



CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

Tp. Hồ Chí Minh, ____ tháng 12, 2020

NHIỆM VỤ TIỂU LUẬN CHUYÊN NGÀNH

Họ tên sinh viên: Nguyễn Huy Cường

MSSV: 17110107

Họ tên sinh viên: Bùi Văn Hà

MSSV: 17110129

Chuyên ngành: Công nghệ thông tin

Lớp: 17110CLST1

Giảng viên hướng dẫn: ThS. TRẦN CÔNG TÚ

Điện thoại: 0983 674 375

Ngày nhận đề tài: 27/08/2020

Ngày nộp đề tài:

1. Tên đề tài: Xây dựng website bán hàng online

2. Các số liệu, tài liệu ban đầu:

Hiện thực thuật toán bằng HTML, CSS, JS, Bootstrap, ASP.NET CORE 3.x, SQL Server, Angular.

Thiết kế trang web bán hàng trực tuyến.

Tham khảo tài liệu của TeDu (Technology Education).

3. Nội dung thực hiện đề tài:

Nghiên cứu các thiết kế được một website bán hàng trực tuyến với đầy đủ các tính năng như: xem sản phẩm, tìm kiếm, quản lý sản phẩm, quản lý giỏ hàng, thống kê bán hàng, thanh toán sau khi mua hàng, quản lý dịch vụ, quản lý nhân viên, ...

Phân tích chức năng của website và lựa chọn phương án thiết kế website phù hợp.

4. Sản phẩm:

Thiết kế và Xây dựng website bán hàng trực tuyến có đầy đủ các chức năng cơ bản.

TRƯỞNG NGÀNH

GIẢNG VIÊN HƯỚNG DẪN

(Kí và ghi rõ họ tên)



CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

Tp. Hồ Chí Minh, ____ tháng 12, 2020

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Họ và tên sinh viên: Nguyễn Huy Cường

Mã số sinh viên: 17110107

Họ và tên sinh viên: Bùi Văn Hà

Mã số sinh viên: 17110129

Ngành: Công nghệ thông tin.

Tên đề tài: Xây dựng website bán hàng online

Họ và tên giáo viên hướng dẫn: ThS. TRẦN CÔNG TÚ

NHẬN XÉT

1. Về nội dung đề tài & khối lượng thực hiện:

.....

.....

.....

2. Ưu điểm:

.....

.....

.....

3. Khuyết điểm:

.....

.....

4. Đề nghị cho bảo vệ hay không?

.....

5. Đánh giá loại:

.....

6. Điểm: (bằng chữ.....)

Tp. Hồ Chí Minh, ngày ____ tháng 12, 2020

GIẢNG VIÊN HƯỚNG DẪN



CỘNG HOÀ XÃ HỘI CHỦ NGHĨA VIỆT NAM
Độc lập – Tự do – Hạnh phúc

Tp. Hồ Chí Minh, ____ tháng 12, 2020

PHIẾU NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

Họ và tên sinh viên: Nguyễn Huy Cường

Mã số sinh viên: 17110107

Họ và tên sinh viên: Bùi Văn Hà

Mã số sinh viên: 17110129

Ngành: Công nghệ thông tin.

Tên đề tài: Xây dựng website bán hàng online

Họ và tên giáo viên phản biện:

NHẬN XÉT

7. Về nội dung đề tài & khối lượng thực hiện:

.....
.....
.....

8. Ưu điểm:

.....
.....
.....

9. Khuyết điểm:

.....
.....
.....

10. Đề nghị cho bảo vệ hay không?

.....

11. Đánh giá loại:

.....

12. Điểm: (bằng chữ.....)

Tp. Hồ Chí Minh, ngày ____ tháng 12, 2020

GIẢNG VIÊN PHẢN BIỆN

LỜI CẢM ƠN

Lời đầu tiên, nhóm thực hiện xin được phép gửi lời cảm ơn chân thành đến khoa Đào tạo Chất Lượng Cao – Trường Đại học Sư phạm Kỹ thuật Thành phố Hồ Chí Minh đã tạo mọi điều kiện thuận lợi nhất cho nhóm thực hiện có cơ hội được tự do tiếp cận, tham khảo, mở rộng thêm kiến thức trong lĩnh vực Công nghệ thông tin nói chung và môn Tiểu Luận Chuyên Ngành nói riêng.

Lời cảm ơn trân trọng nhất nhóm thực hiện xin chân thành gửi đến Thầy **Trần Công Tú** – người đã dùng mọi tâm huyết và tri thức của người Thầy, cùng đồng hành và giảng dạy, hướng dẫn và tạo mọi điều kiện thuận lợi giúp đỡ cho nhóm phát huy hết khả năng cũng như nâng cao kiến thức trong suốt quá trình học tập, đặc biệt là trong quá trình chuẩn bị và thực hiện tiểu luận chuyên ngành. Cảm ơn sự nhiệt tình của Thầy, là động lực vô cùng to lớn giúp nhóm thực hiện tiểu luận có thể thực hiện, kiên trì trong suốt quá trình thực hiện tiểu luận chuyên ngành và khám phá ra những kiến thức mới đầy thú vị và bổ ích liên quan đến đề tài tiểu luận mà cụ thể là đề tài **Xây dựng website bán hàng online**.

Cuối cùng nhóm thực hiện kính chúc quý thầy cô dồi dào sức khỏe, niềm tin để tiếp tục sự nghiệp cao quý của mình là truyền đạt kiến thức cho thế hệ mai sau.

Nhóm xin chân thành cảm ơn.

TPHCM, ngày tháng 12 năm 2020

Sinh viên thực hiện

LỜI MỞ ĐẦU

- Trong nền kinh tế thị trường, các doanh nghiệp, nhất là các doanh nghiệp trong lĩnh vực kinh doanh, đã và đang ngày càng xuất hiện ngày một nhiều. Với thị trường đầy tiềm năng và sôi động, buộc các doanh nghiệp phải cạnh tranh một cách gay gắt và sòng phẳng với nhau.
- Thế kỷ thứ XXI, nền công nghệ ngày càng hiện đại hơn, nhờ có sự kết hợp tốt giữa hai lĩnh vực kinh doanh và công nghệ, nền thương mại điện tử đã gây đòi, nhằm tạo ra một sân chơi mua bán, trao đổi hàng hóa cho người mua và các doanh nghiệp cũng có nơi để cạnh tranh, thu hút các đối tượng khách hàng cho riêng mình.
- Nhận thấy được điều này, những doanh nghiệp kinh doanh đã tìm đến các công ty phần mềm, tạo cho họ những trang website thương mại điện tử cho riêng mình. Tiếp bước cho điều đó, nhóm thực hiện đã thực hiện một đề tài tương tự, xây dựng và ứng dụng một trang website quản lý bán hàng online.

MỤC LỤC

NHIỆM VỤ TIỂU LUẬN CHUYÊN NGÀNH	I
LỜI CẢM ƠN	IV
LỜI MỞ ĐẦU	V
MỤC LỤC.....	VI
DANH MỤC CÁC TỪ, TỪ VIẾT TẮT.....	IX
DANH MỤC BẢNG BIỂU	X
DANH MỤC BIỂU ĐỒ VÀ HÌNH ẢNH	XII
CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI	1
1.1. Giới thiệu chung về đề tài.....	1
1.2. Lý do chọn đề tài	1
1.3. Mục tiêu	1
1.4. Công nghệ sử dụng	1
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	2
2.1. Angular Framework.....	2
2.2. ASP.NET Entity Framework Core	2
2.3. ASP.NET Web API	3
2.4. Microsoft SQL Server	3
CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ	4
3.1. Khảo sát hiện trạng	4
3.1.1. Trang Web Tiki.vn.....	4
3.1.2. Trang Web Lazada.vn.....	7
3.1.3. Trang Web Shopee.vn.....	10
3.1.4. Kết luận.....	13

3.2. Đặc tả hệ thống	14
3.2.1. Mô hình hóa yêu cầu.....	14
3.2.2. Thiết kế Use Case Diagram	15
3.2.3. Đặc tả Use Case	16
3.2.4. Mô tả một số chức năng qua lược đồ tuần tự (Sequence Diagram)	51
3.2.5. Mô tả Cơ sở dữ liệu của hệ thống.....	54
3.3. Thiết kế giao diện	63
3.3.1. Giao diện dành cho “Quản trị viên”	63
3.3.2. Giao diện dành cho “Khách, Thành Viên”	63
CHƯƠNG 4: CÀI ĐẶT SẢN PHẨM.....	64
4.1. Front – End	64
4.1.1. Môi trường phát triển.....	64
4.1.2. Ứng dụng Angular	64
4.2. Back – End.....	71
4.2.1. Cấu trúc Project API	71
4.2.2. Một số hàm (Function) và Procedure	72
PHẦN TỔNG KẾT	77
1. Kết quả đạt được	77
2. Ưu điểm.....	77
3. Nhược điểm.....	77
4. Khó khăn	78
5. Bài học kinh nghiệm	78
6. Hướng phát triển	78
TÀI LIỆU THAM KHẢO.....	80

PHỤ LỤC.....	81
1. Tạo một ứng dụng Angular application	81

DANH MỤC CÁC TỪ, TỪ VIẾT TẮT

Author: Tác giả

Priority: Độ bảo mật

Actor: Nhân vật

Summary: Mục đích

Goal: Kết quả

Trigger: Các ràng buộc

Pre-condition: Tiền điều kiện

Post-condition: Điều kiện trạng thái

Scenario: Các bước xử lý

Exception: Ngoại lệ

DANH MỤC BẢNG BIỂU

Bảng 1: Use Case Xem Danh Sách Sản phẩm	17
Bảng 2: Use Case Xem Chi Tiết Sản phẩm.....	18
Bảng 3: Use Case Tìm kiếm sản phẩm.....	19
Bảng 4 : Use Case Đăng nhập	21
Bảng 5: Use Case Đăng ký	23
Bảng 6: Use Case Xem và Cập nhật thông tin tài khoản	24
Bảng 7: Use Case Xem lịch sử giao dịch	26
Bảng 8: Use Case Thêm sản phẩm vào giỏ hàng	27
Bảng 9: Use Case Xóa sản phẩm đã chọn	29
Bảng 10: Use Case Thanh toán	31
Bảng 11: Use Case Thêm sản phẩm	33
Bảng 12: Use Case Chỉnh sửa thông tin sản phẩm.....	34
Bảng 13: Use Case Xóa sản phẩm.....	36
Bảng 14: Use Case Xem tất cả các giao dịch	37
Bảng 15: Use Case lọc theo Ngày, Tháng, Năm	39
Bảng 16: Use Case Xem thống kê bán sản phẩm theo Tháng.....	41
Bảng 17: Use Case Xem thống kê bán sản phẩm theo Năm	42
Bảng 18: Use Case Thêm nhân viên	44
Bảng 19: Use Case Xét quyền Admin.....	45
Bảng 20: Use Case Vô hiệu hóa quyền Admin	47
Bảng 21: Use Case Xem danh sách đơn hàng	49
Bảng 22: Use Case Nhận đơn hàng	50
Bảng 23: Bảng Sản Phẩm	57
Bảng 24: Bảng Giao Dịch.....	58
Bảng 25: Bảng Hóa Đơn	59
Bảng 26: Bảng Hình Ảnh	60
Bảng 27: Bảng User.....	60
Bảng 28: Bảng Account.....	61

Bảng 29: Bảng SanPhamInHoaDon62

DANH MỤC BIỂU ĐỒ VÀ HÌNH ẢNH

Hình 1: Sơ đồ cấu trúc của SQL Server	3
Hình 2: Sơ đồ chức năng Use Case của hệ thống	15
Hình 3: Lược đồ tuần tự Đăng ký	51
Hình 4: Lược đồ tuần tự Thêm vào giỏ hàng	52
Hình 5: Lược đồ tuần tự Thanh toán.....	53
Hình 6: Sơ đồ liên kết thực thể ERD.....	54
Hình 7: Sơ đồ quan hệ các bảng trong cơ sở dữ liệu.....	55
Hình 8 : Giao diện mẫu Quản trị viên	63
Hình 9: Giao diện mẫu Khách, Thành viên.....	63
Hình 10 : Đoạn code xử lý Đăng Nhập.....	65
Hình 11 : Đoạn code xử lý thêm người dùng mới (Đăng ký)	65
Hình 12 : Đoạn code xử lý hiển thị chi tiết một sản phẩm theo id.....	65
Hình 13 : Đoạn code xử lý thêm sản phẩm vào giỏ hàng	66
Hình 14 : Đoạn code xử lý thanh toán bằng PayPal	67
Hình 15: Cấu trúc giao diện trang Home	67
Hình 16: Cấu trúc giao diện trang Login	68
Hình 17: Cấu trúc giao diện trang Sản Phẩm.....	68
Hình 18: Cấu trúc giao diện trang Chi tiết sản phẩm.....	69
Hình 19: Cấu trúc giao diện trang Giỏ hàng	69
Hình 20: Cấu trúc giao diện trang Thông tin tài khoản.....	70
Hình 21: Cấu trúc giao diện trang Danh sách sản phẩm.....	70
Hình 22: Cấu trúc khai báo Class.....	72
Hình 23: Cấu trúc khai báo hàm API.....	73
Hình 24: Cấu trúc khai báo hàm GET all.....	73
Hình 25: Cấu trúc khai báo hàm GET theo id.....	74
Hình 26: Cấu trúc khai báo hàm POST.....	74
Hình 27: Cấu trúc khai báo hàm PUT.....	75
Hình 28: Cấu trúc khai báo hàm DELETE	76

CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI

1.1. Giới thiệu chung về đề tài

- Đề tài mà nhóm thực hiện xây dựng là thiết kế và lập trình một trang web bán hàng online, mang đầy đủ những đặc điểm, tính chất, chức năng cơ bản của một hệ thống website bán hàng Online.

1.2. Lý do chọn đề tài

- Nhằm giúp cho việc nghiên cứu về một hệ thống trang Website bán hàng, thương mại điện tử, hiểu hơn về cách thức vận hành và thiết kế giao diện cho một trang web bán hàng.
- Tăng cường khả năng lập trình, làm việc nhóm.
- Tiếp thu kiểm các kiến thức về Angular và ASP.Net Core.

1.3. Mục tiêu

- Tạo ra được một hệ thống website bán hàng trực tuyến đơn giản, phù hợp với nhu cầu của người dùng, mang đầy đủ những tính năng cơ bản của một website thương mại điện tử thông dụng.

1.4. Công nghệ sử dụng

- Công nghệ sử dụng cho Front End: Angular.
- Công nghệ sử dụng cho Back End: C# ASP.NET Core Web API
- Công nghệ sử dụng cho CSDL: Microsoft SQL Server.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1. Angular Framework

- Angular là một framework được tạo ra và hỗ trợ từ Google, giúp cho các lập trình viên trong việc lập trình và thiết kế giao diện các trang website^[6]
- Phiên bản sớm nhất của Angular đặt tên là Angular 2. Sau đó đổi tên thành Angular. Sau đó team Angular phát triển một phiên bản mới thường xuyên và bản mới nhất là Angular 10^[6].
- Angular yêu cầu hiểu biết về JavaScript, HTML và CSS, TypeScript và OOP.
- Angular là framework được xây dựng cả ứng dụng Mobile và Desktop. Nó được xây dựng sử dụng JavaScript. Người dùng sẽ có thể sử dụng nó để xây dựng ứng dụng hoàn chỉnh kết hợp với HTML, CSS và JavaScript^[6].
- Angular được load với tính năng Power-packaged. Một số tính năng được liệt kê ra đây như sau:
 - Cơ chế Two-Way Data Binding
 - Hỗ trợ cơ chế Routing mạnh mẽ
 - Mở rộng HTML
 - Thiết kế module hoá
 - Hỗ trợ làm việc với hệ thống Backend
 - Cộng đồng tốt: Angular được hỗ trợ bởi Google và cộng đồng^[6]

2.2. ASP.NET Entity Framework Core

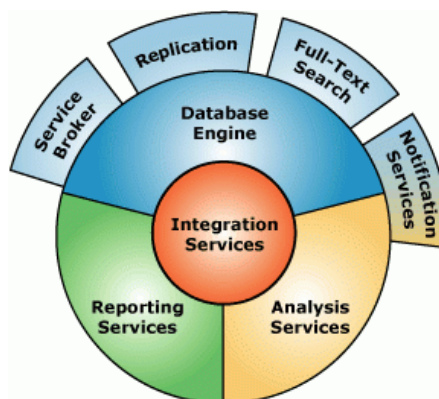
- ASP.NET Entity Framework Core là một web framework mã nguồn và được tối ưu hóa cho cloud để phát triển các ứng dụng web chạy trên nhiều nền tảng như Windows, Linux, Mac,.....^[7]
- Các ứng dụng ASP.NET Entity Framework Core có thể chạy trên .NET Core hoặc trên .NET Framework hoàn chỉnh.^[7]

2.3. ASP.NET Web API

- Web API là một phương thức dùng để cho phép các ứng dụng khác nhau có thể giao tiếp, trao đổi dữ liệu qua lại. Dữ liệu được Web API trả lại thường ở dạng JSON hoặc XML thông qua giao thức HTTP hoặc HTTPS^[8].
- Web API có các đặc trưng như:^[8]
 - Tự động hóa sản phẩm.
 - Khả năng tích hợp linh động.
 - Cập nhật thông tin thời gian thực.
 - Có tiêu chuẩn chung để sử dụng.

2.4. Microsoft SQL Server

- SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS)) sử dụng câu lệnh SQL (Transact-SQL) để trao đổi dữ liệu giữa máy Client và máy cài SQL Server.^[9]
- Các thành phần cơ bản trong SQL Server:
- SQL Server được cấu tạo bởi nhiều thành phần như Database Engine, Reporting Services, Notification Services, Integration Services, Full Text Search Service....^[9]



Hình 1: Sơ đồ cấu trúc của SQL Server

CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ

3.1. Khảo sát hiện trạng

3.1.1. Trang Web Tiki.vn

a. Giới thiệu chung về trang Tiki.vn

- Tiki là một hệ sinh thái thương mại tất cả trong một, gồm các công ty thành viên như:
 - TikiNOW Smart Logistics cung cấp dịch vụ logistics đầu-cuối;
 - Ticketbox mang đến dịch vụ vé sự kiện, xem phim hàng đầu;
 - Đơn vị bán lẻ Tiki Trading và Sàn Giao dịch cung cấp 10 triệu sản phẩm từ 26 ngành hàng phục vụ hàng triệu khách hàng trên toàn quốc.
- Với phương châm hoạt động “Tất cả vì Khách Hàng”, Tiki luôn không ngừng nỗ lực nâng cao chất lượng dịch vụ và sản phẩm, từ đó mang đến trải nghiệm mua sắm trọn vẹn cho Khách Hàng Việt Nam với dịch vụ giao hàng nhanh trong 2 tiếng và ngày hôm sau TikiNOW lần đầu tiên tại Đông Nam Á, cùng cam kết cung cấp hàng chính hãng với chính sách hoàn tiền 111% nếu phát hiện hàng giả, hàng nhái.

-

b. Các tính năng đã khảo sát trong trang web Tiki.vn cho TLCN

Danh mục hàng hóa

- Hiển thị theo từng loại hàng hóa khác nhau hoặc các loại sản phẩm liên quan đến nhau:
 - Có nhiều loại danh mục như: Điện thoại – Máy tính bản, Điện tử - Điện lạnh.
 - Mỗi mục sẽ có các đề mục nhỏ để thu hút khách hàng: Ví dụ: Mục Điện thoại – Máy tính bản sẽ có thêm các mục nhỏ là xu hướng, nổi bật, máy tính bản, phụ kiện điện thoại, ...
- Ngoài ra tiki.vn còn hiển thị danh sách sản phẩm 1 cách đặc biệt:
 - Hiển thị sản phẩm giảm giá.

- Hiện thị nổi bật sản phẩm với số phần trăm được giảm, số tiền trước và sau khi giảm để khách hàng so sánh.
- Hiện thị thời gian giảm giá tính theo giờ và số sản phẩm đã bán được.
- Sau khi đăng nhập và xem sản phẩm thì hệ thống sẽ lưu lại và hiển thị các sản phẩm đã xem trước đó.
- Về mặt thiết kế, trang chủ tiki thiết kế tương đối hài hòa giữa các thành phần, đặc biệt là đánh được vào tâm lý người dùng với việc hiển thị các sản phẩm được giảm giá ở phần giữa trang web.
- Về chức năng tìm kiếm:
 - Khi để con trỏ chuột ở thanh tìm kiếm, hệ thống sẽ hiển thị gợi ý: từ khóa hot, lịch sử tìm kiếm (chỉ xuất hiện khi đã tìm kiếm ít nhất 1 sản phẩm).
 - Tự động gợi ý mỗi khi khách gõ ký tự.
- Về danh sách các sản phẩm:
 - Hiện thị theo mục hoặc theo tên sản phẩm mà người dùng tìm kiếm.
 - Có thể lọc danh sách sản phẩm theo giá, có miễn phí giao hàng hay không, theo thương hiệu, theo nhà cung cấp,...

Giỏ hàng và thanh toán

- Khi người dùng tiến hành chọn mua hàng thì món hàng được chọn sẽ được chuyển vào giỏ hàng (Shopping Cart), trang web sẽ chuyển về đầu trang, hiển thị thông báo “Thêm vào giỏ hàng thành công” cùng nút “xem giỏ hàng và thanh toán”, và được lưu trữ tạm thời trong giỏ hàng.
- Khi sản phẩm đã ở trong giỏ hàng thì người dùng có thể:
 - Thêm / bớt số lượng sản phẩm.
 - Xóa sản phẩm.
 - Hoặc sử dụng chức năng để dành mua sau.
- Trước khi tiến hành thanh toán, người dùng có thể chọn và áp dụng các mã giảm giá đã có.
- Khi thanh toán:

- Đầu tiên, người dùng phải đăng nhập.
- Sau đó, Chọn (đã có địa chỉ giao hàng) hoặc thêm địa chỉ giao hàng.
- Cuối cùng là chọn hình thức giao hàng và thanh toán.

Xem thông tin chi tiết của từng hàng hóa đơn lẻ

- Người dùng có thể xem thông chi tiết sản phẩm như giá cả, thông tin sơ bộ, các dịch vụ liên quan,...
- Người dùng có thể rê chuột lên hình để xem các góc hình được phóng to.
- Có thể viết và xem được các bài nhận xét của khách.
- Và có thể xem được bài mô tả chi tiết về sản phẩm.

Đăng nhập / Đăng ký

- Người dùng thông thường có thể Đăng Ký tài khoản mới để trở thành thành viên của Tiki hoặc Đăng Nhập nếu đã có sẵn tài khoản hoặc đăng nhập bằng tài khoản Facebook, Google, Zalo.

Chăm sóc khách hàng

- Tiki có 1 hệ thống hỗ trợ tên là hotro.tiki.vn, tại đây người dùng có thể tìm kiếm các câu hỏi về vấn đề mà đang cần giải quyết như: tài khoản, đặt hàng thanh toán, giao nhận hàng,...

Quản lý nhà cung cấp

- Với mục “Bán hàng cùng Tiki”, các cá nhân hoặc tổ chức có thể trở thành nhà cung cấp với khách hàng khi Tiki trở thành bên trung gian, tạo cầu nối cho cả bên bán và bên mua.
- Đồng thời các cá nhân có nhu cầu có thể trở thành 1 thành viên của chương trình “Bán hàng hiệu quả cùng Tiki”.

c. Kết Luận

- **Ưu Điểm**

- Thanh toán dễ dàng và nhanh chóng.
- Có thể đăng nhập qua Facebook, Google,.....
- Chức năng tìm kiếm có tự động gợi ý các sản phẩm theo các tiêu chí:
 - Sản phẩm nổi bật
 - Sản phẩm đang được tìm kiếm nhiều
- Thanh tìm kiếm tự động gợi ý từ khóa
- Nếu đang có đơn hàng thì có thể theo dõi đơn hàng
- Người dùng có thể đánh giá sản phẩm khi xem chi tiết sản phẩm
- **Khuyết Điểm**
- Giao diện phân bố chưa đẹp mắt
- Thanh toán bằng xu tiki còn chưa hợp lý

3.1.2. Trang Web Lazada.vn

a. Giới thiệu chung về trang Lazada.vn

- **Lazada.vn** là một trong những trang web bán hàng Online rất nổi tiếng của Việt Nam, cạnh tranh tốt với các đối thủ như Tiki, Sendo, ChoTot,..... Có thể nói Lazada.vn có lượng người dùng cũng như doanh thu và độ tin cậy rất tốt.

b. Các tính năng đã khảo sát trong trang web Lazada.vn cho TLCN

Danh mục hàng hóa

- Hiện thị các tùy chọn về các mặt hàng có sẵn trong trang web theo các đề mục khác nhau:
- Hiện thị các hàng hóa theo từng loại danh mục khác nhau, có liên quan đến tính chất của mặt hàng, Ví dụ như Đồ điện tử sẽ có các danh mục con như Điện thoại, Tablet, Tivi, Laptop, Thiết bị nghe nhìn, Máy quay, máy ảnh, Loa, Giúp sắp xếp các mặt hàng một cách thuận tiện nhất.
- Về mặt thiết kế, trang chủ của Lazada sắp xếp các hiển thị của các mặt hàng theo dạng các ô vuông nhỏ và vừa phải, bố cục trang web gọn gàng màu sắc đơn giản nhưng hài hòa.

Giỏ hàng và thanh toán

- Khi người dùng tiến hành chọn hàng thì món hàng được chọn sẽ được chuyển vào giỏ hàng (Shopping Cart) và được lưu trữ tạm thời trong giỏ hàng.
- Khi người dùng tiến hành thanh toán. Hệ thống sẽ dựa vào thông tin sản phẩm người dùng cần mua, kiểm tra giỏ hàng, yêu cầu người dùng nhập đầy đủ thông tin cá nhân và tiến hành việc thanh toán và đặt lịch nhận hàng.

Xem thông tin chi tiết của từng hàng hóa đơn lẻ

- Với từng mặt hàng riêng lẻ, người dùng hoàn toàn có thể xem các thông tin chi tiết của mặt hàng đó như: Thông tin sản xuất, giá bán, số lượng, địa chỉ nơi bán, hình ảnh về món hàng, những nhận xét, hỏi đáp, bình luận của những khách hàng khác,.... Để tương tác với trang web.

Đăng nhập / Đăng ký

- Người dùng thông thường có thể Đăng Ký tài khoản mới để trở thành thành viên của Lazada hoặc Đăng Nhập nếu đã có sẵn tài khoản.
- Với tài khoản Lazada, bạn sẽ có thêm những tính năng mới như nhận thông báo về khuyến mãi, sự thay đổi của thị trường, Để từ đó người dùng có thể theo dõi các hoạt động đang diễn ra trên trang Web một cách dễ dàng và thuận tiện hơn.

Chăm sóc khách hàng

- Lazada có hệ thống chăm sóc khách hàng, là nơi để các khách hàng có thể giải đáp thắc mắc của mình thông qua việc như:
- Liên hệ với các trung tâm hỗ trợ trực tuyến của Lazada hoặc liên hệ trực tiếp với chi nhánh Lazada
- Tìm những câu hỏi liên quan đến tài khoản đăng nhập. cách thức giao hàng và nhận hàng, cách thực hiện kiểm tra thanh toán, hóa đơn và hàng hóa sau khi nhận được từ bên giao hàng.

- Trò chuyện trực tuyến với cửa hàng thông qua hộp thoại tin nhắn.
- Kết nối với Lazada thông qua các trang mạng xã hội thông dụng như: Facebook, Twitter, Youtube, Linkedin, Pinterest, Google+, Zing Me,.....

Quản lý nhà cung cấp

- Với mục “Bán hàng cùng Lazada”, các cá nhân hoặc tổ chức có thể trở thành nhà cung cấp với khách hàng khi Lazada trở thành bên trung gian, tạo cầu nối cho cả bên bán và bên mua.

c. Kết luận

- Ưu Điểm

- Thanh toán dễ dàng và nhanh chóng.
- Có khả năng đăng nhập và đăng ký bằng nhiều hình thức như Facebook, Google, ... một cách dễ dàng và tiện lợi.
- Chức năng tìm kiếm có tự động gợi ý các sản phẩm theo các tiêu chí:
 - Sản phẩm nổi bật
 - Sản phẩm đang được tìm kiếm nhiều
 - Thanh tìm kiếm tự động gợi ý từ khóa
 - Khi đăng nhập có thể xem các sản phẩm đã xem trước đóNếu đang có đơn hàng thì có thể theo dõi đơn hàng
- Người dùng có thể đăng ký nhận mail thông báo
- Người dùng có thể đánh giá sản phẩm khi xem chi tiết sản phẩm

- Khuyết Điểm

- Giao diện phân bố chưa đẹp mắt
- Thanh toán bằng xu tiki còn chưa hợp lý

3.1.3. Trang Web Shopee.vn

a. Giới thiệu chung về trang Shopee.vn

- Ra mắt năm 2015, nền tảng thương mại Shopee được xây dựng nhằm cung cấp cho người sử dụng những trải nghiệm dễ dàng, an toàn và nhanh chóng khi mua sắm trực tuyến thông qua hệ thống hỗ trợ thanh toán và vận hành vững mạnh.

b. Các tính năng đã khảo sát trong trang web Shopee.vn cho TLCN

Danh mục hàng hóa

- Hiện thị danh mục hàng hóa 1 cách thú vị là hình ảnh sản phẩm đi kèm với tên tạo sự thu hút với khách hàng khi lần đầu tiên đến với trang web. Với vị trí nằm ở đầu body nên tạo người dùng có thể dễ dàng tìm được và sử dụng chức năng tìm kiếm theo danh mục 1 cách dễ dàng.
- Cũng giống với Tiki.vn, shopee.vn cũng Thu hút khách hàng bằng cách hiển thị danh sách sản phẩm được giảm giá cùng với số sản phẩm đã bán được của mặt hàng đó.
- Trang web cũng gợi ý cho khách hàng các sản phẩm được tìm kiếm hàng đầu tại trang web để khách dễ dàng lựa chọn.
- Về chức năng tìm kiếm:
 - o Tự động gợi ý mỗi khi khách gõ ký tự.
- Về danh sách các sản phẩm:
 - o Hiện thị theo mục hoặc theo tên sản phẩm mà người dùng tìm kiếm.
 - o Có thể lọc danh sách sản phẩm theo nơi bán, đơn vị vận chuyển, theo thương hiệu, theo giá, tình trạng, ...

Giỏ hàng và thanh toán

- Sau khi xem và chọn số lượng cần mua, người dùng có 2 sự lựa chọn:
 - o Thêm sản phẩm vào giỏ hàng: sản phẩm sẽ chuyển đến giỏ hàng và người dùng tiếp tục xem thông tin sản phẩm.

- Mua ngay: sản phẩm sẽ chuyển đến giỏ hàng và người dùng chuyển tiếp đến giỏ hàng để tiến hành thanh toán.
- Trước khi thanh toán, người dùng có thể quyết định sử dụng mã giảm giá hiện có, thêm bớt số lượng cần mua, xóa sản phẩm hay sử dụng chức năng tìm kiếm sản phẩm tương tự.
 - Tìm kiếm tương tự: hệ thống hiển thị danh sách các sản phẩm tương tự như sản phẩm mà người dùng đã chọn.
- Tiến hành thanh toán:
 - Bước 1: Chọn địa chỉ (Nếu đã từng mua hàng trước đó)
 - Nếu chưa có thì sẽ điền thông tin đại chỉ giao hàng, Hệ thống có 1 điểm đặc biệt là có thể tự động xác định địa chỉ hiện tại của người mua, giúp việc xác nhận địa chỉ thêm dễ dàng.
 - Tiến hành chọn phương thức thanh toán và đặt hàng.

Xem thông tin chi tiết của từng hàng hóa đơn lẻ

- Người dùng có thể xem thông chi tiết sản phẩm như giá cả, thông tin sơ bộ, thêm bớt số lượng,...
- Có thể viết và xem được các bài nhận xét của khách.
- Có thể xem được bài mô tả chi tiết về sản phẩm.
- Hệ thống cũng gợi ý danh sách các sản phẩm tương tự cũng như các sản phẩm được yêu thích của trang web.
- Người dùng cũng có thể liên hệ trực tiếp với người cung cấp, doanh nghiệp thông qua chức năng “chat ngay”.

Đăng nhập / Đăng ký

- Người dùng thông thường có thể Đăng Ký tài khoản mới để trở thành thành viên của Shopee hoặc Đăng Nhập nếu đã có sẵn tài khoản hoặc đăng nhập bằng tài khoản Facebook, Google, Apple.
- Thành viên của Tiki có các tính năng nổi bật như:

- Theo dõi đơn hàng.
- Tích lũy Shoppe Xu.
- Ví Voucher:
 - Xem được các voucher đang sở hữu và điều kiện để có được các voucher.

Chăm sóc khách hàng

- Khi có câu hỏi cần được giải đáp, người dùng có thể liên hệ với bộ phận hỗ trợ thông qua trang hỗ trợ của shopee là help.shopee.vn. Tại đây người dùng có thể giả đáp thắc mắc thông qua các câu hỏi có sẵn, hoặc liên hệ trực tiếp theo hướng dẫn ở đầu trang web.

Quản lý nhà cung cấp

- Shopee cho phép các nhà cung cấp của họ nhận các câu hỏi trực tiếp từ khách hàng để dễ dàng tư vấn và hỗ trợ khi cần thiết.

c. Kết luận

- Ưu Điểm

- Có thể đăng nhập qua Facebook, Google,.....
- Chức năng tìm kiếm có tự động gợi ý các sản phẩm theo các tiêu chí:
 - Sản phẩm nổi bật
 - Sản phẩm đang được tìm kiếm nhiều
- Thanh tìm kiếm tự động gợi ý từ khóa
- Nếu đang có đơn hàng thì có thể theo dõi đơn hàng
- Người dùng có thể đánh giá sản phẩm khi xem chi tiết sản phẩm

- Khuyết Điểm

- Thanh toán còn khó khăn khi chưa xác minh được phương thức thanh toán.
- Liên tục điều hướng trang web nên chưa thân thiện với người dùng

3.1.4. Kết luận

- Phương pháp cho đề tài Tiểu Luận Chuyên Ngành của nhóm
- Qua các hệ thống trên, nhóm thực hiện rút ra những chức năng cơ bản cho việc xây dựng một trang web bán hàng online ổn định và hợp logic cho đề tài Tiểu luận chuyên ngành sắp tới.
- Với người dùng bình thường
 - Tìm kiếm theo tên sản phẩm, mã sản phẩm, giá tiền, nhãn hiệu, loại sản phẩm, danh mục sản phẩm.
 - Xem thông tin của một sản phẩm bất kỳ (bao gồm thông số kỹ thuật, giá bán, khuyến mãi, chi tiết sản phẩm, đánh giá từ người dùng,...)
 - Đăng nhập , đăng ký (xác thực tài khoản qua Email)
- Với người dùng đã đăng nhập
 - Có mọi quyền hạn của người dùng khách, ngoài ra còn có thể:
 - Xem và cập nhật thông tin tài khoản
 - Xem lịch sử giao dịch
 - Đánh giá sản phẩm
 - Thêm sản phẩm vào giỏ hàng (Xóa đi khi không cần thiết nữa, hoặc có thể Thanh toán khi muốn mua hàng)
- Quản trị viên (Adminstator)
 - Admin thường (Admin)
 - Chịu trách nhiệm quản lý trực tiếp các mặt hàng như: Hệ thống CSDL bên trong Server (có thể Thêm, Xóa, Sửa, hay Cập nhật về những thông tin, khuyến mãi, thông số kỹ thuật của mỗi mặt hàng
 - Quản lý toàn bộ mọi tài khoản người dùng (Bao gồm người dùng đã đăng nhập, Nhân viên giao hàng, Admin nhân viên và cả chính tài khoản Admin tổng)
 - Có thể xem qua các lịch sử giao dịch, thống kê doanh số, doanh thu bán hàng.
 - Admin cấp cao (Super Admin)

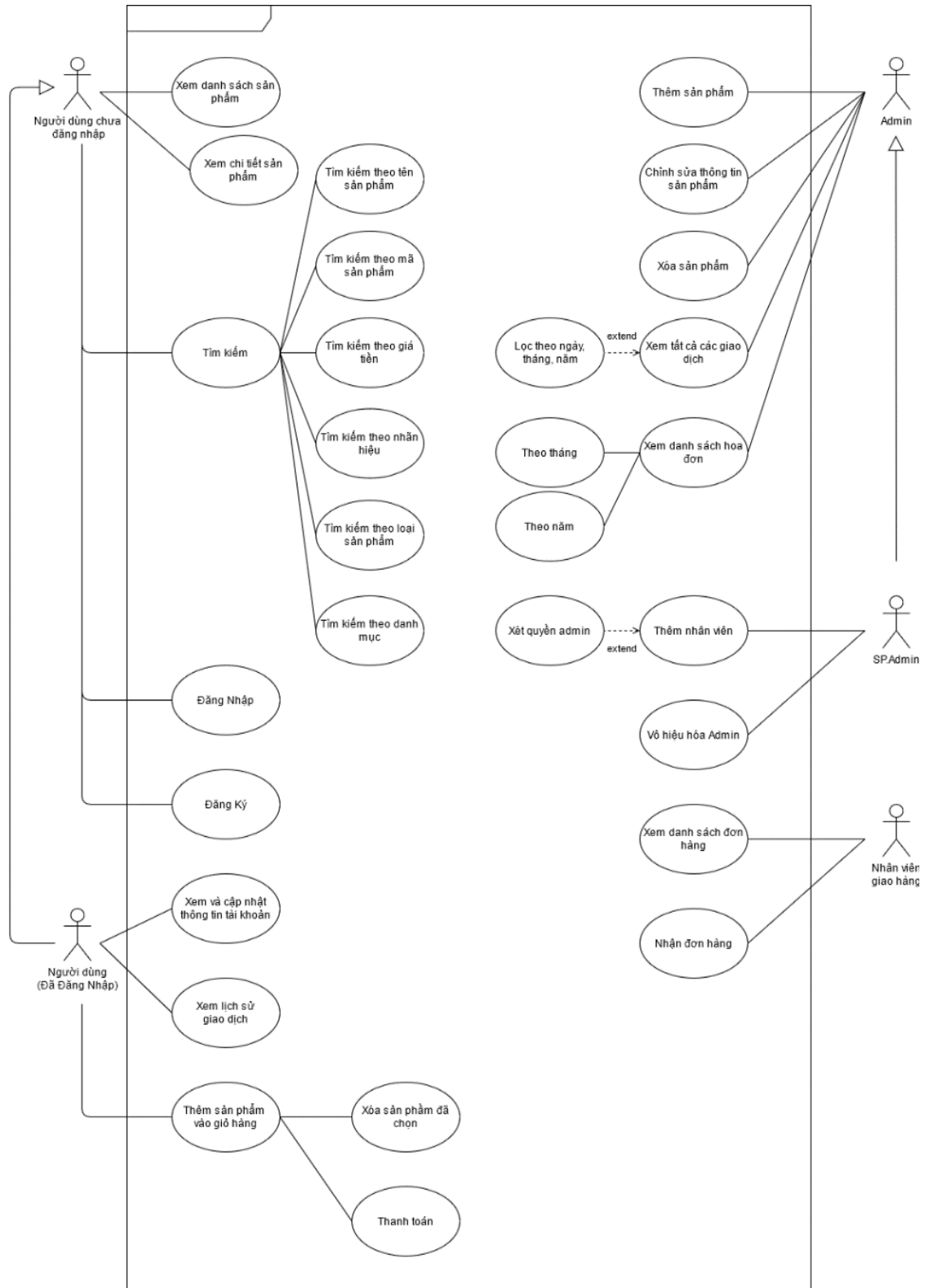
- Có được toàn bộ quyền hạn của Admin thường ở trên
- Ngoài ra, còn có thể xét hoặc bỏ quyền Admin của các Admin cấp thấp hơn (Admin thường)
 - Nhân viên giao hàng
- Họ có thể xem danh sách đơn hàng và nhận đơn hàng đó để đi giao hàng cho khách hàng.

3.2. Đặc tả hệ thống

3.2.1. Mô hình hóa yêu cầu

- Về mặt hình thức: Hệ thống mang mô hình hệ thống Website thương mại điện tử, bán hàng trực tuyến cho người sử dụng, khách hàng thông qua môi trường mạng Internet, có giỏ hàng riêng cho từng khách hàng, có thể thanh toán hàng ngay trên trang web hoặc qua người giao hàng.
- Về mặt kỹ thuật: Hệ thống được thiết kế bằng Angular Framework, xử lý thông qua gọi API từ mô hình .Net Core MVC và Web API.

3.2.2. Thiết kế Use Case Diagram



Hình 2: Sơ đồ chức năng Use Case của hệ thống

3.2.3. Đặc tả Use Case

Use Case Xem Danh Sách Sản Phẩm

USE CASE – ND_01									
Use Case No.	ND_01	Use Case Version	1.0						
Use Case Name	Xem danh sách sản phẩm								
Author	Bùi Văn Hà								
Date	09/2020	Priority	High						
Actor: <ul style="list-style-type: none">• Người dùng chưa đăng nhập• Người dùng đã đăng nhập Summary: <ul style="list-style-type: none">• Usecase cho phép người dùng xem được danh sách sản phẩm khi truy cập vào trang web Goal: <ul style="list-style-type: none">• Người dùng xem được hình ảnh một số sản phẩm được hiển thị trên trang chủ Triggers: <ul style="list-style-type: none">• Thông tin sản phẩm phải trùng khớp với hệ thống Preconditions: <ul style="list-style-type: none">• Người dùng phải truy cập vào trang web Post conditions: <ul style="list-style-type: none">• Success: Hiển thị thành công danh sách sản phẩm• Fail: Hệ thống báo lỗi Main Success Scenario: <table><tr><th>Step</th><th>Actor Action</th><th>System Response</th></tr><tr><td>1</td><td>Người dùng truy cập vào trang website qua URL</td><td>Hiển thị trang chủ của website bán hàng</td></tr></table>				Step	Actor Action	System Response	1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
Step	Actor Action	System Response							
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng							

Alternative Scenario:		
Exceptions:		
No	Actor Action	System Response
1		

Bảng 1: Use Case Xem Danh Sách Sản phẩm

Use Case Xem Chi Tiết Sản Phẩm

USE CASE – ND_02			
Use Case No.	ND_02	Use Case Version	1.0
Use Case Name	Xem danh sách sản phẩm		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
<p>Actor:</p> <ul style="list-style-type: none"> • Người dùng chưa đăng nhập • Người dùng đã đăng nhập <p>Summary:</p> <ul style="list-style-type: none"> • Usecase cho phép người dùng xem được chi tiết của một sản phẩm khi truy cập vào trang web <p>Goal:</p> <ul style="list-style-type: none"> • Người dùng xem được thông tin chi tiết về một sản phẩm được hiển thị trên trang chủ <p>Triggers:</p> <ul style="list-style-type: none"> • Thông tin sản phẩm phải trùng khớp với hệ thống <p>Preconditions:</p> <ul style="list-style-type: none"> • Người dùng phải truy cập vào trang web 			

Post conditions:

- Success: Hiện thị thành công danh sách sản phẩm
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiện thị trang chủ của website bán hàng
2	Người dùng truy cập vào một sản phẩm bất kỳ	Hiện thị thông tin chi tiết của sản phẩm đó

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1		

Bảng 2: Use Case Xem Chi Tiết Sản phẩm

Use Case Tìm Kiếm sản phẩm

USE CASE – ND_03			
Use Case No.	ND_03	Use Case Version	1.0
Use Case Name	Tìm kiếm theo tên sản phẩm		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none"> • Người dùng chưa đăng nhập • Người dùng đã đăng nhập Summary:			

- Usecase cho phép người dùng tìm kiếm thông tin của sản phẩm theo các từ khóa

Goal:

- Người dùng có thể tìm được sản phẩm cần tìm thông qua từ khóa

Triggers:

- Thông tin sản phẩm phải trùng khớp với hệ thống

Preconditions:

- Người dùng phải truy cập vào trang web

Post conditions:

- Success: Hiển thị thành công đúng sản phẩm cần tìm
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng trở vào thanh tìm kiếm trên trang chủ	Thanh tìm kiếm xuất hiện , sẵn sàng để người dùng nhập liệu
3	Người dùng gõ vào từ khóa cần tìm kiếm	Thông tin của sản phẩm cần tìm kiếm được hiển thị trong trang kết quả (Exception 1)

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng gõ vào từ khóa cần tìm kiếm	Hệ thống thông báo không tìm thấy được thông tin do không có dữ liệu

Bảng 3: Use Case Tìm kiếm sản phẩm

Use Case Đăng Nhập

USE CASE – ND_09			
Use Case No.	ND_09	Use Case Version	1.0
Use Case Name	Đăng nhập		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High

Actor:

- Người dùng chưa đăng nhập

Summary:

- Usecase cho phép người dùng chưa đăng nhập đăng nhập vào tài khoản được đăng ký trước đó trên trang web

Goal:

- Người dùng có thể đăng nhập và sử dụng tài khoản của trang web

Triggers:

- Thông tin người dùng phải trùng khớp với hệ thống

Preconditions:

- Người dùng phải đăng ký tài khoản thành công trước khi đăng nhập

Post conditions:

- Success: Hiện thị thông báo đăng nhập thành công
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiện thị trang chủ của website bán hàng
2	Người dùng truy cập vào nút đăng nhập	Hộp thoại đăng nhập xuất hiện

3	Người dùng truy nhập thông tin tài khoản	Hiển thị thông tin tài khoản (Exception 1)
4	Người dùng bấm vào nút Đăng Nhập	Thông báo đăng nhập thành công (Exception 2)

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng truy nhập thông tin tài khoản	Thông báo lỗi thông tin (Mật khẩu quá ngắn, chứa ký tự cấm,...)
2	Người dùng bấm vào nút Đăng Nhập	Thông báo đăng nhập thất bại

Bảng 4 : Use Case Đăng nhập

Use Case Đăng Ký

USE CASE – ND_10			
Use Case No.	ND_10	Use Case Version	1.0
Use Case Name	Đăng ký		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
<p>Actor:</p> <ul style="list-style-type: none"> Người dùng chưa đăng nhập <p>Summary:</p> <ul style="list-style-type: none"> Use case cho phép người dùng đăng ký một tài khoản mới từ trang web <p>Goal:</p>			

- Người dùng có được một tài khoản từ trang web

Triggers:

- Thông tin người dùng phải trùng khớp với hệ thống

Preconditions:

- Người dùng phải truy cập vào trang web

Post conditions:

- Success: Hiển thị đăng ký thành công
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng truy cập vào nút đăng nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng truy cập vào nút Đăng ký tài khoản mới	Hộp thoại đăng ký xuất hiện
4	Người dùng truy nhập thông tin tài khoản mới	Hiển thị thông tin tài khoản (Exception 1)
5	Người dùng bấm vào nút Đăng Ký	Thông báo gửi mã xác nhận qua Gmail (Exception 2)
6	Người dùng xác nhận qua Gmail	Thông báo xác nhận thành công Đăng ký thành công, tài khoản mới được tạo (Exception 3)

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
----	--------------	-----------------

1	Người dùng truy nhập thông tin tài khoản mới	Thông báo lỗi thông tin (Mật khẩu quá ngắn, chứa ký tự cấm,...)
2	Người dùng bấm vào nút Đăng Ký	Thông báo gửi mã xác nhận qua Gmail thất bại
3	Người dùng xác nhận qua Gmail	Thông báo xác nhận thất bại, Đăng ký không thành công.

Bảng 5: Use Case Đăng ký

Use Case Xem và Cập nhật thông tin tài khoản

USE CASE – ND_11			
Use Case No.	ND_11	Use Case Version	1.0
Use Case Name	Xem và cập nhật thông tin sản phẩm		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
<p>Actor:</p> <ul style="list-style-type: none"> Người dùng đã đăng nhập <p>Summary:</p> <ul style="list-style-type: none"> Usecase cho phép người dùng cập nhật những thông tin mới nhất về những sản phẩm của hệ thống <p>Goal:</p> <ul style="list-style-type: none"> Người dùng có thể nắm bắt được thông tin của sản phẩm <p>Triggers:</p> <ul style="list-style-type: none"> Thông tin sản phẩm phải trùng khớp với hệ thống <p>Preconditions:</p> <ul style="list-style-type: none"> Người dùng phải truy cập vào trang web 			

- Người dùng đã đăng nhập thành công

Post conditions:

- Success: Thông tin về sản phẩm được gửi tới người dùng
- Fail: Không hiển thị cập nhật mới nhất cho người dùng

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập	Đăng nhập thành công (Exception 1)
4	Người dùng truy cập vào phần Thông tin tài khoản	Bảng thông tin tài khoản xuất hiện để người dùng xem
5	Người dùng bấm Cập nhật tài khoản	Các thông tin của tài khoản được mở khóa ẩn để sửa thông tin
6	Người dùng tiến hành cập nhật tài khoản	Thông báo cập nhật tài khoản thành công (Exception 2)

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập	Đăng nhập thất bại
2	Người dùng tiến hành cập nhật tài khoản	Thông báo cập nhật tài khoản không thành công

Bảng 6: Use Case Xem và Cập nhật thông tin tài khoản

Use Case Xem Lịch sử giao dịch

USE CASE – ND_12			
Use Case No.	ND_12	Use Case Version	1.0
Use Case Name	Xem lịch sử giao dịch		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High

Actor:

- Người dùng đã đăng nhập

Summary:

- Usecase cho phép người dùng xem lịch sử mua hàng của mình trên trang web

Goal:

- Người dùng nắm bắt được mình đã từng mua những sản phẩm nào trên trang web

Triggers:

- Thông tin trong lịch sử giao dịch phải trùng khớp với hệ thống

Preconditions:

- Người dùng phải truy cập vào trang web
- Người dùng đã đăng nhập thành công

Post conditions:

- Success: Hiển thị thành công lịch sử giao dịch
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện

3	Người dùng tiến hành Đăng Nhập	Đăng nhập thành công (Exception 1)
4	Người dùng truy cập vào phần Thông tin tài khoản	Bảng thông tin tài khoản xuất hiện để người dùng xem
5	Người dùng bấm vào Lịch sử giao dịch	Bản thông tin lịch sử giao dịch sẽ được hiển thị

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập	Đăng nhập thất bại

Bảng 7: Use Case Xem lịch sử giao dịch

Use Case Thêm Sản phẩm vào giỏ hàng

USE CASE – ND_14			
Use Case No.	ND_14	Use Case Version	1.0
Use Case Name	Thêm sản phẩm vào giỏ hàng		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
<p>Actor:</p> <ul style="list-style-type: none"> Người dùng đã đăng nhập <p>Summary:</p> <ul style="list-style-type: none"> Usecase cho phép người dùng thêm một sản phẩm bất kỳ muốn mua vào giỏ hàng <p>Goal:</p> <ul style="list-style-type: none"> Người dùng lưu được sản phẩm của mình vào giỏ hàng <p>Triggers:</p>			

- Không có

Preconditions:

- Người dùng phải truy cập vào trang web
- Người dùng đã đăng nhập thành công

Post conditions:

- Success: Hiển thị thành công sản phẩm trong giỏ hàng
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập	Đăng nhập thành công (Exception 1)
4	Người dùng truy cập vào một sản phẩm bất kỳ	Trang chi tiết sản phẩm xuất hiện
5	Người dùng bấm vào mục Thêm vào giỏ hàng	Hệ thống hiển thị giỏ hàng đã được cập nhật
6	Người dùng tiến hành xem Giỏ hàng của mình	Sản phẩm được chọn đã được thêm vào trong giỏ hàng thành công

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập	Đăng nhập thất bại

Bảng 8: Use Case Thêm sản phẩm vào giỏ hàng

Use Case Xóa sản phẩm đã chọn

USE CASE – ND_15			
Use Case No.	ND_15	Use Case Version	1.0
Use Case Name	Xóa sản phẩm đã chọn		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none">Người dùng đã đăng nhập			
Summary: <ul style="list-style-type: none">Usecase cho phép người dùng xóa sản phẩm khỏi giỏ hàng			
Goal: <ul style="list-style-type: none">Người dùng có sản phẩm bất kỳ ra khỏi giỏ hàng khi không muốn mua nữa			
Triggers: <ul style="list-style-type: none">Không có			
Preconditions: <ul style="list-style-type: none">Người dùng phải truy cập vào trang webNgười dùng đã đăng nhập thành côngSản phẩm phải có sẵn trong giỏ hàng			
Post conditions: <ul style="list-style-type: none">Success: Hiện thị xóa sản phẩm khỏi giỏ hàng thành côngFail: Hệ thống báo lỗi			
Main Success Scenario:			
Step	Actor Action	System Response	

1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập	Đăng nhập thành công (Exception 1)
4	Người dùng truy cập giỏ hàng của mình	Trang giỏ hàng xuất hiện
5	Người dùng bấm vào mục Xóa sản phẩm của sản phẩm cần xóa khỏi giỏ hàng	Hệ thống hiển thị thông báo có muốn xóa sản phẩm này khỏi giỏ hàng hay không
6	Người dùng bấm Có	Thông báo xóa sản phẩm khỏi giỏ hàng thành công

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập	Đăng nhập thất bại

Bảng 9: Use Case Xóa sản phẩm đã chọn

Use Case Thanh Toán

USE CASE – ND_16			
Use Case No.	ND_16	Use Case Version	1.0
Use Case Name	Thanh Toán		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none"> Người dùng đã đăng nhập Summary:			

- Usecase cho phép người dùng thanh toán những sản phẩm đang có trong giỏ hàng

Goal:

- Người dùng thanh toán thành công sản phẩm cần mua trên trang web

Triggers:

- Thông tin sản phẩm phải trùng khớp với hệ thống

Preconditions:

- Người dùng phải truy cập vào trang web
- Người dùng đã đăng nhập thành công
- Sản phẩm đang có sẵn trong giỏ hàng

Post conditions:

- Success: Hiện thị thanh toán thành công
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiện thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập	Đăng nhập thành công (Exception 1)
4	Người dùng truy cập giỏ hàng của mình	Trang giỏ hàng xuất hiện
5	Người dùng bấm vào sản phẩm cần mua	Hệ thống hiển thị thông tin thanh toán
6	Người dùng tiến hành nhập thông tin thanh toán	Hệ thống xác nhận thông tin thanh toán hợp lệ (Exception 2)

7	Người dùng chọn phương thức thanh toán bấm và nút Thanh toán	Hệ thống tạo hóa đơn và xác nhận thanh toán thành công (Exception 3)
Alternative Scenario:		
Exceptions:		
No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập	Đăng nhập thất bại
2	Người dùng tiến hành nhập thông tin thanh toán	Hệ thống xác nhận thông tin thanh toán chưa hợp lệ
3	Người dùng chọn phương thức thanh toán bấm và nút Thanh toán	Hệ thống báo lỗi, thông báo thanh toán không thành công

Bảng 10: Use Case Thanh toán

Use Case Thêm Sản phẩm

USE CASE – AD_01			
Use Case No.	AD_01	Use Case Version	1.0
Use Case Name	Thêm sản phẩm		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none"> Admin Super Admin Summary: <ul style="list-style-type: none"> Usecase cho phép quản trị viên thêm sản phẩm vào trong hệ thống Goal:			

- Một sản phẩm mới được thêm vào trong hệ thống sản phẩm

Triggers:

- Thông tin sản phẩm phải đúng đắn

Preconditions:

- Người sử dụng phải có quyền truy cập quản trị viên

Post conditions:

- Success: Hiển thị thêm sản phẩm thành công
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thành công (Exception 1)
4	Quản trị viên bấm vào danh mục sản phẩm	Danh sách các sản phẩm xuất hiện
5	Quản trị viên bấm vào Thêm sản phẩm mới	Trang thêm sản phẩm mới xuất hiện
6	Quản trị viên nhập thông tin, hình ảnh cho sản phẩm mới	Thông tin, hình ảnh cho sản phẩm mới được thêm vào
7	Quản trị viên bấm vào Thêm	Sản phẩm mới được thêm vào thành công

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
----	--------------	-----------------

1	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thất bại
---	--	--------------------

Bảng 11: Use Case Thêm sản phẩm

Use Case **Chỉnh sửa thông tin sản phẩm**

USE CASE – AD_02			
Use Case No.	AD_02	Use Case Version	1.0
Use Case Name	Chỉnh sửa thông tin sản phẩm		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
<p>Actor:</p> <ul style="list-style-type: none"> Admin Super Admin <p>Summary:</p> <ul style="list-style-type: none"> Usecase cho phép quản trị viên chỉnh sửa những thông tin của sản phẩm trong hệ thống <p>Goal:</p> <ul style="list-style-type: none"> Một sản phẩm được cập nhật thông tin vào trong hệ thống sản phẩm <p>Triggers:</p> <ul style="list-style-type: none"> Thông tin sản phẩm phải đúng đắn <p>Preconditions:</p> <ul style="list-style-type: none"> Người sử dụng phải có quyền truy cập quản trị viên <p>Post conditions:</p> <ul style="list-style-type: none"> Success: Hiện thị cập nhật thông tin sản phẩm thành công Fail: Hệ thống báo lỗi 			

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thành công (Exception 1)
4	Quản trị viên bấm vào danh mục sản phẩm	Danh sách các sản phẩm xuất hiện
5	Quản trị viên bấm vào Sản phẩm cần chỉnh sửa	Thông tin chi tiết sản phẩm được hiển thị
5	Quản trị viên bấm vào Cập nhật thông tin sản phẩm mới	Các thông tin có thể chỉnh sửa
6	Quản trị viên cập nhật thông tin, hình ảnh cho sản phẩm	Thông tin, hình ảnh cho sản phẩm mới được cập nhật lại
7	Quản trị viên bấm Cập nhật	Thông tin của sản phẩm được cập nhật thành công

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thất bại

Bảng 12: Use Case Chỉnh sửa thông tin sản phẩm

Use Case Xóa sản phẩm

USE CASE – AD_03			
Use Case No.	AD_03	Use Case Version	1.0

Use Case Name	Xóa sản phẩm		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none">AdminSuper Admin			
Summary: <ul style="list-style-type: none">Usecase cho phép quản trị viên xóa sản phẩm từ trong hệ thống			
Goal: <ul style="list-style-type: none">Một sản phẩm mới được xóa khỏi hệ thống sản phẩm			
Triggers: <ul style="list-style-type: none">Thông tin sản phẩm phải đúng đắn			
Preconditions: <ul style="list-style-type: none">Người sử dụng phải có quyền truy cập quản trị viên			
Post conditions: <ul style="list-style-type: none">Success: Hiện thị xóa sản phẩm thành côngFail: Hệ thống báo lỗi			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thất bại	
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện	
3	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thành công (Exception 1)	
4	Quản trị viên bấm vào danh mục sản phẩm	Danh sách các sản phẩm xuất hiện	

5	Quản trị viên bấm vào Sản phẩm cần xóa	Thông tin chi tiết sản phẩm được hiển thị
5	Quản trị viên bấm vào Xóa sản phẩm	Thông báo có muốn xóa sản phẩm này hay không
6	Quản trị viên Bấm vào nút Xóa	Thông tin, hình ảnh cho sản phẩm mới được xóa khỏi danh sách sản phẩm

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thất bại

Bảng 13: Use Case Xóa sản phẩm

Use Case Xem Tất cả các giao dịch

USE CASE – AD_04			
Use Case No.	AD_04	Use Case Version	1.0
Use Case Name	Xem tất cả giao dịch		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none"> Admin Super Admin Summary: <ul style="list-style-type: none"> Usecase cho phép quản trị viên xem được toàn bộ lịch sử giao dịch của các người dùng trong hệ thống 			

Goal:

- Quản trị viên quản lý được lịch sử giao dịch của toàn hệ thống

Triggers:

- Thông tin người dùng, thông tin giao dịch phải đúng đắn

Preconditions:

- Người sử dụng phải có quyền truy cập quản trị viên

Post conditions:

- Success: Hiển thị danh sách lịch sử giao dịch của hệ thống
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thành công (Exception 1)
4	Quản trị viên bấm vào danh sách giao dịch	Danh sách các giao dịch xuất hiện

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thất bại

Bảng 14: Use Case Xem tất cả các giao dịch

Use Case lọc theo Ngày, Tháng, Năm

USE CASE – AD_05			
Use Case No.	AD_05	Use Case Version	1.0
Use Case Name	Lọc sản phẩm theo Ngày, Tháng, Năm		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none">AdminSuper Admin			
Summary: <ul style="list-style-type: none">Usecase cho phép quản trị viên lọc ra sản phẩm theo thời gian (Ngày, tháng, năm)			
Goal: <ul style="list-style-type: none">Các sản phẩm được lọc ra theo đúng thứ tự			
Triggers: <ul style="list-style-type: none">Thông tin sản phẩm phải đúng đắn			
Preconditions: <ul style="list-style-type: none">Người sử dụng phải có quyền truy cập quản trị viên			
Post conditions: <ul style="list-style-type: none">Success: Hiển thị danh sách sản phẩm được lọc ra theo thứ tự yêu cầu ban đầuFail: Hệ thống báo lỗi			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng	
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện	

3	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thành công (Exception 1)
4	Quản trị viên bấm vào danh sách sản phẩm	Danh sách các sản phẩm xuất hiện
5	Quản trị bấm vào Lọc sản phẩm	Các sản phẩm được lọc ra từ động theo Ngày, Tháng, Năm

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thất bại

Bảng 15: Use Case lọc theo Ngày, Tháng, Năm

Use Case Xem thống kê bán sản phẩm

Use Case Theo Tháng

USE CASE – AD_06			
Use Case No.	AD_06	Use Case Version	1.0
Use Case Name	Xem thống kê bán sản phẩm theo tháng		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none"> Admin Super Admin Summary:			

- Usecase cho phép quản trị viên xem trình trạng bán hàng trong hệ thống theo tháng

Goal:

- Quản trị viên nắm bắt được thống kê mua bán hàng của hệ thống theo tháng

Triggers:

- Thông tin thống kê phải đúng đắn

Preconditions:

- Người sử dụng phải có quyền truy cập quản trị viên

Post conditions:

- Success: Hiển thị thống kê bán sản phẩm thành công
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thành công (Exception 1)
4	Quản trị viên bấm vào trang thống kê bán hàng	Bảng thống kê bán sản phẩm xuất hiện
5	Quản trị bấm vào Lọc theo Tháng	Bảng thống kê bán sản phẩm được sắp xếp lại theo từng Tháng

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
----	--------------	-----------------

1	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thất bại
---	--	--------------------

Bảng 16: Use Case Xem thống kê bán sản phẩm theo Tháng

Use Case Theo Năm

USE CASE – AD_07			
Use Case No.	AD_07	Use Case Version	1.0
Use Case Name	Xem thống kê bán sản phẩm theo năm		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
<p>Actor:</p> <ul style="list-style-type: none"> Admin Super Admin <p>Summary:</p> <ul style="list-style-type: none"> Usecase cho phép quản trị viên xem trình trạng bán hàng trong hệ thống theo năm <p>Goal:</p> <ul style="list-style-type: none"> Quản trị viên nắm bắt được thống kê mua bán hàng của hệ thống theo năm <p>Triggers:</p> <ul style="list-style-type: none"> Thông tin thống kê phải đúng đắn <p>Preconditions:</p> <ul style="list-style-type: none"> Người sử dụng phải có quyền truy cập quản trị viên <p>Post conditions:</p> <ul style="list-style-type: none"> Success: Hiện thị thống kê bán sản phẩm thành công Fail: Hệ thống báo lỗi 			

Main Success Scenario:		
Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thành công (Exception 1)
4	Quản trị viên bấm vào trang thống kê bán hàng	Bảng thống kê bán sản phẩm xuất hiện
5	Quản trị bấm vào Lọc theo Năm	Bảng thống kê bán sản phẩm được sắp xếp lại theo từng Năm
Alternative Scenario:		
Exceptions:		
No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Quản trị viên	Đăng nhập thất bại

Bảng 17: Use Case Xem thống kê bán sản phẩm theo Năm

Use Case Thêm nhân viên

USE CASE – AD_08			
Use Case No.	AD_08	Use Case Version	1.0
Use Case Name	Thêm nhân viên		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none"> Super Admin Summary:			

- Usecase cho phép quản trị viên thêm một nhân viên mới vào hệ thống

Goal:

- Quản trị viên thêm mới một nhân viên

Triggers:

- Thông tin nhân viên phải đúng đắn

Preconditions:

- Người sử dụng phải có quyền truy cập quản trị viên cao nhất

Post conditions:

- Success: Hiển thị thêm nhân viên thành công
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Super Admin	Đăng nhập thành công (Exception 1)
4	Super Admin bấm vào trang danh sách Nhân viên	Bảng danh sách nhân viên xuất hiện
5	Super Admin bấm vào Thêm nhân viên mới	Trang thêm nhân viên mới xuất hiện
6	Super Admin điền thông tin nhân viên mới	Thông tin của tài khoản nhân viên được hiển thị
7	Super Admin bấm vào Thêm mới	Tài khoản nhân viên mới được thêm thành công (Exception 2)

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Super Admin	Đăng nhập thất bại
2	Super Admin bấm vào Thêm mới	Tài khoản nhân viên mới chưa được thêm vào

Bảng 18: Use Case Thêm nhân viên

Use Case Xét quyền Admin

USE CASE – AD_09			
Use Case No.	AD_09	Use Case Version	1.0
Use Case Name	Xét quyền Admin		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
<p>Actor:</p> <ul style="list-style-type: none"> Super Admin <p>Summary:</p> <ul style="list-style-type: none"> Usecase cho phép quản trị viên đặt quyền quản trị viên cho nhân viên <p>Goal:</p> <ul style="list-style-type: none"> Nhân viên được trở thành quản trị viên <p>Triggers:</p> <ul style="list-style-type: none"> Thông tin nhân viên phải đúng đắn <p>Preconditions:</p> <ul style="list-style-type: none"> Người sử dụng phải có quyền truy cập quản trị viên cao nhất <p>Post conditions:</p> <ul style="list-style-type: none"> Success: Hiện thị thêm quyền thành công 			

- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Super Admin	Đăng nhập thành công (Exception 1)
4	Super Admin bấm vào trang danh sách Nhân viên	Bảng danh sách nhân viên xuất hiện
5	Super Admin bấm vào một tài khoản nhân viên bất kỳ	Trang tài khoản nhân viên xuất hiện
6	Super Admin thêm quyền Admin cho nhân viên	Quyền của nhân viên được cập nhật thành công (Exception 2)

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Super Admin	Đăng nhập thất bại
2	Super Admin thêm quyền Admin cho nhân viên	Quyền của nhân viên chưa được cập nhật thành công

Bảng 19: Use Case Xét quyền Admin

Use Case Vô hiệu hóa quyền Admin

USE CASE – AD_10			
Use Case No.	AD_10	Use Case Version	1.0

Use Case Name	Vô hiệu hóa quyền Admin		
Author	Nguyễn Huy Cường		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none">Super Admin			
Summary: <ul style="list-style-type: none">Usecase cho phép quản trị viên hủy quyền quản trị viên của một nhân viên			
Goal: <ul style="list-style-type: none">Quản trị viên hủy đi quyền quản trị viên của nhân viên			
Triggers: <ul style="list-style-type: none">Thông tin nhân viên phải đúng đắn			
Preconditions: <ul style="list-style-type: none">Người sử dụng phải có quyền truy cập quản trị viên cao nhất			
Post conditions: <ul style="list-style-type: none">Success: Hiện thị xóa quyền thành côngFail: Hệ thống báo lỗi			
Main Success Scenario:			
Step	Actor Action	System Response	
1	Người dùng truy cập vào trang website qua URL	Hiện thị trang chủ của website bán hàng	
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện	
3	Người dùng tiến hành Đăng Nhập với quyền Super Admin	Đăng nhập thành công (Exception 1)	
4	Super Admin bấm vào trang danh sách Nhân viên	Bảng danh sách nhân viên xuất hiện	

5	Super Admin bấm vào một tài khoản nhân viên bất kỳ	Trang tài khoản nhân viên xuất hiện
6	Super Admin xóa quyền Admin cho nhân viên	Quyền của nhân viên được cập nhật thành công (Exception 2)

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Super Admin	Đăng nhập thất bại
2	Super Admin xóa quyền Admin cho nhân viên	Quyền của nhân viên chưa được cập nhật thành công

Bảng 20: Use Case Vô hiệu hóa quyền Admin

Use Case Xem Danh sách đơn hàng

USE CASE – GH_01			
Use Case No.	GH_01	Use Case Version	1.0
Use Case Name	Xem danh sách giao hàng		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
<p>Actor:</p> <ul style="list-style-type: none"> Nhân viên giao hàng <p>Summary:</p> <ul style="list-style-type: none"> Usecase cho phép nhân viên giao hàng xem trước đơn hàng của khách hàng trước khi tiến hành giao hàng <p>Goal:</p>			

- Nhân viên giao hàng xem trước đơn giao hàng

Triggers:

- Thông tin đơn hàng phải trùng khớp với hệ thống

Preconditions:

- Người dùng phải truy cập vào trang web
- Người dùng có quyền là nhân viên giao hàng
- Phải có hóa đơn giao hàng

Post conditions:

- Success: Xem thành công đơn hàng
- Fail: Hệ thống báo lỗi

Main Success Scenario:

Step	Actor Action	System Response
1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Nhân viên giao hàng	Đăng nhập thành công (Exception 1)
4	Nhân viên giao hàng bấm vào trang danh sách đơn hàng	Bảng danh sách đơn hàng xuất hiện

Alternative Scenario:

Exceptions:

No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Nhân viên giao hàng	Đăng nhập thất bại

Bảng 21: Use Case Xem danh sách đơn hàng

Use Case Nhận đơn hàng

USE CASE – GH_02			
Use Case No.	GH_02	Use Case Version	1.0
Use Case Name	Nhận đơn hàng		
Author	Bùi Văn Hà		
Date	09/2020	Priority	High
Actor: <ul style="list-style-type: none">Nhân viên giao hàng			
Summary: <ul style="list-style-type: none">Usecase cho phép nhân viên giao hàng nhận đơn hàng của khách hàng trước khi tiến hành giao hàng			
Goal: <ul style="list-style-type: none">Nhân viên giao hàng nhận đơn giao hàng			
Triggers: <ul style="list-style-type: none">Thông tin đơn hàng phải trùng khớp với hệ thống			
Preconditions: <ul style="list-style-type: none">Người dùng phải truy cập vào trang webNgười dùng có quyền là nhân viên giao hàngPhải có hóa đơn giao hàng			
Post conditions: <ul style="list-style-type: none">Success: Nhận thành công đơn hàngFail: Hệ thống báo lỗi			
Main Success Scenario:			
Step	Actor Action	System Response	

1	Người dùng truy cập vào trang website qua URL	Hiển thị trang chủ của website bán hàng
2	Người dùng bấm vào Đăng Nhập	Hộp thoại đăng nhập xuất hiện
3	Người dùng tiến hành Đăng Nhập với quyền Nhân viên giao hàng	Đăng nhập thành công (Exception 1)
4	Nhân viên giao hàng bấm vào trang danh sách đơn hàng	Bảng danh sách đơn hàng xuất hiện
5	Nhân viên giao hàng chọn Đơn hàng cần nhận	Thông báo có muốn Nhận đơn hàng này hay không
6	Nhân viên giao hàng chọn Đồng ý	Thông báo Đơn hàng đã được giao

Alternative Scenario:

Exceptions:

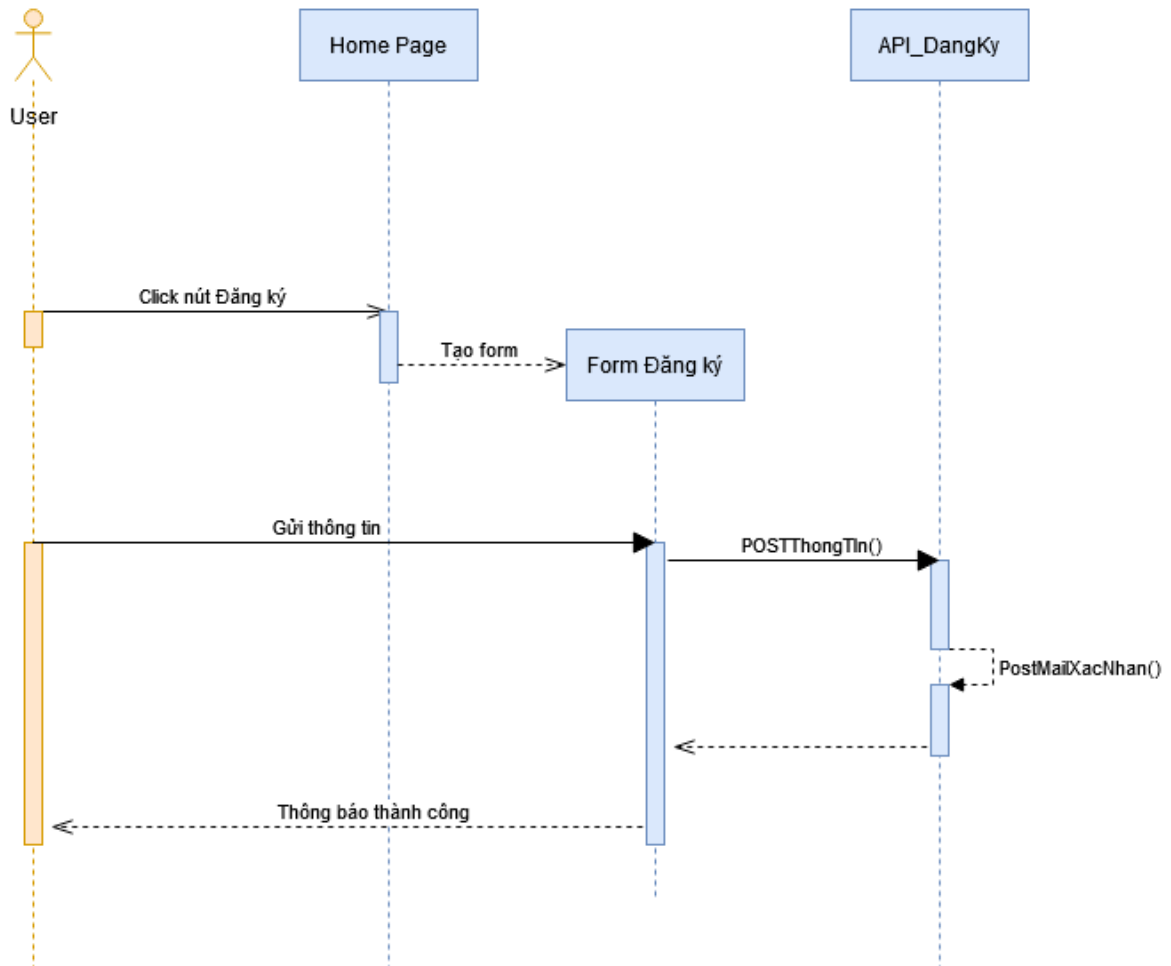
No	Actor Action	System Response
1	Người dùng tiến hành Đăng Nhập với quyền Nhân viên giao hàng	Đăng nhập thất bại

Bảng 22: Use Case Nhận đơn hàng

3.2.4. Mô tả một số chức năng qua lược đồ tuần tự (Sequence Diagram)

Chức năng đăng ký

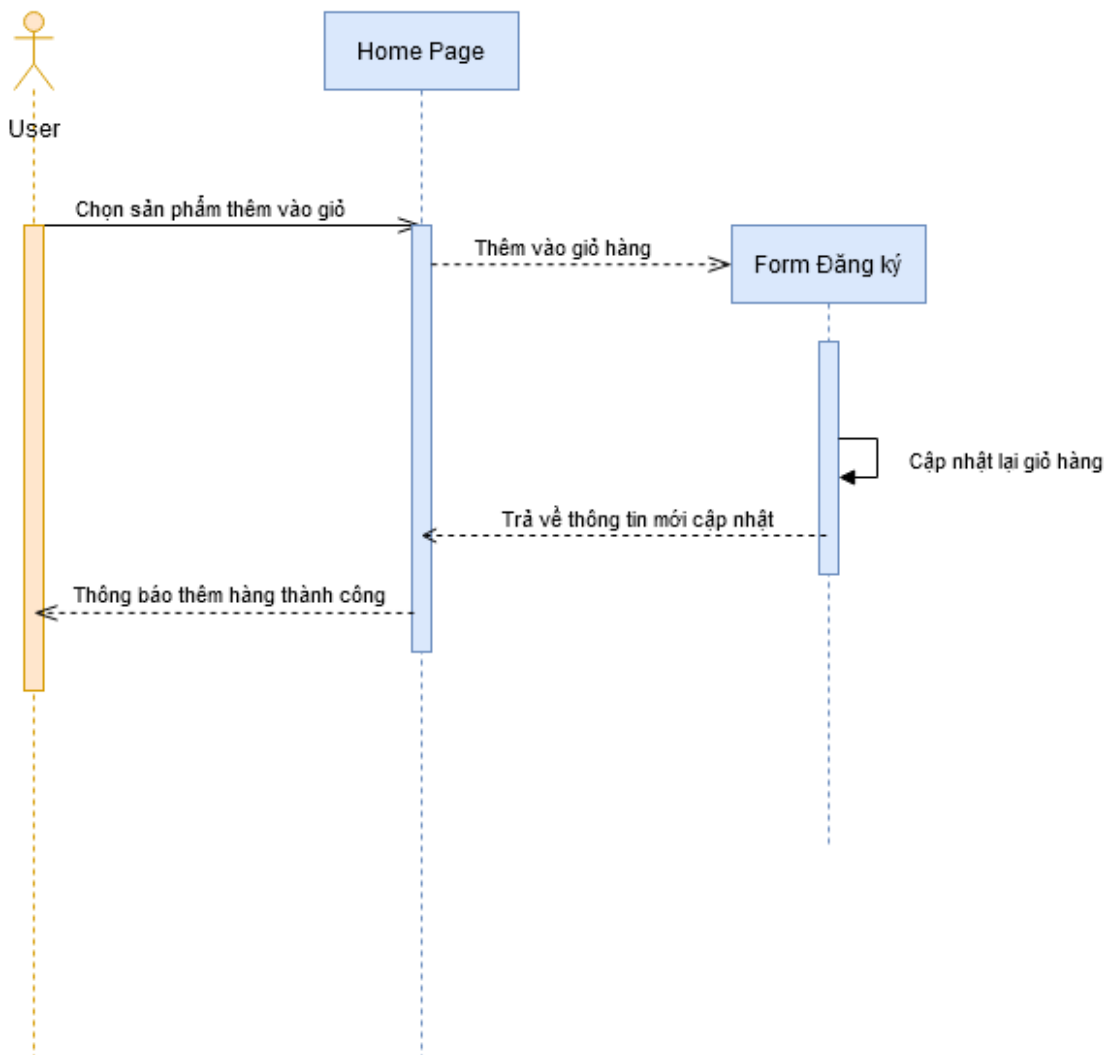
Đăng ký



Hình 3: Lược đồ tuần tự Đăng ký

Chức năng thêm vào giỏ hàng

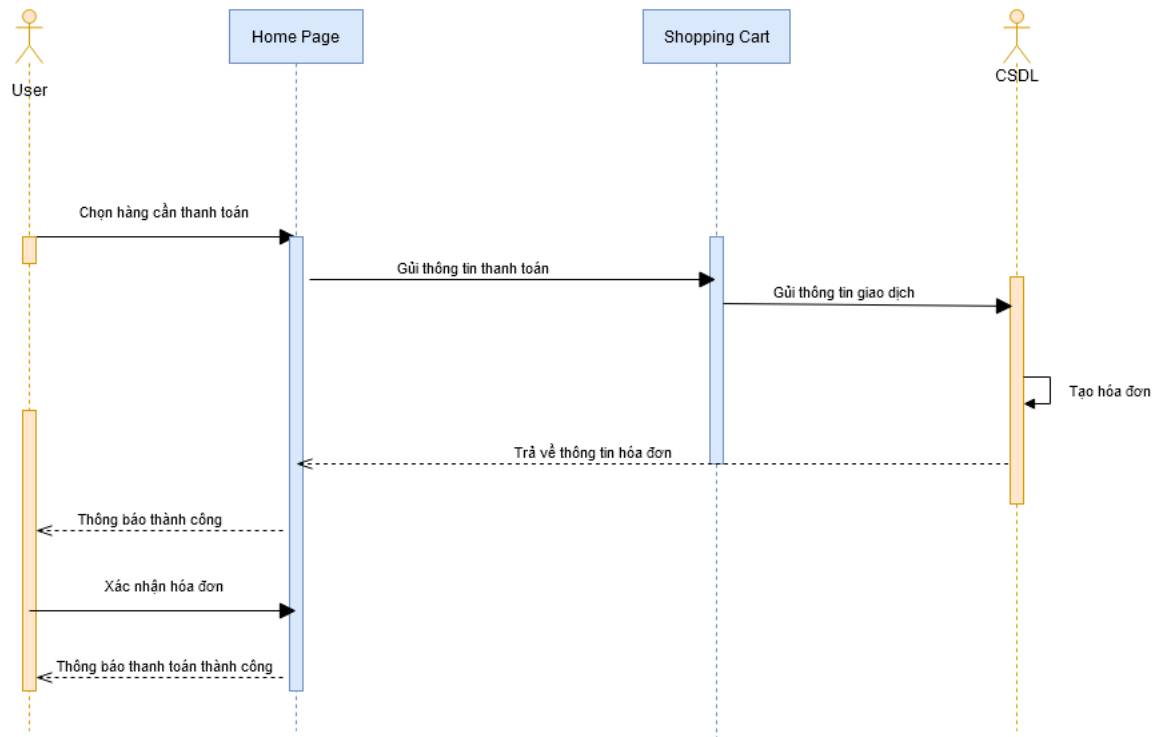
Thêm hàng vào giỏ hàng



Hình 4: Lược đồ tuần tự Thêm vào giỏ hàng

Chức năng thanh toán

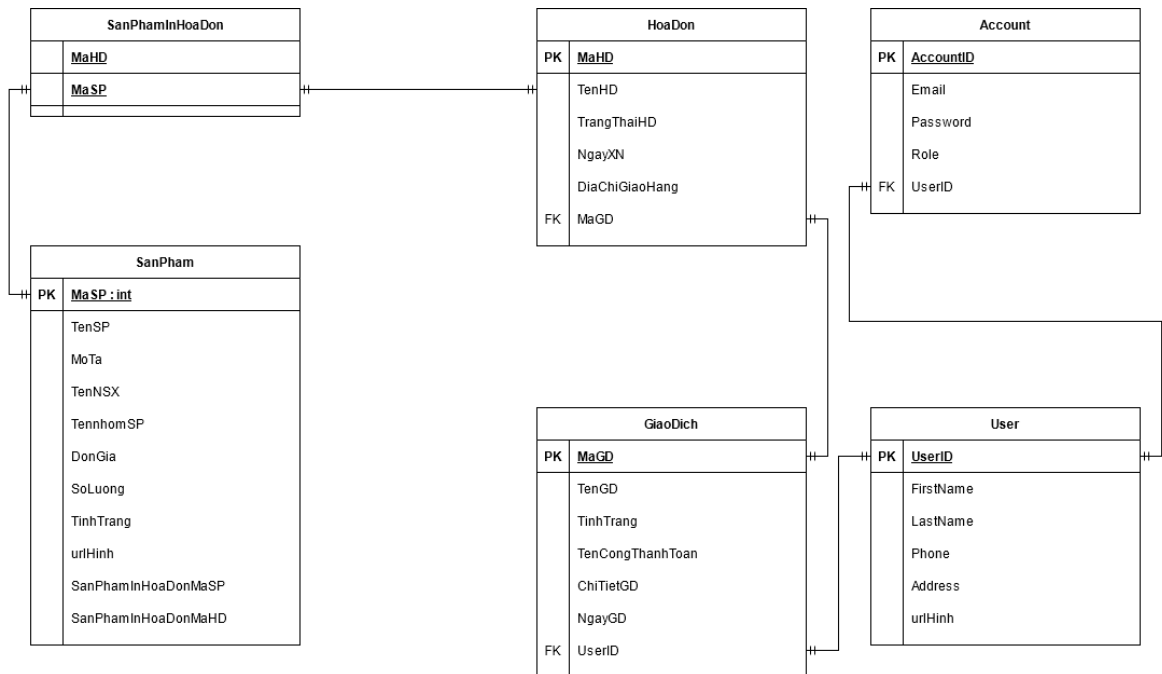
Thanh toán



Hình 5: Lược đồ tuần tự Thanh toán

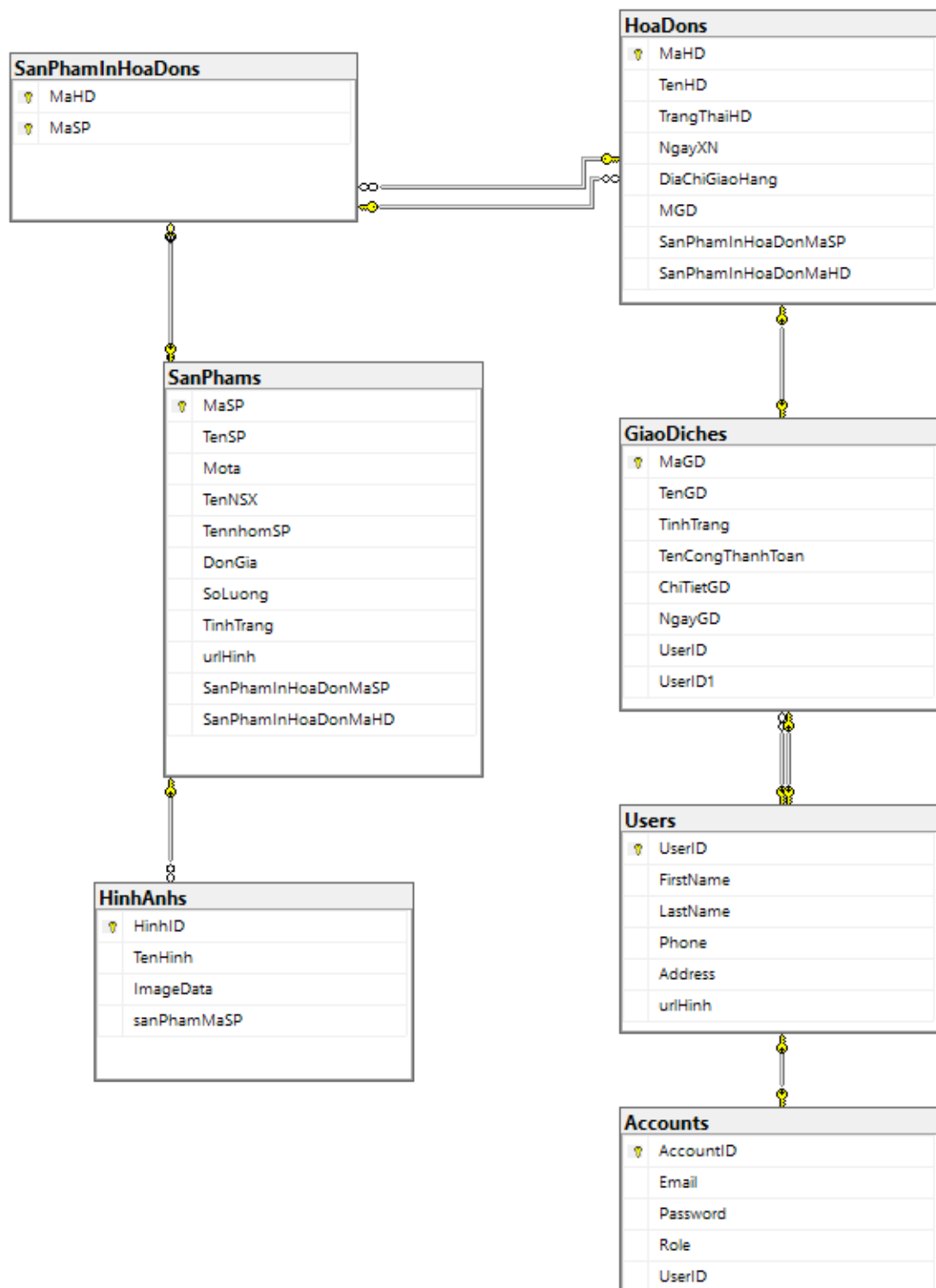
3.2.5. Mô tả Cơ sở dữ liệu của hệ thống

Mô hình liên kết thực thể (Sơ đồ ERD)



Hình 6: Sơ đồ liên kết thực thể ERD

Mô hình quan hệ các bảng trong Cơ sở dữ liệu (Sơ đồ Database Diagram)



Hình 7: Sơ đồ quan hệ các bảng trong cơ sở dữ liệu

Mô tả các bảng trong CSDL

Bảng Sản Phẩm

Tên field	Kiểu dữ liệu	Chi tiết	Khóa chính / Khóa ngoại
MaSP	int	Mã sản phẩm	Khóa chính
TenSP	Nvarchar(MAX)	Tên sản phẩm	
MoTa	Nvarchar(MAX)	Mô tả về sản phẩm	
TennhomSP	Nvarchar(MAX)	Tên nhóm Sản Phẩm chứa sản phẩm này	
DonGia	float	Đơn giá của sản phẩm	
SoLuong	int	Số lượng hiện có của sản phẩm	
TinhTrang	Nvarchar(MAX)	Tình trạng hiện tại của sản phẩm	

urlHinh	Nvarchar(MAX)	Hình ảnh của sản phẩm	
SanphamInHoaDonMaSP	int	Mã sản phẩm từ bảng HoaDon	Khóa ngoại
SanphamInHoaDonMaHD	int	Mã hóa đơn từ bảng HoaDon	Khóa ngoại

Bảng 23: Bảng Sản Phẩm

Bảng Giao Dịch

Tên field	Kiểu dữ liệu	Chi tiết	Khóa chính / Khóa ngoại
MaGD	int	Mã giao dịch	Khóa chính
TenGD	Nvarchar(MAX)	Tên giao dịch	
TinhTrang	Nvarchar(MAX)	Tình trạng giao dịch	
TenCongThanhToan	Nvarchar(MAX)	Tên loại thanh toán	

ChiTietGD	Nvarchar(MAX)	Chi tiết thông tin giao dịch	
NgayGD	datetime	Ngày thực hiện giao dịch	
UserID	int		Khóa ngoại
UserID1	int		Khóa ngoại

Bảng 24: Bảng Giao Dịch

Bảng Hóa Đơn

Tên field	Kiểu dữ liệu	Chi tiết	Khóa chính / Khóa ngoại
MaHD	int	Mã hóa đơn	Khóa chính
TenHD	Nvarchar(MAX)	Tên hóa đơn	
TrangThaiHD	Nvarchar(MAX)	Trạng thái của hóa đơn	

NgàyXN	Datetime2	Ngày hóa đơn được xác nhận	
DiaChiGiaoHang	Nvarchar(MAX)	Thông tin địa chỉ để giao hàng	
MGD	Int	Mã giao dịch	Khóa ngoại
SanphamInHoaDonMaSP	int	Mã sản phẩm từ bảng HoaDon	Khóa ngoại
SanphamInHoaDonMaHD	int	Mã hóa đơn từ bảng HoaDon	Khóa ngoại

Bảng 25: Bảng Hóa Đơn

Bảng Hình Ảnh

Tên field	Kiểu dữ liệu	Chi tiết	Khóa chính / Khóa ngoại
HinhID	int	Mã hình ảnh	Khóa chính
TenHinh	Nvarchar(MAX)	Tên hình ảnh	

ImageData	Vabinary(max)	Dữ liệu hình ảnh	
sanPhamMaSP	Int	Mã sản phẩm từ Sản Phẩm	Khóa ngoại

Bảng 26: Bảng Hình Ảnh

Bảng User

Tên field	Kiểu dữ liệu	Chi tiết	Khóa chính / Khóa ngoại
UserID	int	ID người dùng	Khóa chính
FirstName	Nvarchar(MAX)	Họ và tên đệm	
LastName	Nvarchar(MAX)	Tên người dùng	
Phone	Nvarchar(MAX)	Số điện thoại	
Address	Nvarchar(MAX)	Địa chỉ	
urlHinh	Nvarchar(MAX)	Hình đại diện cho tài khoản	

Bảng 27: Bảng User

Bảng Account

Tên field	Kiểu dữ liệu	Chi tiết	Khóa chính / Khóa ngoại
AccountID	int	Mã tài khoản	Khóa chính
Email	Nvarchar(MAX)	Email tài khoản sử dụng	
Password	Nvarchar(MAX)	Mật khẩu	
Role	Nvarchar(MAX)	Quyền hạn	
UserID	Nvarchar(50)	ID người dùng	Khóa ngoại

Bảng 28: Bảng Account






Bảng SanPhamInHoaDon (bảng nhiều nhiều N-N)

Tên field	Kiểu dữ liệu	Chi tiết	Khóa chính / Khóa ngoại
MaHD	int	Mã hóa đơn	Khóa chính
MaSP	int	Mã sản phẩm	Khóa chính

Bảng 29: Bảng SanPhamInHoaDon

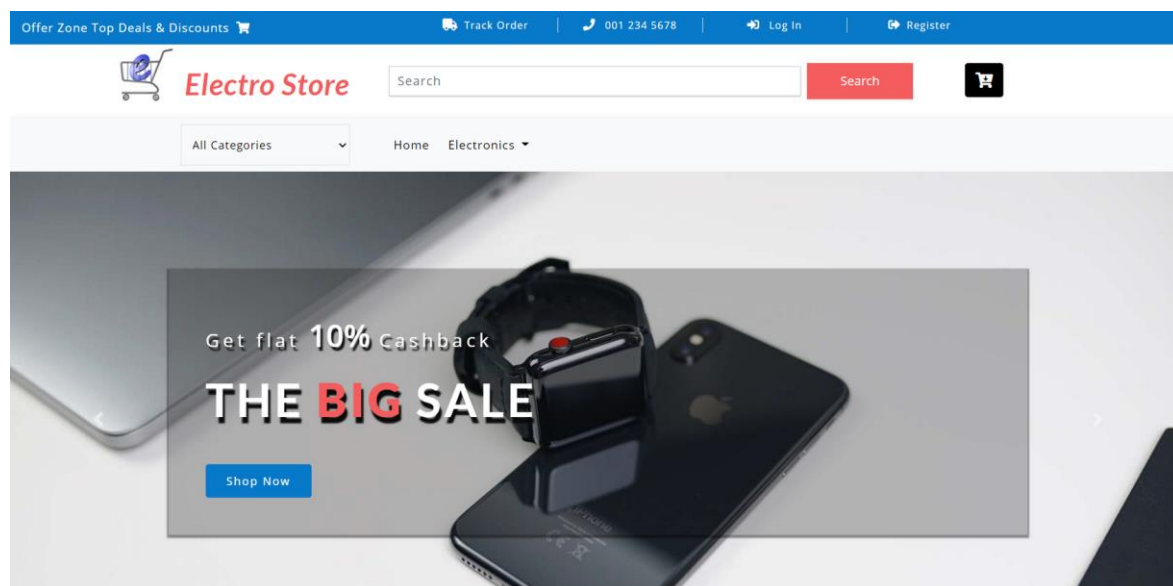
3.3. Thiết kế giao diện

3.3.1. Giao diện dành cho “Quản trị viên”

ADMIN 							
Dashboard Report Product Accounts Log Out							
+ Add Product							
Product	Quality	Product Name	Price	Manufacturer	Brand	Condition	
	9	SAMSUNG GALAXY J7	3999998 đ	SAMSUNG	PHONE	Old	Details Remove
	10	IPHONE X	10000000 đ	APPLE	PHONE	New	Details Remove
	10	OPPO A37f	10000000 đ	OPPO	PHONE	New	Details Remove
	5	Sony 80 cm (32 inches)	15000000 đ	SONY	Audio	New	Details Remove

Hình 8 : Giao diện mẫu Quản trị viên

3.3.2. Giao diện dành cho “Khách, Thành Viên”



Hình 9: Giao diện mẫu Khách, Thành viên

CHƯƠNG 4: CÀI ĐẶT SẢN PHẨM

4.1. Front – End

4.1.1. Môi trường phát triển

- Front End sử dụng Angular Framework để phát triển
- Cài đặt các công cụ, môi trường sau:
 - Windows 10 64-bit.
 - Node.js.
 - npm.
 - Visual Studio Code

4.1.2. Ứng dụng Angular

Cấu trúc ứng dụng^[1]

Các đoạn code xử lý chính trong ứng dụng Angular

- Login: Tạo giao diện và xử lý chức năng đăng nhập:
- Sử dụng *authenticationService* để xác nhận đăng nhập và *currentUserValue.role* để kiểm tra phân quyền của tài khoản đang đăng nhập

```
login = () => {
  console.log(this.email + this.password);
  this.authenticationService.login(this.email, this.password).subscribe(
    async (data) => {
      if (data != null && data.email) {
        localStorage.setItem('email', data.email);
        localStorage.setItem('password', data.password);
        localStorage.setItem('isLoggedIn', data.isLoggedIn = 'true');
        console.log('login Success');
        if (this.authenticationService.currentUserValue.role === 'Admin')
        {
          this.router.navigateByUrl('/dashboard');
        }
      }
      else{
        this.router.navigateByUrl('');
      }
    } else{
      console.log('login fail');
    }
  },
  (err) => console.error(err)
);
}
```

Hình 10 : Đoạn code xử lý Đăng Nhập

- Register: Thêm người dùng mới bằng xử lý đồng bộ

```
public createUser = async () => {
  try
  {
    console.log(this.FirstName, this.LastName, this.Phone, this.Address);
    const user = new User();
    user.FirstName = this.FirstName;
    user.LastName = this.LastName;
    user.Phone = this.Phone;
    user.Address = this.Address;
    console.log(user);
    const result = await this.registerservice.postuser(user) as any;
    console.log(result);
    localStorage.setItem('user', JSON.stringify(result));
    const userinfo = localStorage.getItem('user');
    const saveuser = JSON.parse(userinfo || '{}');
    const account = new Account();
    account.email = this.Email;
    account.password = this.Password;
    account.role = this.Role;
    account.userID = result.userID;
    console.log(account);
    const result2 = await this.registerservice.addAccount(account) as any;
    console.log(result2);
    this.router.navigateByURL('/');
  }
  catch (e)
  {
    return e;
  }
}
```

Hình 11 : Đoạn code xử lý thêm người dùng mới (Đăng ký)

- ProductDetail: Gọi đồng bộ qua API từ *localStorage*^[10]

```
getProductDetail = async (id: any) => {
  this.detailProduct = await this.productDetailService.getProductDetail(id) as Product[];
  console.log(this.detailProduct);
  if (this.productDetailService.checkLocalStorage() === true)
  {
    localStorage.removeItem('productdetail');
  }
  const productdetail = JSON.parse(localStorage.getItem('productdetail') || '[]');
  productdetail.push(this.detailProduct);
  localStorage.setItem('productdetail', JSON.stringify(productdetail));
  this.router.navigateByURL('/product-detail');
}
```

Hình 12 : Đoạn code xử lý hiển thị chi tiết một sản phẩm theo id

- Cart: sử dụng localStorage và JSON.parse với dữ liệu ở dạng JSON

```
public addCart = (item: any) => {  
    item.quantity = 1;  
    item.donGia = parseFloat(item.donGia);  
    const cart = JSON.parse(localStorage.getItem('cart') || '[]');  
    console.log(cart);  
    let flag = 0;  
    if (cart.length > 0) {  
        // sau khi add 1 product bất kì  
        cart.forEach((el: any) => {  
            if (el.maSP === item.maSP) {  
                el.quantity += 1;  
                el.donGia = el.donGia * el.quantity;  
                flag = 1;  
                return;  
            }  
        });  
        if (flag === 0) {  
            cart.push(item);  
        }  
    } else {  
        cart.push(item);  
    }  
    console.log(cart);  
    localStorage.setItem('cart', JSON.stringify(cart));  
}
```

Hình 13 : Đoạn code xử lý thêm sản phẩm vào giỏ hàng

- Paypal : Sử dụng các hàm event gọi từ ứng dụng qua PayPal^[5]

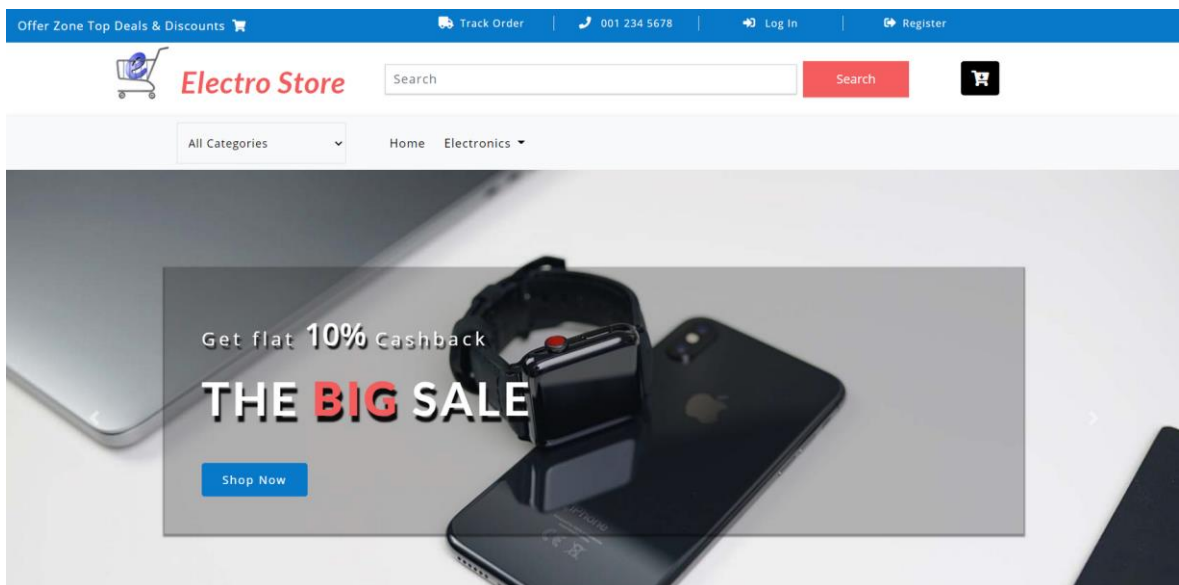
```
onApprove: (data, actions) => {
  console.log('onApprove - transaction was approved, but not authorized', data, actions);
  actions.order.get().then((details: any) => {
    console.log('onApprove - you can get full order details inside onApprove: ', details);
  });
  this.router.navigateByUrl('/cart');
  localStorage.removeItem('cart');
},
onClientAuthorization: (data) => {
  console.log('onClientAuthorization - you should probably inform your server about completed transaction at this point', data);
  this.showSuccess = true;
},
onCancel: (data, actions) => {
  console.log('OnCancel', data, actions);
},
onError: err => {
  console.log('OnError', err);
},
onClick: (data, actions) => {
  console.log('onClick', data, actions);
},
};
```

Hình 14 : Đoạn code xử lý thanh toán bằng PayPal

Các giao diện chức năng chính

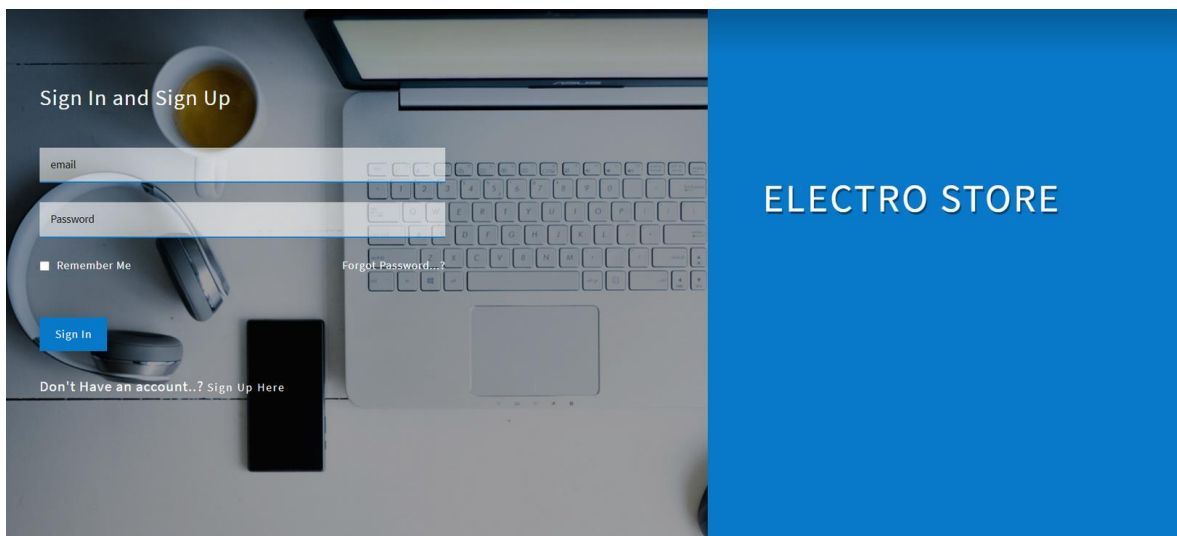
- Giao diện bao gồm các chức năng chính như:

Trang Home



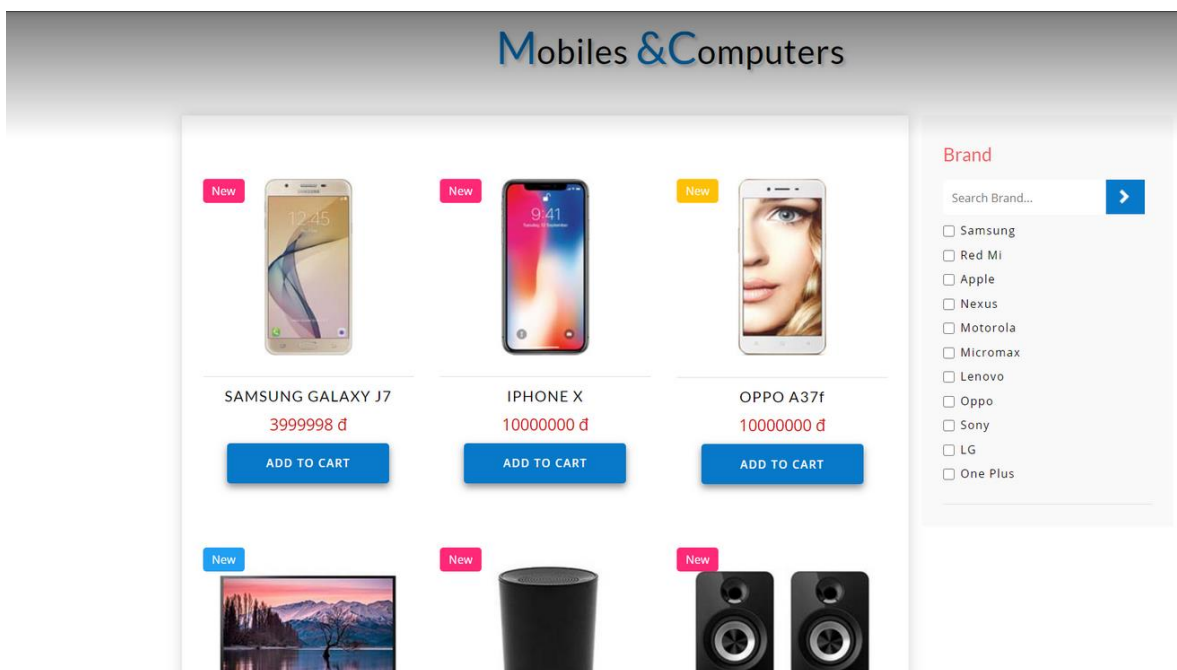
Hình 15: Cấu trúc giao diện trang Home

Trang Login



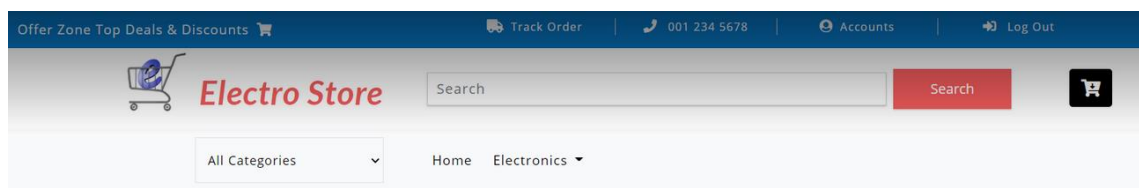
Hình 16: Cấu trúc giao diện trang Login

Trang Sản phẩm



Hình 17: Cấu trúc giao diện trang Sản Phẩm

Trang Chi tiết sản phẩm



Single Page



SAMSUNG GALAXY J7

3999998đ

Manufacturer: SAMSUNG

Free delivery

Cash on Delivery Eligible.

Shipping Speed to Delivery.

1 Year Manufacturer Warranty

ADD TO CART

Hình 18: Cấu trúc giao diện trang Chi tiết sản phẩm

Trang Giỏ hàng

Checkout

Your shopping cart contains: 1 Products

Product	Quantity	Product Name	Price	Remove
	1	SAMSUNG GALAXY J7	3999998 đ	
Grand Total:			\$3,999,998.00 đ	

Hình 19: Cấu trúc giao diện trang Giỏ hàng

Trang thông tin tài khoản


Offer Zone Top Deals & Discounts

Track Order

001 234 5678

Accounts

Log Out



Electro Store


Search

Search


All Categories

Home Electronics

Thanh



Change Image



Chọn tệp Không có tệp nào được chọn

First name

Thanh

Last name

Luyện

Phone

0943218973

Address

Bình Phước

Save

Change Pass Work

Hình 20: Cấu trúc giao diện trang Thông tin tài khoản

Trang Danh Sách sản phẩm

ADMIN

Dashboard





Report

Product

Accounts

Log Out

Add Product

Product	Quality	Product Name	Price	Manufacturer	Brand	Condition	
	9	SAMSUNG GALAXY J7	3999998 đ	SAMSUNG	PHONE	Old	<div>Details Remove</div>
	10	IPHONE X	10000000 đ	APPLE	PHONE	New	<div>Details Remove</div>
	10	OPPO A37f	10000000 đ	OPPO	PHONE	New	<div>Details Remove</div>
	5	Sony 80 cm (32 inches)	15000000 đ	SONY	Audio	New	<div>Details Remove</div>

Hình 21: Cấu trúc giao diện trang Danh sách sản phẩm

4.2. Back – End

4.2.1. Cấu trúc Project API

- Hệ thống Back End của đề tài được dựa trên nền tảng ASP.NET Core Web API với cây thư mục như sau:

Dependencies

- Là nơi chứa các thư viện để sử dụng trong việc lập trình cho hệ thống

Controllers

- Là thư mục chứa các trình điều khiển cho các Model của CSDL

Migrations

- Là thư mục chứa các file config cơ sở dữ liệu (database) cho mỗi lần thực thi Code First thông qua hai cơ chế là thêm Migration bằng câu lệnh: add migration và Cập nhật lại CSDL thông qua câu lệnh: update database

Model

- Là thư mục chứa các class khai báo về các trường của các Table trong cơ sở dữ liệu

File appsettings.json

- Là file cấu hình các hàm cho hệ thống

File Program.cs

- Là file hàm Main của chương trình

File Startup.cs

- Là file cấu hình Class sẽ xuất hiện khi bắt đầu thực thi chương trình

4.2.2. Một số hàm (Function) và Procedure

a. Hàm khai báo trong class Model

```
1 using System.Collections.Generic;
2 using System.ComponentModel.DataAnnotations;
3 using System.ComponentModel.DataAnnotations.Schema;
4
5 namespace Backend.Model
6 {
7     10 references
8     public class SanPham
9     {
10         [Key]
11         [DatabaseGenerated(DatabaseGeneratedOption.Identity)]
12         4 references
13         public int MaSP { get; set; }
14         0 references
15         public string TenSP { get; set; }
16         0 references
17         public string Mota { get; set; }
18         0 references
19         public string TenNSX { get; set; }
20         0 references
21         public string TennhomSP { get; set; }
22         0 references
23         public double DonGia { get; set; }
24         0 references
25         public int SoLuong { get; set; }
26         0 references
27         public string TinhTrang { get; set; }
28         0 references
29         public string urlHinh { get; set; }
30
31         // public HinhAnh hinhAnh { get; set; }
32         0 references
33         public SanPhamInHoaDon SanPhamInHoaDon { get; set; }
34
35         1 reference
36         public ICollection<SanPhamInHoaDon> SanPhamInHoaDons { get; set; }
37     }
38 }
```

Hình 22: Cấu trúc khai báo Class

b. Hàm API trong file Controller^[2]

Khai báo class, tạo Route

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using Microsoft.AspNetCore.Http;
using Microsoft.AspNetCore.Mvc;
using Microsoft.EntityFrameworkCore;
using Backend.Model;

namespace Backend.Controllers
{
    [Route("api/[controller]")]
    [ApiController]
    public class SanPhamsController : ControllerBase
    {
        private readonly ApplicationDbContext _context;

        public SanPhamsController(ApplicationDbContext context)
        {
            _context = context;
        }
    }
}
```

Hình 23: Cấu trúc khai báo hàm API

Hàm API GET all

```
// GET: api/SanPhams
[HttpGet]
public async Task<ActionResult<IEnumerable<SanPham>>> GetSanPhams()
{
    return await _context.SanPhams.ToListAsync();
}
```

Hình 24: Cấu trúc khai báo hàm GET all

Hàm API GET theo id

```
// GET: api/SanPhams/5
[HttpGet("{id}")]
0 references
public async Task<ActionResult<SanPham>> GetSanPham(int id)
{
    var sanPham = await _context.SanPhams.FindAsync(id);

    if (sanPham == null)
    {
        return NotFound();
    }

    return sanPham;
}
```

Hình 25: Cấu trúc khai báo hàm GET theo id

Hàm API POST

```
// POST: api/SanPhams
// To protect from overposting attacks, please enable the specific properties you want to bind to, for
// more details see https://aka.ms/RazorPagesCRUD.
[HttpPost]
0 references
public async Task<ActionResult<SanPham>> PostSanPham(SanPham sanPham)
{
    _context.SanPhams.Add(sanPham);
    await _context.SaveChangesAsync();

    return CreatedAtAction("GetSanPham", new { id = sanPham.MaSP }, sanPham);
}
```

Hình 26: Cấu trúc khai báo hàm POST

Hàm API PUT

```
// PUT: api/SanPhams/5
// To protect from overposting attacks, please enable the specific pro
// more details see https://aka.ms/RazorPagesCRUD.
[HttpPut("{id}")]
// references
public async Task<IActionResult> PutSanPham(int id, SanPham sanPham)
{
    if (id != sanPham.MaSP)
    {
        return BadRequest();
    }

    _context.Entry(sanPham).State = EntityState.Modified;

    try
    {
        await _context.SaveChangesAsync();
    }
    catch (DbUpdateConcurrencyException)
    {
        if (!SanPhamExists(id))
        {
            return NotFound();
        }
        else
        {
            throw;
        }
    }

    return NoContent();
}
```

Hình 27: Cấu trúc khai báo hàm PUT

Hàm API DELETE

```
// DELETE: api/SanPhams/5
[HttpDelete("{id}")]
0 references
public async Task<ActionResult<SanPham>> DeleteSanPham(int id)
{
    var sanPham = await _context.SanPhams.FindAsync(id);
    if (sanPham == null)
    {
        return NotFound();
    }

    _context.SanPhams.Remove(sanPham);
    await _context.SaveChangesAsync();

    return sanPham;
}
```

Hình 28: Cấu trúc khai báo hàm DELETE

PHẦN TỔNG KẾT

1. Kết quả đạt được

- Sau khi thực hiện đề tài, nhóm thực hiện đã có được những kết quả, thành tựu nhất định:
- Tạo ra được một trang website bán hàng thương mại điện tử với các chức năng cơ bản như:
- User (Người dùng):
 - Đăng nhập trực tiếp / thông qua Facebook, Google
 - Đăng ký / Xác thực thông qua Email / Quản lý tài khoản của mình
 - Xem Sản phẩm / Chi tiết sản phẩm
 - Thêm hàng vào giỏ hàng^[3]
 - Tìm kiếm sản phẩm^[4]
 - Thanh toán trực tiếp / thông qua Paypal
- Admin (Quản trị viên):
 - Quản lý người dùng / Phân quyền tài khoản
 - Quản lý sản phẩm / Quản lý bán hàng
 - Quản lý giỏ hàng / Thanh toán
 - Quản lý giao dịch / Quản lý hóa đơn
- Có cho mình những kiến thức về việc thực hiện một trang website thông qua việc kết nối giữa Client và Server
- Biết cách thực hiện một đề tài Tiểu Luận Chuyên Ngành đầy đủ

2. Ưu điểm

- Giao diện người dùng thân thiện, đơn giản, dễ sử dụng
- Các bộ dữ liệu được quản lý chặt chẽ với nhau
- Các chức năng được miêu tả tỉ mỉ, dễ tiếp cận
- Chạy được trên nhiều nền tảng trình duyệt Web như: Firefox, Chrome, Edge

3. Nhược điểm

- Hệ thống còn ít các tính năng nâng cao, có tính logic như:

- Chatbot bằng Digital Flow: Nhóm đã thực hiện nhưng gặp khó khăn
- Gọi ý mua hàng / Hàng hóa nổi bật: Nhóm đã thực hiện nhưng gặp khó khăn
- Các tính năng còn ở mức cơ bản, thông dụng

4. Khó khăn

- Trong quá trình thực hiện đề tài nhóm thực hiện nhóm đã gặp phải những khó khăn nhất định:
- Khó tiếp cận ban đầu với Angular Framework
- Mất nhiều thời gian trong việc hiểu cách thức hoạt động của việc gọi các chức năng thông qua các hàm API
- Khó khăn trong việc phối hợp giữa ASP.NET Core với Angular
- Khó khăn trong công tác kiểm thử và sửa lỗi
- Khó khăn trong việc tiếp cận và thực hiện các chức năng nâng cao
- Thời gian hạn chế khiến việc hoàn thiện sản phẩm vẫn chưa được chỉnh chu nhất có thể

5. Bài học kinh nghiệm

- Sau khi thực hiện đề tài, nhóm thực hiện đề tài đã rút ra được những bài học kinh nghiệm có ích:
- Xây dựng trang website thương mại điện tử có thể giúp mang lại lợi nhuận cho người xây dựng khi có thể tiếp tục phát triển và ứng dụng trang web một cách rộng rãi
- Biết cách xây dựng một hệ thống bán hàng thương mại điện tử và hiểu được cách thức vận hành của trang web.
- Hiểu hơn về cách thức phối hợp giữa các nền tảng, công nghệ lập trình.

6. Hướng phát triển

- Với hệ thống hiện tại, nhóm thực hiện đề tài có thể tiếp thu và cải thiện sản phẩm của mình sao cho hợp lý như:
- Cải thiện tốc độ mở trang web, cải thiện thời gian chờ phản hồi
- Cải thiện giao diện người dùng UI/UX sao cho đơn giản hơn nhưng tinh tế hơn

- Cải thiện các tính năng sẵn có, đặc biệt là các tính năng về giỏ hàng, thanh toán, đánh giá sản phẩm, thống kê bán hàng, quản lý đơn hàng.
- Thêm được một số tính năng nâng cao như: Chatbot, Giao hàng thông qua Grab, GoViet, Gợi ý mua hàng, hàng hóa nổi bật, SEO.

TÀI LIỆU THAM KHẢO

[1] Tham khảo cấu trúc và cách viết của Angular Project

<https://angular.io/>

[2] Tham khảo các phương thức sử dụng hàm API trong .NET Core

<https://docs.microsoft.com/en-us/aspnet/core/tutorials/first-web-api?view=aspnetcore-5.0&tabs=visual-studio>

[3] Tham khảo cách chuyển đổi (Convert) một hình ảnh sang dạng base64 trong Angular

<https://stackoverflow.com/questions/42482951/converting-an-image-to-base64-in-angular-2>

[4] Tham khảo cách lọc sản phẩm, tìm kiếm trong Angular

<https://www.npmjs.com/package/ng2-search-filter>

[5] Tham khảo cách tích hợp Paypal vào trong dịch vụ thanh toán

<https://www.npmjs.com/package/ngx-paypal>

[6] Tham chiếu cho cơ sở lý thuyết về Angular

<https://blog.tinohost.com/angular-la-gi/>

[7] Tham chiếu cho cơ sở lý thuyết về .NET Entity Framework Core

<https://netcore.vn/bai-viet/tong-quan-ve-aspnet-core>

[8] Tham chiếu cho cơ sở lý thuyết về .NET Core Web API

<https://topdev.vn/blog/api-la-gi/#ftoc-heading-3>

[9] Tham chiếu cho cơ sở lý thuyết về Microsoft SQL Server

<http://sqladvice.com/tong-quan-ve-sql-server/>

[10] Tham khảo cách lưu dữ liệu vào localStorage

<https://stackoverflow.com/questions/51536262/angular-6-saving-data-to-local-storage>

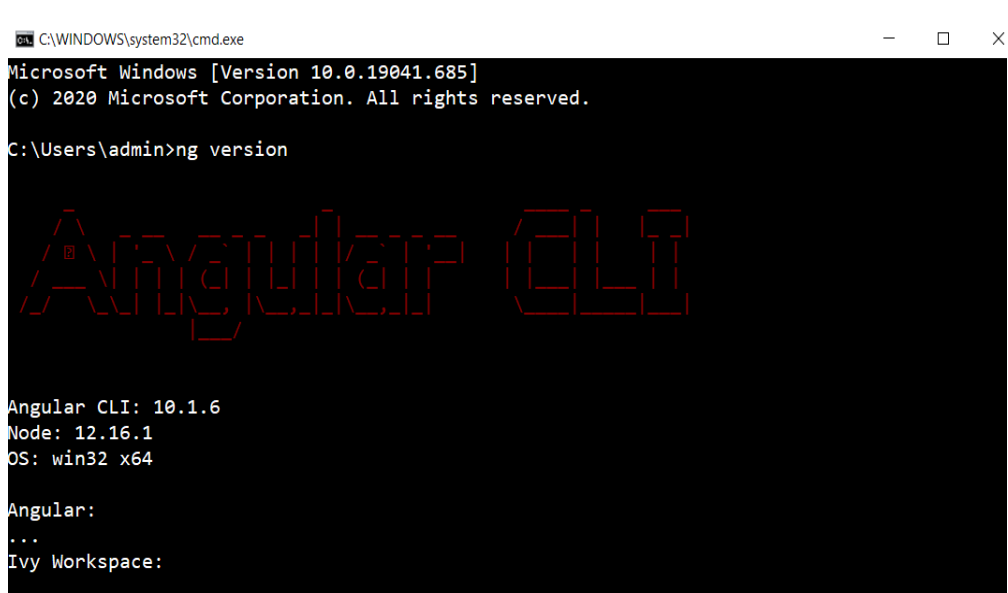
PHỤ LỤC

1. Tạo một ứng dụng Angular application

- Công cụ Angular CLI - Angular Command Line Interface - ra đời hỗ trợ khởi tạo, sinh các component cũng như kiểm thử và triển khai ứng dụng.
- Angular CLI thực chất là một Node.js package, cài đặt thông qua trình quản lý package của Node.js - npm. Mở cửa sổ console và gõ vào như sau:

```
npm install -g @angular/cli
```

- Kiểm tra cài đặt thành công và số phiên bản bằng cách nhập:



```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows [Version 10.0.19041.685]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\Users\admin>ng version

Angular CLI
Angular CLI: 10.1.6
Node: 12.16.1
OS: win32 x64

Angular:
...
Ivy Workspace:
```

version

Kiểm tra môi trường Angular trong máy tính

- Để tạo một Angular Application, ta gõ câu lệnh sau vào Terminal của Visual Studio Code: `ng new first-angular-app`

```
SUN-ASTERISK\ngo.tien.phat@b121065-pc:/var/www/html$ ng new first-angular-app
? Would you like to add Angular routing? Yes
? Which stylesheet format would you like to use? Less [ http://lesscss.org
CREATE first-angular-app/README.md (1033 bytes)
CREATE first-angular-app/.editorconfig (274 bytes)
CREATE first-angular-app/.gitignore (631 bytes)
CREATE first-angular-app/angular.json (3742 bytes)
CREATE first-angular-app/package.json (1260 bytes)
CREATE first-angular-app/tsconfig.json (458 bytes)
CREATE first-angular-app/tslint.json (3185 bytes)
CREATE first-angular-app/.browserslistrc (853 bytes)
CREATE first-angular-app/karma.conf.js (1029 bytes)
CREATE first-angular-app/tsconfig.app.json (287 bytes)
CREATE first-angular-app/tsconfig.spec.json (333 bytes)
CREATE first-angular-app/src/favicon.ico (948 bytes)
CREATE first-angular-app/src/index.html (301 bytes)
CREATE first-angular-app/src/main.ts (372 bytes)
CREATE first-angular-app/src/polyfills.ts (2835 bytes)
CREATE first-angular-app/src/styles.less (80 bytes)
CREATE first-angular-app/src/test.ts (753 bytes)
CREATE first-angular-app/src/assets/.gitkeep (0 bytes)
CREATE first-angular-app/src/environments/environment.prod.ts (51 bytes)
CREATE first-angular-app/src/environments/environment.ts (662 bytes)
CREATE first-angular-app/src/app/app-routing.module.ts (245 bytes)
CREATE first-angular-app/src/app/app.module.ts (393 bytes)
CREATE first-angular-app/src/app/app.component.less (0 bytes)
CREATE first-angular-app/src/app/app.component.html (25757 bytes)
CREATE first-angular-app/src/app/app.component.spec.ts (1090 bytes)
CREATE first-angular-app/src/app/app.component.ts (222 bytes)
CREATE first-angular-app/e2e/protractor.conf.js (869 bytes)
CREATE first-angular-app/e2e/tsconfig.json (294 bytes)
CREATE first-angular-app/e2e/src/app.e2e-spec.ts (650 bytes)
CREATE first-angular-app/e2e/src/app.po.ts (301 bytes)
✓ Packages installed successfully.
  Successfully initialized git.
```

Tải và cài đặt các Module

- Đợi sau hệ thống tải về đầy đủ các module cần thiết, ta đã có một Project angular cơ bản để bắt đầu thực hiện công việc của mình.