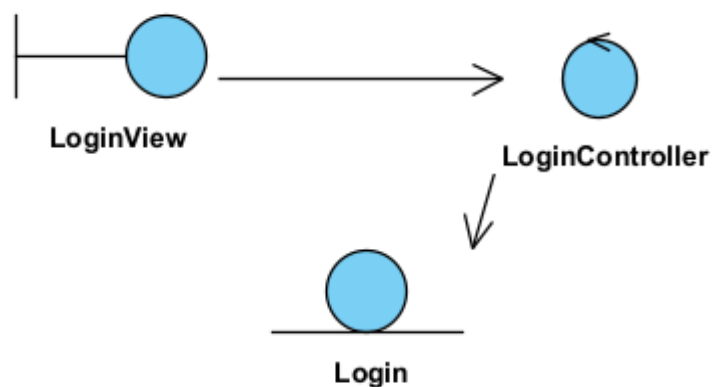
2	Bill_id	Int	Foreign_key	Mã đơn hàng
3	Product_id	Int	Foreign_key	Mã sản phẩm
4	Quantity	Int	-	Số lượng
5	Price	Float	-	Giá
6	Subtotal	Float	-	Tổng tiền phụ

Bảng 3. 19: Cấu trúc bảng provider

STT	Tên trường	Kiểu dữ liệu	Ràng buộc	Ghi chú
1	Provider_id	Int	Primary key, auto-increment	Mã nhà cung cấp
2	Provider_name	Varchar	Not Null	Tên nhà cung cấp
3	Email	Varchar	Unique index	Email
4	Create_at	Timestamp	-	Ngày tạo
5	Update_at	Timestamp	-	Ngày cập nhật
6	Phone_number	Varchar	-	Số điện thoại
7	Status	Tinyint	-	Trạng thái
8	Address	Varchar	-	Địa chỉ

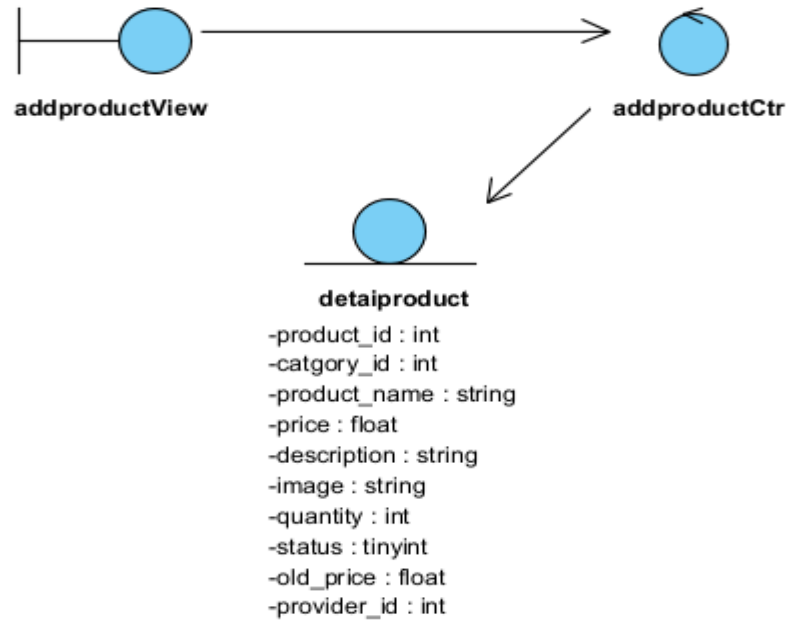
3.3.3. Thiết kế lớp đối tượng

a) Biểu đồ lớp VOPC của các ca sử dụng

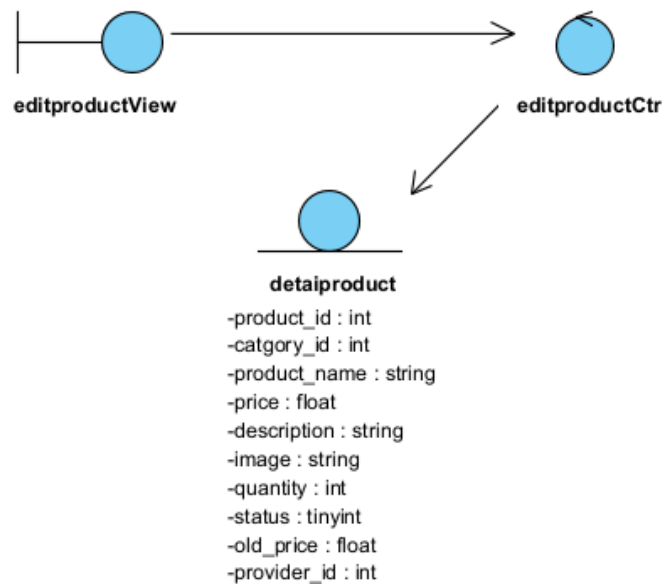


Hình 3. 12: Biểu đồ VOPC của use case Đăng Nhập

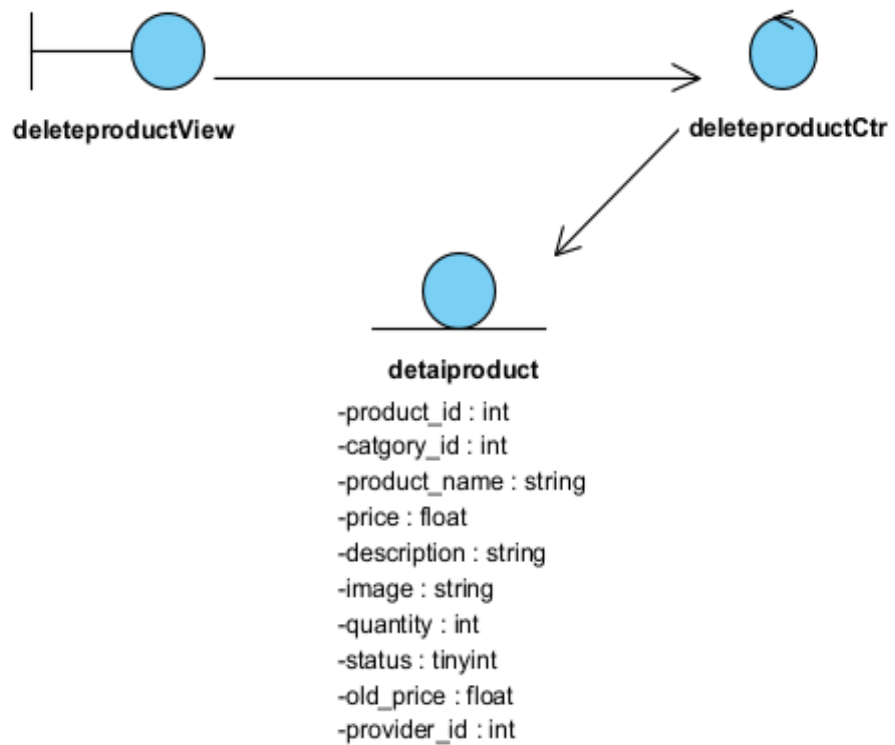
- Biểu đồ phân tích của use case quản lý sản phẩm



Hình 3. 13: Biểu đồ VOPC của thêm sản phẩm

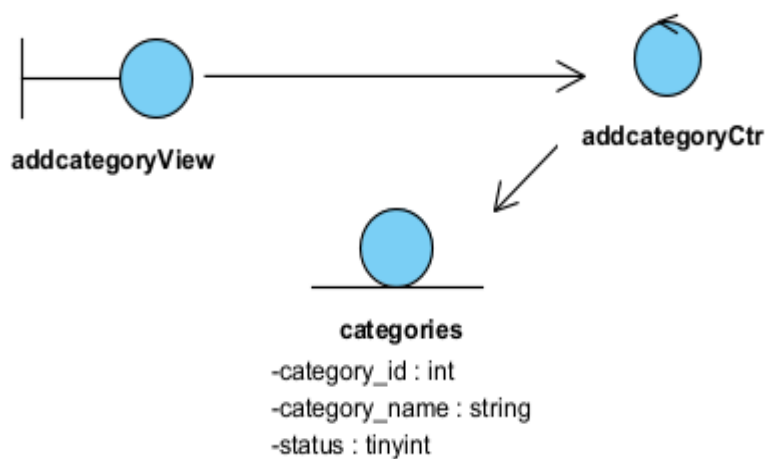


Hình 3. 14: Biểu đồ VOPC của sửa sản phẩm

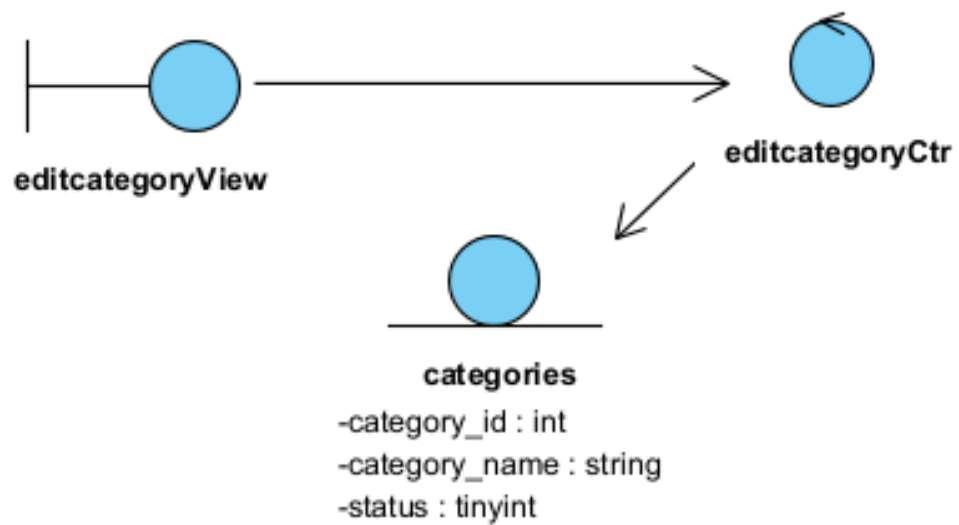


Hình 3. 15: Biểu đồ VOPC của xóa sản phẩm

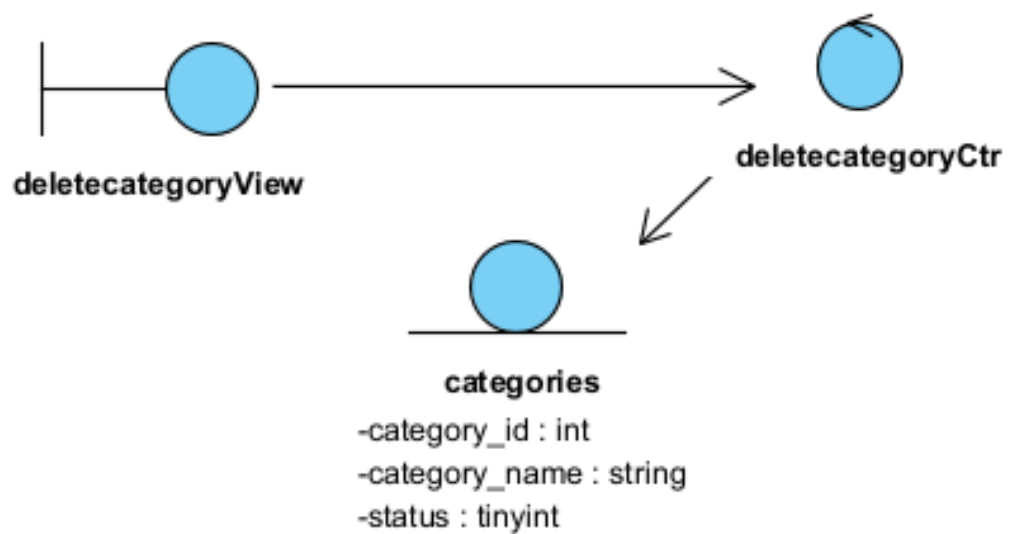
- Biểu đồ phân tích của use case quản lý loại sản phẩm



Hình 3. 16: Biểu đồ VOPC của use thêm loại sản phẩm

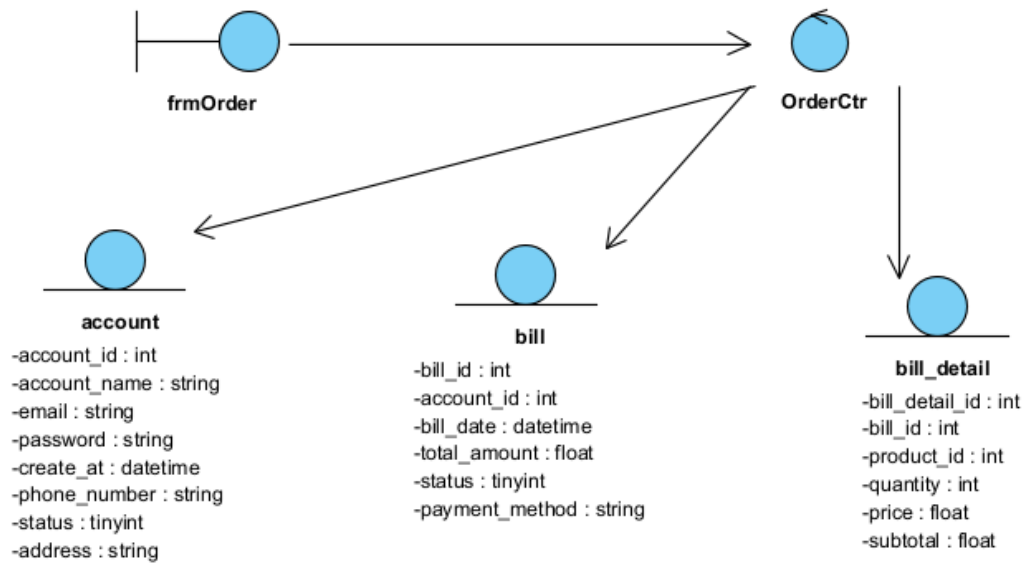


Hình 3. 17: Biểu đồ VOPC của use case sửa loại sản phẩm



Hình 3. 18: Biểu đồ VOPC của use case xóa loại sản phẩm

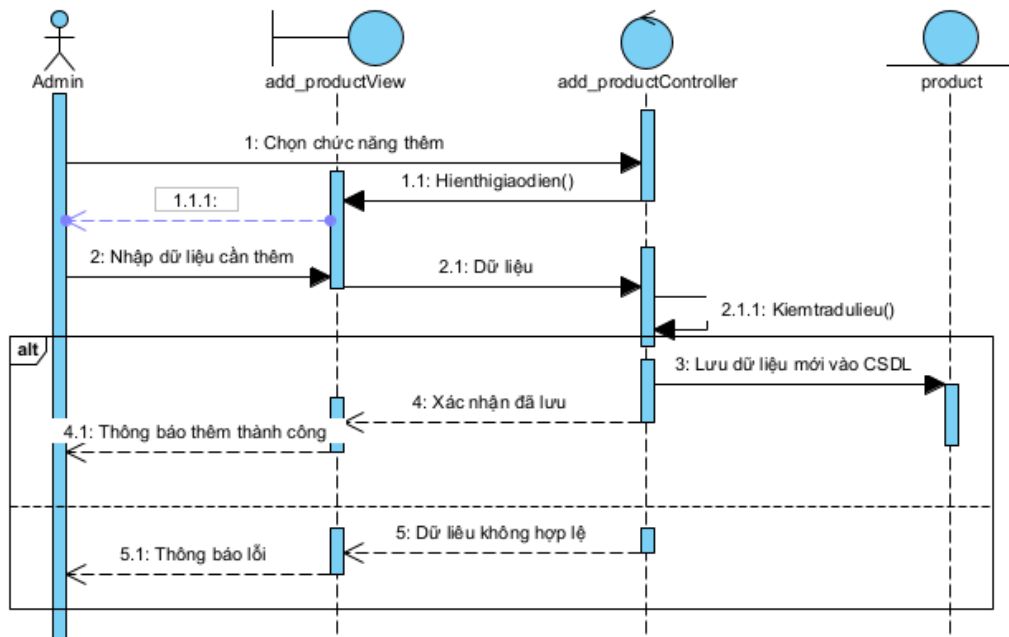
- Biểu đồ phân tích của use case đặt hàng



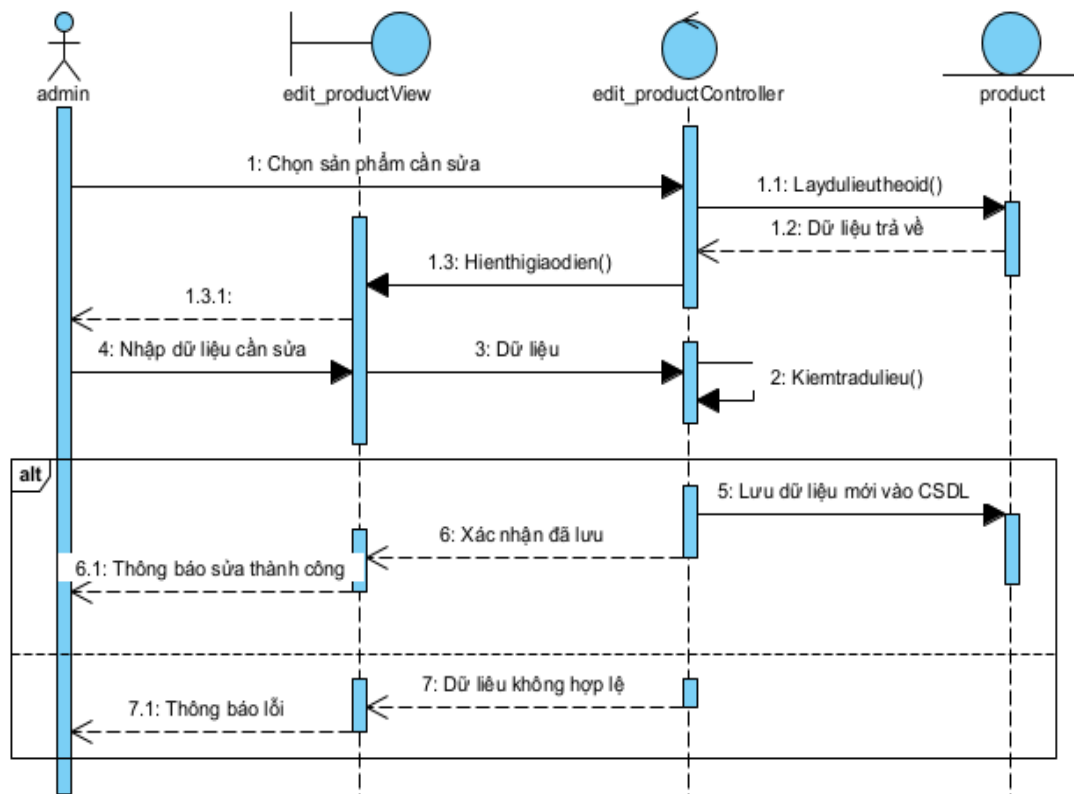
Hình 3. 19: Biểu đồ VOPC của use case đặt hàng

b) Biểu đồ tuần tự

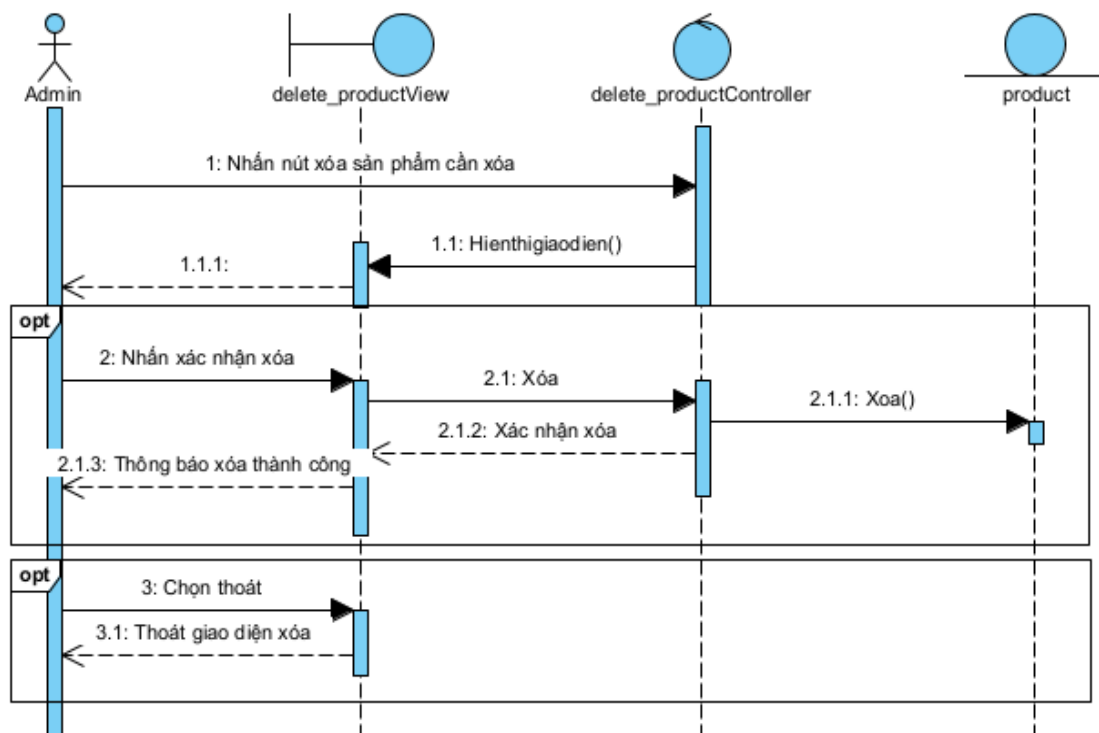
- Biểu đồ tuần tự của use case quản lý sản phẩm



Bảng 3. 20: Biểu đồ tuần tự của thêm sản phẩm

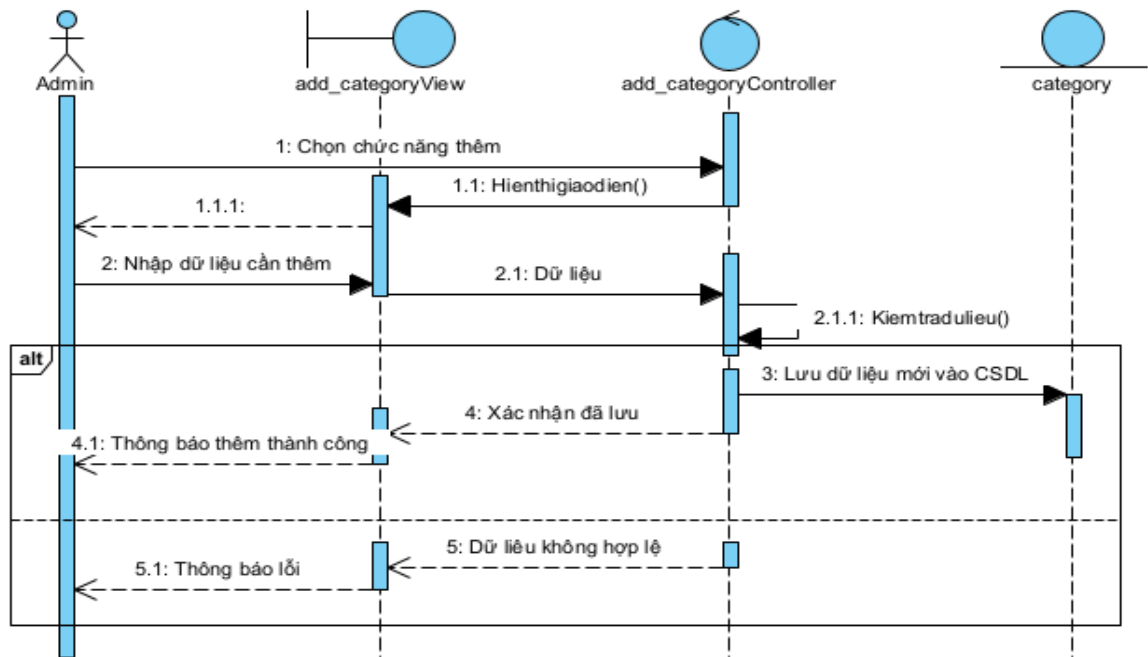


Hình 3. 20: Biểu đồ tuần tự của use case sửa sản phẩm

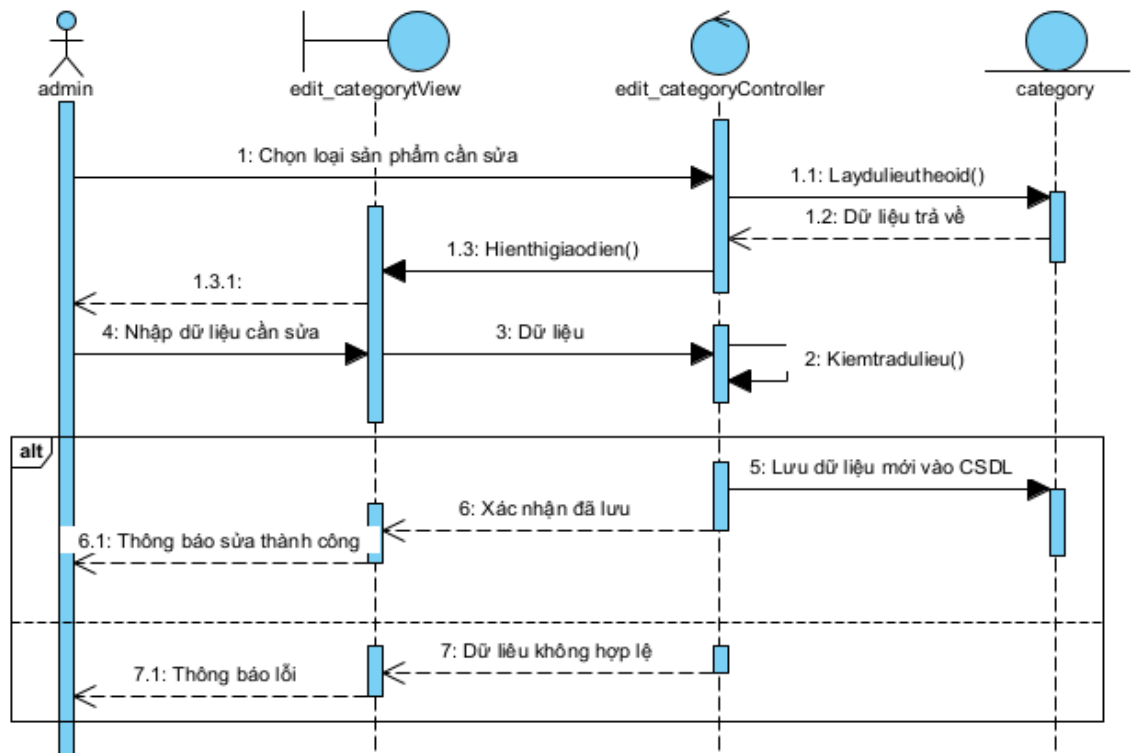


Hình 3. 21: Biểu đồ tuần tự của xóa sản phẩm

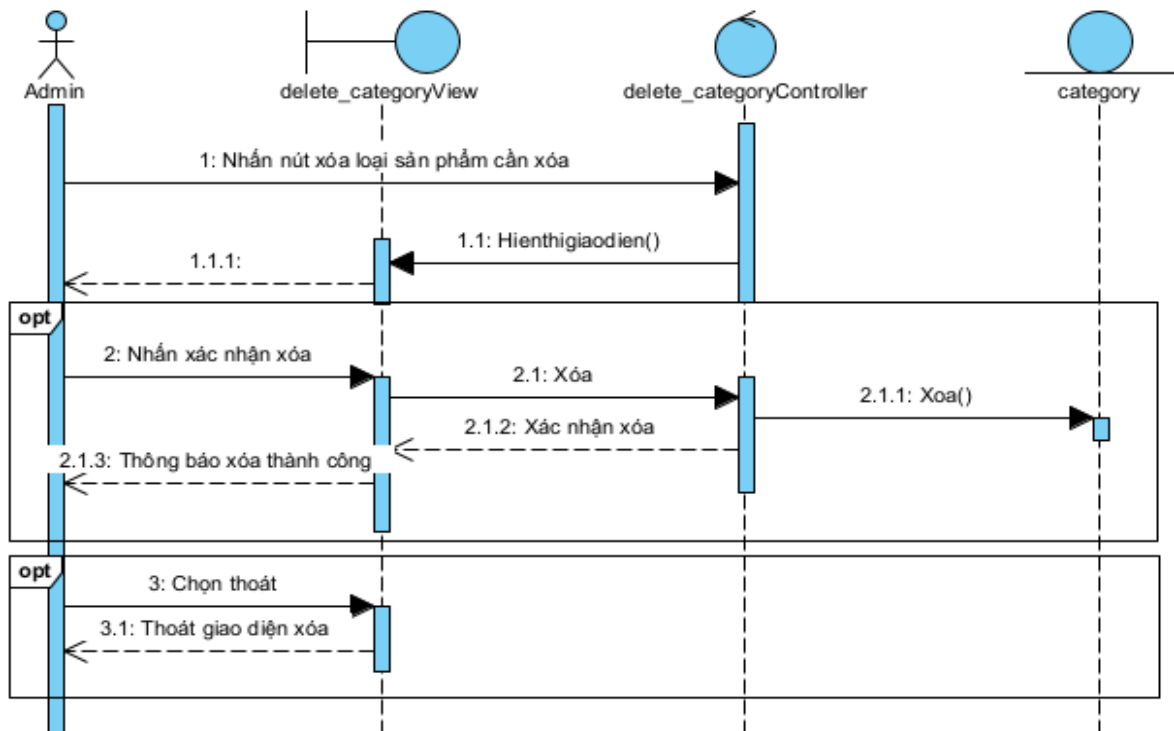
- Biểu đồ tuần tự của use case quản lý loại sản phẩm



Hình 3. 22: Biểu đồ tuần tự của use thêm loại sản phẩm



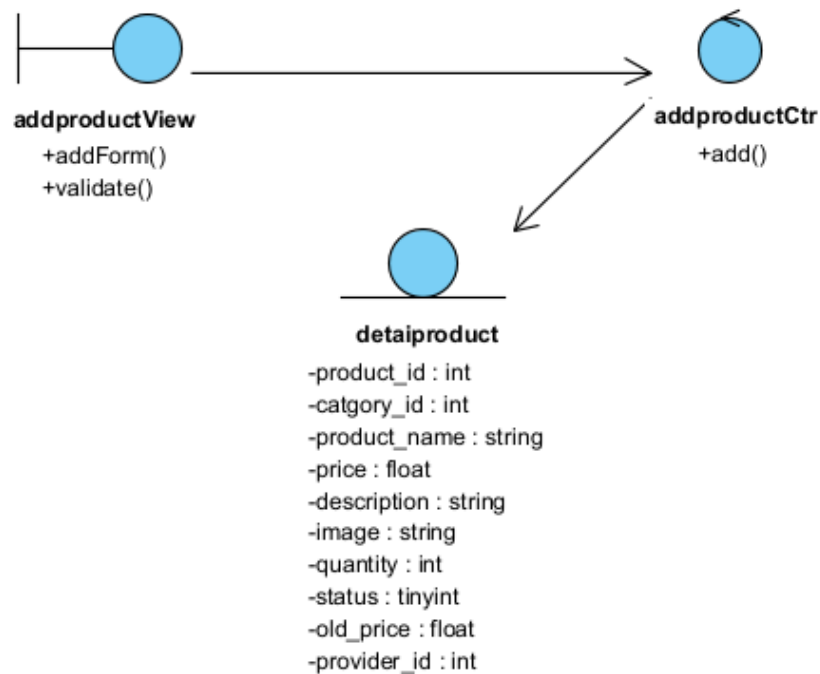
Hình 3. 23: Biểu đồ tuần tự của use case sửa loại sản phẩm



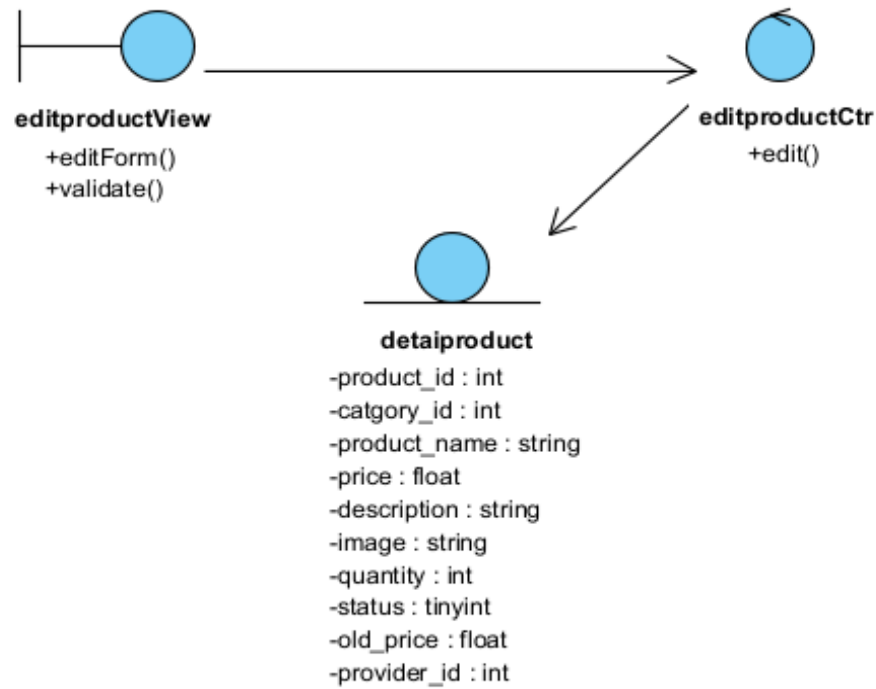
Hình 3. 24: Biểu đồ tuần tự của use case xóa loại sản phẩm

c) Biểu đồ lớp chi tiết

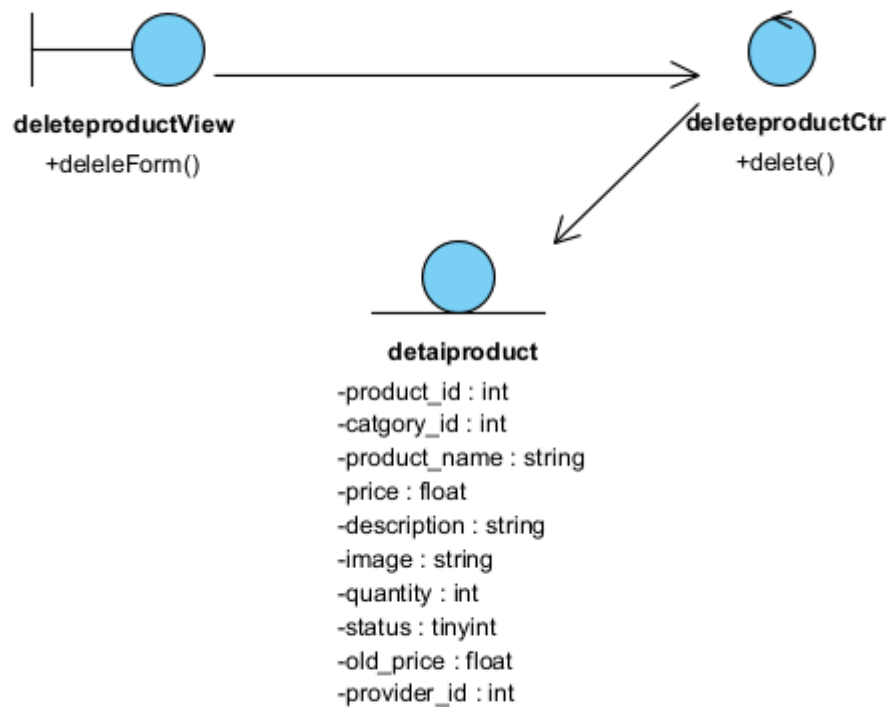
- Biểu đồ lớp thiết kế của use case quản lý sản phẩm



Hình 3. 25: Biểu đồ lớp thiết kế của thêm sản phẩm

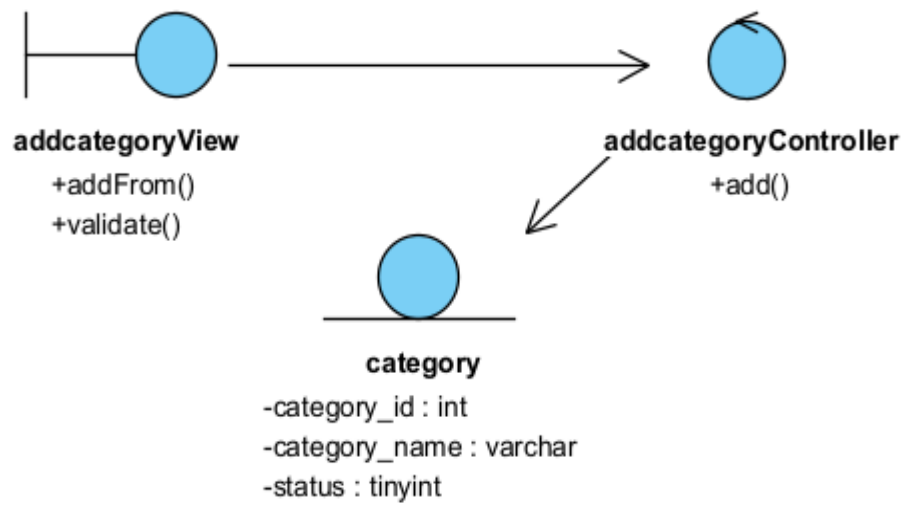


Hình 3. 26: Biểu đồ lớp thiết kế của use case sửa sản phẩm

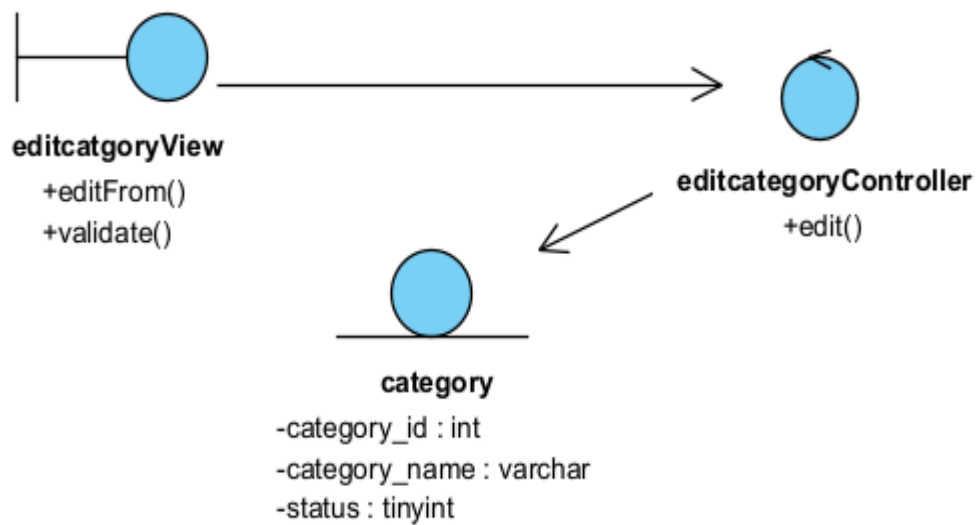


Hình 3. 27: Biểu đồ lớp thiết kế của xóa sản phẩm

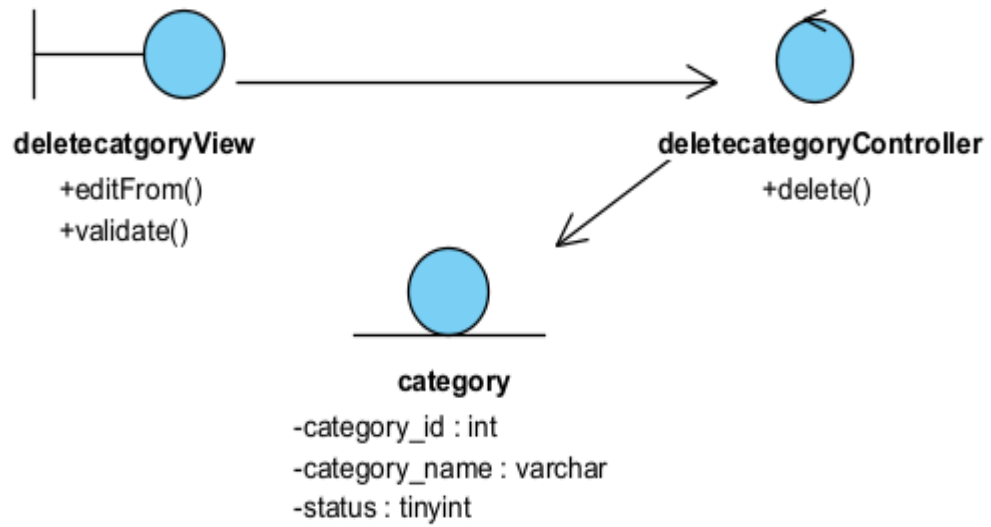
- Biểu đồ lớp thiết kế của use case quản lý loại sản phẩm



Hình 3. 28: Biểu đồ lớp thiết kế của use thêm loại sản phẩm



Hình 3. 29: Biểu đồ lớp thiết kế của use case sửa loại sản phẩm



Hình 3. 30: Biểu đồ lớp thiết kế của use case xóa loại sản phẩm

3.3.4. Thiết kế giao diện

❖ Giao diện trang quản trị

TT	Mã loại sản phẩm	Tên loại sản phẩm	Chức năng
1	1	Ghế	Hiển thị
2	2	Bàn	Hiển thị
3	4	Gối	Hiển thị
4	5	Tranh	Hiển thị
5	6	Đồ gốm	Hiển thị

Hình 3. 31: Bảng quản lý danh mục

Simple Tables <small>preview of simple tables</small>									
Quản lý sản phẩm Thêm mới									
	Mã sản phẩm	Loại	Ảnh sản phẩm	Tên sản phẩm	Giá cũ/giá khuyến mại	Số lượng	Mô tả	Trang thái	Chức năng
1	85	Ghế		sofa mẫu A01	Cũ: 2.000.000 đ/khuyến mại: 1.900.000 đ	33	Chất liệu: <ul style="list-style-type: none">Khung ghế: Gỗ sồi Nga đã qua xử lý, chống mối mọt, cong vênh.Mặt ngồi và tựa lưng: Da bò thật 100%, mềm mại, đàn hồi và thoáng khí.Chân ghế: Thép cao cấp, mạ crom sáng bóng, chịu lực tốt.	Hiện thị	
2	86	Ghế		sofa mẫu b01 màu xám	Cũ: 1.500.000 đ/khuyến mại: 1.400.000 đ	5	Ưu điểm: <ul style="list-style-type: none">Chất liệu cao cấp, độ bền cao.Thiết kế sang trọng, đẳng cấp.Mang đến sự thoải mái tối đa cho người sử dụng.Dễ dàng vệ sinh và bảo quản.	Hiện thị	
3	87	Giường		Bộ chân ga gối đệm họa tiết đen trắng màu a00	Cũ: 5.000.000 đ/khuyến mại: 4.900.000 đ	5	Kích thước: Chân có nhiều kích thước khác nhau, phổ biến nhất là 200x220cm, 180x200cm và 160x200cm. Chất liệu: Chân được làm từ nhiều chất liệu khác nhau như cotton, lụa, nỉ, nhung, v.v. Mỗi chất liệu có ưu và nhược điểm riêng. Cotton: Thoáng mát, thấm hút mồ hôi tốt, phù hợp cho mọi mùa sử dụng. Lụa: Mềm mại, sang trọng, mang lại cảm giác mát mẻ trong mùa hè. Nỉ: Giữ ấm tốt, phù hợp cho mùa đông. Nhung: Mềm mại, êm ái, mang lại cảm giác ấm áp trong mùa	Hiện thị	

Hình 3. 32: Quản lý sản phẩm

Thông tin hóa đơn

Mã hóa đơn: 61

Tên người dùng: Phạm Thị Hương

Email: buihaihiiep123@gmail.com

Số điện thoại: 0333256738

Địa chỉ: Hưng yên

Ngày đặt: 6/8/2024, 11:24:46 AM

Tổng tiền: 570000

Phương thức thanh toán: Thanh toán khi nhận hàng

Trạng thái: Chưa xác nhận

Chi tiết sản phẩm

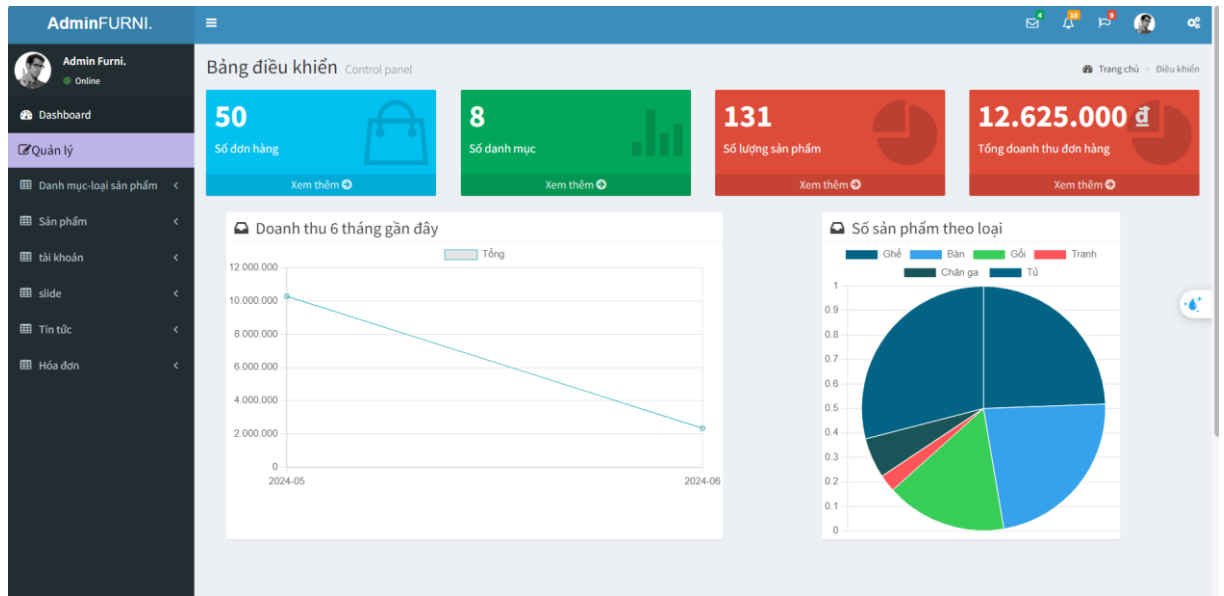
Tên sản phẩm: Bộ tranh treo tường phong cách nghệ thuật cách điệu M01

Số lượng: 1

Giá: 570000

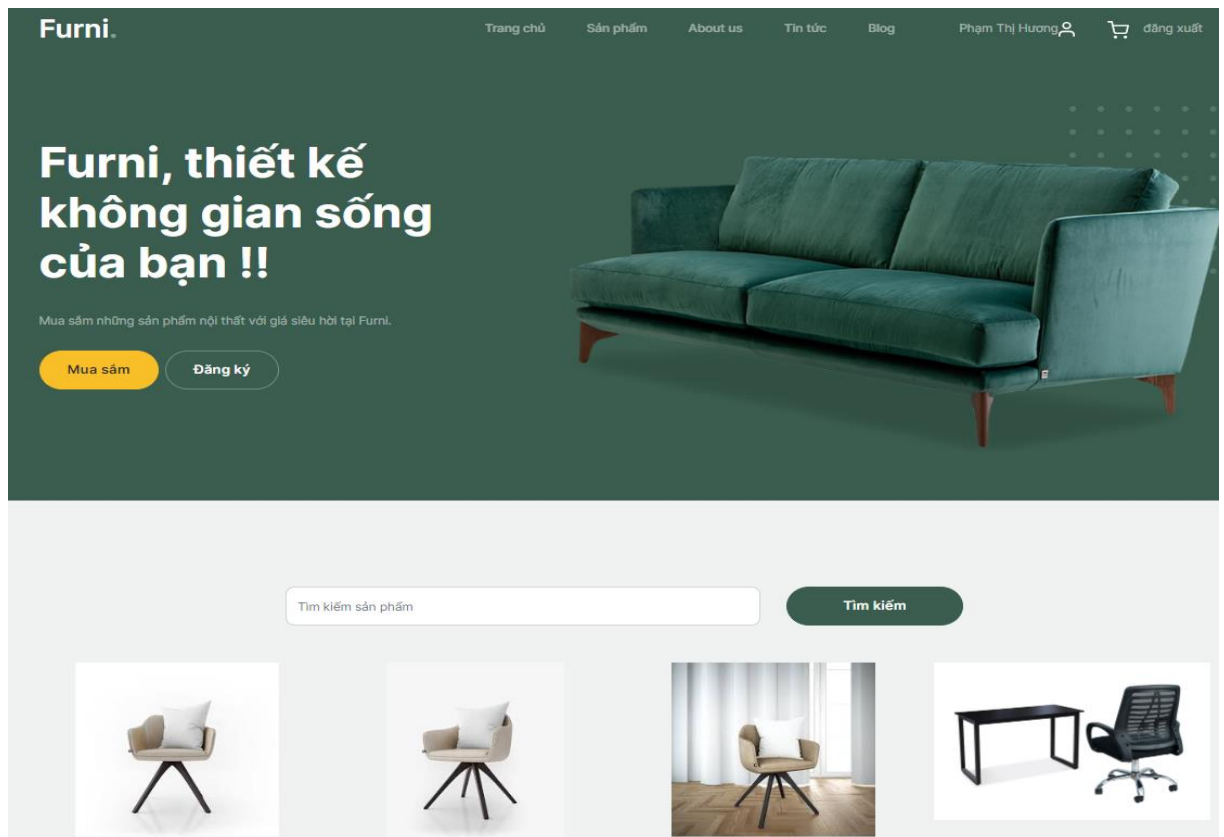
Tổng cộng: 570000

Hình 3. 33: Xem chi tiết hóa đơn

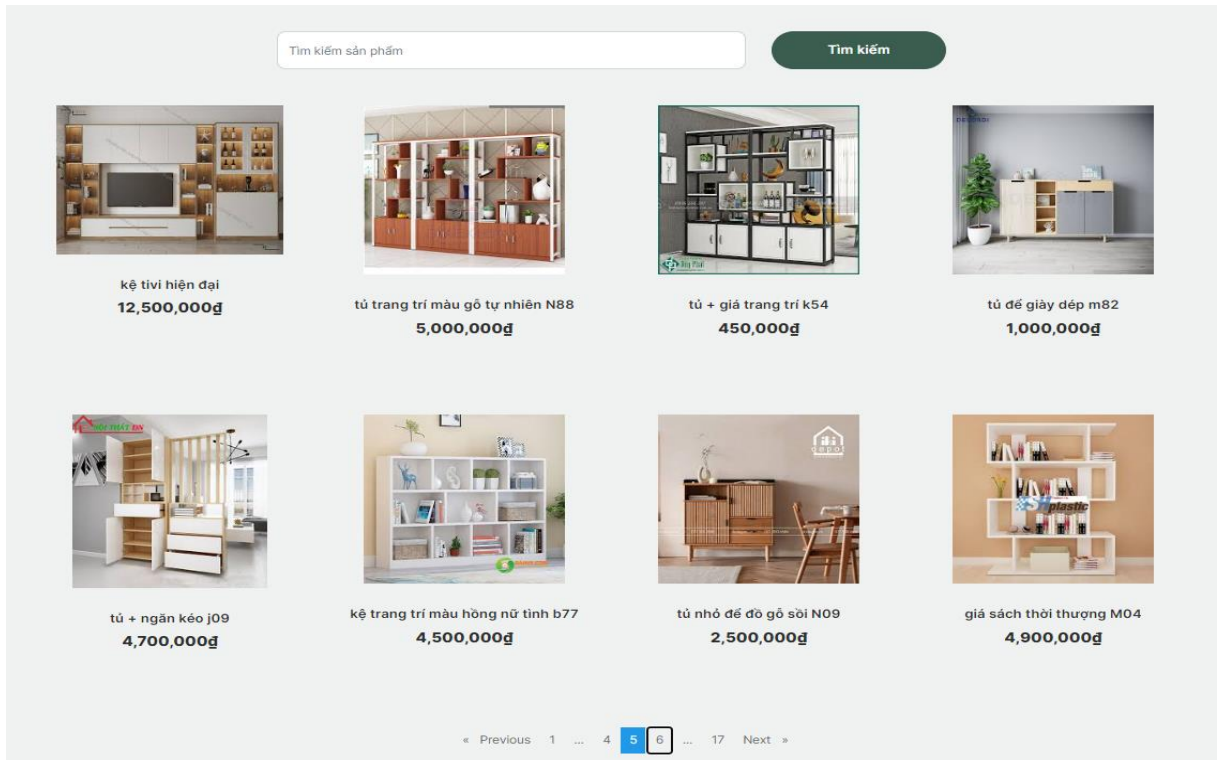


Hình 3. 34: Quản lý thống kê

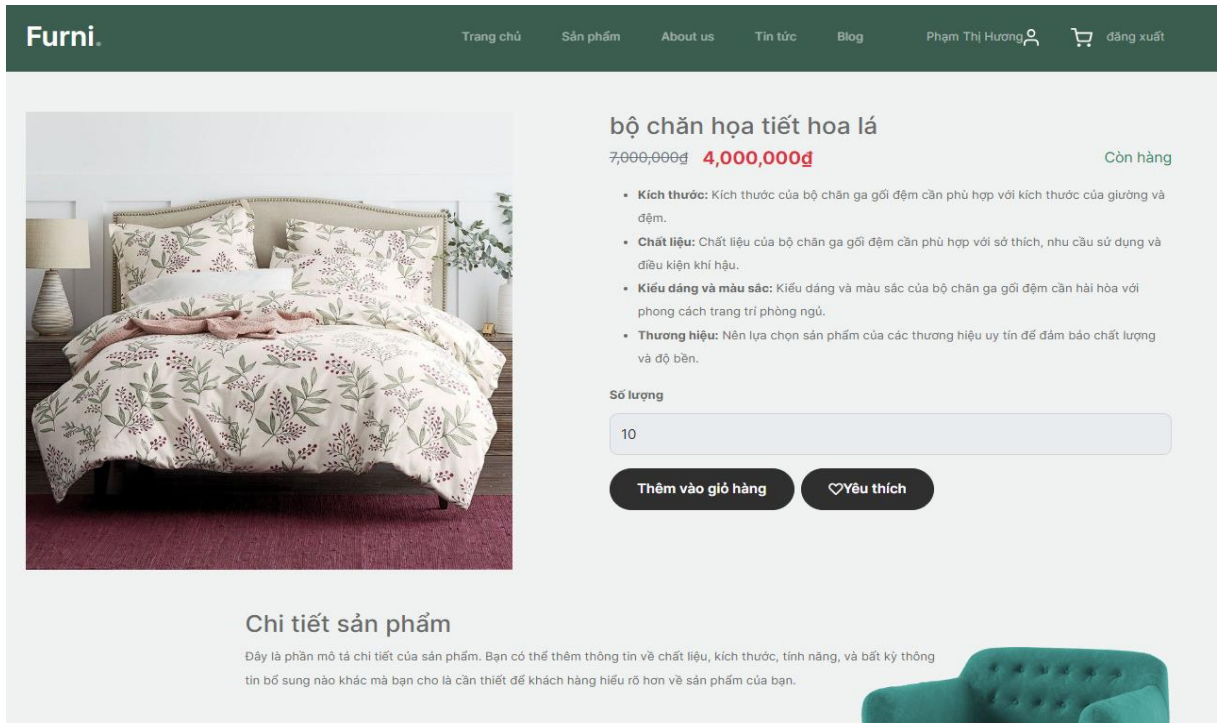
❖ Giao diện trang người dùng



Hình 3. 35: Trang chủ



Hình 3. 36: Trang sản phẩm



Hình 3. 37: Trang chi tiết sản phẩm

Bạn có nhu cầu mua hàng hay sử dụng các chức năng khác? [Nhấp chuột](#) để đăng nhập!

Chi tiết hóa đơn

Họ tên *

Phạm Thị Hương

Địa chỉ *

Hưng yên

Email *

buihaihiiep123@gmail.com


Số điện thoại *

0333256738

Phương Thức Thanh toán *


Chọn phương thức thanh toán
VNPAY
Thanh toán khi nhận hàng

Your Order

Ảnh	Sản phẩm	Số tiền
	bộ chăn họa tiết hoa lá x 1	4,000,000đ

Tổng thanh toán: 4,000,000đ

Đặt hàng



Hình 3. 38: Trang thanh toán

CHƯƠNG 4: TRIỂN KHAI WEBSITE

4.1. Triển khai các chức năng cho ứng dụng

- Phương pháp xây dựng API
 - Xây dựng API theo hướng thủ tục
 - Tạo các đường link API trong file routes
- Đặc tả các API
 - API phân hệ quản trị

Bảng 4. 1: API phân hệ quản trị

STT	Chức năng	Phương thức	API	Mô tả
1	Thêm sản phẩm	Post	http://localhost:3000/product-add	Api thêm sản phẩm
2	Sửa sản phẩm	Post	http://localhost:3000/product-edit/:id	Api sửa sản phẩm
3	Xóa sản phẩm	Delete	http://localhost:3000/product-delete/:id	Api xóa sản phẩm
4	Tìm kiếm sản phẩm	Get	http://localhost:3000/product/?name	Api tìm kiếm sản phẩm
5	Đăng nhập	Post	http://localhost:3000/login	Api đăng nhập
6	Tìm kiếm người dùng	Get	http://localhost:3000/account/?name	Api tìm kiếm người dùng
14	Cập nhật loại sản phẩm	Post	http://localhost:3000/category-edit/:id	Api cập nhật loại sản phẩm

STT	Chức năng	Phương thức	API	Mô tả
15	Xem loại sản phẩm	Get	http://localhost:3000/category-edit/:id	Api xem loại sản phẩm
16	Tìm kiếm loại sản phẩm	Get	http://localhost:3000/category/?name	Api tìm kiếm loại sản phẩm
17	Xóa loại sản phẩm	Delete	http://localhost:3000/category-deletet/:id	Api xóa loại sản phẩm
18	Thêm loại sản phẩm	Post	http://localhost:3000/category-add	Api thống kê tổng doanh thu
19	Xóa tin tức	Post	http://localhost:3000/new-delete/:id	Api xóa tin tức
20	Xem tin tức	Get	http://localhost:3000/new-edit/:id	Api xem tin tức
21	Thêm nhà cung cấp	Post	http://localhost:3000/provider-add	Api thêm nhà cung cấp
22	Sửa nhà cung cấp	Post	http://localhost:3000/provider-edit/:id	Api sửa nhà cung cấp
23	Xem nhà cung cấp	Get	http://localhost:3000/provider-edit/:id	Api xem nhà cung cấp
24	Xóa nhà cung cấp	Post	http://localhost:3000/provider-delete/:id	Api xóa nhà cung cấp

STT	Chức năng	Phương thức	API	Mô tả
25	Thêm mới tin tức	Post	http://localhost:3000/new	Api tin tức
26	Cập nhật tin tức	Get	http://localhost:3000/new/:id	Api sửa tin tức
27	Thống kê số đơn hàng đã bán	Get	http://localhost:3000	Api thống kê số đơn hàng đã bán
28	Thống kê tổng sản phẩm	Get	http://localhost:3000	Api thống kê tổng sản phẩm
29	Thống kê tổng doanh thu	Get	http://localhost:3000	Api thống kê tổng doanh thu
30	Thống kê số đơn hàng mới	Get	http://localhost:3000	Api thống kê số đơn hàng mới
31	Thống kê số lượng sản phẩm	Get	http://localhost:3000	Api thống kê số lượng sản phẩm theo loại

STT	Chức năng	Phương thức	API	Mô tả
	phẩm theo loại			
32	Thông kê top sản phẩm bán chạy	Get	http://localhost:3000	Api thống kê top sản phẩm bán chạy

- API phân hệ người dùng

Bảng 4. 2: API phân hệ người dùng

STT	Chức năng	Phương thức	API	Mô tả
1	Đăng ký	Post	http://localhost:3000/signupUSER	Api đăng ký
2	Đăng nhập	Post	http://localhost:3000/loginUSER	Api đăng nhập
2	Quên mật khẩu	Post	http://localhost:3000/forgotPasswordUSER	Api quên mật khẩu
3	Xem thông tin	Get	http://localhost:3000/getUSERById	Api xem thông tin người dùng

STT	Chức năng	Phương thức	API	Mô tả
4	Thay đổi mật khẩu	Post	http://localhost:3000/changePasswordUSER	Api thay đổi mật khẩu
5	Xem danh sách sản phẩm	Get	http://localhost:3000/api/product	Api trang danh sách sản phẩm
6	Xem chi tiết sản phẩm	Get	http://localhost:3000/api/product/:id	Api trang chi tiết sản phẩm
7	Xem danh sách sản phẩm theo loại	Get	http://localhost:3000/api/product/category/:id	Api danh sách sản phẩm theo loại
8	Hiển thị tin tức	Get	http://localhost:3000/api/new	Api hiển thị tin tức

STT	Chức năng	Phương thức	API	Mô tả
9	Xem tin tức	Get	http://localhost:3000/api/new/:id	Api xem tin tức
10	Yêu thích sản phẩm	Post	http://localhost:3000/api/favorite	Api yêu thích sản phẩm
11	Đặt hàng	Post	http://localhost:3000/api/createbill	Api đặt hàng

4.2. Xây dựng các chức năng

4.2.1. Các chức năng phân hệ người dùng

Bảng 4. 3: Chức năng nghiệp vụ phân hệ người dùng

Chức năng	Ý tưởng xây dựng	Xử lý nghiệp vụ
Đăng nhập	Cung cấp giao diện đăng nhập cho người dùng.	Xác thực thông tin đăng nhập, kiểm tra tính hợp lệ và cung cấp quyền truy cập tương ứng.
Đăng ký	Cung cấp giao diện đăng ký mới tài khoản.	Xác thực thông tin đăng ký, kiểm tra tính hợp lệ và tạo tài khoản mới trong hệ thống.
Quên mật khẩu	Cung cấp giao diện lấy lại mật khẩu tài khoản.	Xác thực thông tin, kiểm tra tính hợp lệ và gửi lại mật khẩu.
Xem danh sách sản phẩm	Hiển thị danh sách sản phẩm có sẵn trong hệ thống.	Truy vấn cơ sở dữ liệu từ API để lấy thông tin sản phẩm và hiển thị danh sách cho người dùng.

Chức năng	Ý tưởng xây dựng	Xử lý nghiệp vụ
Tìm kiếm sản phẩm	Cho phép người dùng tìm kiếm sản phẩm theo các tiêu chí như hãng, giá, danh mục.	Xử lý yêu cầu tìm kiếm, truy vấn cơ sở dữ liệu từ API và gọi API hiển thị kết quả tìm kiếm cho người dùng.
Xem chi tiết sản phẩm	Hiển thị thông tin chi tiết về một sản phẩm cụ thể.	Truy vấn cơ sở dữ liệu từ API để lấy thông tin chi tiết và hiển thị cho người dùng.
Thêm sản phẩm vào giỏ hàng	Cho phép người dùng thêm sản phẩm vào giỏ hàng.	Cập nhật giỏ hàng của người dùng trong cơ sở dữ liệu từ API và gọi API hiển thị thông tin giỏ hàng.
Đặt hàng	Xử lý đơn đặt hàng và thông tin giao hàng.	Kiểm tra tính hợp lệ của đơn hàng, cập nhật cơ sở dữ liệu bên API và xử lý thông tin giao hàng.
Xem đơn hàng	Hiển thị thông tin chi tiết về các đơn hàng đã đặt.	Truy vấn cơ sở dữ liệu từ API và gọi API hiển thị thông tin đơn hàng cho người dùng.
Quản lý thông tin cá nhân	Cho phép người dùng cập nhật thông tin cá nhân và địa chỉ giao hàng.	Cập nhật thông tin trong cơ sở dữ liệu từ API và đảm bảo tính bảo mật của thông tin người dùng.

4.2.2. Các chức năng phân hệ quản trị

Bảng 4. 4: Chức năng phân hệ quản trị

Chức năng	Ý tưởng xây dựng	Xử lý nghiệp vụ
Quản lý danh mục sản phẩm	Cho phép quản trị viên thêm, sửa đổi và xóa danh mục sản phẩm.	Cập nhật cơ sở dữ liệu danh mục sản phẩm từ API và đảm bảo tính nhất quán của dữ liệu.
Quản lý sản phẩm	Cho phép quản trị viên thêm, sửa đổi và xóa thông tin sản phẩm.	Cập nhật cơ sở dữ liệu sản phẩm từ API và đảm bảo tính nhất quán của dữ liệu.
Quản lý đơn hàng	Hiển thị danh sách đơn hàng và chi tiết của từng đơn hàng.	Truy vấn cơ sở dữ liệu đơn hàng từ API và gọi API hiển thị thông tin đơn hàng cho quản trị viên.
Quản lý tài khoản	Hiển thị danh sách tài khoản.	Truy vấn cơ sở dữ liệu tài khoản từ API và hiển thị thông tin tài khoản cho quản trị viên.
Thống kê	Hệ thống sẽ tự động thống kê và hiển thị dữ liệu lên trang dashboard.	Gọi API, thực hiện các chức năng thống kê thống kê về lượng truy cập trang web, số lượng người dùng, số lượng tin đăng, và các thông tin khác liên quan đến hoạt động của trang web.

4.3. Kiểm thử và triển khai ứng dụng

4.3.1. Kiểm thử chức năng đăng nhập

Bảng 4. 5: Chức năng phân hệ quản trị

ID	Trường hợp thử nghiệm	Tiền điều kiện	Thủ tục trường hợp thử nghiệm	Kết quả dự kiến	Kết quả thực tế
Login 1	Nhập user và password hợp lệ và kiểm tra tồn tại trong CSDL.	None	1. Mở trang đăng nhập 2. Nhập user, password hợp lệ 3. Nhấn nút đăng nhập	- Đăng nhập thành công thì chuyển tới trang dashboard - Tồn tại user, password trong csdl	Pass
Login 2	Nhập user và password không hợp lệ và kiểm tra tồn tại trong CSDL.	None	1. Mở trang đăng nhập 2. Nhập user, password hợp lệ 3. Nhấn nút đăng nhập	- Đăng nhập không thành công thì hiển thị thông báo - Không tồn tại user, password trong csdl	Fail
Login 3	Nhập user và để trống password	None	1. Mở trang đăng nhập 2. Nhập user, password hợp lệ 3. Nhấn nút đăng nhập	- Đăng nhập không thành công thì hiển thị thông báo - User hoặc password không chính xác	Fail
Login 4	Nhập password và để trống user	None	1. Mở trang đăng nhập 2. Nhập user, password hợp lệ 3. Nhấn nút đăng nhập	- Đăng nhập không thành công thì hiển thị thông báo - User hoặc password không chính xác	Fail

4.3.2. Kiểm thử chức năng quản lý sản phẩm

❖ Kiểm thử chức năng thêm sản phẩm

Bảng 4. 6: Kiểm thử chức năng thêm sản phẩm

STT	Trường hợp thử nghiệm	Tiền điều kiện	Thủ tục trường hợp thử nghiệm	Kết quả dự kiến	Kết quả thực tế
Thêm sản phẩm 1	Nhập thông tin sản phẩm	None	1. Mở modal thêm sản phẩm 2. Nhập thông tin sản phẩm 3. Nhấn nút submit	- Lưu thành công thì hiển thị thông báo - Hiển thị sản phẩm vừa thêm lên danh sách.	Pass
Thêm sản phẩm 2	Không nhập thông tin sản phẩm	None	1. Mở modal thêm sản phẩm 2. Để trống các trường thông tin 3. Nhấn nút submit	- Hiển thị thông báo lưu không thành công - Hiển thị validate trên modal thêm sản phẩm.	Fail
Thêm sản phẩm 3	Nhập thiếu thông tin sản phẩm	None	1. Mở modal thêm sản phẩm 2. Nhập thiếu 1 trường thông tin sản phẩm. 3. Nhấn nút submit	- Hiển thị thông báo lưu không thành công - Hiển thị validate trên modal thêm sản phẩm.	Fail

❖ Kiểm thử chức năng cập nhật sản phẩm

Bảng 4. 7: Kiểm thử chức năng cập nhật sản phẩm

ID	Trường hợp thử nghiệm	Tiền điều kiện	Thủ tục trường hợp thử nghiệm	Kết quả dự kiến	Kết quả thực tế
Cập nhật sản phẩm 1	Cập nhật thông tin sản phẩm các trường không để trống.	None	1. Mở trang modal cập nhật sản phẩm 2. Chỉnh sửa thông sản phẩm 3. Nhấn nút submit	- Lưu thành công thì hiển thị thông báo - Hiển thị sản phẩm vừa cập nhật lên danh sách.	Pass
Cập nhật sản phẩm 2	Cập nhật thông tin sản phẩm để trống các trường thông tin.	None	1. Mở modal cập nhật sản phẩm 2. Để trống các trường thông tin 3. Nhấn nút submit	- Hiển thị thông báo lưu không thành công - Hiển thị validate trên modal thêm tin.	Fail
Cập nhật sản phẩm 3	Nhập thiếu thông tin sản phẩm	None	1. Mở modal cập nhật sản phẩm 2. Nhập thiếu 1 trường thông tin sản phẩm. 3. Nhấn nút submit	- Hiển thị thông báo lưu không thành công - Hiển thị validate trên modal thêm tin.	Fail

❖ Kiểm thử chức năng xóa sản phẩm

Bảng 4. 8: Kiểm thử chức năng xóa sản phẩm

ID	Trường hợp thử nghiệm	Tiền điều kiện	Thủ tục trường hợp thử nghiệm	Kết quả dự kiến	Kết quả thực tế
Xóa sản phẩm 1	Yêu cầu xóa sản phẩm	None	1. Hiện thị modal thông báo xác nhận xóa sản phẩm 2. Nhấn ok	- Hiện thị thông báo đã xóa sản phẩm - Xóa sản phẩm trên danh sách.	Pass
Xóa tin 2	Cập nhật thông tin sản phẩm để trống các trường thông tin.	None	1. Hiện thị modal thông báo xác nhận xóa sản phẩm 2. Nhấn cancel	- Hiện thị thông báo chưa xóa sản phẩm - Không xóa sản phẩm trên danh sách.	Pass

❖ Kiểm thử chức năng tìm kiếm sản phẩm

Bảng 4. 9: Kiểm thử chức năng tìm kiếm

ID	Trường hợp thử nghiệm	Tiền điều kiện	Thủ tục trường hợp thử nghiệm	Kết quả dự kiến	Kết quả thực tế
Tìm kiếm sản phẩm	Nhập từ khóa và yêu cầu tìm kiếm.	None	1. Tìm kiếm sản phẩm	- Hiện thị sản phẩm cần tìm lên danh sách.	Pass

KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

Tổng kết kết quả

Trong quá trình thực hiện dự án, em đã hoàn thành xây dựng một website bán hàng, cụ thể là bán đồ nội thất cho cửa hàng Furni, bao gồm cả hệ thống quản lý nội dung và phân hệ dành cho người dùng. Qua việc phát triển website này, em đã hiểu rõ được kiến thức về lập trình và tăng cường kỹ năng thực hành, cụ thể là:

- Hiểu rõ quy trình và các chức năng cần thiết cho một website thương mại điện tử như: quản lý sản phẩm, người dùng, thống kê doanh thu và đơn hàng.
- Phát triển và tối ưu hóa mã nguồn, cũng như hiểu cách thực hiện các công việc liên quan đến quản lý nhập/xuất kho và thống kê.

Giới hạn

Tuy nhiên, với kiến thức và kinh nghiệm hiện tại, em nhận thức rằng vẫn còn nhiều hạn chế cần khắc phục:

- Không thể tối ưu hóa toàn bộ chức năng của website.
- Một số chức năng chưa hoàn thiện do khả năng lập trình và kinh nghiệm còn hạn chế.

Hướng phát triển:

Để cải thiện những hạn chế này, hướng phát triển của em trong thời gian tới gồm:

- Hoàn thiện các chức năng chưa hoàn chỉnh và tối ưu hóa chất lượng mã nguồn cho website.
- Tiếp tục học hỏi, nâng cao kiến thức và kỹ năng lập trình để có thể đạt được mục tiêu đề ra.

Cuối cùng, việc thực hiện dự án này không chỉ giúp em cải thiện và phát triển kỹ năng lập trình, mà còn giúp em hiểu rõ hơn về quy trình hoạt động của một website thương mại điện tử. Em hy vọng rằng với những kinh nghiệm đã học được, tôi sẽ tiếp tục phát triển và nâng cao kỹ năng của mình trong tương lai.

TÀI LIỆU THAM KHẢO

- [1] Chu Thị Minh Huệ, *Phân tích thiết kế hướng đối tượng với UML*, Khoa Công nghệ thông tin, Đại học Sư Phạm Kỹ Thuật Hưng Yên.
- [2] Chu Thị Minh Huệ, *công nghệ phần mềm*, Khoa Công nghệ thông tin, Đại học Sư Phạm Kỹ Thuật Hưng Yên.
- [3] Ngô Thanh Huyền, *chuyên đề phân tích thiết kế thuật toán*, Khoa Công nghệ thông tin, Đại học Sư Phạm Kỹ Thuật Hưng Yên.
- [4] Nguyễn Hữu Đông, *Lập trình Web API*, Khoa Công nghệ thông tin, Đại học Sư Phạm Kỹ Thuật Hưng Yên.
- [5] Vũ Xuân Thắng, *Chuyên đề Công nghệ web và ứng dụng*, Khoa Công nghệ thông tin, Đại học Sư Phạm Kỹ Thuật Hưng Yên.
- [6] Website: <https://angular.io/>