

HỌ VÀ TÊN SINH VIÊN: BÙI ĐÌNH CƯỜNG

NGÀNH: KỸ THUẬT PHẦN MỀM

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

ĐỒ ÁN ĐẠI HỌC KỸ THUẬT PHẦN MỀM

XÂY DỰNG WEBSITE THƯƠNG MẠI ĐIỆN TỬ
NEST SỬ DỤNG CÔNG NGHỆ LARAVEL FRAMEWORK

Giảng viên hướng dẫn: TS. Lê Xuân Hùng

Lớp: KTPM04 Khóa: K16

Sinh viên thực hiện: Bùi Đình Cường

Mã số sinh viên: 2021608308

Hà Nội – Năm 2024

BỘ CÔNG THƯƠNG
TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI

ĐỒ ÁN ĐẠI HỌC KỸ THUẬT PHẦN MỀM

XÂY DỰNG WEBSITE THƯƠNG MẠI ĐIỆN TỬ
NEST SỬ DỤNG CÔNG NGHỆ LARAVEL FRAMEWORK

Giảng viên hướng dẫn: TS. Lê Xuân Hùng
Lớp: KTPM04 Khóa: K16
Sinh viên thực hiện: Bùi Đình Cường
Mã số sinh viên: 2021608308

Hà Nội – Năm 2024

LỜI CẢM ƠN

Sau quá trình học tập và rèn luyện tại Khoa Công nghệ thông tin – Trường Đại học Công nghiệp Hà Nội, bản thân em đã được các thầy cô truyền đạt, cung cấp và trang bị những kiến thức cơ bản và chuyên sâu về công nghệ thông tin. Nhờ đó, em có thể hiểu sâu hơn về những hoạt động hệ thống thương mại điện tử, đồng thời cũng góp phần bồi dưỡng khả năng tư duy, xử lý vấn đề của bản thân. Tuy nhiên, thực tế đôi khi không hoàn toàn giống như lý thuyết. Do đó, em nhận thấy quá trình thực tập thực tế là cơ hội để khám phá sự khác nhau, giúp em biết cách ứng dụng những gì đã học tại nhà trường và tích lũy kinh nghiệm cũng như những kỹ năng nghề nghiệp để có thể phát triển công việc sau này.

Em xin gửi lời biết ơn sâu sắc đến thầy giáo – **T.S Lê Xuân Hùng**, người hướng dẫn trực tiếp và đóng góp ý kiến giúp em hoàn thành đề tài này.

Đề tài đồ án tốt nghiệp lần này là đúc kết kiến thức, kinh nghiệm mà em nhận được dưới sự truyền đạt của các thầy cô giáo trong suốt bốn năm đại học. Trong thời gian 9 tuần thực hiện đồ án tốt nghiệp, em đã hoàn thành tất cả các mục tiêu được đặt ra trong phiếu giao đề tài cũng như những yêu cầu từ giảng viên hướng dẫn. Em mong muốn nhận được sự góp ý của thầy cô để đề tài thêm phần hoàn thiện.

Em xin chân thành cảm ơn!

Sinh viên thực hiện

Bùi Đình Cường

MỤC LỤC

DANH MỤC CÁC TỪ VIẾT TẮT.....	v
DANH MỤC CÁC HÌNH VẼ.....	vi
DANH MỤC CÁC BẢNG	viii
1. Tên đề tài.....	1
2. Lý do chọn đề tài.	1
3. Mục tiêu đề tài.	2
4. Đối tượng và phạm vi.	2
5. Nội dung nghiên cứu.....	2
6. Kết quả mong muốn.....	4
7. Bố cục đề tài.	5
CHƯƠNG 1 – CƠ SỞ LÝ THUYẾT VÀ KHẢO SÁT BÀI TOÁN.....	6
1.1. Cơ sở lý thuyết.....	6
1.1.1. Tổng quan về Laravel Framework.	6
1.1.2. Tổng quan về mô hình MVC.....	10
1.1.3. Tổng quan về Ajax.	15
1.1.4. Tổng quan về SCSS.....	17
1.1.5. Tổng quan về Bootstrap 5.	19
1.2. Khảo sát bài toán.....	21
1.2.1. Giới thiệu bài toán.	21
1.2.2. Hệ thống thực đang vận hành.....	22
1.2.3. Khảo sát yêu cầu người dùng.	23
CHƯƠNG 2 – PHÂN TÍCH THIẾT KẾ HỆ THỐNG.....	24
2.1. Giới thiệu về hệ thống.	24
2.1.1. Mô tả bài toán.....	24
2.1.2. Quy trình xây dựng hệ thống.....	24
2.1.3. Kết quả mong muốn đạt được.	26
2.2. Phân tích yêu cầu.	26
2.2.1. Yêu cầu về hệ thống.	26
2.2.2. Yêu cầu phía người dùng.	28
2.2.3. Vai trò của người dùng.....	28
2.2.4. Yêu cầu chức năng.	29
2.2.5. Yêu cầu phi chức năng.	30

2.2.6. Tài liệu đặc tả yêu cầu.....	30
2.3. Phân tích hệ thống.	32
2.3.1. Mô hình hóa chức năng hệ thống.	32
2.3.2. Mô tả chi tiết use case phía khách hàng.	33
2.3.3. Mô tả chi tiết use case phía nhà cung cấp.	37
2.3.4. Mô tả chi tiết use case phía quản trị viên.	40
2.3.5. Biểu đồ trình tự và biểu đồ lớp phân tích.....	43
2.3.6. Thiết kế hệ thống cơ sở dữ liệu.	48
2.3.7. Thiết kế giao diện hệ thống.	49
CHƯƠNG 3 – CÀI ĐẶT HỆ THỐNG VÀ KẾT QUẢ ĐẠT ĐƯỢC	53
3.1. Kết quả đạt được.	53
3.1.1. Giao diện phía khách hàng (customer).....	53
3.1.2. Giao diện phía nhà cung cấp (supplier).....	55
3.1.3. Giao diện phía quản trị viên (admin).	57
3.2. Kế hoạch kiểm thử.	59
3.2.1. Giới thiệu.....	59
3.2.2. Tài liệu tham khảo.....	60
3.2.3. Lịch trình công việc.....	61
3.2.4. Những yêu cầu về tài nguyên.....	61
3.2.5. Phạm vi kiểm thử.	62
3.2.6. Chiến lược kiểm thử.....	62
3.2.7. Đánh giá rủi ro.....	64
3.2.8. Điều kiện chấp nhận kiểm thử.....	65
3.2.9. Theo dõi lỗi.	65
3.3. Trường hợp kiểm thử.....	66
3.3.1. Kiểm thử chức năng.	66
3.3.2. Kiểm thử phi chức năng.	69
3.3.3. Kiểm thử giao diện.....	71
3.4. Báo cáo kiểm thử.	73
3.4.1. Thông tin chung.	73
3.4.2. Mục tiêu.....	74
3.4.3. Phạm vi.....	74
3.4.4. Tổng quan kết quả kiểm thử.....	75
3.4.5. Chi tiết kết quả kiểm thử.	75

3.4.6. Kết luận.	78
KẾT LUẬN	80
TÀI LIỆU THAM KHẢO	82
PHỤ LỤC	83

DANH MỤC CÁC TỪ VIẾT TẮT

STT	Từ viết tắt	Giải thích
1	HTTP	HyperText Transfer Protocol
2	HTML	Hyper Text Markup Language
3	JSON	JavaScript Object Notation
4	XML	Extensible Markup Language
5	MVC	Model – View – Controller
6	UC	Use case
7	CSDL	Cơ sở dữ liệu

DANH MỤC CÁC HÌNH VẼ

Hình 1. 1. Laravel Framework.	6
Hình 1. 2. Mô hình MVC.	11
Hình 1. 3. Ajax.	15
Hình 1. 4. SCSS.	17
Hình 1. 5. Bootstrap 5.	19
Hình 2. 1. Biểu đồ Usecase.	32
Hình 2. 2. Biểu đồ trình tự UC <Tìm kiếm>.	43
Hình 2. 3. Biểu đồ lớp phân tích UC <Tìm kiếm>.	43
Hình 2. 4. Biểu đồ trình tự UC <Quản lý giỏ hàng>.	44
Hình 2. 5. Biểu đồ lớp phân tích UC <Quản lý giỏ hàng>.	44
Hình 2. 6. Biểu đồ trình tự UC <Thanh toán>.	45
Hình 2. 7. Biểu đồ lớp phân tích UC <Thanh toán>.	45
Hình 2. 8. Biểu đồ trình tự UC <Quản lý sản phẩm>.	46
Hình 2. 9. Biểu đồ lớp phân tích UC <Quản lý sản phẩm>.	47
Hình 2. 10. Biểu đồ trình tự UC <Thống kê>.	47
Hình 2. 11. Biểu đồ lớp phân tích UC <Thống kê>.	48
Hình 2. 12. Database relationship.	48
Hình 2. 13. Thiết kế màn hình Trang chủ.	49
Hình 2. 14. Thiết kế màn hình Danh mục sản phẩm.	49
Hình 2. 15. Thiết kế màn hình Xem sản phẩm.	50
Hình 2. 16. Thiết kế màn hình giỏ hàng.	50
Hình 2. 17. Thiết kế màn hình Thanh toán.	51
Hình 2. 18. Thiết kế màn hình Quản trị viên.	51
Hình 2. 19. Thiết kế màn hình Quản lý đơn hàng.	52
Hình 3. 1. Màn hình Trang chủ.	53
Hình 3. 2. Màn hình Xem danh mục sản phẩm.	53
Hình 3. 3. Màn hình Xem sản phẩm.	54
Hình 3. 4. Màn hình Giỏ hàng.	54

Hình 3. 5. Màn hình Thanh toán.	55
Hình 3. 6. Màn hình Bảng điều khiển – Nhà cung cấp.	55
Hình 3. 7. Màn hình Quản lý sản phẩm – Nhà cung cấp.	56
Hình 3. 8. Màn hình Trang chủ cửa hàng.	56
Hình 3. 9. Màn hình Quản lý tài khoản.	57
Hình 3. 10. Màn hình Cài đặt.	57
Hình 3. 11. Màn hình Bảng điều khiển – Quản trị viên.	57
Hình 3. 12. Màn hình Thống kê.	58
Hình 3. 13. Màn hình Quản lý sản phẩm – Quản trị viên.	58
Hình 3. 14. Màn hình Quản lý đơn hàng.	58
Hình 3. 15. Màn hình Cài đặt.	59

DANH MỤC CÁC BẢNG

Bảng 2. 1. Phân quyền và yêu cầu chức năng.....	29
Bảng 2. 2. Mô tả UC <Đăng nhập - Khách hàng>.....	33
Bảng 2. 3. Mô tả UC <Đăng ký – Khách hàng>.....	34
Bảng 2. 4. Mô tả UC <Tìm kiếm>.	34
Bảng 2. 5. Mô tả UC <Xem sản phẩm>.....	35
Bảng 2. 6. Mô tả UC <Quản lý giỏ hàng>.	36
Bảng 2. 7. Mô tả UC <Thanh toán>.....	36
Bảng 2. 8. Mô tả UC < Đăng nhập – Nhà cung cấp>.	37
Bảng 2. 9. Mô tả UC < Đăng ký – Nhà cung cấp>.....	38
Bảng 2. 10. Mô tả UC <Thêm sản phẩm – Nhà cung cấp>.	39
Bảng 2. 11. Mô tả UC <Thống kê – Nhà cung cấp>.	39
Bảng 2. 12. Mô tả UC <Đăng nhập – Quản trị viên>.	40
Bảng 2. 13. Mô tả UC <Thống kê – Quản trị viên>.	41
Bảng 2. 14. Mô tả UC <Quản lý sản phẩm>.....	41
Bảng 2. 15. Mô tả UC <Quản lý người dùng>.....	42
Bảng 2. 16. Mô tả UC <User – Quản lý người dùng>.....	42
Bảng 3. 1. Bảng ghi nhận thay đổi tài liệu.	59
Bảng 3. 2. Định nghĩa và các từ viết tắt.	60
Bảng 3. 3. Tài liệu tham khảo.	60
Bảng 3. 4. Lịch trình công việc.	61
Bảng 3. 5. Yêu cầu về phần cứng.....	61
Bảng 3. 6. Yêu cầu về phần mềm.....	61
Bảng 3. 7. Công cụ kiểm thử.....	61
Bảng 3. 8. Nhân sự kiểm thử.....	62
Bảng 3. 9. Kiểm thử chức năng.....	63
Bảng 3. 10. Kiểm thử giao diện.	63
Bảng 3. 11. Các mức đánh giá rủi ro.....	64
Bảng 3. 12. Các mức phân loại lỗi.	65

Bảng 3. 13. Kiểm thử chức năng.....	66
Bảng 3. 14. Kiểm thử phi chức năng.	69
Bảng 3. 15. Kiểm thử giao diện.	71
Bảng 3. 16. Tổng quan kết quả kiểm thử.	75
Bảng 3. 17. Chi tiết kết quả kiểm thử.	75

MỞ ĐẦU

1. Tên đề tài.

“Xây dựng website thương mại điện tử Nest sử dụng công nghệ Laravel Framework.”

2. Lý do chọn đề tài.

Trong thời đại số không ngừng phát triển như hiện nay, ngành công nghệ thông tin đã có những bước tiến vượt bậc để tiến sâu hơn vào trong nhu cầu hằng ngày của đại đa số người dùng. Công nghệ đã, đang và sẽ được áp dụng ngày một cụ thể hơn vào các lĩnh vực của đời sống nhằm mục đích tạo ra những giá trị thực sự và nâng cao chất lượng dịch vụ.

Đi cùng với tốc độ phát triển chóng mặt của thời kỳ 4.0 là khối lượng công việc khổng lồ thường ngày. Từ đó, những giải pháp hữu hiệu có thể giúp người dùng tối ưu thời gian được nhiều doanh nghiệp, tổ chức đưa vào nghiên cứu và áp dụng. Vì thế những doanh nghiệp, tổ chức hay kể cả người tiêu dùng đều mong muốn những thứ tối ưu có thể xử lý được trên những nền tảng công nghệ - nơi mà mọi người có thể ngồi tại một địa điểm cụ thể nhưng vẫn thực hiện được các tác vụ mong muốn từ xa.

Việc lựa chọn đề tài ***“Xây dựng website thương mại điện tử Nest sử dụng công nghệ Laravel Framework”*** cũng mở ra một loạt các cơ hội thú vị trong việc thử nghiệm và áp dụng các công nghệ mới. Laravel không chỉ là một framework phổ biến mà còn được cộng đồng phát triển liên tục, điều này đồng nghĩa với việc sử dụng các tính năng và công nghệ mới nhất trong quá trình phát triển. Điều này có thể bao gồm việc tích hợp các API thanh toán mới, áp dụng các kỹ thuật xử lý dữ liệu tiên tiến, hoặc thử nghiệm các cách tiếp cận mới trong việc tối ưu hóa trải nghiệm người dùng. Điều này không chỉ làm phong phú thêm cho kiến thức và kỹ năng của tác giả mà còn tạo ra một sản phẩm cuối cùng với hiệu suất và tính năng đáp ứng đến các tiêu chuẩn và yêu cầu mới nhất của thị trường. Do đó, việc lựa chọn đề tài này không chỉ là một bước đi đúng đắn mà còn là một cơ hội để khám phá và đóng góp vào sự phát triển của cộng đồng công nghệ.

3. Mục tiêu đề tài.

Đề tài “*Xây dựng website thương mại điện tử Nest sử dụng công nghệ Laravel Framework*” được xây dựng dựa trên những mục tiêu sau:

- Hiểu biết những kiến thức cần thiết và xây dựng giao diện trang web, phân tích thiết kế hệ thống, cơ sở dữ liệu.
- Nắm được những kiến thức cơ bản về PHP, Laravel Framework, MySQL, mô hình MVC,... cùng các kiến thức và công cụ liên quan.
- Cần dựa trên những tiêu chí, khảo sát từ khách hàng để xây dựng nên hệ thống và chức năng.
- Vượt qua giai đoạn xây dựng, kiểm thử để có thể đi vào hoạt động.

4. Đối tượng và phạm vi.

- Công nghệ phát triển phần mềm: Laravel Framework.
- Mô hình MVC.
- Ngôn ngữ: PHP.
- Công cụ:
 - + Visual Studio Code: Xây dựng chương trình.
 - + Laragon: Chạy chương trình.
 - + Case Studio 2: Vẽ biểu đồ trình tự.
 - + Rational Rose: Vẽ biểu đồ use case.
 - + Figma: Thiết kế giao diện hệ thống.

5. Nội dung nghiên cứu.

- Giới thiệu:
 - + Mục tiêu của đề tài: Giới thiệu về mục tiêu nghiên cứu, tầm quan trọng và lý do lựa chọn đề tài.
 - + Phạm vi nghiên cứu: Xác định phạm vi và giới hạn của đề tài.
- Tổng quan về thương mại điện tử và Laravel Framework:
 - + Khái niệm về thương mại điện tử: Định nghĩa, lợi ích, và xu hướng phát triển của thương mại điện tử.

- + Tổng quan về Laravel Framework: Giới thiệu về Laravel, các tính năng và ưu điểm của Laravel trong việc phát triển ứng dụng web.

- Phân tích yêu cầu hệ thống:

- + Phân tích yêu cầu chức năng: Xác định các chức năng chính của hệ thống (quản lý sản phẩm, quản lý nhà cung cấp, quản lý đơn hàng, thanh toán, vận chuyển, v.v.).

- + Phân tích yêu cầu phi chức năng: Xác định các yêu cầu về hiệu năng, bảo mật, khả năng mở rộng và giao diện người dùng.

- Thiết kế hệ thống:

- + Kiến trúc hệ thống: Mô tả kiến trúc tổng thể của hệ thống, bao gồm cả mô hình MVC (Model-View-Controller) trong Laravel.

- + Thiết kế cơ sở dữ liệu: Thiết kế các bảng, mối quan hệ giữa các bảng và các trường dữ liệu chính.

- + Thiết kế giao diện người dùng: Mô tả các giao diện chính của hệ thống, bao gồm trang chủ, trang sản phẩm, trang giỏ hàng, trang thanh toán, v.v.

- Triển khai hệ thống:

- + Cài đặt môi trường phát triển: Hướng dẫn cài đặt Laravel, PHP, MySQL và các công cụ cần thiết khác.

- + Xây dựng các tính năng chính: Mô tả chi tiết quá trình phát triển các chức năng chính của hệ thống.

- + Tích hợp hệ thống: Mô tả quá trình tích hợp các thành phần của hệ thống, bao gồm cả tích hợp các dịch vụ bên thứ ba (ví dụ: dịch vụ thanh toán trực tuyến).

- Kiểm thử và đánh giá hệ thống:

- + Kế hoạch kiểm thử: Xây dựng kế hoạch kiểm thử bao gồm kiểm thử chức năng, kiểm thử giao diện.

- + Thực hiện kiểm thử: Thực hiện các bài kiểm thử đã lập kế hoạch và ghi nhận kết quả.

+ Đánh giá hệ thống: Đánh giá kết quả kiểm thử, xác định các vấn đề cần khắc phục và đề xuất các cải tiến.

- Kết luận và hướng phát triển: Tóm tắt kết quả đạt được của đề tài, những đóng góp và hạn chế. Đề xuất các hướng phát triển tiếp theo cho hệ thống, bao gồm cả các tính năng mới và cải tiến hiệu năng.

- Tài liệu tham khảo: Liệt kê các tài liệu, sách, bài báo và nguồn tham khảo đã sử dụng trong quá trình nghiên cứu và phát triển hệ thống.

6. Kết quả mong muốn.

- Tuân thủ các tiêu chuẩn phát triển phần mềm: Dự án tuân thủ các tiêu chuẩn phát triển phần mềm như PSR (PHP Standards Recommendations) để đảm bảo mã nguồn sạch sẽ, dễ bảo trì và mở rộng.

- Tài liệu đầy đủ: Dự án cung cấp tài liệu chi tiết bao gồm tài liệu kỹ thuật, hướng dẫn sử dụng và tài liệu cho nhà phát triển, giúp người dùng và các nhà phát triển khác dễ dàng hiểu và sử dụng hệ thống.

- Giao diện thân thiện, đẹp mắt.

- Responsive Design: Đảm bảo giao diện hiển thị tốt trên mọi thiết bị, từ máy tính để bàn, laptop đến điện thoại di động và máy tính bảng.

- Tính nhất quán và đơn giản: Thiết kế giao diện đơn giản, dễ hiểu với các thành phần nhất quán, giúp người dùng dễ dàng tương tác với hệ thống.

- Chủ đề tùy chỉnh: Cho phép người dùng thay đổi chủ đề, màu sắc và bố cục của giao diện theo sở thích cá nhân.

- Chức năng thông minh, dễ dàng sử dụng:

+ Tìm kiếm và gợi ý thông minh: Hệ thống cung cấp chức năng tìm kiếm nhanh chóng và chính xác, cùng với các gợi ý sản phẩm dựa trên lịch sử tìm kiếm và mua sắm của người dùng.

+ Quản lý sản phẩm dễ dàng: Giao diện quản lý sản phẩm đơn giản, cho phép nhà cung cấp thêm, sửa và xóa sản phẩm một cách nhanh chóng.

+ Quản lý đơn hàng và khách hàng hiệu quả: Cung cấp các công cụ quản lý đơn hàng và khách hàng, bao gồm theo dõi trạng thái đơn hàng, quản lý thông tin khách hàng và lịch sử mua hàng.

➔ Bằng cách phát triển và nhấn mạnh các yếu tố trên, đề tài sẽ trở nên toàn diện và chuyên nghiệp hơn, đáp ứng tốt các yêu cầu của một dự án thương mại điện tử hiện đại và thân thiện với người dùng.

7. Bố cục đề tài.

Nội dung bài báo cáo chia thành 3 chương:

- **Chương 1 – Cơ sở lý thuyết và khảo sát bài toán:** Tìm hiểu về những công nghệ sẽ sử dụng trong đề tài. Tiến hành khảo sát, lấy ý kiến thực tế từ người dùng.

- **Chương 2 – Phân tích thiết kế hệ thống:** Mô tả bài toán, phân tích thiết kế, yêu cầu và chức năng của website. Vẽ phác thảo giao diện.

- **Chương 3 – Cài đặt hệ thống và kết quả đạt được:** Trình bày những hình ảnh của website. Kiểm thử hệ thống.

CHƯƠNG 1 – CƠ SỞ LÝ THUYẾT VÀ KHẢO SÁT BÀI TOÁN

1.1. Cơ sở lý thuyết.

1.1.1. Tổng quan về Laravel Framework.

Laravel Framework là một công cụ mạnh mẽ và linh hoạt cho việc phát triển các ứng dụng web. Được ra đời từ năm 2011, Laravel nhanh chóng trở thành một trong những framework phổ biến nhất trong cộng đồng phát triển web. Điều đặc biệt về Laravel là sự dễ sử dụng và tính linh hoạt. Với cú pháp rõ ràng và trực quan, người phát triển có thể nhanh chóng tạo ra các ứng dụng web phức tạp mà không cần phải đối mặt với nhiều rắc rối kỹ thuật.

Một trong những điểm mạnh của Laravel là Eloquent ORM, cho phép tương tác với cơ sở dữ liệu một cách dễ dàng thông qua các đối tượng PHP, giúp giảm thiểu việc viết mã và tăng tính bảo trì của mã nguồn. Ngoài ra, Blade Template Engine của Laravel cung cấp một cách tiếp cận linh hoạt và hiệu quả cho việc phát triển giao diện người dùng, giúp tạo ra các trang web động và hấp dẫn.

Không chỉ là một công cụ phát triển, Laravel còn là một cộng đồng lớn và tích cực, với nhiều tài liệu, hướng dẫn và nguồn tài nguyên mở. Sự hỗ trợ mạnh mẽ từ cộng đồng cùng với các công cụ và dịch vụ hỗ trợ đã giúp Laravel trở thành lựa chọn hàng đầu cho việc phát triển các ứng dụng web hiện đại và mạnh mẽ.



Hình 1. 1. Laravel Framework.

1.1.1.1. Một số tính năng của Laravel Framework.

Laravel Framework có nhiều tính năng mạnh mẽ giúp người phát triển xây dựng các ứng dụng web hiệu quả và linh hoạt. Dưới đây là một số tính năng nổi bật của Laravel:

- Routing: Laravel cung cấp một cách tiếp cận linh hoạt và mạnh mẽ cho việc xử lý routing. Định nghĩa và quản lý các route trong Laravel rất dễ dàng, giúp tạo ra các URL đẹp và dễ đọc.

- Eloquent ORM: Laravel đi kèm với Eloquent ORM, một công cụ mạnh mẽ cho phép tương tác với cơ sở dữ liệu bằng cách sử dụng các đối tượng PHP. Điều này giúp giảm thiểu việc viết các truy vấn SQL thủ công và tạo ra mã nguồn dễ đọc và dễ bảo trì.

- Middleware: Middleware trong Laravel cho phép bạn áp dụng các lớp trung gian (middleware) cho các yêu cầu HTTP. Điều này cho phép kiểm soát quyền truy cập, xác thực, và xử lý các yêu cầu trước khi chúng được chuyển đến các route.

- Blade Template Engine: Blade là một template engine mạnh mẽ trong Laravel, cung cấp các công cụ linh hoạt để phát triển giao diện người dùng. Blade cho phép sử dụng các lệnh điều kiện, vòng lặp, và kế thừa giao diện một cách dễ dàng.

- Authentication và Authorization: Laravel cung cấp các tính năng tích hợp sẵn cho việc xác thực người dùng và kiểm soát quyền truy cập. Việc xây dựng hệ thống đăng nhập và phân quyền trở nên đơn giản với các tính năng được tích hợp sẵn trong framework.

- Caching: Laravel hỗ trợ nhiều phương pháp caching để tăng hiệu suất ứng dụng. Tính năng này cho phép lưu trữ các dữ liệu trung gian hoặc kết quả tính toán trước đó để giảm thiểu thời gian phản hồi của ứng dụng.

- Queues và Background Processing: Laravel cung cấp hỗ trợ cho việc xử lý các tác vụ ở nền (background tasks) thông qua hệ thống queues. Điều này giúp tăng khả năng mở rộng của ứng dụng và cải thiện trải nghiệm người dùng.

➔ Tính năng của Laravel không chỉ dừng lại ở đây, nhưng những tính năng này đã làm nổi bật và giúp Laravel trở thành một trong những framework phát triển web hàng đầu hiện nay.

1.1.1.2. Cấu trúc cơ bản của Laravel Framework.

Cấu trúc cơ bản của Laravel Framework bao gồm các thư mục và tập tin quan trọng sau đây:

- Thư mục “app”: Thư mục này chứa mã nguồn của ứng dụng, bao gồm các Controllers, Models, và Services.
- Thư mục “config”: Chứa các tập tin cấu hình của ứng dụng, như cấu hình database, mail, session, và các cài đặt khác.
- Thư mục “database”: Chứa các tập tin migration để tạo và quản lý cơ sở dữ liệu, cũng như các seeders để điền dữ liệu mẫu vào cơ sở dữ liệu.
- Thư mục “public”: Thư mục này chứa các tập tin như index.php, css, js, hình ảnh và các tài nguyên tĩnh khác được truy cập trực tiếp từ client.
- Thư mục “resources”: Chứa các tài nguyên không phải mã nguồn như các file view, file CSS và Javascript, và các file ngôn ngữ.
- Thư mục “routes”: Chứa các file route để định tuyến các yêu cầu HTTP đến các hành động của ứng dụng.
- Thư mục “storage”: Thư mục này chứa các file cache, sessions, logs và các file được sinh ra bởi ứng dụng.
- Thư mục “tests”: Chứa các file test cho việc kiểm thử ứng dụng.
- Tập tin “artisan”: Tập tin này là một công cụ dòng lệnh để thực hiện các tác vụ như tạo controllers, models, migrations, và chạy các lệnh khác của Laravel.
- Tập tin “.env”: Tập tin cấu hình môi trường, chứa các biến môi trường như cấu hình database, thông tin xác thực, và các cài đặt khác.

➔ Cấu trúc này giúp tổ chức ứng dụng một cách rõ ràng và dễ quản lý, đồng thời cũng làm cho việc phát triển và bảo trì ứng dụng trở nên dễ dàng hơn.

1.1.1.3. Ưu điểm và nhược điểm của Laravel Framework.

Ưu điểm:

Laravel Framework có nhiều ưu điểm đáng chú ý, làm cho nó trở thành một lựa chọn phổ biến cho việc phát triển ứng dụng web:

- **Tính Linh Hoạt và Dễ Sử Dụng:** Laravel cung cấp cú pháp rõ ràng và dễ hiểu, giúp người phát triển dễ dàng làm quen và nhanh chóng bắt đầu phát triển ứng dụng. Các tính năng như routing, database migration, và authentication được xây dựng một cách logic và linh hoạt.

- **Tính Bảo Mật Cao:** Laravel cung cấp các tính năng bảo mật tích hợp sẵn như xác thực người dùng, kiểm soát quyền truy cập, và bảo vệ chống các cuộc tấn công CSRF, giúp bảo vệ ứng dụng khỏi các lỗ hổng bảo mật phổ biến.

- **Tính Mở Rộng và Khả Mở Cao:** Laravel hỗ trợ một cách dễ dàng và linh hoạt cho việc mở rộng tính năng của ứng dụng thông qua việc sử dụng các gói mở rộng (packages) và thư viện bên ngoài. Nó cũng hỗ trợ tích hợp với các công nghệ như Redis, Elasticsearch, và Memcached.

- **Blade Template Engine:** Blade là một template engine mạnh mẽ, cho phép phát triển giao diện người dùng một cách dễ dàng và hiệu quả. Blade cung cấp các tính năng như kế thừa giao diện, các điều kiện và vòng lặp, giúp tạo ra các trang web động và hấp dẫn.

- **Eloquent ORM:** Laravel đi kèm với Eloquent ORM, một công cụ mạnh mẽ cho phép tương tác với cơ sở dữ liệu bằng cách sử dụng các đối tượng PHP. Điều này giúp giảm thiểu lặp lại mã và tạo ra mã nguồn dễ đọc và dễ bảo trì.

- **Cộng Đồng Hỗ Trợ Lớn:** Laravel có một cộng đồng lớn và tích cực, với nhiều tài liệu, hướng dẫn, và nguồn tài nguyên mở. Sự hỗ trợ từ cộng đồng giúp giải quyết các vấn đề nhanh chóng và hiệu quả.

➔ Tóm lại, Laravel Framework không chỉ mang lại sự linh hoạt và dễ sử dụng mà còn có tính bảo mật cao, khả năng mở rộng và một cộng đồng hỗ trợ

manh mẽ. Điều này làm cho Laravel trở thành một lựa chọn hàng đầu cho việc phát triển các ứng dụng web hiện đại và mạnh mẽ.

Nhược điểm:

Mặc dù Laravel Framework có nhiều ưu điểm, nhưng cũng tồn tại một số nhược điểm cần xem xét:

- Học Hỏi Ban Đầu: Mặc dù Laravel có cú pháp dễ hiểu và linh hoạt, nhưng việc học và làm quen ban đầu vẫn có thể đòi hỏi một khoảng thời gian đáng kể đối với những người mới bắt đầu. Điều này có thể gây khó khăn cho các nhà phát triển mới hoặc nhóm không có nhiều kinh nghiệm với framework.

- Hiệu Suất: Mặc dù Laravel cung cấp nhiều tính năng để tối ưu hóa hiệu suất ứng dụng, nhưng đôi khi vẫn có thể gặp phải các vấn đề về hiệu suất đặc biệt là khi xử lý lượng truy cập lớn. Việc tinh chỉnh và tối ưu hiệu suất có thể đòi hỏi kiến thức chuyên sâu và kinh nghiệm kỹ thuật.

- Phiên Bản Mới và Tương Thích Ngược: Khi Laravel phát hành các phiên bản mới, có thể có sự thay đổi trong cú pháp và các tính năng, dẫn đến vấn đề tương thích ngược với các ứng dụng Laravel hiện có. Việc cập nhật lên phiên bản mới có thể đòi hỏi thời gian và công sức để điều chỉnh và kiểm tra tương thích.

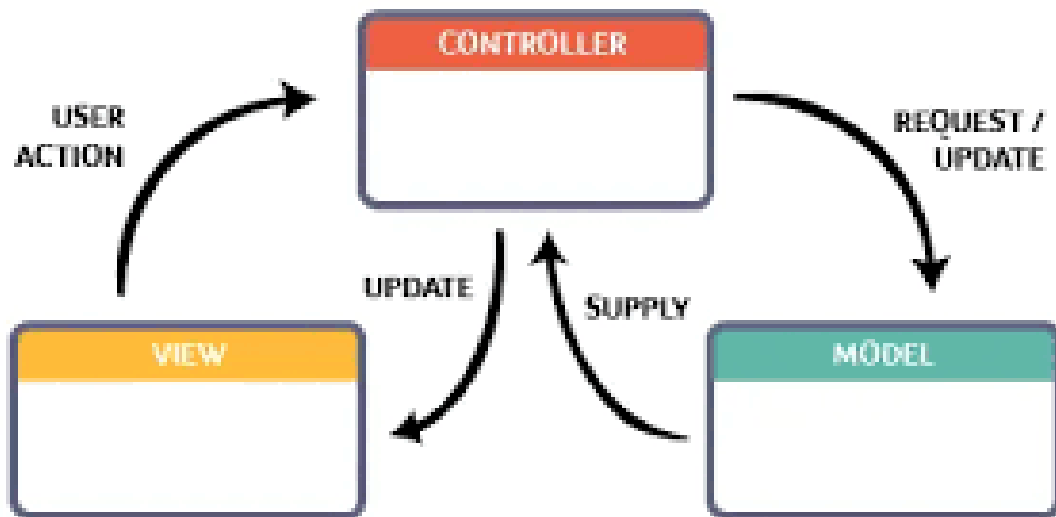
➔ Tóm lại, mặc dù Laravel là một framework mạnh mẽ và phổ biến, nhưng vẫn tồn tại một số nhược điểm cần xem xét khi lựa chọn sử dụng. Điều này cần được cân nhắc và quản lý cẩn thận trong quá trình phát triển và triển khai ứng dụng.

1.1.2. Tổng quan về mô hình MVC.

Mô hình MVC (Model-View-Controller) là một khung kiến trúc quan trọng trong phát triển phần mềm, đặc biệt là trong việc xây dựng các ứng dụng web. Mô hình này được phổ biến bởi sự phân chia rõ ràng và tổ chức logic của ứng dụng thành ba thành phần chính: Model, View và Controller.

Mô hình MVC tách biệt logic kinh doanh, giao diện người dùng và dữ liệu, tạo ra một cách tiếp cận có cấu trúc và tổ chức cho việc phát triển ứng

dụng. Điều này giúp mã nguồn dễ bảo trì, mở rộng và tái sử dụng, đồng thời cung cấp một cách tiếp cận hiệu quả cho việc quản lý luồng điều khiển của ứng dụng. Mô hình MVC đã trở thành một trong những kiến trúc phổ biến nhất được sử dụng trong phát triển phần mềm, đặc biệt là trong lĩnh vực phát triển các ứng dụng web.



Hình 1. 2. Mô hình MVC.

Trong mô hình MVC (Model-View-Controller), mỗi thành phần đều có trách nhiệm riêng biệt và thực hiện các chức năng cụ thể. Dưới đây là mô tả của từng thành phần:

- Model (M):

- + Đại diện cho dữ liệu và logic liên quan đến dữ liệu.
- + Thực hiện các thao tác cơ sở dữ liệu: truy vấn, cập nhật, xóa,...
- + Chứa các phương thức để thực hiện xử lý logic kinh doanh và truy vấn dữ liệu từ cơ sở dữ liệu.
- + Đôi khi có thể bao gồm các lớp hoặc các đối tượng đại diện cho các đối tượng trong cơ sở dữ liệu.

- View (V):

- + Đại diện cho giao diện người dùng (UI) của ứng dụng.
- + Thực hiện việc hiển thị dữ liệu từ model cho người dùng cuối.

- + Không chứa logic kinh doanh phức tạp và thường chỉ chịu trách nhiệm hiển thị dữ liệu theo cách định dạng phù hợp.

- Controller (C):

- + Đại diện cho các hành động và logic điều khiển của ứng dụng.
- + Tiếp nhận các yêu cầu từ người dùng thông qua các route và tương tác với model và view tương ứng.
- + Chịu trách nhiệm xử lý yêu cầu từ người dùng, thực hiện logic kinh doanh và gửi dữ liệu đến view để hiển thị cho người dùng.

Mỗi thành phần trong mô hình MVC làm việc một cách độc lập với nhau, đồng thời tạo ra một môi trường làm việc có tổ chức và dễ bảo trì. Sự phân chia rõ ràng giữa các thành phần giúp giảm sự phụ thuộc giữa chúng, từ đó tăng tính linh hoạt và khả năng mở rộng của ứng dụng.

1.1.2.1. Luồng đi trong mô hình MVC.

Trong mô hình MVC (Model-View-Controller), luồng đi của ứng dụng diễn ra như sau:

- Người dùng tương tác với View (1): Người dùng tương tác với giao diện người dùng của ứng dụng, thường thông qua các trình duyệt web hoặc ứng dụng di động.
- Yêu cầu gửi đến Controller (2): Khi người dùng thực hiện một hành động, chẳng hạn như nhấn vào một nút hoặc điền vào một biểu mẫu, yêu cầu được gửi từ View tới Controller.
- Controller xử lý yêu cầu (3): Controller nhận yêu cầu từ View và quyết định hành động cần thực hiện. Nó có thể gọi các phương thức trong Model để truy vấn hoặc cập nhật dữ liệu, sau đó chuyển kết quả đến View.
- Controller gửi dữ liệu đến View (4): Sau khi xử lý yêu cầu, Controller gửi dữ liệu cần thiết cho View để hiển thị cho người dùng.
- View hiển thị dữ liệu (5): View nhận dữ liệu từ Controller và sử dụng nó để hiển thị giao diện người dùng phản hồi lại cho người dùng.

- Người dùng tương tác tiếp (6): Người dùng có thể tiếp tục tương tác với giao diện người dùng, quá trình này có thể lặp lại từ bước 2 đến bước 5.

➔ Luồng đi trong mô hình MVC giúp tách biệt logic kinh doanh, giao diện người dùng và dữ liệu, giúp mã nguồn dễ bảo trì và mở rộng. Đồng thời, nó cũng cung cấp một cách tiếp cận tổ chức và có cấu trúc cho việc phát triển ứng dụng.

1.1.2.2. Ưu điểm và nhược điểm của mô hình MVC.

Ưu điểm:

- Phân chia rõ ràng các thành phần: MVC phân chia ứng dụng thành ba thành phần chính: Model (mô hình), View (giao diện) và Controller (bộ điều khiển). Điều này giúp dễ dàng quản lý mã nguồn và hiểu rõ cấu trúc của dự án.

- Tính tái sử dụng cao: Nhờ phân chia thành phần rõ ràng, bạn có thể tái sử dụng các phần của ứng dụng một cách dễ dàng. Ví dụ, bạn có thể sử dụng cùng một model cho nhiều giao diện khác nhau.

- Dễ bảo trì: Với việc phân chia rõ ràng, việc bảo trì và cập nhật mã nguồn trở nên đơn giản hơn. Bạn có thể chỉnh sửa một thành phần mà không ảnh hưởng đến các thành phần khác.

- Thích ứng tốt với thay đổi: Khi yêu cầu của dự án thay đổi, việc chỉnh sửa hoặc thay thế một phần của MVC thường dễ dàng hơn so với các kiến trúc khác.

- Phát triển đồng thời: Đội ngũ phát triển có thể làm việc đồng thời trên các phần khác nhau của ứng dụng mà không gây ra xung đột đáng kể.

- Tách biệt logic và giao diện: Việc tách biệt logic ứng dụng (Controller và Model) và giao diện người dùng (View) giúp làm rõ ràng hóa và tăng tính linh hoạt trong việc phát triển và bảo trì ứng dụng.

- Kiểm thử dễ dàng: Với các thành phần được phân chia rõ ràng, việc kiểm thử trở nên dễ dàng hơn và hiệu quả hơn.

Nhược điểm:

- Phức tạp khi dự án nhỏ: Trong các dự án nhỏ, việc triển khai một kiến trúc phức tạp như MVC có thể làm tăng độ phức tạp và làm chậm quá trình phát triển.

- Khó khăn trong việc hiểu và triển khai ban đầu: Đối với những người mới bắt đầu, việc hiểu và triển khai MVC có thể gặp khó khăn do sự phức tạp của cấu trúc và quy tắc.

- Lỗi gắn kết mạnh mẽ giữa các thành phần: Trong MVC, Controller thường liên kết chặt chẽ với View và Model, dẫn đến việc thay đổi một phần có thể ảnh hưởng đến các phần khác mà không dễ dàng kiểm soát được.

1.1.2.3. Ứng dụng mô hình MVC vào lập trình.

Mô hình MVC (Model-View-Controller) là một phương pháp tổ chức mã nguồn trong phát triển phần mềm, và nó có thể được áp dụng vào nhiều ngôn ngữ lập trình và nền tảng khác nhau. Dưới đây là một ví dụ về cách áp dụng MVC trong lập trình web sử dụng ngôn ngữ PHP:

- Model (Mô hình):

+ Model đại diện cho dữ liệu của ứng dụng và các logic xử lý liên quan. Trong một ứng dụng web, Model thường tương tác với cơ sở dữ liệu để lấy và lưu trữ dữ liệu.

+ Ví dụ: Một model trong ứng dụng blog có thể chứa các phương thức để lấy bài viết từ cơ sở dữ liệu, thêm mới bài viết, cập nhật hoặc xóa bài viết.

- View (Giao diện):

+ View là phần hiển thị dữ liệu cho người dùng cuối. Nó có thể là các trang HTML, các mẫu (templates) hoặc các thành phần giao diện khác.

+ Ví dụ: Một view trong ứng dụng blog có thể là một trang web HTML hiển thị danh sách bài viết hoặc một form để thêm mới bài viết.

- Controller (Bộ điều khiển):

+ Controller là trung gian giữa Model và View. Nó nhận yêu cầu từ người dùng thông qua giao diện và xử lý yêu cầu bằng cách gọi các phương thức trong Model, sau đó chuyển dữ liệu đã xử lý đến View để hiển thị.

+ Ví dụ: Một controller trong ứng dụng blog có thể chứa các phương thức để hiển thị danh sách bài viết, thêm mới bài viết, hoặc hiển thị chi tiết của một bài viết cụ thể.

1.1.3. Tổng quan về Ajax.

Ajax (Asynchronous JavaScript and XML) là một kỹ thuật lập trình web cho phép gửi và nhận dữ liệu từ máy chủ mà không cần phải làm mới hoàn toàn trang web. Thay vì gửi yêu cầu đến máy chủ và chờ đợi phản hồi, Ajax cho phép các yêu cầu và phản hồi diễn ra bất đồng bộ, điều này giúp cải thiện trải nghiệm người dùng và tăng hiệu suất của ứng dụng web.



Hình 1. 3. Ajax.

1.1.3.1. Cách thức hoạt động của Ajax.

Gửi yêu cầu:

- Tạo yêu cầu: JavaScript tạo một yêu cầu HTTP, thông thường là sử dụng đối tượng XMLHttpRequest (XHR) hoặc Fetch API. Yêu cầu này có thể là GET, POST, PUT, DELETE hoặc các phương thức HTTP khác.

- Cấu hình yêu cầu: Thiết lập các tham số cho yêu cầu như URL của máy chủ, phương thức HTTP, dữ liệu cần gửi (nếu có), và các cài đặt khác như headers.

- Gửi yêu cầu: JavaScript gửi yêu cầu HTTP đến máy chủ thông qua đối tượng XMLHttpRequest hoặc Fetch API. Yêu cầu này được gửi đi không đồng bộ, nghĩa là trang web không chờ đợi phản hồi từ máy chủ mà tiếp tục thực hiện các tác vụ khác.

Xử lý yêu cầu tại máy chủ:

- Tiếp nhận yêu cầu: Máy chủ nhận yêu cầu từ trình duyệt web và xử lý yêu cầu theo các logic được định nghĩa trong ứng dụng web.
- Xử lý yêu cầu: Máy chủ thực hiện các xử lý như truy vấn cơ sở dữ liệu, xử lý logic kinh doanh, hoặc lấy dữ liệu từ các nguồn khác.
- Tạo phản hồi: Sau khi xử lý yêu cầu, máy chủ tạo ra một phản hồi HTTP chứa dữ liệu được yêu cầu hoặc thông báo lỗi (nếu có). Dữ liệu này thường được trả về dưới dạng JSON, XML hoặc HTML.

Xử lý phản hồi:

- Nhận phản hồi: Trình duyệt web nhận phản hồi từ máy chủ khi nó được trả về.
- Xử lý phản hồi: JavaScript xử lý phản hồi từ máy chủ. Điều này có thể bao gồm việc hiển thị dữ liệu lên trang web, cập nhật giao diện người dùng, hoặc thực hiện các hành động khác dựa trên dữ liệu nhận được.

1.1.3.2. Ưu điểm và nhược điểm của Ajax.**Ưu điểm:**

- Trải nghiệm người dùng tốt hơn: Giảm thời gian phản hồi và làm mới trang web.
- Tăng hiệu suất: Giảm lượng dữ liệu truyền qua mạng và tải lại trang.
- Tiết kiệm băng thông: Giảm lượng dữ liệu truyền qua mạng.
- Tính tương tác cao: Cho phép tương tác với máy chủ mà không làm mất trạng thái của trang web.

Nhược điểm:

- Khó khăn trong việc xử lý lỗi: Xử lý lỗi trong Ajax có thể phức tạp hơn do việc quản lý các yêu cầu và phản hồi không đồng bộ.
- Yêu cầu tăng tải cho máy chủ: Ajax có thể gây tăng tải cho máy chủ nếu không được quản lý cẩn thận.

1.1.4. Tổng quan về SCSS.

SCSS (Sassy CSS) là một ngôn ngữ mở rộng của CSS có cú pháp tương tự như CSS thông thường nhưng với nhiều tính năng mở rộng và linh hoạt hơn



Hình 1. 4. SCSS.

1.1.4.1. Tính mở rộng của SCSS.

- Biến: SCSS cho phép định nghĩa và sử dụng biến trong mã nguồn CSS, giúp tái sử dụng mã nguồn và dễ dàng thay đổi các giá trị màu sắc, kích thước, v.v.
- Nested Syntax: SCSS cho phép viết CSS dạng lồng nhau, giúp tạo ra cấu trúc cây dễ đọc và hiểu hơn.
- Mixins: Mixin là một khối mã có thể tái sử dụng trong SCSS, giúp viết lại các đoạn mã CSS phức tạp một cách ngắn gọn.
- Sass Functions và Operators: SCSS hỗ trợ các hàm và toán tử, giúp tính toán các giá trị và thực hiện các thao tác phức tạp trong mã CSS.
- Importing: SCSS cho phép import các file CSS và SCSS khác vào trong một file SCSS, giúp phân chia và quản lý mã nguồn một cách dễ dàng.

- Extend/Inheritance: SCSS cung cấp tính năng kế thừa, cho phép một lớp CSS kế thừa các thuộc tính từ một lớp khác, giúp giảm lặp lại mã nguồn.

1.1.4.2. Biên dịch SCSS.

- Sử dụng Command Line:

+ Sass Command Line Interface (CLI): Cài đặt Sass CLI thông qua npm (Node Package Manager): **npm install -g sass**.

+ Sử dụng lệnh sass để biên dịch SCSS thành CSS: **sass input.scss output.css**.

+ Cài đặt node-sass thông qua npm: **npm install -g node-sass**.

+ SCSS thành CSS: **node-sass input.scss -ooutput.css**.

- Sử dụng Gulp hoặc Grunt: Có thể sử dụng các công cụ quản lý nhiệm vụ như Gulp hoặc Grunt để tự động biên dịch SCSS thành CSS và thực hiện nhiều tác vụ khác trong quy trình phát triển.

1.1.4.3. Ưu điểm và nhược điểm của SCSS.

SCSS (Sassy CSS) là một ngôn ngữ mở rộng của CSS, và như mọi công nghệ khác, nó có những ưu và nhược điểm riêng.

Ưu điểm:

- Dễ hiểu và sử dụng: SCSS sử dụng cú pháp tương tự CSS, do đó dễ dàng học và triển khai cho những người đã quen với CSS.

- Tính mở rộng và linh hoạt: SCSS cung cấp nhiều tính năng mạnh mẽ như biến, nested syntax, mixins, functions, và extend/inheritance, giúp tạo ra mã nguồn CSS linh hoạt và dễ bảo trì.

- Tái sử dụng mã nguồn: Các tính năng như biến và mixins giúp tái sử dụng mã nguồn CSS, giảm thiểu lặp lại mã nguồn và tăng hiệu suất phát triển.

- Cộng đồng lớn và hỗ trợ tốt: SCSS được sử dụng rộng rãi và có một cộng đồng lớn, điều này có nghĩa là có nhiều tài liệu, ví dụ, và hỗ trợ trực tuyến có sẵn để giúp bạn khi cần.

- Hỗ trợ tốt cho Quản lý dự án: SCSS cho phép bạn chia nhỏ và tổ chức mã nguồn CSS thành các module riêng biệt, giúp quản lý dự án dễ dàng hơn.

Nhược điểm:

- Tăng thêm bước biên dịch: Để sử dụng SCSS, bạn cần phải biên dịch mã nguồn SCSS thành CSS trước khi triển khai, điều này làm tăng thêm một bước trong quy trình phát triển.

- Tăng kích thước tệp tin: Do SCSS chứa nhiều tính năng và cú pháp mở rộng, các tệp tin SCSS có thể lớn hơn so với tệp tin CSS tương đương.

- Có thể phức tạp khi sử dụng sai: Mặc dù SCSS cung cấp nhiều tính năng mạnh mẽ, nhưng nếu không sử dụng đúng cách, có thể dẫn đến mã nguồn phức tạp và khó bảo trì.

➔ SCSS là một công cụ mạnh mẽ và linh hoạt cho việc viết CSS, nhưng cũng có nhược điểm riêng cần được xem xét khi lựa chọn và triển khai.

1.1.5. Tổng quan về Bootstrap 5.

Bootstrap 5 là một framework front-end mạnh mẽ và phổ biến được sử dụng để xây dựng các giao diện web thân thiện với người dùng và dễ bảo trì. Với Bootstrap 5, việc phát triển các trang web đáp ứng và hiện đại trở nên dễ dàng hơn bao giờ hết. Với sự hỗ trợ của CSS Grid và Flexbox, Bootstrap 5 cho phép bạn tạo ra các bố cục linh hoạt và tinh tế. Nó cung cấp một loạt các thành phần và giao diện người dùng sẵn có như nút, biểu mẫu, thẻ, thanh điều hướng và nhiều hơn nữa, giúp giảm thời gian phát triển và tăng hiệu quả.



Hình 1. 5. Bootstrap 5.

1.1.5.1. Tính năng của Bootstrap 5.

- Thiết kế đáp ứng: Bootstrap 5 hỗ trợ thiết kế đáp ứng, giúp các trang web tự động thích ứng và hiển thị tốt trên các thiết bị khác nhau như máy tính, máy tính bảng và điện thoại di động.

- CSS Grid và Flexbox: Bootstrap 5 sử dụng CSS Grid và Flexbox để quản lý cấu trúc trang và bố cục, giúp tạo ra các giao diện linh hoạt và dễ bảo trì.

- Các thành phần và giao diện người dùng: Bootstrap 5 đi kèm với một loạt các thành phần và giao diện người dùng sẵn có như nút, biểu mẫu, thẻ, thanh điều hướng, tiêu đề, chân trang và nhiều hơn nữa, giúp giảm thời gian phát triển và tăng hiệu quả.

- Tích hợp JavaScript: Bootstrap 5 cung cấp các plugin JavaScript tích hợp sẵn như Carousel, Modal, Tooltip, Popover, Collapse và Scrollspy để tạo ra các tính năng tương tác động trên trang web.

- Tùy chỉnh dễ dàng: Bootstrap 5 cho phép tùy chỉnh dễ dàng thông qua việc điều chỉnh biến Sass, và bạn cũng có thể chọn lọc các thành phần cần sử dụng để giảm kích thước của tệp CSS.

- Tương thích với trình duyệt: Bootstrap 5 tương thích với hầu hết các trình duyệt web phổ biến như Chrome, Firefox, Safari, Edge và Internet Explorer 11+.

- Tốc độ và hiệu suất: Bootstrap 5 tối ưu hóa để tối ưu hóa hiệu suất và tốc độ tải trang, giúp tăng trải nghiệm người dùng và tối ưu hóa SEO.

1.1.5.2. Ưu điểm và nhược điểm của Bootstrap 5.

Ưu điểm:

- Tốc độ phát triển nhanh chóng: Bootstrap 5 cung cấp một loạt các thành phần và giao diện người dùng sẵn có, giúp giảm thời gian và công sức cần thiết để xây dựng giao diện web từ đầu.

- Thiết kế đáp ứng: Framework này hỗ trợ thiết kế đáp ứng, giúp các trang web tự động thích ứng và hiển thị tốt trên các thiết bị khác nhau.

- Tính linh hoạt: Bootstrap 5 cho phép tùy chỉnh dễ dàng thông qua việc điều chỉnh biến Sass và chọn lọc các thành phần cần sử dụng, giúp giảm kích thước của tệp CSS và tối ưu hóa hiệu suất.

Nhược điểm:

- Trang bị khá nặng: Bootstrap 5 cung cấp một loạt các tính năng và giao diện sẵn có, dẫn đến tệp CSS lớn và trang bị khá nặng, đặc biệt khi chỉ sử dụng một phần nhỏ của chúng.

- Giao diện trang web đồng nhất: Do Bootstrap 5 được sử dụng rộng rãi, nhiều trang web có thể có cảm giác giống nhau hoặc một lượng lớn giao diện web có thể trở nên quá giống nhau.

- Khó tùy chỉnh chi tiết: Mặc dù Bootstrap 5 cho phép tùy chỉnh dễ dàng, nhưng có thể khó để điều chỉnh các phần tử nhỏ hoặc thay đổi hoàn toàn giao diện mà không làm ảnh hưởng đến cấu trúc của framework.

1.2. Khảo sát bài toán.

1.2.1. Giới thiệu bài toán.

Bài toán này tập trung vào việc xây dựng một trang web thương mại điện tử (multi-vendor e-commerce platform) với tên gọi Nest và sử dụng công nghệ Laravel Framework. Dự án được phác thảo một cách tổng quan như sau.

Tổng quan về dự án:

- Tên dự án: Nest - Website thương mại điện tử.
- Mô tả: Nest là một nền tảng thương mại điện tử cho phép nhiều nhà cung cấp đăng ký và bán sản phẩm của họ trên cùng một trang web. Khách hàng có thể tìm kiếm, xem và mua các sản phẩm từ các nhà cung cấp khác nhau.

- Chức năng cơ bản:

- + Đăng ký và đăng nhập cho người dùng.
- + Hiển thị danh sách sản phẩm từ nhiều nhà cung cấp khác nhau.
- + Tìm kiếm và lọc sản phẩm.
- + Xem chi tiết sản phẩm và thêm vào giỏ hàng.
- + Thanh toán và đặt hàng.

- + Quản lý đơn hàng cho cả người dùng và nhà cung cấp.
- + Thống kê.

- Công nghệ và Framework: Sử dụng Laravel Framework để xây dựng backend của ứng dụng và cùng các công nghệ web front-end như HTML, CSS, Bootstrap, JavaScript (có thể sử dụng thêm Vue.js hoặc React) để xây dựng giao diện người dùng.

Công việc cần thực hiện:

- Phân tích yêu cầu: Hiểu rõ yêu cầu của dự án, bao gồm chức năng, giao diện người dùng, và các tính năng đặc biệt như tích hợp thanh toán, quản lý đơn hàng, v.v.

- Thiết kế cơ sở dữ liệu: Thiết kế cấu trúc cơ sở dữ liệu cho các thực thể như người dùng, sản phẩm, đơn hàng, v.v.

- Phát triển backend: Xây dựng các API và logic backend bằng Laravel Framework để xử lý yêu cầu từ phía client, quản lý cơ sở dữ liệu và tích hợp các tính năng như đăng nhập, đăng ký, thanh toán, v.v.

- Phát triển frontend: Xây dựng giao diện người dùng bằng HTML, CSS, Bootstrap và JavaScript, có thể sử dụng các thư viện hoặc framework như Vue.js hoặc React để tạo ra trải nghiệm người dùng tốt hơn.

- Kiểm thử và Debugging: Kiểm tra tính năng và xử lý lỗi trên cả backend và frontend để đảm bảo hoạt động ổn định và mượt mà.

- Triển khai và duy trì: Triển khai ứng dụng lên một máy chủ và duy trì nó sau khi đã hoàn thành để đảm bảo ứng dụng luôn hoạt động một cách ổn định và an toàn.

➔ Xây dựng một website thương mại điện tử như Nest sử dụng Laravel Framework là một dự án thú vị và có thể mang lại nhiều cơ hội phát triển.

1.2.2. Hệ thống thực đang vận hành.

- Một số nghiệp vụ thường xuyên:

- + Cho phép xem thông tin sản phẩm: Người dùng có thể xem chi tiết sản phẩm mong muốn, thêm vào giỏ hàng, thanh toán,...

- + Đề cử những sản phẩm tương đồng.
- Các nghiệp vụ định kỳ:
 - + Cập nhật thông tin sản phẩm mới nhất.
 - + Thống kê định kỳ hằng tháng.

1.2.3. Khảo sát yêu cầu người dùng.

- Đường dẫn tới bài khảo sát:

https://docs.google.com/forms/d/1LdTni3zYUznuc3aln2UgJQ24JnJInGxF__JKTgsg8Hs/edit#responses

Kết luận nhu cầu cho hệ thống sau khảo sát:

- Đối với quản trị viên:
 - + Quản lý tài khoản người dùng.
 - + Thống kê.
 - + Quyền quản trị cao nhất trong hệ thống
- Đối với nhà cung cấp:
 - + Quản lý tài khoản nhà cung cấp.
 - + Thống kê liên quan đến cửa hàng.
 - + Cung cấp những sản phẩm phù hợp với khách hàng.
- Đối với khách hàng:
 - + Quản lý tài khoản cá nhân.
 - + Xem sản phẩm, thông tin chi tiết sản phẩm.
 - + Giỏ hàng, thanh toán.

- Ngoài ra, hệ thống cần được thiết kế sao cho dễ hiểu, giao diện trực quan mang lại trải nghiệm dễ dàng cho người sử dụng. Điều quan trọng là phải đảm bảo thông tin cá nhân được an toàn tuyệt đối. Đồng thời phải linh hoạt cho người quản trị.

CHƯƠNG 2 – PHÂN TÍCH THIẾT KẾ HỆ THỐNG

2.1. Giới thiệu về hệ thống.

2.1.1. Mô tả bài toán.

Đề tài được đặt ra nhằm xây dựng một trang web thương mại điện tử với tên gọi Nest, được phát triển bằng công nghệ Laravel Framework. Trang web này sẽ cung cấp một nền tảng trực tuyến cho các nhà cung cấp để quản lý, đăng ký và bán các sản phẩm của họ. Đồng thời, Nest cũng là nơi cho phép khách hàng tìm kiếm, mua sắm và đặt hàng từ một loạt các nhà cung cấp khác nhau.

Chức năng chính của Nest bao gồm việc đăng ký và đăng nhập tài khoản cho cả khách hàng, nhà cung cấp và admin; quản lý sản phẩm với khả năng thêm, sửa đổi và xóa sản phẩm, cũng như quản lý thông tin sản phẩm và giá cả. Để tối ưu trải nghiệm mua sắm, trang web cung cấp công cụ tìm kiếm và lọc sản phẩm, giúp khách hàng dễ dàng tìm kiếm và chọn lựa sản phẩm phù hợp. Ngoài ra, Nest cũng hỗ trợ tính năng giỏ hàng và thanh toán, cho phép khách hàng thêm sản phẩm vào giỏ hàng, xem và chỉnh sửa giỏ hàng, sau đó tiến hành thanh toán khi đã hoàn tất mua sắm. Đồng thời, trang web cung cấp cho cả khách hàng và nhà cung cấp chức năng quản lý đơn hàng, bao gồm xem chi tiết đơn hàng, trạng thái đơn hàng và cập nhật thông tin vận chuyển.

Cuối cùng, Nest còn tích hợp tính năng đánh giá và nhận xét sản phẩm, cho phép khách hàng đánh giá và viết nhận xét về sản phẩm mà họ đã mua, giúp khách hàng khác có thể tham khảo trước khi quyết định mua hàng. Sử dụng công nghệ Laravel Framework, bài toán này được giải quyết một cách linh hoạt, an toàn và hiệu quả, mang lại trải nghiệm mua sắm trực tuyến tốt nhất cho khách hàng và nhà cung cấp.

2.1.2. Quy trình xây dựng hệ thống.

Quy trình xây dựng hệ thống là một loạt các bước được thực hiện để phát triển một sản phẩm hoặc ứng dụng từ ý tưởng ban đầu đến sản phẩm hoàn chỉnh.

- Thu thập yêu cầu và phân tích:

+ Thu thập thông tin từ khách hàng hoặc người sử dụng để hiểu rõ yêu cầu và mục tiêu của hệ thống.

+ Phân tích yêu cầu và xác định các tính năng và chức năng cần thiết cho hệ thống.

- Thiết kế:

+ Xây dựng một thiết kế chi tiết của hệ thống, bao gồm cấu trúc cơ sở dữ liệu, giao diện người dùng, và các chức năng và tính năng chi tiết.

+ Xác định công nghệ và framework phù hợp cho việc phát triển hệ thống.

- Phát triển:

+ Phát triển mã nguồn của hệ thống theo thiết kế đã được xác định, bao gồm cả phần back-end và front-end.

+ Kiểm tra và triển khai các tính năng và chức năng theo từng giai đoạn để đảm bảo tính ổn định và chất lượng của hệ thống.

- Kiểm thử:

+ Thực hiện các bài kiểm thử để đảm bảo tính đúng đắn, hiệu suất và bảo mật của hệ thống.

+ Kiểm tra tích hợp để đảm bảo các thành phần của hệ thống hoạt động cùng nhau một cách mượt mà.

- Triển khai:

+ Triển khai hệ thống vào môi trường sản xuất hoặc môi trường thực tế mà người dùng sẽ truy cập.

+ Tiến hành cài đặt, cấu hình và kiểm tra lại hệ thống để đảm bảo hoạt động ổn định.

- Vận hành và bảo trì:

+ Theo dõi và quản lý hoạt động của hệ thống, bao gồm giám sát hiệu suất và khắc phục sự cố khi cần thiết.

+ Cập nhật và bảo trì hệ thống để đảm bảo tính bảo mật và hiệu suất liên tục.

Quy trình này có thể được điều chỉnh và tùy biến tùy thuộc vào yêu cầu cụ thể của dự án và môi trường phát triển. Điều quan trọng là duy trì sự linh hoạt và làm việc chặt chẽ với khách hàng và các bên liên quan để đảm bảo thành công của dự án.

2.1.3. Kết quả mong muốn đạt được.

Kết quả mong muốn đạt được từ việc xây dựng hệ thống thương mại điện tử Nest sử dụng công nghệ Laravel Framework bao gồm:

- Trải nghiệm người dùng tốt: Tạo ra một trải nghiệm mua sắm trực tuyến dễ sử dụng và thú vị cho người dùng cuối, từ việc tìm kiếm sản phẩm đến thanh toán và nhận hàng.

- Quản lý sản phẩm hiệu quả: Cung cấp cho nhà cung cấp một nền tảng quản lý sản phẩm hiệu quả để đăng ký, quản lý và bán hàng trực tuyến.

- Tính linh hoạt và tùy chỉnh: Cho phép tùy chỉnh dễ dàng và linh hoạt cho cả người dùng và nhà cung cấp để đáp ứng các yêu cầu, mong muốn.

- Tính ổn định và bảo mật: Đảm bảo hệ thống hoạt động ổn định và an toàn, bảo vệ thông tin cá nhân của người dùng và dữ liệu của nhà cung cấp.

- Tính mở rộng và mở đóng: Xây dựng hệ thống có khả năng mở rộng để có thể thêm mới tính năng và mở rộng quy mô theo thời gian.

- Tăng cường doanh số bán hàng: Giúp tăng cường doanh số bán hàng cho các nhà cung cấp bằng cách cung cấp một kênh bán hàng trực tuyến hiệu quả và tiện lợi.

- Tính tương tác và phản hồi: Tạo điều kiện cho việc tương tác và phản hồi giữa khách hàng và nhà cung cấp thông qua các tính năng như đánh giá, nhận xét và hỗ trợ khách hàng.

- Tăng cường thương hiệu: Xây dựng và tăng cường thương hiệu của Nest như một điểm đến tin cậy và uy tín trong lĩnh vực thương mại điện tử.

2.2. Phân tích yêu cầu.

2.2.1. Yêu cầu về hệ thống.

Website thương mại điện tử Nest hướng tới một hệ thống có giao diện bắt mắt, dễ sử dụng, dễ quản lý và đem lại trải nghiệm tốt nhất cho người

dùng. Bên cạnh đó, hệ thống cũng hướng tới sự tối ưu hiệu suất, tiết kiệm thời gian nhất cho người sử dụng. Một số yêu cầu cơ bản mà hệ thống của website thương mại điện tử Nest cần đáp ứng:

- Quản lý sản phẩm: Hệ thống cần có khả năng quản lý các sản phẩm từ nhiều nhà cung cấp khác nhau, bao gồm thông tin sản phẩm, hình ảnh, giá cả và mô tả.

- Tìm kiếm và lọc sản phẩm: Cung cấp tính năng tìm kiếm nhanh chóng và lọc sản phẩm theo nhiều tiêu chí khác nhau như danh mục, giá cả, đánh giá của khách hàng, v.v.

- Quản lý đơn hàng: Hệ thống cần có khả năng quản lý đơn hàng từ khách hàng, bao gồm xác nhận đơn hàng, giao hàng và cập nhật trạng thái đơn hàng.

- Thanh toán và xử lý thanh toán: Hệ thống cần tích hợp các cổng thanh toán an toàn và tiện lợi để khách hàng có thể thanh toán mua hàng một cách dễ dàng và an toàn.

- Quản lý người dùng và tài khoản: Hệ thống cần cung cấp chức năng đăng ký, đăng nhập và quản lý tài khoản cho khách hàng, bao gồm cập nhật thông tin cá nhân và lịch sử mua hàng.

- Tính năng đánh giá và phản hồi: Cho phép khách hàng đánh giá và viết nhận xét về các sản phẩm, cũng như phản hồi về trải nghiệm mua sắm của họ trên trang web.

- Bảo mật: Đảm bảo an toàn cho dữ liệu cá nhân của khách hàng và thông tin thanh toán thông qua các biện pháp bảo mật như mã hóa SSL và cơ chế xác thực hai yếu tố.

- Hiệu suất và mở rộng: Đảm bảo hệ thống có hiệu suất cao và có thể mở rộng để đáp ứng nhu cầu ngày càng tăng của khách hàng và doanh nghiệp.

- Tích hợp vận chuyển: Tích hợp các dịch vụ vận chuyển để khách hàng có thể chọn lựa phương thức vận chuyển phù hợp và theo dõi đơn hàng của mình.

2.2.2. Yêu cầu phía người dùng.

- Dễ sử dụng: Giao diện người dùng thân thiện và dễ sử dụng để khách hàng có thể tìm kiếm sản phẩm, thêm vào giỏ hàng và thanh toán một cách dễ dàng.
- Đa dạng sản phẩm: Một loạt các sản phẩm từ nhiều nhà cung cấp khác nhau để khách hàng có nhiều lựa chọn.
- Tìm kiếm và lọc sản phẩm: Cung cấp công cụ tìm kiếm và lọc sản phẩm hiệu quả để khách hàng có thể nhanh chóng tìm thấy những sản phẩm mà họ quan tâm.
- Thông tin chi tiết sản phẩm: Cung cấp thông tin chi tiết và hình ảnh chất lượng cao về sản phẩm để khách hàng có thể đánh giá và quyết định mua hàng.
- Thanh toán an toàn: Hỗ trợ nhiều phương thức thanh toán an toàn và tiện lợi để khách hàng có thể lựa chọn.
- Tài khoản người dùng: Cho phép khách hàng đăng ký tài khoản để quản lý thông tin cá nhân, theo dõi lịch sử mua hàng và đánh giá sản phẩm.
- Phản hồi và đánh giá: Cho phép khách hàng đánh giá và viết nhận xét về sản phẩm mà họ đã mua, giúp khách hàng khác có thêm thông tin để quyết định mua hàng.
- Vận chuyển và giao hàng: Cung cấp thông tin về các phương thức vận chuyển có sẵn và thời gian giao hàng ước tính để người dùng có thể lựa chọn phù hợp.
- Bảo mật thông tin: Bảo vệ thông tin cá nhân và thông tin thanh toán của khách hàng thông qua các biện pháp bảo mật như mã hóa SSL và cơ chế xác thực.

2.2.3. Vai trò của người dùng.

- Hệ thống được phân quyền thành 3 nhóm chính: Quản trị viên (admin), nhà cung cấp (supplier) và khách hàng (customer).
- Mô tả cho nhóm người dùng:

+ Quản trị viên (admin): Đây là vai trò cao nhất trong hệ thống. Người quản trị có toàn quyền quản lý, thao tác, chỉnh sửa, cập nhật, thêm mới toàn bộ thông tin trong hệ thống.

+ Nhà cung cấp (supplier): Đây là vai trò được cung cấp cho nhóm người dùng cho nhu cầu hợp tác với website để buôn bán những mặt hàng cụ thể cho nhóm đối người khách hàng. Có mọi quyền tương ứng như quản trị viên trong khu vực cửa hàng.

+ Khách hàng (customer): Đây là vai trò của nhóm người mua trong hệ thống. Được cung cấp những quyền cơ bản như tìm kiếm, xem thông tin chi tiết sản phẩm, thanh toán, đánh giá sản phẩm.

2.2.4. Yêu cầu chức năng.

Bảng 2. 1. Phân quyền và yêu cầu chức năng.

Vai trò	Yêu cầu chức năng
Quản trị viên (admin)	<ul style="list-style-type: none"> - Đăng nhập, đăng ký. - Quản lý tài khoản. - Quản lý nhà cung cấp. - Quản lý khách hàng. - Thống kê.
Nhà cung cấp (supplier)	<ul style="list-style-type: none"> - Đăng nhập, đăng ký - Thêm, sửa, xóa sản phẩm. - Thống kê. - Quản lý tài khoản bán hàng.
Khách hàng (customer)	<ul style="list-style-type: none"> - Đăng nhập, đăng ký. - Tìm kiếm. - Xem sản phẩm và thông tin chi tiết sản phẩm. - Giỏ hàng. - Thanh toán. - Đánh giá sản phẩm. - Quản lý tài khoản cá nhân.

2.2.5. Yêu cầu phi chức năng.

Yêu cầu phi chức năng của trang web thương mại điện tử "Nest" bao gồm các yêu cầu không liên quan trực tiếp đến tính năng của hệ thống, nhưng vẫn đóng vai trò quan trọng trong việc cung cấp trải nghiệm tổng thể cho người dùng. Dưới đây là một số yêu cầu phi chức năng:

- Tốc độ tải trang: Đảm bảo trang web có thời gian tải nhanh để người dùng không cảm thấy bị chậm chạp hoặc mất kiên nhẫn.

- Responsive design: Thiết kế trang web để phản hồi tốt trên các thiết bị di động và máy tính bảng để người dùng có thể trải nghiệm trang web một cách thoải mái trên mọi thiết bị.

- Thiết kế hấp dẫn: Sử dụng thiết kế đẹp mắt và hấp dẫn để thu hút người dùng và tạo ra ấn tượng tích cực ngay từ lần đầu tiên họ truy cập trang web.

- Chất lượng hình ảnh: Sử dụng hình ảnh chất lượng cao để hiển thị sản phẩm một cách rõ ràng và hấp dẫn.

- Chính sách trả hàng và bảo hành: Cung cấp thông tin rõ ràng và chi tiết về chính sách trả hàng và bảo hành để tạo sự tin tưởng cho người dùng.

- Đánh giá và nhận xét từ người dùng: Cho phép người dùng đánh giá và viết nhận xét về sản phẩm để tạo ra một cộng đồng mua sắm trực tuyến tích cực và tin cậy.

- Hướng dẫn sử dụng và hỗ trợ: Cung cấp hướng dẫn sử dụng sản phẩm và hỗ trợ kỹ thuật cho người dùng khi cần thiết.

2.2.6. Tài liệu đặc tả yêu cầu.

- Mục đích của website: Trang web thương mại điện tử "Nest" nhằm mục đích tạo ra một nền tảng trực tuyến cho việc mua sắm các sản phẩm từ nhiều nhà cung cấp khác nhau. Trang web sẽ cung cấp một trải nghiệm mua sắm dễ dàng, thuận tiện và đa dạng cho người dùng.

- Mô tả chung:

- + Trang web có giao diện bắt mắt và dễ sử dụng.

- + Khách hàng có thể tìm kiếm các thông tin về những sản phẩm mong muốn bằng cách sử dụng thanh tìm kiếm hoặc duyệt qua các danh mục.

- + Khách hàng có thể xem chi tiết thông tin và để lại bình luận.

- Các tính năng của trang web:

- + Tìm kiếm thông tin: Khách hàng có thể nhập từ khóa tìm kiếm để tìm thông tin và xem kết quả phù hợp.

- + Xem chi tiết thông tin: Khách hàng có thể xem thông tin chi tiết sản phẩm như tên sản phẩm, mô tả, hình ảnh, đơn giá.

- Yêu cầu kỹ thuật:

- + Trang web phải hoạt động tốt trên các trình duyệt web phổ biến như Chrome, Firefox, Safari và Edge.

- + Trang web phải đảm bảo tính bảo mật của dữ liệu người dùng.

- + Trang web phải đảm bảo tính đáp ứng và tối ưu hóa trải nghiệm người dùng trên các thiết bị khác nhau, từ máy tính để bàn đến thiết bị di động.

- + Các chức năng phải hoạt động đúng và chính xác.

- + Thời gian tải trang web phải nhanh để đảm bảo trải nghiệm người dùng tốt.

2.3. Phân tích hệ thống.

2.3.1. Mô hình hóa chức năng hệ thống.



Hình 2. 1. Biểu đồ Usecase.

2.3.2. Mô tả chi tiết use case phía khách hàng.

2.3.2.1. Mô tả UC <Đăng nhập – Khách hàng>.

Bảng 2. 2. Mô tả UC <Đăng nhập - Khách hàng>.

Mô tả	Use case này cho phép người dùng đăng nhập tài khoản đã đăng ký của mình để truy cập vào hệ thống.
Tác nhân	Khách hàng.
Tiền điều kiện	Khách hàng phải đăng ký tài khoản trước đó.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Khách hàng truy cập vào ứng dụng, hệ thống sẽ yêu cầu khách hàng đăng nhập. Đăng nhập tài khoản và mật khẩu. 2. Nhập username và password, kích vào sign in, hệ thống sẽ kiểm tra tài khoản và mật khẩu rồi đưa khách hàng về màn hình chính. Use case kết thúc.
Luồng rẽ nhánh	<ol style="list-style-type: none"> 1. Nếu khách hàng nhập sai tên đăng nhập, hệ thống sẽ hiển thị một thông báo lỗi. Khách hàng có thể chọn thoát website hoặc nhập lại. Use case kết thúc. 2. Nếu khách hàng nhập mật khẩu sai, hệ thống sẽ hiển thị một thông báo lỗi. Khách hàng có thể nhập lại mật khẩu, thoát website hoặc tìm lại mật khẩu bằng cách nhấn “Quên mật khẩu”. Use case kết thúc. 3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.2.2. Mô tả UC <Đăng ký – Khách hàng>.

Bảng 2. 3. Mô tả UC <Đăng ký – Khách hàng>.

Mô tả	Use case này cho phép khách hàng đăng ký tài khoản mới để đăng nhập thực hiện các chức năng khác của hệ thống.
Tác nhân	Khách hàng.
Tiền điều kiện	Không.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Khách hàng muốn sử dụng chức năng của trang web, hệ thống sẽ yêu cầu khách hàng tạo một tài khoản mới. 2. Nhập username và password, kích vào “Đăng ký”, hệ thống sẽ kiểm tra tên đăng nhập và mật khẩu rồi đưa khách hàng về màn hình chính. Use case kết thúc.
Luồng rẽ nhánh	<ol style="list-style-type: none"> 1. Nếu khách hàng nhập một tên tài khoản trùng với một tên đã có trong database, hệ thống sẽ hiển thị một thông báo tên tài khoản đã tồn tại và yêu cầu khách hàng chọn một tên khác. Khách hàng có thể chọn thoát website hoặc nhập lại. Use case kết thúc. 2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.2.3. Mô tả UC <Tìm kiếm>.

Bảng 2. 4. Mô tả UC <Tìm kiếm>.

Mô tả	Use case này cho phép khách hàng tìm kiếm những sản phẩm mong muốn.
Tác nhân	Khách hàng.
Tiền điều kiện	Không.
Luồng sự kiện	<ol style="list-style-type: none"> 1. Khách hàng ấn vào thanh tìm kiếm, rồi nhập. Hệ thống sẽ lấy

kiện chính	dữ liệu từ bảng “ec_products” và cho ra kết quả là những sản phẩm có tên gần giống nhất với từ đã nhập vào. Use case kết thúc.
Luồng rẽ nhánh	1. Nếu khách hàng nhập một hay một chuỗi ký tự chưa có trong database, hệ thống sẽ đưa ra thông báo “không tìm thấy sản phẩm tương tự”, khách hàng có thể nhập lại hoặc sử dụng một chức năng khác. Use case kết thúc. 2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.2.4. Mô tả UC <Xem sản phẩm>.

Bảng 2. 5. Mô tả UC <Xem sản phẩm>.

Mô tả	Use case này cho phép người dùng theo dõi những thông tin mong muốn.
Tác nhân	Người dùng.
Tiền điều kiện	Không.
Luồng sự kiện chính	1. Tại màn hình trang chủ, khách hàng có thể tùy chọn sản phẩm để xem thông tin hoặc tìm kiếm trên thanh tìm kiếm bằng cách nhấn vào sản phẩm mong muốn. 2. Sau khi khách hàng chọn được sản phẩm, màn hình sẽ chuyển sang thông tin chi tiết sản phẩm. Use case kết thúc.
Luồng rẽ nhánh	1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.2.5. Mô tả UC <Quản lý giỏ hàng>.

Bảng 2. 6. Mô tả UC <Quản lý giỏ hàng>.

Mô tả	Use case này cho phép khách hàng thêm sản phẩm vào giỏ hàng để tiến hành thanh toán.
Tác nhân	Khách hàng.
Tiền điều kiện	Khách hàng phải đăng nhập trước đó.
Luồng sự kiện chính	1. Tại màn hình trang chủ hoặc màn hình xem sản phẩm, khách hàng ấn nút “Thêm vào giỏ hàng” tại sản phẩm đã chọn. 2. Sau đó, hệ thống sẽ thêm sản phẩm vào bảng “EC_CART” sau đó sản phẩm được lựa chọn sẽ hiện thị ở giỏ hàng. Use case kết thúc.
Luồng rẽ nhánh	1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.2.6. Mô tả UC <Thanh toán>.

Bảng 2. 7. Mô tả UC <Thanh toán>.

Mô tả	Use case này cho phép khách hàng thanh toán những sản phẩm muốn mua có trong giỏ hàng.
Tác nhân	Khách hàng.
Tiền điều kiện	Giỏ hàng phải có sản phẩm trước đó.
Luồng sự kiện chính	1. Trong màn hình giỏ hàng, khách hàng chọn những sản phẩm muốn thanh toán rồi ấn nút “Tiến hành thanh toán”. 2. Hệ thống sẽ chuyển từ màn hình giỏ hàng sang màn hình thanh toán. Tại đây khách hàng nhập địa chỉ nhận hàng tại những trường địa chỉ thông tin, chọn phương thức thanh toán mong muốn rồi tiến hành thanh toán. Use case kết thúc.

Luồng rẽ nhánh	<p>1. Nếu khách hàng nhập một địa chỉ không tồn tại trên bản đồ, hệ thống sẽ báo lỗi và yêu cầu khách hàng nhập lại. Use case kết thúc.</p> <p>2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.</p>
Hậu điều kiện	Không.

2.3.3. Mô tả chi tiết use case phía nhà cung cấp.

2.3.3.1. Mô tả UC <Đăng nhập – Nhà cung cấp>.

Bảng 2. 8. Mô tả UC < Đăng nhập – Nhà cung cấp>.

Mô tả	Use case này cho phép nhà cung cấp đăng nhập tài khoản đã đăng ký của mình để truy cập vào hệ thống.
Tác nhân	Nhà cung cấp.
Tiền điều kiện	Nhà cung cấp phải đăng ký tài khoản trước đó.
Luồng sự kiện chính	<p>1. Nhà cung cấp truy cập vào ứng dụng, hệ thống sẽ yêu cầu nhà cung cấp đăng nhập. Đăng nhập tài khoản và mật khẩu.</p> <p>2. Nhập username và password, kích vào sign in, hệ thống sẽ kiểm tra tài khoản và mật khẩu rồi đưa nhà cung cấp về màn hình chính. Use case kết thúc.</p>
Luồng rẽ nhánh	<p>1. Nếu nhà cung cấp nhập sai tên đăng nhập, hệ thống sẽ hiển thị một thông báo lỗi. Nhà cung cấp có thể chọn thoát website hoặc nhập lại. Use case kết thúc.</p> <p>2. Nếu nhà cung cấp nhập mật khẩu sai, hệ thống sẽ hiển thị một thông báo lỗi. Nhà cung cấp có thể nhập lại mật khẩu, thoát website hoặc tìm lại mật khẩu bằng cách nhấn “Quên mật khẩu”. Use case kết thúc.</p> <p>3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu</p>

	không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.3.2. Mô tả UC <Đăng ký – Nhà cung cấp>.

Bảng 2. 9. Mô tả UC < Đăng ký – Nhà cung cấp>.

Mô tả	Use case này cho phép nhà cung cấp đăng ký tài khoản mới để đăng nhập thực hiện các chức năng khác của hệ thống.
Tác nhân	Nhà cung cấp.
Tiền điều kiện	Không.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Nhà cung cấp muốn sử dụng chức năng của trang web, hệ thống sẽ yêu cầu nhà cung cấp tạo một tài khoản mới. 2. Nhập username và password, kích vào “Đăng ký”, hệ thống sẽ kiểm tra tên đăng nhập và mật khẩu rồi đưa nhà cung cấp về màn hình chính. Use case kết thúc.
Luồng rẽ nhánh	<ol style="list-style-type: none"> 1. Nếu nhà cung cấp nhập một tên tài khoản trùng với một tên đã có trong database, hệ thống sẽ hiển thị một thông báo tên tài khoản đã tồn tại và yêu cầu nhà cung cấp chọn một tên khác. Nhà cung cấp có thể chọn thoát website hoặc nhập lại. Use case kết thúc. 2. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.3.3. Mô tả UC <Thêm sản phẩm – Nhà cung cấp>.

Bảng 2. 10. Mô tả UC <Thêm sản phẩm – Nhà cung cấp>.

Mô tả	Use case này cho phép nhà cung cấp thêm sản phẩm muốn bán vào cửa hàng trực tuyến trên website.
Tác nhân	Nhà cung cấp.
Tiền điều kiện	Nhà cung cấp phải đăng nhập trước đó và đã được quản trị viên phê duyệt chức năng bán hàng.
Luồng sự kiện chính	1. Nhà cung cấp chọn mục “Product” trong bảng điều khiển của người bán hàng. 2. Chọn “Create” để thêm sản phẩm mới. Nhập những thông tin cần thiết về sản phẩm rồi ấn “Save”. Hệ thống sẽ đẩy sản phẩm lên cửa hàng trực tuyến. Use case kết thúc.
Luồng rẽ nhánh	1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.3.4. Mô tả UC <Thống kê – Nhà cung cấp>.

Bảng 2. 11. Mô tả UC <Thống kê – Nhà cung cấp>.

Mô tả	Use case này cho phép nhà cung cấp theo dõi thống kê trong cửa hàng trực tuyến.
Tác nhân	Nhà cung cấp.
Tiền điều kiện	Nhà cung cấp phải đăng nhập trước đó và đã được quản trị viên phê duyệt chức năng bán hàng.
Luồng sự kiện chính	1. Nhà cung cấp chọn mục “Dashboard” trong bảng điều khiển của người bán hàng. 2. Hệ thống sẽ chuyển tới màn hình thống kê, nơi nhà cung cấp có thể xem thông tin thống kê về cửa hàng trực tuyến. Use case kết thúc.

Luồng rẽ nhánh	1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.4. Mô tả chi tiết use case phía quản trị viên.

2.3.4.1. Mô tả UC <Đăng nhập – Quản trị viên>.

Bảng 2. 12. Mô tả UC <Đăng nhập – Quản trị viên>.

Mô tả	Use case này cho phép quản trị viên đăng nhập tài khoản đã đăng ký của mình để truy cập vào hệ thống.
Tác nhân	Quản trị viên.
Tiền điều kiện	Không.
Luồng sự kiện chính	<p>1. Quản trị viên truy cập vào ứng dụng, hệ thống sẽ yêu cầu quản trị viên đăng nhập. Đăng nhập tài khoản (User name) và mật khẩu.</p> <p>2. Nhập username và password, kích vào sign in, hệ thống sẽ kiểm tra tài khoản và mật khẩu rồi đưa quản trị viên về màn hình chính. Use case kết thúc.</p>
Luồng rẽ nhánh	<p>1. Nếu Quản trị viên nhập tên đăng nhập sai, hệ thống sẽ hiển thị một thông báo lỗi. Quản trị viên có thể chọn thoát website hoặc nhập lại. Use case kết thúc.</p> <p>2. Nếu quản trị viên nhập mật khẩu sai, hệ thống sẽ hiển thị một thông báo lỗi. Quản trị viên có thể nhập lại mật khẩu, thoát ứng dụng hoặc tìm lại mật khẩu bằng cách nhấn “Quên mật khẩu”. Use case kết thúc.</p> <p>3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.</p>
Hậu điều kiện	Không.

2.3.4.2. Mô tả UC <Thống kê – Quản trị viên>.

Bảng 2. 13. Mô tả UC <Thống kê – Quản trị viên>.

Mô tả	Use case này cho phép quản trị viên dùng để xem thống kê hệ thống.
Tác nhân	Quản trị viên.
Tiền điều kiện	Quản trị viên phải đăng nhập trước đó.
Luồng sự kiện chính	1. Tại màn hình chính, người quản trị nhấn vào “Người dùng” hệ thống sẽ hiển thị một số chức năng. 2. Sau khi người quản trị nhấn vào “Người dùng”, tại màn hình “Người dùng” người quản trị nhấn vào “Dashboard”. Khi đó hệ thống đưa tới màn hình xem “Dashboard”. Use case kết thúc.
Luồng rẽ nhánh	1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.4.3. Mô tả UC <Quản lý sản phẩm>.

Bảng 2. 14. Mô tả UC <Quản lý sản phẩm>.

Mô tả	Use case này cho phép người quản trị dùng để xem danh sách hình ảnh đã có.
Người thực hiện	Quản trị viên.
Tiền điều kiện	Quản trị viên phải đăng nhập trước đó.
Luồng sự kiện chính	1. Tại màn hình trang quản trị, quản trị viên chọn mục “Product”, hệ thống sẽ hiển thị tất cả sản phẩm có trên website 2. Tại đây quản trị viên có thể thực hiện những quyền cho phép như thêm, sửa, xóa sản phẩm. Use case kết thúc.

Luồng rẽ nhánh	1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

2.3.4.4. Mô tả UC <Quản lý người dùng>.

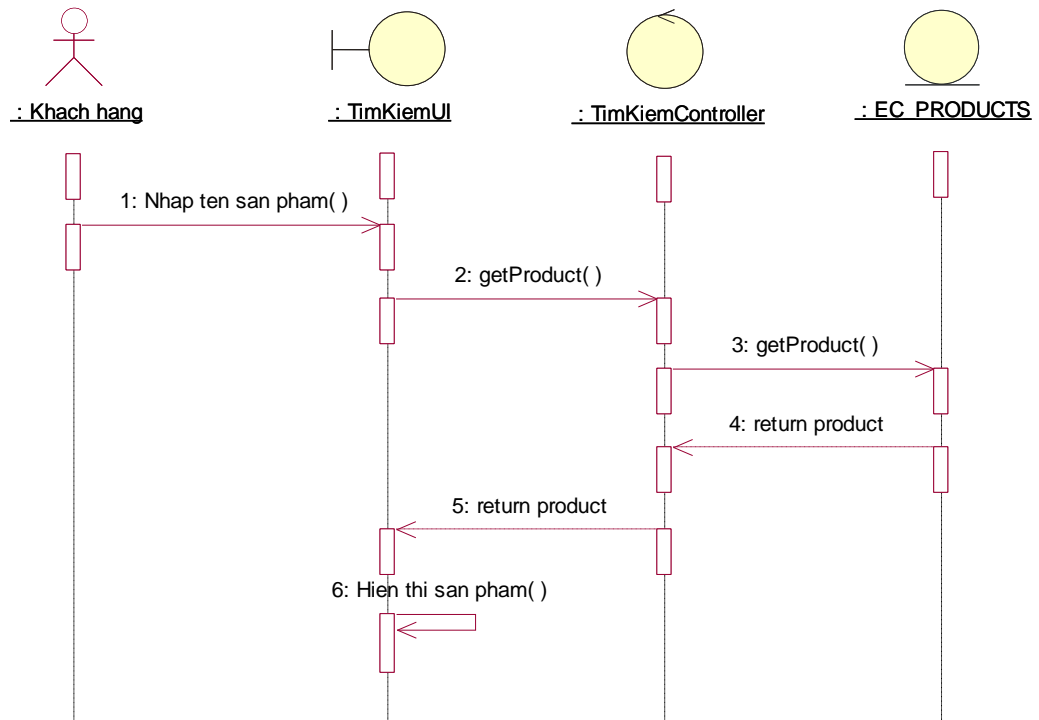
Bảng 2. 15. Mô tả UC <Quản lý người dùng>.

Mô tả	Use case này cho phép quản trị viên dùng để xem danh sách người dùng đã có.
Tác nhân	Quản trị viên.
Tiền điều kiện	Quản trị viên phải đăng nhập trước đó.
Luồng sự kiện chính	1. Tại màn hình chính, quản trị viên chọn mục “Users”. 2. Khi đó hệ thống đưa quản trị viên tới màn hình xem “Danh sách người dùng”. Use case kết thúc.
Luồng rẽ nhánh	1. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ hiển thị thông báo lỗi.
Hậu điều kiện	Không.

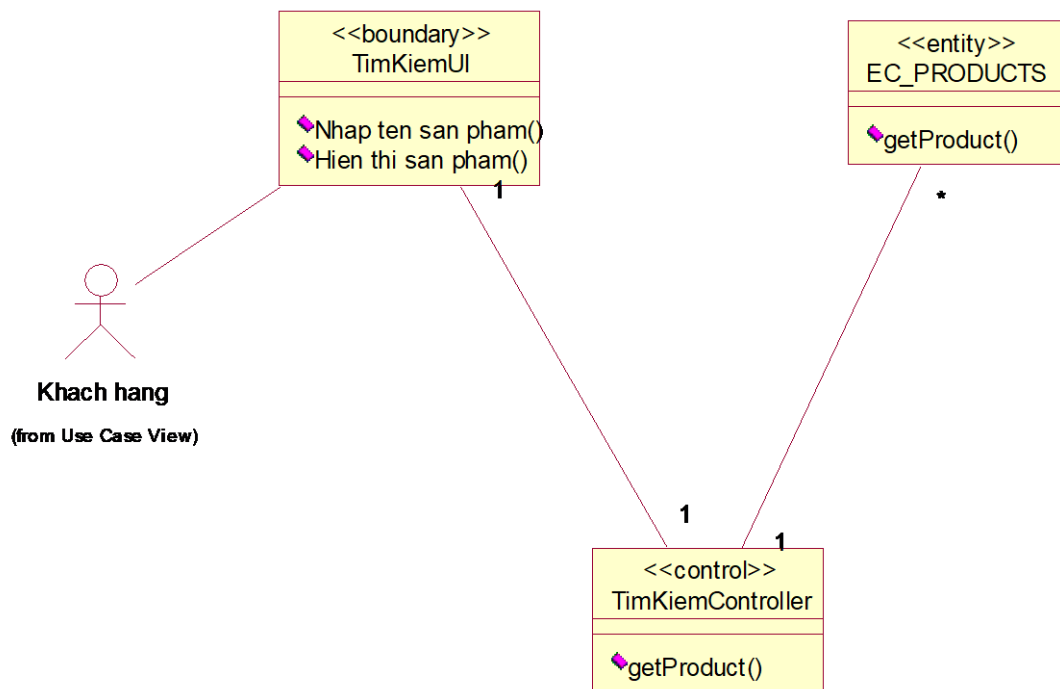
Bảng 2. 16. Mô tả UC <User – Quản lý người dùng>.

2.3.5. Biểu đồ trình tự và biểu đồ lớp phân tích.

2.3.5.1. UC <Tìm kiếm>.

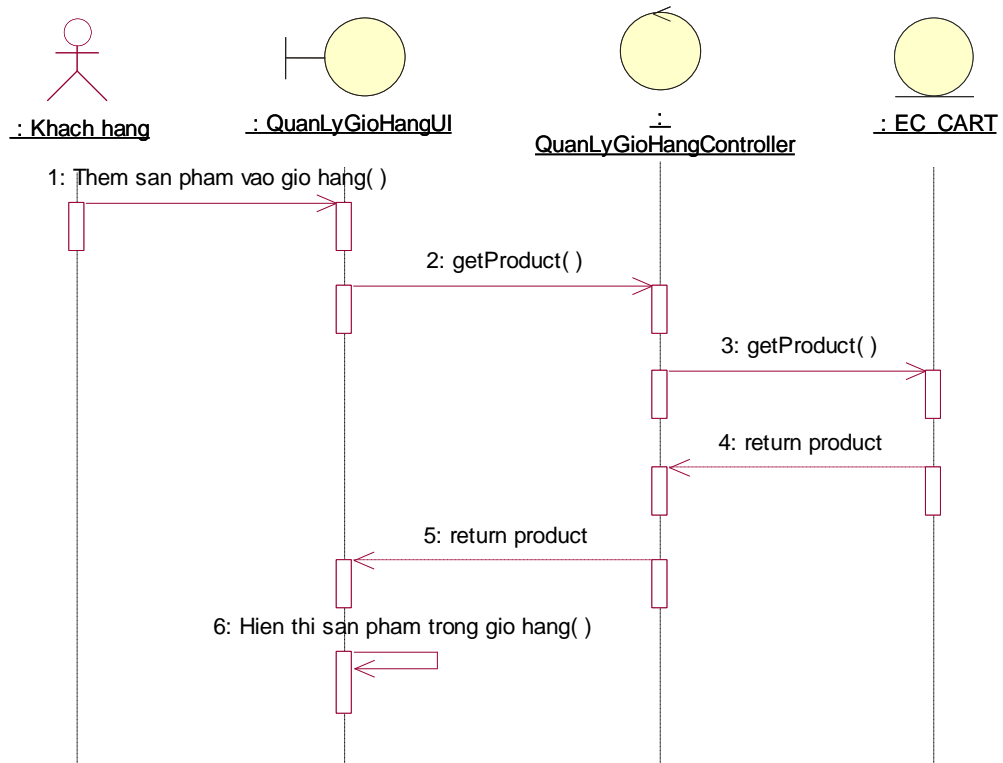


Hình 2. 2. Biểu đồ trình tự UC <Tìm kiếm>.

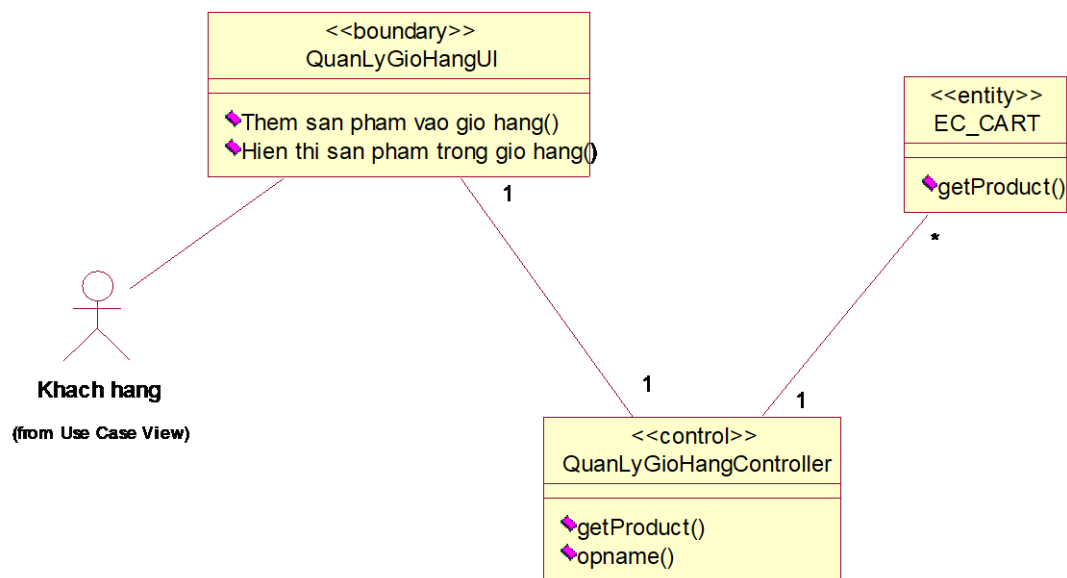


Hình 2. 3. Biểu đồ lớp phân tích UC <Tìm kiếm>.

2.3.5.2. UC <Quản lý giỏ hàng>.

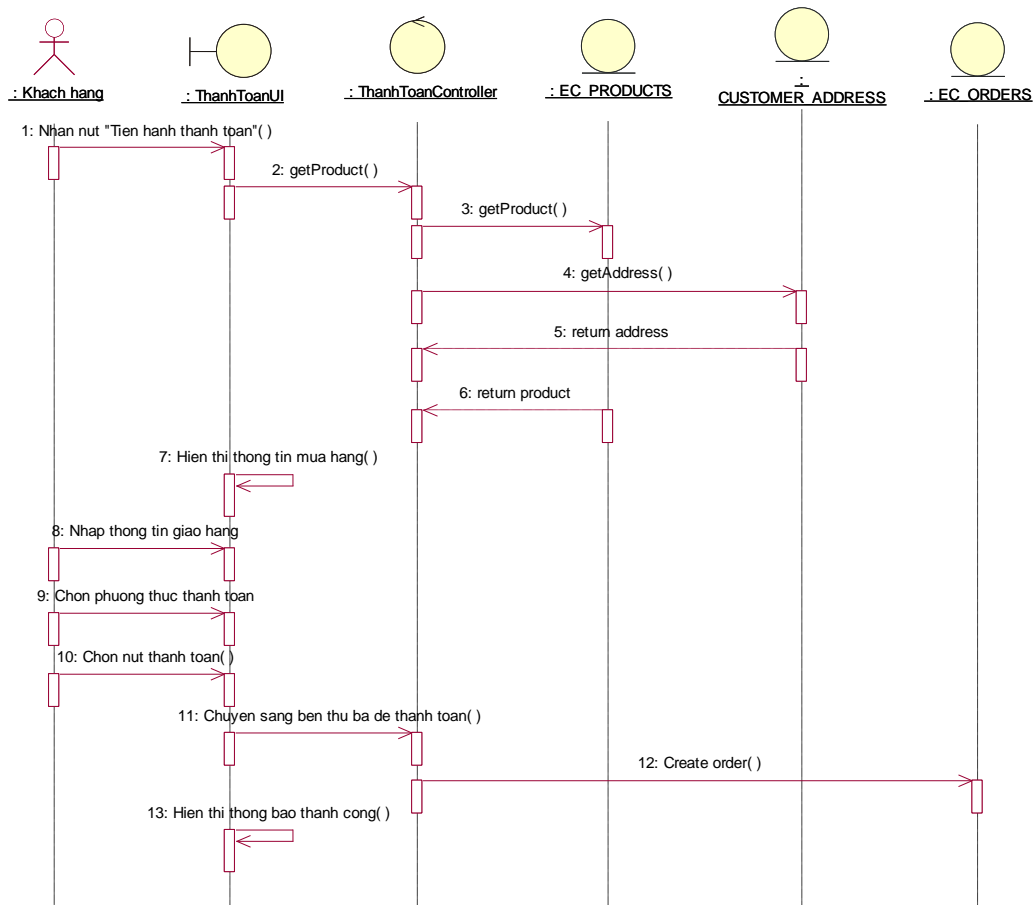


Hình 2. 4. Biểu đồ trình tự UC <Quản lý giỏ hàng>.

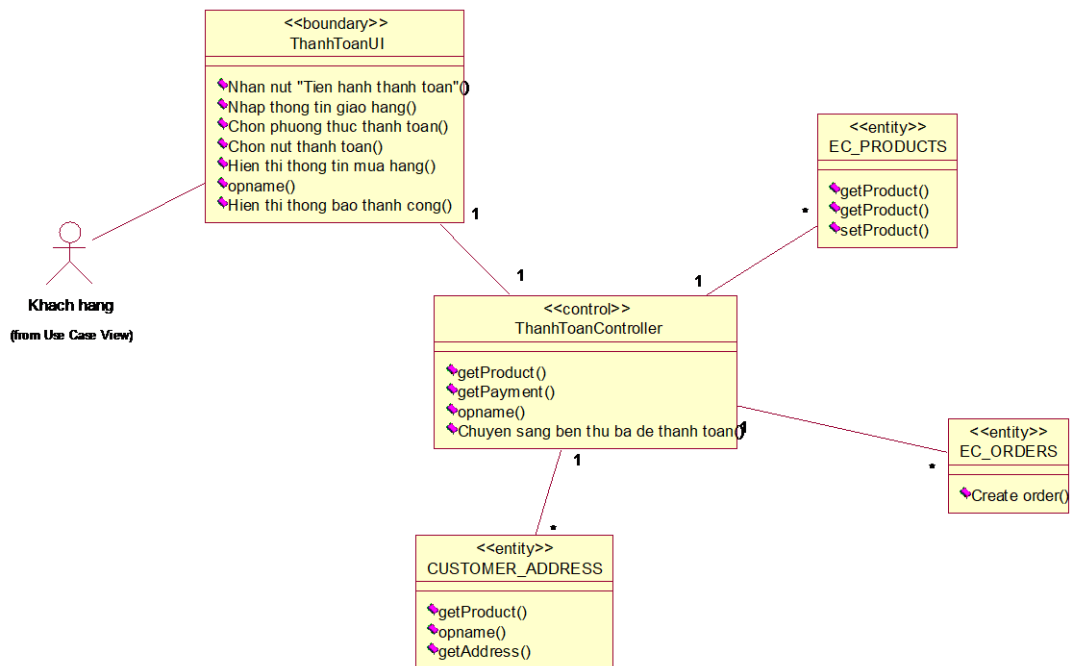


Hình 2. 5. Biểu đồ lớp phân tích UC <Quản lý giỏ hàng>.

2.3.5.3. UC <Thanh toán>.

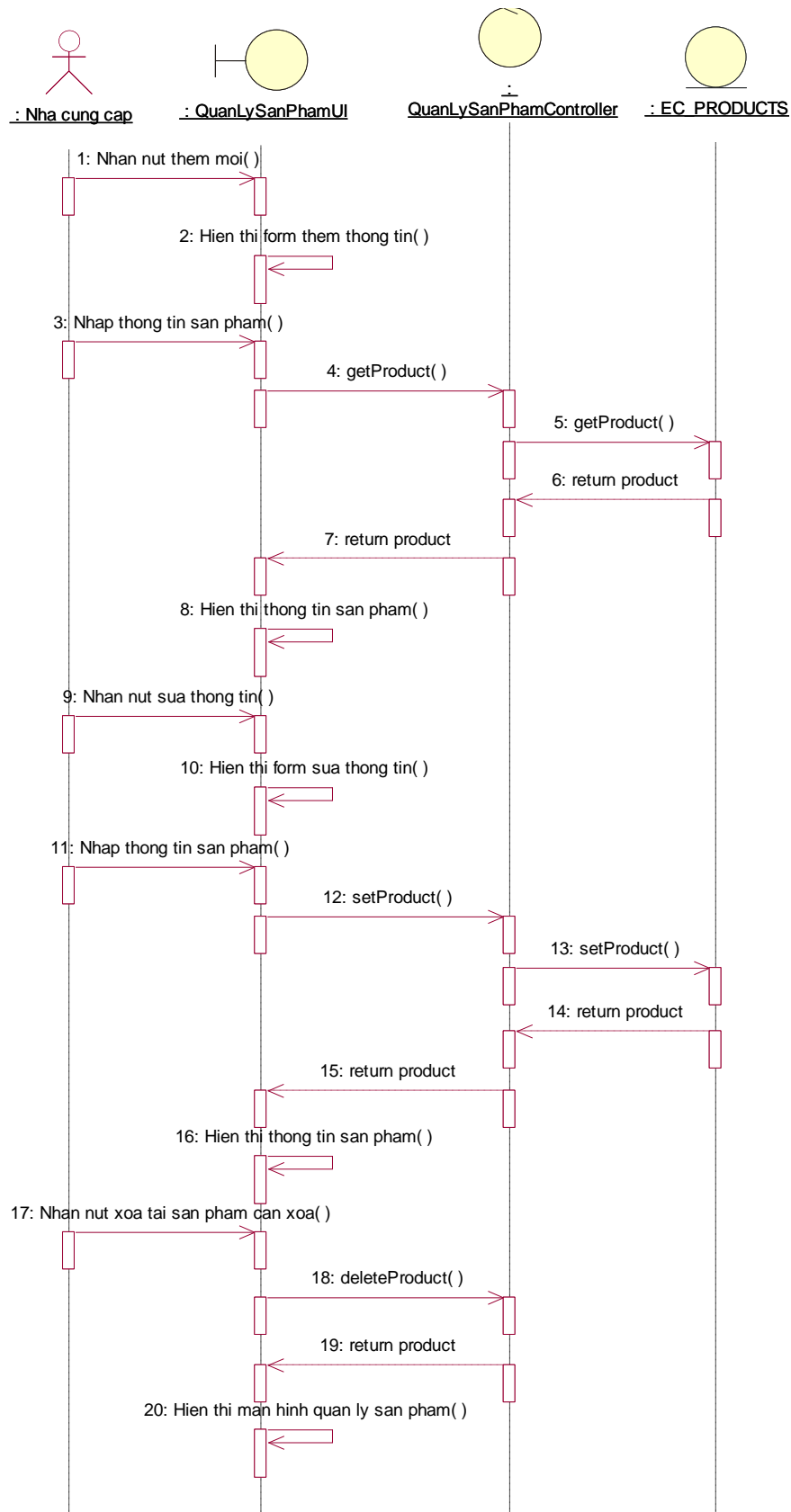


Hình 2. 6. Biểu đồ trình tự UC <Thanh toán>.

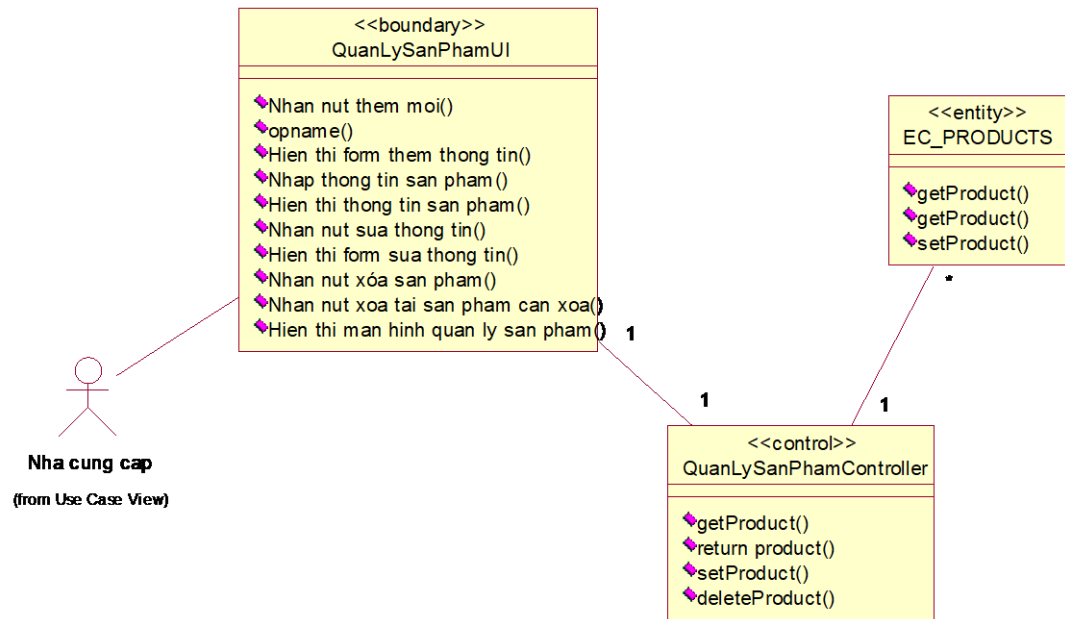


Hình 2. 7. Biểu đồ lớp phân tích UC <Thanh toán>.

2.3.5.4. UC <Quản lý sản phẩm>.

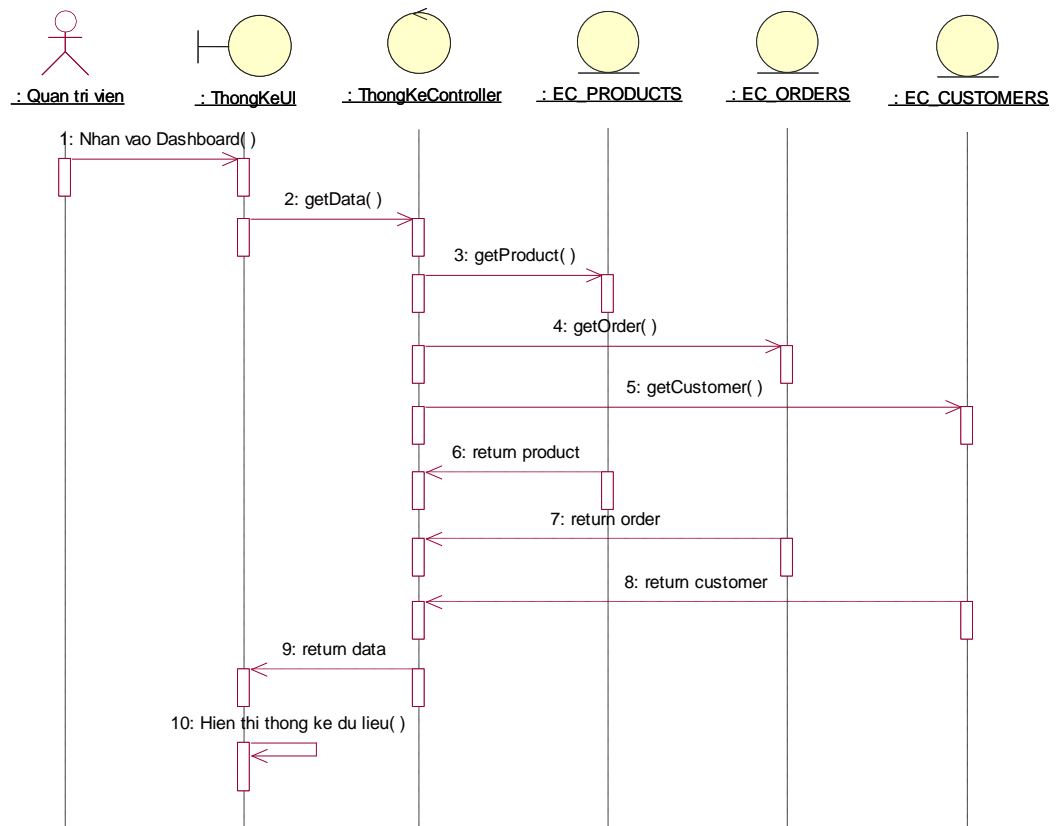


Hình 2. 8. Biểu đồ trình tự UC <Quản lý sản phẩm>.

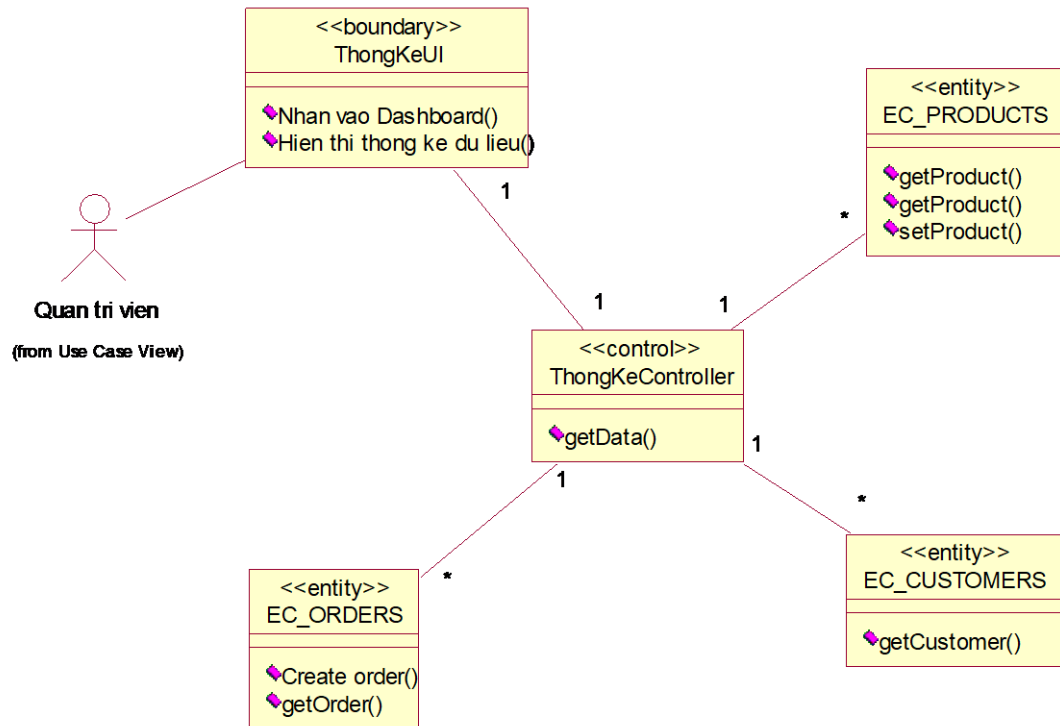


Hình 2. 9. Biểu đồ lớp phân tích UC <Quản lý sản phẩm>.

2.3.5.5. UC <Thống kê>.

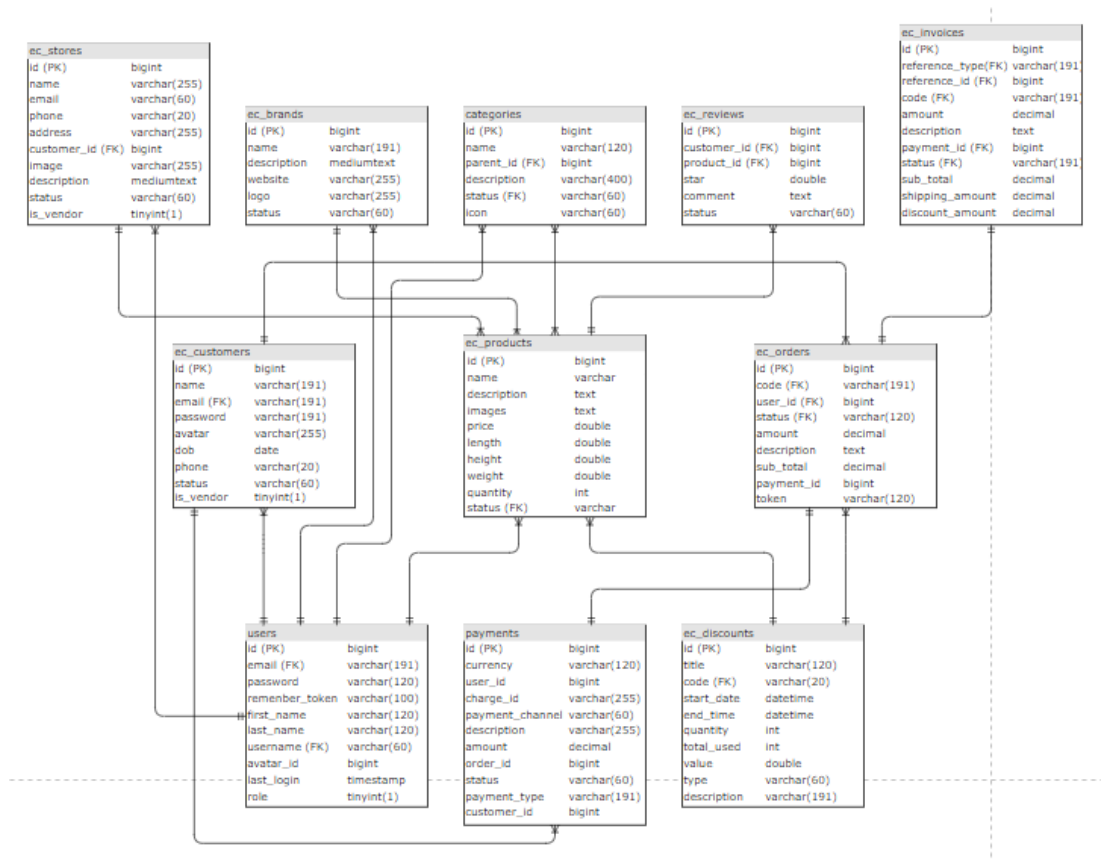


Hình 2. 10. Biểu đồ trình tự UC <Thống kê>.



Hình 2. 11. Biểu đồ lớp phân tích UC <Thống kê>.

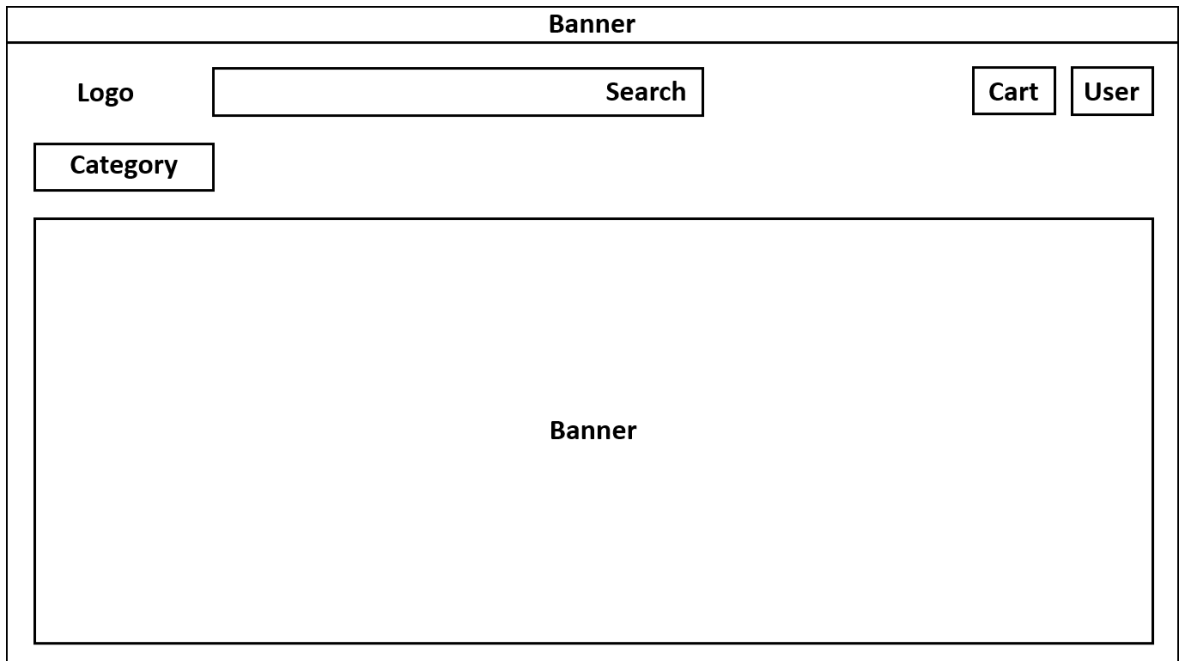
2.3.6. Thiết kế hệ thống cơ sở dữ liệu.



Hình 2. 12. Database relationship.

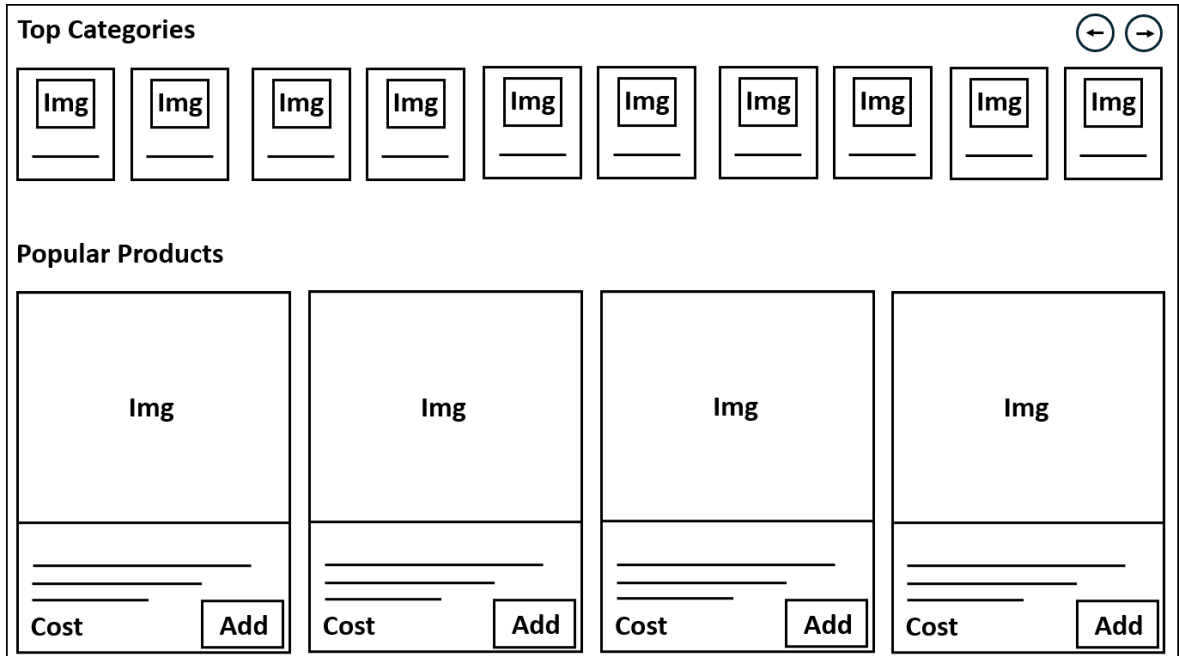
2.3.7. Thiết kế giao diện hệ thống.

2.3.7.1. Màn hình Trang chủ.



Hình 2. 13. Thiết kế màn hình Trang chủ.

2.3.7.2. Màn hình Danh mục sản phẩm.



Hình 2. 14. Thiết kế màn hình Danh mục sản phẩm.

2.3.7.3. Màn hình Xem sản phẩm.

Category		
<div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div>	<div> <div>Img</div> <div> <div>Img</div> <div>Img</div> </div> </div>	<div> <div>_____</div> <div>Cost</div> <div>_____</div> <div>_____</div> <div>_____</div> <div> <div></div> <div></div> </div> <div>Quantity</div> <div>Add</div> <div>Buy</div> </div>

Hình 2. 15. Thiết kế màn hình Xem sản phẩm.

2.3.7.4. Màn hình Giỏ hàng.

Banner				
Logo	Search	Cart	User	
Category				
Your Cart				
Product	Unit Price	Quantity	Subtotal	Remove
<div> <div>Img</div> <div> <div>_____</div> <div>_____</div> </div> </div>				
Continue Shopping				<div> <div>_____</div> <div>_____</div> <div>Proceed to Checkout</div> </div>

Hình 2. 16. Thiết kế màn hình giỏ hàng.

2.3.7.5. Màn hình Thanh toán.

<div>Logo</div> <hr/> <div>Shipping information</div> <div></div> <div></div> <div>Payment method</div> <div></div> <div>Other order</div> <div></div> <div>Checkout</div>	<div>Product</div> <div></div>
--	--------------------------------

Hình 2. 17. Thiết kế màn hình Thanh toán.

2.3.7.6. Màn hình Quản trị viên.

Logo	Admin
Dashboard	<div> <div></div> <div></div> <div></div> <div></div> </div> <div></div> <div> <div></div> <div></div> </div>

Hình 2. 18. Thiết kế màn hình Quản trị viên.

2.3.7.7. Màn hình Quản lý đơn hàng.

Logo		Admin	
Ecommere	<div><div></div>Search<div></div><div></div></div>		
	<div><div></div><div></div><div></div></div>		

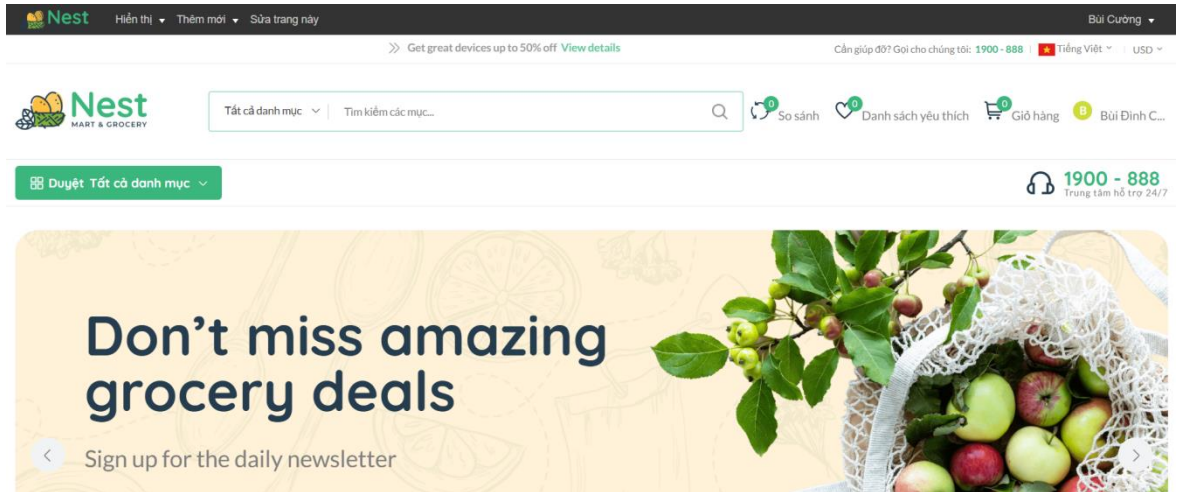
Hình 2. 19. Thiết kế màn hình Quản lý đơn hàng.

CHƯƠNG 3 – CÀI ĐẶT HỆ THỐNG VÀ KẾT QUẢ ĐẠT ĐƯỢC

3.1. Kết quả đạt được.

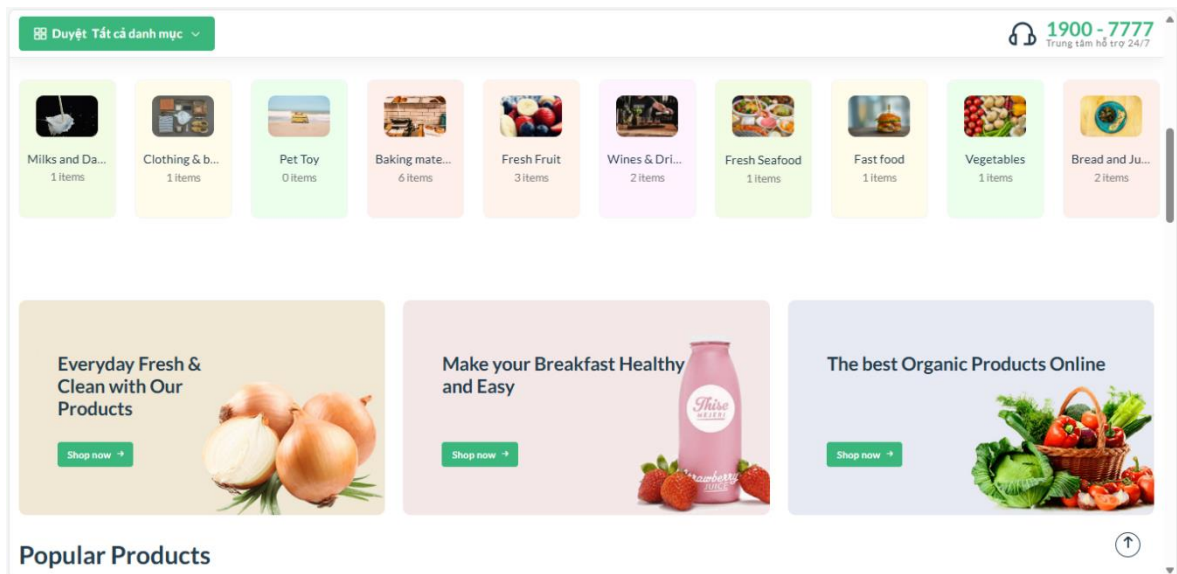
3.1.1. Giao diện phía khách hàng (customer).

3.1.1.1. Màn hình Trang chủ.



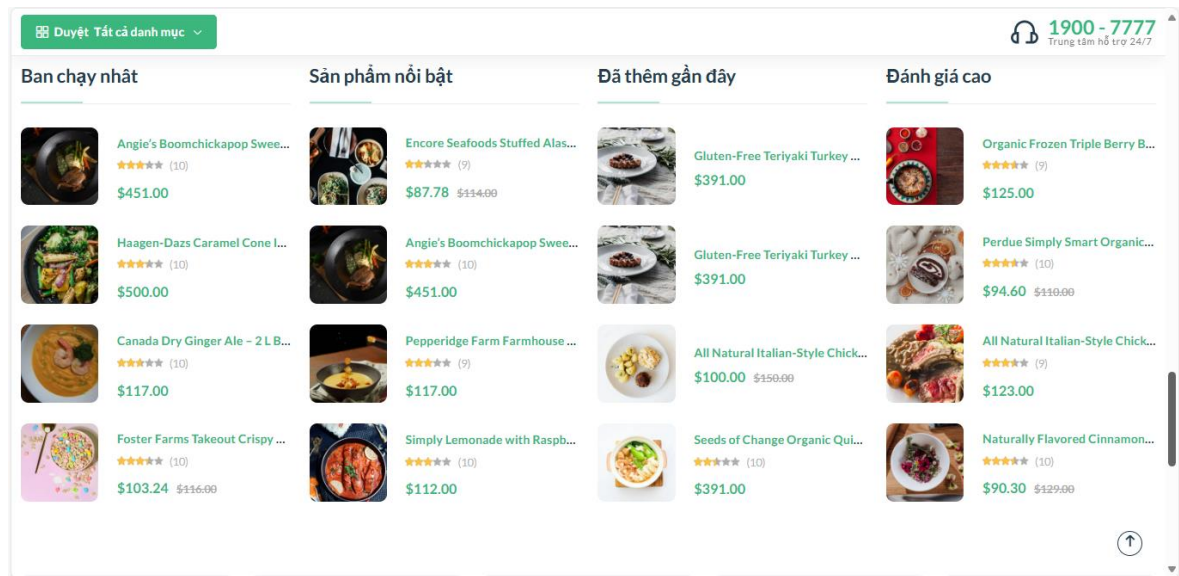
Hình 3. 1. Màn hình Trang chủ.

3.1.1.2. Màn hình Xem danh mục sản phẩm.



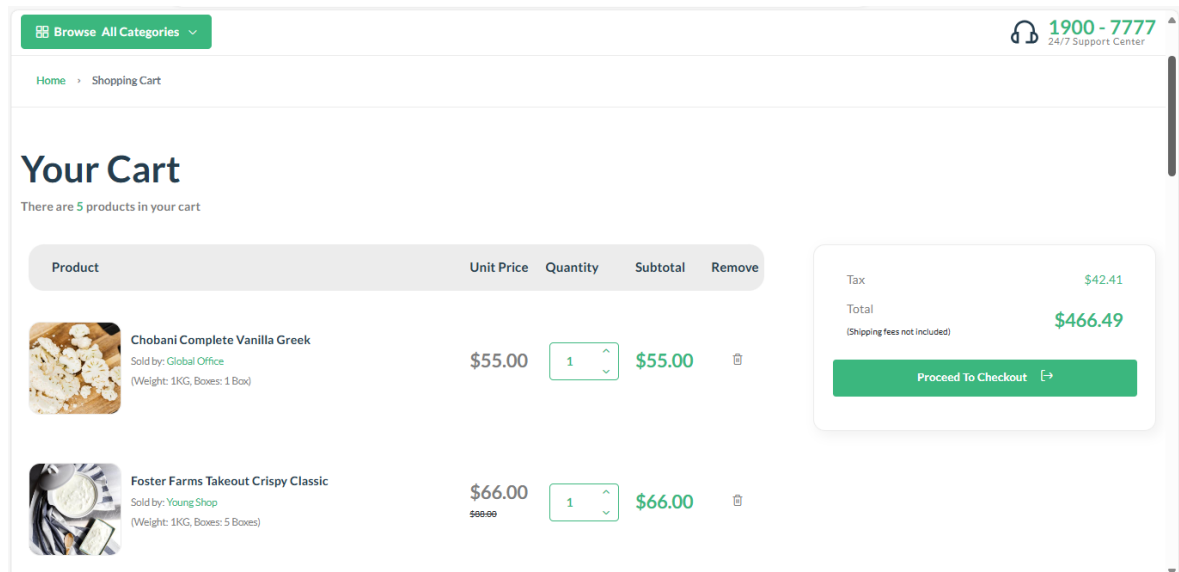
Hình 3. 2. Màn hình Xem danh mục sản phẩm.

3.1.1.3. Màn hình Xem sản phẩm.



Hình 3. 3. Màn hình Xem sản phẩm.

3.1.1.4. Màn hình Giỏ hàng.



Hình 3. 4. Màn hình Giỏ hàng.

3.1.1.5. Màn hình Thanh toán.

Thông tin vận chuyển

Họ và tên

Bùi Đình Cường

E-mail

cuongbuidinh003@gmail.com

Điện thoại

0968655898

Quốc gia

Chọn quốc gia...

Bang

Thành phố

Địa chỉ

Phương thức thanh toán

☒ Pay online via Paypal

You will be redirected to PayPal to complete the payment.

☐ Thanh toán khi nhận hàng (COD)

☐ Chuyển khoản qua ngân hàng

Ghi chú đặt hàng

Ghi chú về đơn đặt hàng của bạn, ví dụ: Chỉ giao hàng giữ hành chính

☐ Yêu cầu xuất hóa đơn công ty (Vui lòng điền thông tin công ty bạn để nhận hóa đơn)?

[← Quay lại giỏ hàng](#)
[Tiến hành thanh toán](#)

1

Encore Seafoods Stuffed Alaskan

(Weight: 3KG, Boxes: 5 Boxes)

\$87.78

Phương thức vận chuyển:

☒ Free delivery - **Miễn phí vận chuyển**

Tạm tính:

\$87.78

Thuế (VAT - 10%):

\$8.78

Phí vận chuyển:

\$0.00

Tổng cộng:

\$96.56

[Bạn có mã phiếu giảm giá?](#)

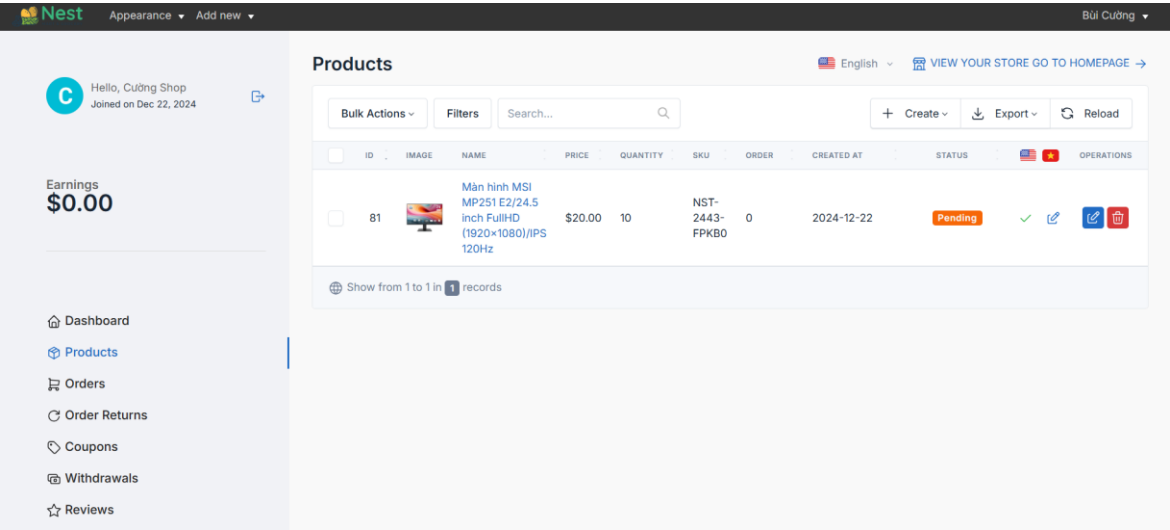
Hình 3. 5. Màn hình Thanh toán.

3.1.2. Giao diện phía nhà cung cấp (supplier).

3.1.2.1. Màn hình Bảng điều khiển – Nhà cung cấp.

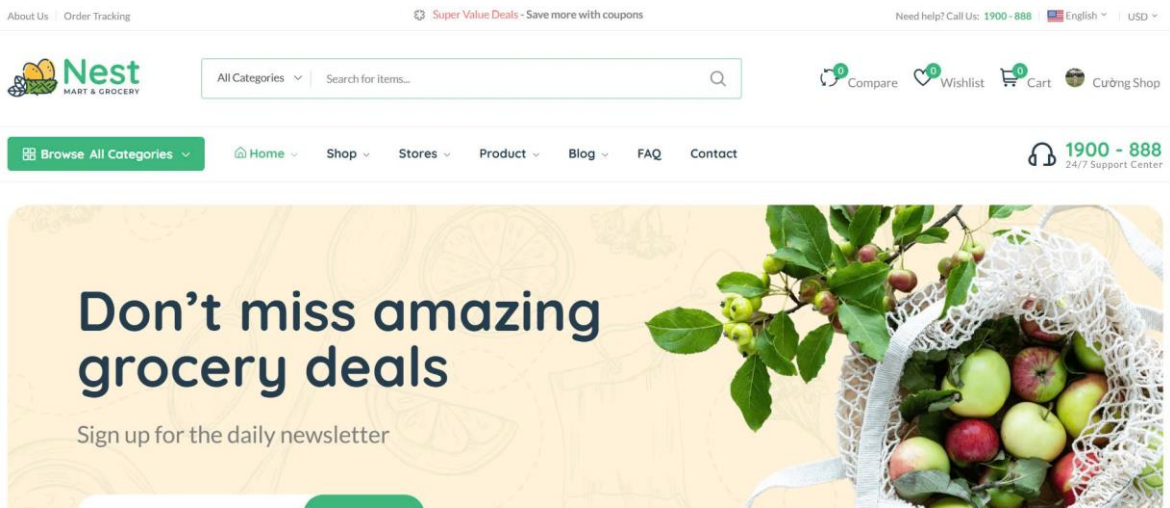
Hình 3. 6. Màn hình Bảng điều khiển – Nhà cung cấp.

3.1.2.2. Màn hình Quản lý sản phẩm – Nhà cung cấp.



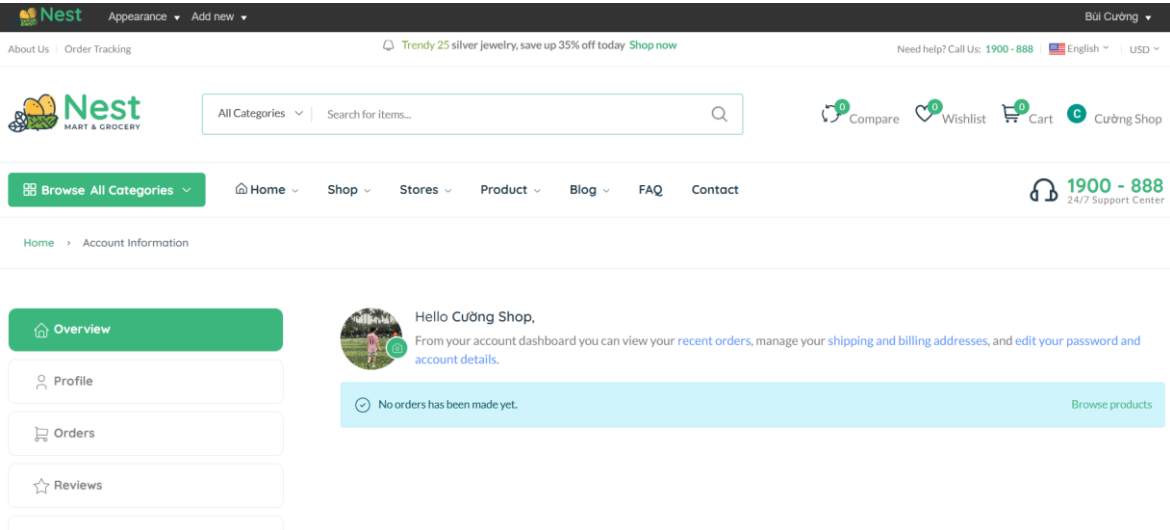
Hình 3. 7. Màn hình Quản lý sản phẩm – Nhà cung cấp.

3.1.2.3. Màn hình Trang chủ cửa hàng.



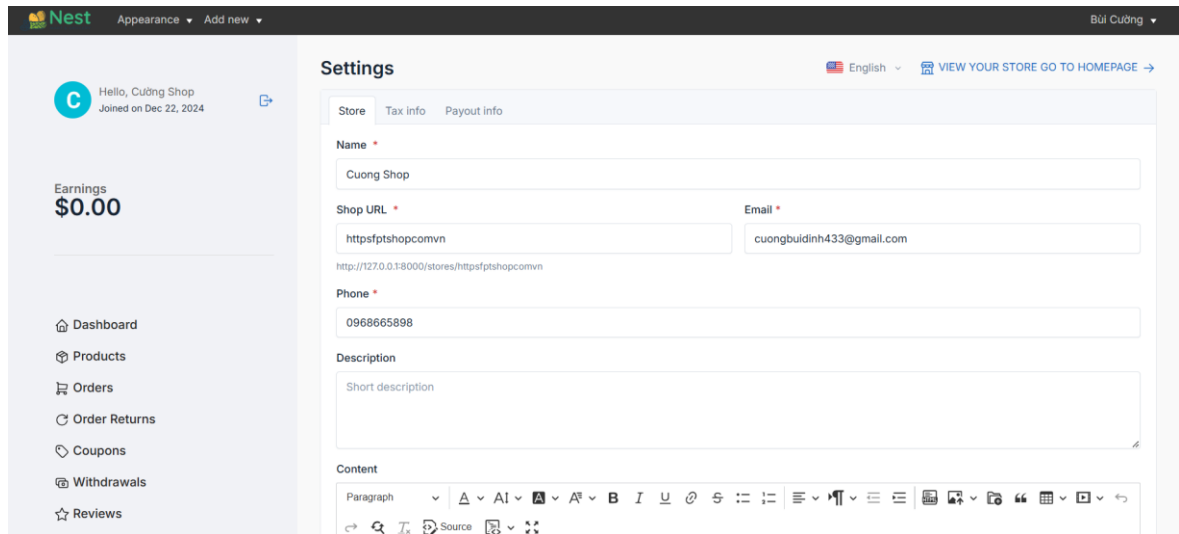
Hình 3. 8. Màn hình Trang chủ cửa hàng.

3.1.2.4. Màn hình Quản lý tài khoản.



Hình 3. 9. Màn hình Quản lý tài khoản.

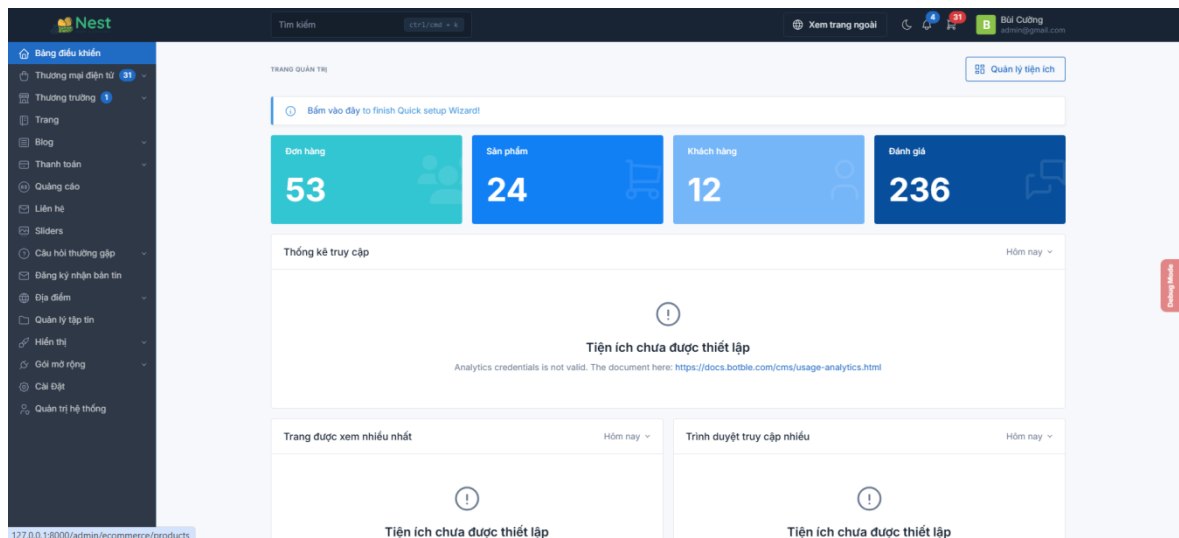
3.1.2.5. Màn hình Cài đặt.



Hình 3. 10. Màn hình Cài đặt.

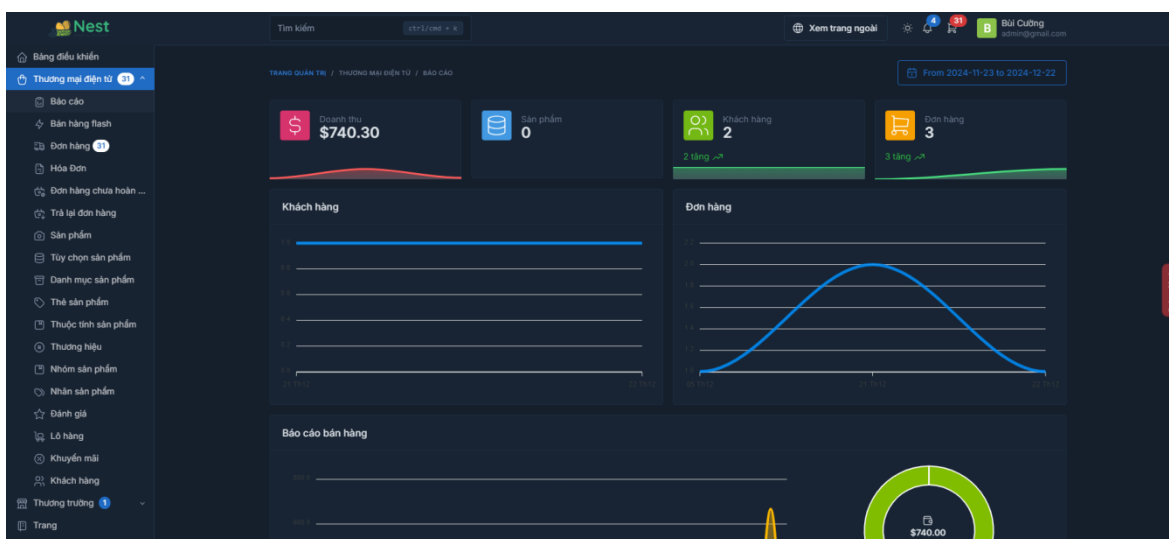
3.1.3. Giao diện phía quản trị viên (admin).

3.1.3.1. Màn hình Bảng điều khiển – Quản trị viên.



Hình 3. 11. Màn hình Bảng điều khiển – Quản trị viên.

3.1.3.2. Màn hình Thống kê.



Hình 3. 12. Màn hình Thống kê.

3.1.3.3. Màn hình Quản lý sản phẩm – Quản trị viên.

The product management interface displays a table of products. The table has columns for ID, Name, Price, Status, Quantity, Product Code, Date, and Action. The table lists several products including 'Signature Wood-Fired Mushroom and Caramelized', 'Colorful Banana', 'Organic Cage-Free Grade A Large Brown Eggs', 'Chen Watermelon (Digital)', 'Purpure Simply Smart Organics', and 'Simply Lemonade with Raspberry'.

ID	Hình ảnh	Sản phẩm	Giá cơ bản	Tình trạng tồn kho	Số lượng	Mã sản phẩm	Thời gian	Ngày tạo	Trạng thái	Tác vụ
24		Signature Wood-Fired Mushroom and Caramelized — Thuộc vật chất	\$93.48 \$114.00	Còn hàng	17	HS-158-A0	0	2024-02-24	Đã xuất bản	
23		Colorful Banana — Thuộc vật chất	\$121.00	Còn hàng	40	HS-110-A0	0	2024-02-24	Đã xuất bản	
22		Organic Cage-Free Grade A Large Brown Eggs — Thuộc vật chất	\$116.00	Còn hàng	19	HS-178-A0	0	2024-02-24	Đã xuất bản	
21		Chen Watermelon (Digital) — Điện tử	\$126.00	Còn hàng	38	HS-161-A0	0	2024-02-24	Đã xuất bản	
20		Purpure Simply Smart Organics Gluten Free — Thuộc vật chất	\$94.80 \$110.00	Còn hàng	44	HS-122-A0	0	2024-02-24	Đã xuất bản	
19		Simply Lemonade with Raspberry	\$112.00	Còn hàng	85	HS-121-A0	0	2024-02-24	Đã xuất bản	

Hình 3. 13. Màn hình Quản lý sản phẩm – Quản trị viên.

3.1.3.4. Màn hình Quản lý đơn hàng.

The order management interface displays a table of orders. The table has columns for ID, Customer Name, Email, Phone Number, Total Amount, Product Price, Shipping Fee, Payment Method, and Action. The table lists several orders including those from 'Bùi Cường', 'admin', 'Mr. Davonte Borer', and 'Miss Callie Zboncak'.

ID	Khách hàng	E-mail	Số điện thoại	Tổng cộng	Giá sản phẩm	Phí vận chuyển	Phương thức thanh toán	Tác vụ
53	Bùi Cường	cuong@gmail.com	123456789	\$242.00	\$22.00	\$0.00	PayPal	
52	Bùi Cường	cuong@gmail.com	0123456789	\$425.70	\$38.70	\$0.00	PayPal	
51	admin	nguyenvana@gmail.com	0847734562	\$72.80	\$6.60	\$0.00	PayPal	
50	Mr. Davonte Borer	brand79@example.com	+17045049936	\$112.00	\$0.00	\$0.00	Razorpay	
49	Mr. Davonte Borer	brand79@example.com	+17045049936	\$782.00	\$0.00	\$0.00	Thanh toán khi giao hàng (COD)	
48	Miss Callie Zboncak	nhilpert@example.org	+14129420298	\$232.00	\$0.00	\$0.00	Stripe	
47	Miss Callie Zboncak	nhilpert@example.org	+14129420298	\$330.00	\$0.00	\$0.00	PayPal	
46	Miss Callie Zboncak	nhilpert@example.org	+14129420298	\$224.00	\$0.00	\$0.00	SeCommerce	
45	Miss Callie Zboncak	nhilpert@example.org	+14129420298	\$250.00	\$0.00	\$0.00	PayPal	

Hình 3. 14. Màn hình Quản lý đơn hàng.

- Liệt kê những yêu cầu cho việc kiểm thử (Test Requirements).
- Những chiến lược kiểm thử nên được sử dụng.
- Ước lượng những yêu cầu về tài nguyên và chi phí cho việc kiểm thử.
- Những tài liệu được lập sau khi hoàn thành việc kiểm thử.

3.2.1.2. Tổng quan.

Website thương mại điện tử là một nền tảng trực tuyến cho phép khách hàng mua sắm và giao dịch các sản phẩm cần thiết thông qua nền tảng trực tuyến. Đây là một hình thức mua sắm tiện lợi và linh hoạt, cho phép người dùng lựa chọn và đặt hàng từ một loạt các sản phẩm có sẵn trên trang web.

3.2.1.3. Phạm vi.

Tài liệu kế hoạch kiểm thử này được áp dụng cho việc kiểm thử chức năng và giao diện của Website thương mại điện tử Nest được đặc tả trong tài liệu học phần Đồ án tốt nghiệp.

3.2.1.4. Định nghĩa và các từ viết tắt.

Bảng 3. 2. Định nghĩa và các từ viết tắt.

Thuật ngữ	Định nghĩa
Test Designer/Tester	Người thiết kế kiểm thử/Kiểm thử viên
Từ viết tắt	Mô tả
UC	Use case
TC	Test case
GUI	Giao diện người dùng
ST	System test
UAT	User acceptance test
TR	Test report

3.2.1.5. Đối tượng sử dụng tài liệu.

Tài liệu này được sử dụng cho Test Manager, Test Designer và Tester.

3.2.2. Tài liệu tham khảo.

Bảng 3. 3. Tài liệu tham khảo.

Tên tài liệu	Tác giả	Phiên bản	Ngày duyệt
Tài liệu mô tả yêu cầu hệ thống	Bùi Đình Cường	05	31/11/2024
Mẫu kế hoạch kiểm thử	Bùi Đình Cường	01	31/11/2024

Mẫu test case	Bùi Đình Cường	03	31/11/2024
Mẫu test plan	Bùi Đình Cường	02	31/11/2024
Mẫu test report	Bùi Đình Cường	03	31/11/2024

3.2.3. Lịch trình công việc.

Bảng 3. 4. Lịch trình công việc.

Mốc công việc	Sản phẩm	Thời gian	Bắt đầu	Kết thúc
Lập kế hoạch kiểm thử	Test plan	2 ngày	31/11/2024	02/12/2024
Xem lại các tài liệu	Test plan	2 ngày	02/12/2024	04/12/2024
Thiết kế các testcase	Test case	1 ngày	04/12/2024	06/12/2024
Viết các testcase	Test case	2 ngày	07/12/2024	09/12/2024
Xem lại các testcase	Test case	1 ngày	10/12/2024	11/12/2024
Thực thi các testcase	Test case	1 ngày	12/12/2024	13/12/2024
Ghi nhận và đánh giá kết quả kiểm thử	Test report	2 ngày	14/12/2024	16/12/2024

3.2.4. Những yêu cầu về tài nguyên.

3.2.4.1. Phần cứng.

Bảng 3. 5. Yêu cầu về phần cứng.

CPU	RAM	HDD	Architecture
Intel Core i5, 2.3GHz	4GB	360GB	64 bit

3.2.4.2. Phần mềm.

Bảng 3. 6. Yêu cầu về phần mềm.

Tên phần mềm	Phiên bản	Loại
Microsoft Windows 11 Professional	11	Hệ điều hành

3.2.4.3. Công cụ kiểm thử.

Bảng 3. 7. Công cụ kiểm thử.

Hoạt động	Công cụ	Nhà cung cấp	Phiên bản
Quản lý Test Case	Microsoft Office Excel	Microsoft	2024
Quản lý phiên bản	Microsoft Office Word	Microsoft	2024
Theo dõi lỗi	Microsoft Office Excel	Microsoft	2024

3.2.4.4. Môi trường kiểm thử.

- Máy tính cá nhân có cài phần mềm MySQL Workbench và Visual Studio Code.
- Các giao diện của hệ thống được kiểm tra trên các hệ điều hành Windows, MacOS, Linux.

3.2.4.5. Nhân sự.

Bảng 3. 8. Nhân sự kiểm thử.

Nhân sự	Vai trò
Bùi Đình Cường	Test Designer / Tester: Lập kế hoạch kiểm thử, quản lý tiến độ hoạt động kiểm thử, thiết kế các test case bổ sung và thực thi các test case bổ sung.
	Test Designer / Tester: Thiết kế và viết các test case, thực thi các test case cho các chức năng, xem lại Test Plan.

3.2.5. Phạm vi kiểm thử.

3.2.5.1. Chức năng được kiểm thử.

- Đăng nhập: Kiểm tra chức năng đăng nhập với tài khoản đã tạo.
- Đăng ký: Tạo tài khoản mới với thông tin của người dùng.
- Giỏ hàng: Kiểm tra chức năng giỏ hàng của tài khoản đã tạo.

3.2.5.2. Giao diện được kiểm thử.

- Đăng nhập.
- Tạo tài khoản.
- Giỏ hàng.

3.2.6. Chiến lược kiểm thử.

3.2.6.1. Các giai đoạn kiểm thử.

Kiểm thử ở mức hệ thống (system test):

- Dùng kiểu kiểm thử thủ công (manual test) bao gồm kiểm thử giao diện và chức năng.

- Việc kiểm thử chỉ bắt đầu khi đã hoàn thiện bộ test case để kiểm thử giao diện và chức năng.

- Thiết kế test case theo phương pháp phân vùng tương đương.
- Chỉ thực hiện kiểm thử hồi quy, không thực hiện kiểm thử lại.

3.2.6.2. Các loại kiểm thử.

- Kiểm thử chức năng.

Bảng 3. 9. Kiểm thử chức năng.

Mục đích kiểm tra	Đảm bảo các chức năng được kiểm tra hoạt động chính xác theo đặc tả yêu cầu
Kỹ thuật	Thực thi tất cả các trường hợp có thể cho mỗi nhóm chức năng, sử dụng dữ liệu hợp lệ và không hợp lệ để xác định: <ul style="list-style-type: none"> - Kết quả mong đợi khi dữ liệu hợp lệ được sử dụng. - Cảnh báo phù hợp hiện ra khi dữ liệu không hợp lệ được sử dụng.
Tiêu chuẩn dừng	Tất cả các test case đã được thiết kế đều được thực thi. Tất cả các lỗi tìm thấy đều được ghi nhận lý do rõ ràng để có thể giúp cho lập trình viên khắc phục.
Chịu trách nhiệm kiểm thử	Test Designer / Tester
Cách kiểm thử	Kiểm thử bằng tay thủ công, tuân tự theo các bước được định nghĩa trong test case
Xử lý ngoại lệ	Liệt kê tất cả các vấn đề liên quan phát sinh trong quá trình thực thi kiểm thử.

- Kiểm thử giao diện.

Bảng 3. 10. Kiểm thử giao diện.

Mục đích kiểm tra	Đảm bảo các giao diện được hiển thị đúng với thiết kế về kiểu hiển thị, tính hợp lệ về dữ liệu, tính thích ứng khi chạy trên các trình duyệt khác nhau
--------------------------	--

Kỹ thuật	Thực thi tất cả các trường hợp có thể cho các hệ điều hành khác nhau, sử dụng dữ liệu hợp lệ và không hợp lệ để xác định: - Kết quả mong đợi khi dữ liệu hợp lệ được sử dụng. - Cảnh báo phù hợp hiện ra khi dữ liệu không hợp lệ được sử dụng.
Tiêu chuẩn dừng	Tất cả các test case đã được thiết kế đều được thực thi. Tất cả các lỗi tìm thấy đều được ghi nhận lý do rõ ràng để có thể giúp cho lập trình viên khắc phục.
Chịu trách nhiệm kiểm thử	Test Designer / Tester
Cách kiểm thử	Kiểm thử bằng tay thủ công, tuân tự theo các bước được định nghĩa trong test case
Xử lý ngoại lệ	Liệt kê tất cả các vấn đề liên quan phát sinh trong quá trình thực thi kiểm thử.

3.2.7. Đánh giá rủi ro.

Bảng 3. 11. Các mức đánh giá rủi ro.

STT	Rủi ro	Cách khắc phục	Mức độ rủi ro
1	Thay đổi yêu cầu làm ảnh hưởng đến nguồn nhân lực và chiến lược test.	Lập lại plan sao cho phù hợp với lịch trình thực tế khi thay đổi yêu cầu, có thể chọn cách tăng thêm nguồn nhân lực cho dự án, hoặc tăng thời gian làm việc ngoài giờ cho nhân viên.	Cao
2	Sản phẩm mà lập trình viên thực hiện không kịp theo thời gian như lịch trình đề ra.	Yêu cầu cập nhật tiến độ công việc thường xuyên để quản lý kịp thời các thay đổi về thời gian và kỹ thuật.	Cao

3	Thiếu nguồn nhân lực do có thành viên đột ngột xin nghỉ.	Bổ sung thêm nguồn nhân lực mới hoặc tăng giờ làm việc ngoài giờ.	Thấp
---	--	---	------

3.2.8. Điều kiện chấp nhận kiểm thử.

- Tỷ lệ test case đạt (passed): > 90%
- Tỷ lệ test case không đạt (failed): < 10%
- Hệ thống chạy ổn định trên các hệ điều hành khác nhau (Windows, MacOS, Linux).

3.2.9. Theo dõi lỗi.

3.2.9.1. Phân loại lỗi.

Bảng 3. 12. Các mức phân loại lỗi.

Mức độ nghiêm trọng	Đặc tả lỗi
High	<ul style="list-style-type: none"> - Không thể đăng nhập - Không thể tạo tài khoản - Không thể đăng xuất - Không thể hiển thị ảnh được chọn - Không thể hiển thị video theo thời gian thực - Không đóng được thông báo lỗi
Medium	<ul style="list-style-type: none"> - Bỏ trống trường tên đăng nhập hoặc mật khẩu nhưng không thông báo lỗi. - Không điều chỉnh được kích thước ảnh hiển thị.
Low	<ul style="list-style-type: none"> - Xuất hiện lỗi chính tả trên màn hình - Vỡ form - Thông báo lỗi không đúng đặc tả - Không thể điều hướng màn hình bằng phím tắt

3.2.9.2. Quy trình xử lý lỗi.

- Ghi nhận lại các lỗi được tìm thấy trong quá trình kiểm thử.
- Xác nhận lại lỗi
- Sửa lỗi

- Kiểm thử hồi quy.
- Cập nhật trạng thái lỗi.

3.3. Trường hợp kiểm thử.

3.3.1. Kiểm thử chức năng.

Bảng 3. 13. Kiểm thử chức năng.

STT	Test case ID	Mô tả	Dữ liệu đầu vào	Kết quả đầu ra	Trạng thái
Chức năng đăng nhập					
1	TC01	Đăng nhập với tài khoản hợp lệ	Username: user1 Password: pass123	Chuyển hướng đến trang chủ	Passed
2	TC02	Đăng nhập với mật khẩu sai	Username: user1 Password: failpass	Hiển thị thông báo lỗi "Sai mật khẩu"	Passed
Chức năng đăng ký					
3	TC03	Đăng ký tài khoản mới	Username: newuser1 Email: n@gmail.com Password: newpass	Hiển thị thông báo "Đăng ký thành công" và chuyển hướng đến trang đăng nhập	Passed
4	TC04	Đăng ký tài khoản với email đã tồn tại	Username: newuser1 Email: old@gmail.com Password: newpass	Hiển thị thông báo lỗi "Email đã được sử dụng"	Passed
5	TC05	Reset mật khẩu	Email: user1@example.com	Hiển thị thông báo "Vui lòng"	Passed

				kiểm tra email để lấy lại mật khẩu"	
Chức năng thêm sản phẩm					
6	TC06	Thêm sản phẩm mới với dữ liệu hợp lệ	Tên sản phẩm: A Giá: 100000 Danh mục: Điện tử Mô tả: Chất lượng cao	Hiển thị thông báo "Thêm sản phẩm thành công"	Passed
7	TC07	Thêm sản phẩm mới thiếu tên sản phẩm	Giá: 100000 Danh mục: Điện tử Mô tả: Chất lượng cao	Hiển thị thông báo lỗi "Tên sản phẩm không được để trống"	Passed
8	TC08	Thêm sản phẩm mới với giá trị âm	Tên sản phẩm: B Giá: -50000 Danh mục: Thời trang Mô tả: Giảm giá	Hiển thị thông báo lỗi "Giá sản phẩm không hợp lệ"	Passed
Chức năng đặt hàng					
9	TC09	Đặt hàng với giỏ hàng có sản phẩm	Sản phẩm: A Số lượng: 2 Địa chỉ giao hàng: 123 Street	Hiển thị thông báo "Đặt hàng thành công"	Passed
10	TC10	Đặt hàng với giỏ hàng trống	Giỏ hàng: Trống	Hiển thị thông báo lỗi "Giỏ hàng của bạn đang trống"	Passed
11	TC11	Đặt hàng với địa chỉ giao hàng sai	Sản phẩm: B Số lượng: 1 Địa chỉ giao hàng: ***	Hiển thị thông báo lỗi "Địa chỉ giao hàng không hợp lệ"	Passed

Chức năng tìm kiếm sản phẩm					
12	TC12	Tìm kiếm sản phẩm có trong kho	Từ khóa: "Đồ ăn"	Hiển thị danh sách sản phẩm liên quan	Passed
13	TC13	Tìm kiếm sản phẩm không có trong kho	Từ khóa: "Máy bay"	Hiển thị thông báo "Không tìm thấy sản phẩm"	Passed
14	TC14	Tìm kiếm sản phẩm với từ khóa rỗng	Từ khóa: ""	Hiển thị thông báo lỗi "Vui lòng nhập từ khóa"	Passed
Chức năng quản lý đơn hàng của nhà cung cấp					
15	TC15	Xem danh sách đơn hàng	Nhà cung cấp: Z	Hiển thị danh sách các đơn hàng của nhà cung cấp	Passed
16	TC16	Cập nhật trạng thái đơn hàng	Mã đơn hàng: 12345 Trạng thái: Đang giao	Hiển thị thông báo "Cập nhật trạng thái thành công"	Passed
17	TC17	Cập nhật trạng thái đơn hàng với mã sai	Mã đơn hàng: 99999 Trạng thái: Đang giao	Hiển thị thông báo lỗi "Không tìm thấy đơn hàng"	Passed
Chức năng quản lý tài khoản người dùng					
18	TC18	Thay đổi thông tin tài khoản	Tên: Người dùng 1 Email: u1@gmail.com	Hiển thị thông báo "Cập nhật thông tin thành công"	Passed

19	TC19	Thay đổi thông tin tài khoản với email đã tồn tại	Tên: Người dùng 2 Email: u2@gmail.com	Hiển thị thông báo lỗi "Email đã được sử dụng"	Passed
20	TC20	Đổi mật khẩu	Mật khẩu cũ: oldpass Mật khẩu mới: new123	Hiển thị thông báo "Đổi mật khẩu thành công"	Passed
21	TC21	Đổi mật khẩu với mật khẩu cũ sai	Mật khẩu cũ: failpass Mật khẩu mới: new123	Hiển thị thông báo lỗi "Mật khẩu cũ không đúng"	Passed

3.3.2. Kiểm thử phi chức năng.

Bảng 3. 14. Kiểm thử phi chức năng.

STT	Test case ID	Mô tả	Dữ liệu đầu vào	Kết quả đầu ra	Trạng thái
Hiệu năng (Performance)					
1	NFR 01	Kiểm thử thời gian tải trang chủ	Truy cập trang chủ	Thời gian tải trang dưới 2 giây	Passed
2	NFR 02	Kiểm thử thời gian tải trang sản phẩm	Truy cập trang sản phẩm cụ thể	Thời gian tải trang dưới 2 giây	Passed
3	NFR 03	Kiểm thử thời gian xử lý đặt hàng	Thực hiện đặt hàng	Thời gian xử lý đơn hàng dưới 3 giây	Passed

Bảo mật (Security)					
4	NFR 04	Kiểm thử SQL Injection trên trang đăng nhập	Username: ' OR '1'='1 Password: ' OR '1'='1	Hệ thống từ chối truy cập và hiển thị lỗi	Passed
5	NFR 05	Kiểm thử XSS trên trang sản phẩm	Tên sản phẩm: <script>alert('XSS')</script>	Hệ thống không thực thi mã JavaScript	Passed
Khả dụng (Availability)					
6	NFR 06	Kiểm thử tính sẵn sàng của hệ thống	Không có	Hệ thống hoạt động ổn định trong 1 giờ liên tục	Passed
7	NFR 07	Kiểm thử khôi phục hệ thống sau sự cố	Gây ra sự cố	Hệ thống khôi phục hoàn toàn trong vòng 1 phút	Passed
Tính tương thích (Compatibility)					
8	NFR 08	Kiểm thử tương thích với trình duyệt Chrome	Sử dụng trình duyệt Chrome	Hệ thống hoạt động bình thường	Passed
9	NFR 09	Kiểm thử tương thích với trình duyệt Edge	Sử dụng trình duyệt Edge	Hệ thống hoạt động bình thường	Passed
Tính khả dụng (Usability)					
10	NFR 10	Kiểm thử khả năng điều hướng trang chủ dễ dàng	Người dùng mới	Người dùng có thể tìm thấy các mục chính trong vòng 3 phút	Passed

11	NFR 11	Kiểm thử độ phản hồi của giao diện người dùng	Thực hiện các thao tác trên giao diện	Hệ thống phản hồi trong vòng 1 giây cho các thao tác	Passed
----	-----------	---	---------------------------------------	--	--------

3.3.3. Kiểm thử giao diện.

Bảng 3. 15. Kiểm thử giao diện.

STT	Test case ID	Mô tả	Dữ liệu đầu vào	Kết quả đầu ra	Trạng thái
Màn hình trang chủ					
1	UI_TC01	Kiểm tra hiển thị chính xác các danh mục sản phẩm	Truy cập trang chủ	Các danh mục sản phẩm hiển thị đầy đủ	Passed
2	UI_TC02	Kiểm tra tính năng tìm kiếm sản phẩm	Nhập từ khóa tìm kiếm vào ô tìm kiếm	Các sản phẩm liên quan được hiển thị đúng	Passed
Màn hình xem sản phẩm					
3	UI_TC03	Kiểm tra hiển thị chi tiết sản phẩm	Truy cập trang chi tiết của một sản phẩm	Thông tin sản phẩm hiển thị đầy đủ	Passed
4	UI_TC04	Kiểm tra tính năng thêm sản phẩm vào giỏ hàng	Nhấn nút "Thêm vào giỏ hàng" trên trang chi tiết sản phẩm	Sản phẩm được thêm vào giỏ hàng thành công	Passed
Màn hình giỏ hàng					
5	UI_TC05	Kiểm tra hiển thị các sản phẩm trong giỏ hàng	Truy cập trang giỏ hàng	Các sản phẩm trong giỏ hàng hiển thị đúng	Passed

6	UI_ TC06	Kiểm tra tính năng cập nhật số lượng sản phẩm trong giỏ hàng	Thay đổi số lượng của một sản phẩm trong giỏ hàng	Số lượng sản phẩm trong giỏ hàng được cập nhật thành công	Passed
Màn hình thanh toán					
7	UI_ TC07	Kiểm tra tính năng điền thông tin thanh toán	Điền thông tin thanh toán trên trang thanh toán	Thông tin thanh toán điền đầy đủ và chính xác	Passed
8	UI_ TC08	Kiểm tra tính năng xác nhận đơn hàng	Nhấn nút xác nhận đơn hàng	Đơn hàng được xác nhận thành công	Passed
Màn hình tìm kiếm					
9	UI_ TC09	Kiểm tra hiển thị kết quả tìm kiếm	Tìm kiếm sản phẩm	Kết quả tìm kiếm hiển thị chính xác và đầy đủ	Passed
10	UI_ TC10	Kiểm tra thông báo khi không tìm thấy kết quả	Tìm kiếm không có kết quả	Hiển thị thông báo "Không tìm thấy kết quả nào"	Passed
Màn hình danh mục sản phẩm					
11	UI_ TC11	Kiểm tra hiển thị các sản phẩm thuộc danh mục	Chọn một danh mục	Các sản phẩm thuộc danh mục hiển thị đúng	Passed
12	UI_ TC12	Kiểm tra hiển thị số lượng	Trang danh mục sản phẩm	Số lượng sản phẩm hiển thị	Passed

		sản phẩm trên trang		phù hợp với cấu hình	
Màn hình chi tiết sản phẩm					
13	UI_ TC13	Kiểm tra hiển thị hình ảnh và mô tả sản phẩm	Xem chi tiết sản phẩm	Hình ảnh và mô tả sản phẩm hiển thị đúng	Passed
14	UI_ TC14	Kiểm tra tính năng phóng to hình ảnh sản phẩm	Click vào hình ảnh	Hình ảnh được phóng to với kích thước lớn	Passed
Màn hình quản lý tài khoản					
15	UI_ TC15	Kiểm tra hiển thị thông tin người dùng đăng nhập	Đăng nhập tài khoản	Thông tin người dùng hiển thị đúng và đầy đủ	Passed
16	UI_ TC16	Kiểm tra tính năng đăng xuất	Nhấn nút đăng xuất	Người dùng được đăng xuất khỏi tài khoản	Passed
Màn hình quản lý đơn hàng					
17	UI_ TC17	Kiểm tra hiển thị danh sách đơn hàng	Truy cập trang quản lý đơn hàng	Danh sách đơn hàng hiển thị đúng	Passed
18	UI_ TC18	Kiểm tra tính năng xem chi tiết đơn hàng	Click vào một đơn hàng	Chi tiết đơn hàng hiển thị đúng và đầy đủ	Passed

3.4. Báo cáo kiểm thử.

3.4.1. Thông tin chung.

- Tên dự án: Nest - Website thương mại điện tử.
- Phiên bản kiểm thử: 1.1.7.

- Ngày lập báo cáo: 16/12/2024.
- Trưởng nhóm kiểm thử: Bùi Đình Cường.
- Thành viên đội kiểm thử: Bùi Đình Cường.

3.4.2. Mục tiêu.

Mục tiêu của báo cáo kiểm thử này là cung cấp thông tin về kết quả kiểm thử của phiên bản 1.1.7, xác định các lỗi và đề xuất các biện pháp khắc phục nhằm đảm bảo chất lượng sản phẩm.

3.4.3. Phạm vi.

Phạm vi kiểm thử bao gồm các chức năng và tính năng của hệ thống, các yêu cầu đã được kiểm thử và những yêu cầu chưa được kiểm thử:

- Kiểm Thử Chức Năng:
 - + Chức năng đăng nhập.
 - + Chức năng đăng ký.
 - + Chức năng thêm sản phẩm.
 - + Chức năng đặt hàng.
 - + Chức năng tìm kiếm sản phẩm.
 - + Chức năng quản lý đơn hàng của nhà cung cấp.
 - + Chức năng quản lý tài khoản người dùng.
- Kiểm Thử Phi Chức Năng:
 - + Hiệu năng (Performance).
 - + Bảo mật (Security).
 - + Khả dụng (Availability).
 - + Tính tương thích (Compatibility).
 - + Tính khả dụng (Usability).
- Kiểm thử giao diện:
 - + Màn hình trang chủ.
 - + Màn hình xem sản phẩm.
 - + Màn hình giỏ hàng.
 - + Màn hình thanh toán.

- + Màn hình tìm kiếm.
- + Màn hình danh mục sản phẩm.
- + Màn hình chi tiết sản phẩm.
- + Màn hình quản lý tài khoản.
- + Màn hình quản lý đơn hàng.

3.4.4. Tổng quan kết quả kiểm thử.

Bảng 3. 16. Tổng quan kết quả kiểm thử.

Loại kiểm thử	Tổng số ca kiểm thử	Đã qua	Không qua	Bị chặn	Chưa thực hiện
Kiểm thử chức năng	21	21	0	0	0
Kiểm thử phi chức năng	11	11	0	0	0
Kiểm thử giao diện	18	18	0	0	0
Tổng số	50	50	0	0	0

3.4.5. Chi tiết kết quả kiểm thử.

Bảng 3. 17. Chi tiết kết quả kiểm thử.

Kiểm thử chức năng			
ID test case	Mô tả	Trạng thái	Ghi chú
TC01	Đăng nhập với tài khoản hợp lệ	Passed	
TC02	Đăng nhập với mật khẩu sai	Passed	
TC03	Đăng ký tài khoản mới	Passed	
TC04	Đăng ký tài khoản với email đã tồn tại	Passed	
TC05	Reset mật khẩu	Passed	
TC06	Thêm sản phẩm mới với dữ liệu hợp lệ	Passed	
TC07	Thêm sản phẩm mới thiếu tên sản phẩm	Passed	

TC08	Thêm sản phẩm mới với giá trị âm	Passed	
TC09	Đặt hàng với giỏ hàng có sản phẩm	Passed	
TC10	Đặt hàng với giỏ hàng trống	Passed	
TC11	Đặt hàng với địa chỉ giao hàng sai	Passed	
TC12	Tìm kiếm sản phẩm có trong kho	Passed	
TC13	Tìm kiếm sản phẩm không có trong kho	Passed	
TC14	Tìm kiếm sản phẩm với từ khóa rỗng	Passed	
TC15	Xem danh sách đơn hàng	Passed	
TC16	Cập nhật trạng thái đơn hàng	Passed	
TC17	Cập nhật trạng thái đơn hàng với mã sai	Passed	
TC18	Thay đổi thông tin tài khoản	Passed	
TC19	Thay đổi thông tin tài khoản với email đã tồn tại	Passed	
TC20	Đổi mật khẩu	Passed	
TC21	Đổi mật khẩu với mật khẩu cũ sai	Passed	
Kiểm thử phi chức năng			
ID Test case	Mô tả	Trạng thái	Ghi chú
NFR01	Kiểm thử thời gian tải trang chủ	Passed	
NFR02	Kiểm thử thời gian tải trang sản phẩm	Passed	
NFR03	Kiểm thử thời gian xử lý đặt hàng	Passed	
NFR04	Kiểm thử SQL Injection trên trang đăng nhập	Passed	
NFR05	Kiểm thử XSS trên trang sản phẩm	Passed	
NFR06	Kiểm thử tính sẵn sàng của hệ thống	Passed	
NFR07	Kiểm thử khôi phục hệ thống sau sự cố	Passed	

NFR08	Kiểm thử tương thích với trình duyệt Chrome	Passed	
NFR09	Kiểm thử tương thích với trình duyệt Edge	Passed	
NFR10	Kiểm thử khả năng điều hướng trang chủ dễ dàng	Passed	
NFR11	Kiểm thử độ phản hồi của giao diện người dùng	Passed	
Kiểm thử giao diện			
ID Test case	Mô tả	Trạng thái	Ghi chú
UI_ TC01	Kiểm tra hiển thị chính xác các danh mục sản phẩm	Passed	
UI_ TC02	Kiểm tra tính năng tìm kiếm sản phẩm	Passed	
UI_ TC03	Kiểm tra hiển thị chi tiết sản phẩm	Passed	
UI_ TC04	Kiểm tra tính năng thêm sản phẩm vào giỏ hàng	Passed	
UI_ TC05	Kiểm tra hiển thị các sản phẩm trong giỏ hàng	Passed	
UI_ TC06	Kiểm tra tính năng cập nhật số lượng sản phẩm trong giỏ hàng	Passed	
UI_ TC07	Kiểm tra tính năng điền thông tin thanh toán	Passed	
UI_ TC08	Kiểm tra tính năng xác nhận đơn hàng	Passed	
UI_ TC09	Kiểm tra hiển thị kết quả tìm kiếm	Passed	
UI_ TC10	Kiểm tra thông báo khi không tìm	Passed	

TC10	thấy kết quả		
UI_ TC11	Kiểm tra hiển thị các sản phẩm thuộc danh mục	Passed	
UI_ TC12	Kiểm tra hiển thị số lượng sản phẩm trên trang	Passed	
UI_ TC13	Kiểm tra hiển thị hình ảnh và mô tả sản phẩm	Passed	
UI_ TC14	Kiểm tra tính năng phóng to hình ảnh sản phẩm	Passed	
UI_ TC15	Kiểm tra hiển thị thông tin người dùng đăng nhập	Passed	
UI_ TC16	Kiểm tra tính năng đăng xuất	Passed	
UI_ TC17	Kiểm tra hiển thị danh sách đơn hàng	Passed	
UI_ TC18	Kiểm tra tính năng xem chi tiết đơn hàng	Passed	

3.4.6. Kết luận.

- Đánh Giá Chất Lượng: Hệ thống nhìn chung đạt yêu cầu chất lượng đề ra, với các chức năng chính như thêm sản phẩm vào giỏ hàng, cập nhật số lượng sản phẩm và tính tổng giá trị đơn hàng đều hoạt động chính xác. Những lỗi trên chỉ ảnh hưởng nhỏ đến trải nghiệm người dùng trong một số trường hợp đặc biệt.

- Tỷ lệ thành công: 100%.

- Tỷ lệ thất bại: 0%.

- Kế Hoạch Hành Động Tiếp Theo:

+ Cần xem xét tối ưu hóa hiệu suất tải hình ảnh để cải thiện trải nghiệm người dùng khi mạng yếu.

- + Thực hiện kiểm thử hiệu suất để đảm bảo rằng các tối ưu hóa được thực hiện không ảnh hưởng đến các chức năng khác của hệ thống.
- + Tiếp tục giám sát và cải thiện hệ thống dựa trên phản hồi của người dùng và kết quả kiểm thử.

KẾT LUẬN

Dự án tài “*Xây dựng website thương mại điện tử Nest sử dụng công nghệ Laravel Framework*” đã đạt được những thành tựu đáng kể trong việc hiểu biết và ứng dụng công nghệ, cũng như cung cấp một trải nghiệm tích hợp và thuận lợi cho người dùng. Tuy nhiên, để phát triển hơn nữa và giữ vững vị thế trong thị trường ngày càng cạnh tranh, dự án cần tiếp tục tập trung vào các mục tiêu và hướng phát triển sau:

- Mở rộng tính năng và khả năng tích hợp:

+ Tích hợp thanh toán và giao hàng: Cung cấp nhiều phương thức thanh toán và lựa chọn giao hàng linh hoạt để tăng sự thuận tiện cho khách hàng.

+ Tích hợp tính năng xem trước sản phẩm: Cho phép người dùng xem trước sản phẩm trực tuyến, tăng khả năng quyết định mua hàng.

- Tối ưu trải nghiệm người dùng:

+ Tối ưu hóa giao diện và trải nghiệm người dùng: Cải thiện giao diện người dùng để tăng tính thẩm mỹ và dễ sử dụng.

+ Tăng cường tính tương tác: Thêm các tính năng tương tác như bình luận, đánh giá sản phẩm để tạo ra cộng đồng mạng tích cực.

- Phát triển chiến lược marketing:

+ Tối ưu hóa SEO: Tăng cường hoạt động SEO để nâng cao vị trí của website trên các công cụ tìm kiếm, tăng lưu lượng truy cập.

+ Chạy chiến dịch quảng cáo: Sử dụng quảng cáo trên mạng xã hội và các kênh khác để tăng tầm nhìn và thu hút khách hàng mới.

- Quản lý và phát triển hệ thống:

+ Bảo trì và nâng cấp hệ thống: Tiếp tục bảo trì và nâng cấp hệ thống để đảm bảo hoạt động ổn định và hiệu suất cao.

+ Phát triển tính mở rộng: Đảm bảo hệ thống có khả năng mở rộng để đối phó với sự tăng trưởng của dữ liệu và người dùng.

- Phát triển hệ thống phân tích và báo cáo:

+ Phát triển hệ thống phân tích dữ liệu: Xây dựng hệ thống phân tích dữ liệu sâu hơn để cung cấp thông tin chi tiết và phân tích sâu về hành vi người dùng và hiệu suất sản phẩm.

+ Tạo ra báo cáo chuyên sâu: Phát triển báo cáo và phân tích chi tiết hơn về doanh số bán hàng, hoạt động của khách hàng để định hình chiến lược kinh doanh.

➔ Bằng cách tập trung vào những mục tiêu và hướng phát triển này, dự án sẽ tiếp tục đem lại giá trị và thu hút sự quan tâm của người dùng và doanh nghiệp, từ đó đạt được sự thành công và bền vững trong thị trường thương mại điện tử.

TÀI LIỆU THAM KHẢO

- [1]. Matt Stauffer (2019), “Laravel: Up & Running: A Framework for Building Modern PHP Apps”, O'Reilly Media, Sebastopol, California.
- [2]. Sang Nguyen (2018), “Learning Laravel: The Easiest Way”, Leanpub.
- [3]. Jaseem Abbas (2021), “Beginning Laravel: A Beginner's Guide to Application Development with Laravel 5.3”, Apress.
- [4]. Terry Matula (2013), “Laravel Application Development Cookbook”, Packt Publishing, Birmingham.
- [5]. Shawn McCool, Taylor Otwell (2015), “Laravel: From Apprentice To Artisan”, Leanpub.
- [6]. Tim Downey (2016), “Guide to Laravel: The PHP Framework for Web Artisans”, Springer, New York, NY.
- [7]. Nathan Wu (2021), “Mastering Laravel: Building Modern Web Applications Using PHP Framework”, Packt Publishing, Birmingham.

PHỤ LỤC

[1]. Link github: https://github.com/C1223bd/DATN_BDC_2021608308

[2]. Link tiến độ: <https://trello.com/b/SgtPDoLL/datnbdc2021608308>