**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT HƯNG YÊN**

****

**ĐỒ ÁN 5**

**XÂY DỰNG WEBSITE GIỚI THIỆU VÀ ĐẶT HÀNG**

**PHỤ KIỆN ĐIỆN TỬ**

NGÀNH: CÔNG NGHỆ THÔNG TIN

CHUYÊN NGÀNH: KỸ THUẬT PHẦN MỀM

SINH VIÊN: **NGUYỄN MINH QUANG**

MÃ LỚP: **101177**

HƯỚNG DẪN: **VŨ XUÂN THẮNG**

**HƯNG YÊN – 2020**

**NHẬN XÉT**

**Nhận xét của giảng viên hướng dẫn:**

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

**GIẢNG VIÊN HƯỚNG DẪN**

*(Ký và ghi rõ họ tên)*

**LỜI CAM ĐOAN**

Em xin cam đoan đồ án “Xây dựng website giới thiệu và đặt hàng phụ kiện” là kết quả thực hiện của bản thân em dưới sự hướng dẫn của thầy giáo Vũ Xuân Thắng.

Những phần sử dụng tài liệu tham khảo trong đồ án đã được nêu rõ trong phần tài liệu tham khảo. Các kết quả trình bày trong đồ án và chương trình xây dựng được hoàn toàn là kết quả do bản thân em thực hiện.

Nếu vi phạm lời cam đoan này, em xin chịu hoàn toàn trách nhiệm trước khoa và nhà trường.

*Hưng Yên, ngày … tháng … năm…..*

SINH VIÊN

…………………………………..

**LỜI CẢM ƠN**

Để có thể hoàn thành đồ án này, lời đầu tiên em xin phép gửi lời cảm ơn tới bộ môn Công nghệ phần mềm, Khoa Công nghệ thông tin – Trường Đại học Sư phạm Kỹ thuật Hưng Yên đã tạo điều kiện thuận lợi cho em thực hiện đồ án môn học này.

Đặc biệt em xin chân thành cảm ơn thầy giáo Vũ Xuân Thắng đã rất tận tình hướng dẫn, chỉ bảo em trong suốt thời gian thực hiện đồ án vừa qua.

Em cũng xin chân thành cảm ơn tất cả các Thầy, các Cô trong Trường đã tận tình giảng dạy, trang bị cho em những kiến thức cần thiết, quý báu để giúp em thực hiện được đồ án này.

Mặc dù em đã có cố gắng, nhưng với trình độ còn hạn chế, trong quá trình thực hiện đề tài không tránh khỏi những thiếu sót. Em hi vọng sẽ nhận được những ý kiến nhận xét, góp ý của các Thầy giáo, Cô giáo về những kết quả triển khai trong đồ án.

Em xin trân trọng cảm ơn!

**MỤC LỤC**

[DANH MỤC CÁC THUẬT NGỮ 7](#_Toc55249187)

[DANH MỤC CÁC BẢNG 8](#_Toc55249188)

[DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ 9](#_Toc55249189)

[CHƯƠNG 1: TỔNG QUAN VỀ ĐỀ TÀI 11](#_Toc55249190)

[1.1. Lý do chọn đề tài 11](#_Toc55249191)

[1.2. Mục tiêu của đề tài 11](#_Toc55249192)

[1.2.1 Mục tiêu tổng quát 11](#_Toc55249193)

[1.2.2 Mục tiêu cụ thể 12](#_Toc55249194)

[1.3. Giới hạn và phạm vi của đề tài 13](#_Toc55249195)

[1.3.1 Đối tượng nghiên cứu 13](#_Toc55249196)

[1.3.2 Phạm vi nghiên cứu 13](#_Toc55249197)

[1.4. Nội dung thực hiện 13](#_Toc55249198)

[1.5. Phương pháp tiếp cận 14](#_Toc55249199)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 15](#_Toc55249200)

[2.1. Quy trình phát triển phần mềm 15](#_Toc55249201)

[2.2. Phương pháp phân tích thiết kế hướng đối tượng 18](#_Toc55249202)

[2.4. Tổng quan về Angular 22](#_Toc55249203)

[2.5. Lập trình Web API NET CORE 24](#_Toc55249204)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 27](#_Toc55249205)

[3.1 Đặc tả yêu cầu phần mềm 27](#_Toc55249206)

[3.1.1 Các yêu cầu chức năng 28](#_Toc55249207)

[3.1.2 Biểu đồ lớp thực thể 43](#_Toc55249208)

[3.1.3 Các yêu cầu phi chức năng 47](#_Toc55249209)

[3.2 Thiết kế hệ thống 48](#_Toc55249210)

[3.2.1 Thiết kế cơ sở dữ liệu 48](#_Toc55249211)

[3.2.2 Thiết kế lớp đối tượng 51](#_Toc55249212)

[3.2.3 Thiết kế giao diện 61](#_Toc55249213)

[CHƯƠNG 4: TRIỂN KHAI WEBSITE 66](#_Toc55249214)

[4.1 Triển khai các chức năng cho phân hệ người dùng 66](#_Toc55249215)

[4.1.1 Trang chủ 66](#_Toc55249216)

[4.1.2 Trang Xem Sản phẩm 67](#_Toc55249217)

[4.1.3 Trang Quản lý giỏ hàng 68](#_Toc55249218)

[4.2 Triển khai các chức năng cho phân hệ quản trị nội dung 69](#_Toc55249219)

[4.3 Kiểm thử và triển khai ứng dụng 70](#_Toc55249220)

[4.3.1 Kiểm thử 70](#_Toc55249221)

[4.3.2 Đóng gói ứng dụng 75](#_Toc55249222)

[4.3.3 Triển khai ứng dụng 75](#_Toc55249223)

[KẾT LUẬN 77](#_Toc55249224)

[TÀI LIỆU THAM KHẢO 78](#_Toc55249225)

DANH MỤC CÁC THUẬT NGỮ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Từ viết tắt | Cụm từ tiếng anh | Diễn giải |
| 1 | MVC | Model View Control | Mẫu thiết kế MVC |
| 2 | SQL | Structured Query Language | Ngôn ngữ truy vấn dữ liệu |
| 3 | API | Application Programing Interface | Lập trình giao diện ứng dụng |
| 4 | REST | Representation State Transfer | Tiêu chuẩn thiết kế API |
| 5 | HTTP | Hypertext Transfer Protocol | Giao thức truyền siêu văn bản |
| 6 | SPA | Single Page Application | Ứng dụng trang đơn |

DANH MỤC CÁC BẢNG

[Bảng 3.1 Chức năng của phân hệ quản trị nội dung 30](#_Toc56195760)

[Bảng 3.2 Chức năng của phân hệ người dùng 35](#_Toc56195761)

[Bảng 3.3 Bảng danh mục sản phẩm 46](#_Toc56195762)

[Bảng 3.4 Bảng sản phẩm 47](#_Toc56195763)

[Bảng 3.5 Bảng khách hàng 47](#_Toc56195764)

[Bảng 3.6 Bảng hóa đơn 48](#_Toc56195765)

[Bảng 3.7 Bảng chi tiết hóa đơn 48](#_Toc56195766)

[Bảng 3.8 Bảng người dùng 49](#_Toc56195767)

[Bảng 3.9 Bảng tin tức 49](#_Toc56195768)

[Bảng 3.10 Cấu trúc bảng danh mục sản phẩm 51](#_Toc56195769)

[Bảng 3.11 Cấu trúc bảng sản phẩm 52](#_Toc56195770)

[Bảng 3.12 Cấu trúc bảng khách hàng 52](#_Toc56195771)

[Bảng 3.13 Cấu trúc bảng hóa đơn 53](#_Toc56195772)

[Bảng 3.14 Cấu trúc bảng chi tiết hóa đơn 53](#_Toc56195773)

[Bảng 3.15 Cấu trúc bảng người dùng 54](#_Toc56195774)

[Bảng 3.16 Cấu trúc bảng tin tức 54](#_Toc56195775)

DANH MỤC CÁC HÌNH VẼ, ĐỒ THỊ

[Hình: 3.1 Biểu đồ UseCase tổng quát trang quản trị 30](#_Toc56348257)

[Hình: 3.2 Use Case quản lý sản phẩm 30](#_Toc56348258)

[Hình: 3.3 UserCase quản lý danh mục sản phẩm 31](#_Toc56348259)

[Hình: 3.4 Use Case quản lý người dùng 31](#_Toc56348260)

[Hình: 3.5 Use Case quản lý đơn hàng 32](#_Toc56348261)

[Hình: 3.6 Use Case quản lý tin tức 32](#_Toc56348262)

[Hình: 3.7 Use Case tổng quát phân hệ trang người dùng 37](#_Toc56348263)

[Hình: 3.8 Use Case xem sản phẩm 38](#_Toc56348264)

[Hình: 3.9 Use Case tìm kiếm sản phẩm 38](#_Toc56348265)

[Hình: 3.10 Use Case quản lý giỏ hàng 39](#_Toc56348266)

[Hình: 3.11 Use Case quản lý đơn hàng 39](#_Toc56348267)

[Hình: 3.12 Biểu đồ lớp thực thể hệ thống 48](#_Toc56348268)

[Hình: 3.13: Mô hình cơ sở dữ liệu quan hệ 49](#_Toc56348269)

[Hình: 3.14 Biểu đồ lớp VOPC của Use Case quản lý sản phẩm 53](#_Toc56348270)

[Hình: 3.15 Biều đồ lớp VOPC của UseCase thêm sản phẩm 53](#_Toc56348271)

[Hình: 3.16 Biều đồ lớp VOPC của UseCase sửa sản phẩm 54](#_Toc56348272)

[Hình: 3.17 Biều đồ lớp VOPC của UseCase xóa sản phẩm 54](#_Toc56348273)

[Hình: 3.18 Biều đồ lớp VOPC của UseCase đặt hàng 55](#_Toc56348274)

[Hình: 3.19 Biểu đồ tuần tự của UseCase quản lý thiết sản phẩm 55](#_Toc56348275)

[Hình: 3.20 Biểu đồ tuần tự của Use case thêm sản phẩm 56](#_Toc56348276)

[Hình: 3.21 Biểu đồ tuần tự của Use case sửa sản phẩm 57](#_Toc56348277)

[Hình: 3.22 Biểu đồ tuần tự của Use case xóa thiết sản phẩm 57](#_Toc56348278)

[Hình: 3.23 Biểu đồ tuần tự của Use case xem sản phẩm theo loại 58](#_Toc56348279)

[Hình: 3.24 Biểu đồ tuần tự của Use case xem chi tiết sản phẩm 58](#_Toc56348280)

[Hình: 3.25 Biểu đồ tuần tự của Use case đặt hàng 59](#_Toc56348281)

[Hình: 3.26 Biểu đồ lớp chi tiết VOPC quản lý sản phẩm 59](#_Toc56348282)

[Hình: 3.27 Biểu đồ lớp chi tiết VOPC thêm sản phẩm 60](#_Toc56348283)

[Hình: 3.28 Biểu đồ lớp chi tiết VOPC sửa sản phẩm 60](#_Toc56348284)

[Hình: 3.29 Biểu đồ lớp chi tiết VOPC xóa sản phẩm 61](#_Toc56348285)

[Hình: 3.30 Biểu đồ lớp chi tiết VOPC đặt hàng 61](#_Toc56348286)

[Hình: 3.31 Giao diện trang quản lý sản phẩm 62](#_Toc56348287)

[Hình: 3.32 Giao diện trang quản lý người dùng 63](#_Toc56348288)

[Hình: 3.33 Giao diện trang quản lý đơn hàng 63](#_Toc56348289)

[Hình: 3.34 Giao diện trang loại sản phẩm 64](#_Toc56348290)

[Hình: 3.35 Giao diện trang giỏ hàng 65](#_Toc56348291)

[Hình: 3.36 Giao diện trang đặt hàng 66](#_Toc56348292)

# TỔNG QUAN VỀ ĐỀ TÀI

* 1. Lý do chọn đề tài

Hiện nay, các công nghệ tiên tiến phát triển ngày càng mạnh mẽ và được ứng dụng ngày càng nhiều vào các lĩnh vực kinh tế, sản xuất cũng như đời sống thường nhật của con người. Một điểm tiêu biểu trong việc phát triển các công nghệ đó phải kể đến việc ứng dụng công nghệ thông tin vào hầu khắp các hoạt động. Nhờ đó, các công việc được thực hiện nhanh, chính xác và đạt kết quả cao hơn rất nhiều.

Khi xã hội ngày càng phát triển, mức sống của người dân được nâng cao thu nhập kinh tế ngày càng được cải thiện thì những phụ điện tử không còn trở nên xa lạ với mọi người. Tuy nhiên, với cuộc sống ngày càng bận rộn như hiện nay thì việc muốn mua những phụ kiện điện tử mình cần thiết thì người tiêu dùng phải đến tận cửa hàng để chọn lựa vì thế sẽ mất khá nhiều thời gian và công sức.

Cùng với các lý do nêu trên, qua tìm hiểu em được biết việc ứng dụng giới thiệu và đặt hàng phụ kiện điện tử trực tuyến sẽ giúp cho khách hàng giảm bớt được thời gian và công sức phải đến tận cửa hàng để mua. Muốn lựa chọn cho mình những phụ kiện điện tử ưng ý phù hợp với túi tiền thì khách hàng chỉ cần ngồi bên chiếc máy tính hoặc điện thoại có nối mạng Internet là có thế mua được mặt hàng mình cần.

Do đó em đã chọn thực hiện đề tài: “Xây dựng Website giới thiệu và đặt hàng phụ kiện điện tử trực tuyến” Phần mềm được xây dựng với định hướng giúp việc mua sắm phụ kiện điện tử của khách hàng dễ dàng và nhanh gọn hơn, không mất thời gian và công sức.

1.2. Mục tiêu của đề tài

1.2.1 Mục tiêu tổng quát

Mục tiêu cuối cùng của đề tài là vận dụng các kiến thức đã tìm hiểu xây dựng được website giới thiệu và đặt hàng phụ kiện điện tử với đầy đủ các chức năng cơ bản.

Chủ cửa hàng có thể thêm, xóa, chỉnh sửa thông tin sản phẩm, danh mục.

Khách hàng có thể vào website để xem thông tin sản phẩm được hiển thị trên website của cửa hàng và có thể đặt hàng trực tiếp thông qua chức năng giỏ hàng trên website.

Website thông báo thông tin hóa đơn chi tiết cho khách hàng khi khách hàng đặt hàng.

Khách hàng có thể phản hồi thông qua chức năng bình luận hoặc thông tin cửa hàng hiển thị trên website.

Cho phép tìm kiếm thông tin sản phẩm.

1.2.2 Mục tiêu cụ thể

*Về kiến thức:*

* Trình bày được các hoạt động trong quy trình phát triển một sản phẩm phần mềm.
* Trình bày được phương pháp phân tích thiết kế hướng đối tượng và các biểu đồ UML tương ứng trong các giai đoạn đặc tả yêu cầu, phân tích, thiết kế phần mềm.
* Phân tích được các xu hướng lập trình Web mới.
* Trình bày được các kiến thức về thiết kế và lập trình phía front-end sử dụng Angular.
* Trình bày được các kiến thức về lập trình sử dụng Web API trên nền tảng .NET Core và các framework khác.

*Về kỹ năng:*

* Vận dụng được các kiến thức phân tích thiết kế hướng đối tượng để xây dựng các biểu đồ ca sử dụng, lớp thực thể, lớp phân tích, biểu đồ tuần tự, lớp thiết kế chi tiết cho hệ thống website phát triển.
* Sử dụng và quản trị được cơ sở dữ liệu theo thiết kế trên hệ quản trị cơ sở dữ liệu SQL Server.
* Vận dụng được ngôn ngữ thiết kế và lập trình web phía front-end (Angular) và back-end (Web API) để xây dựng website theo thiết kế.
* Kiểm thử và triển khai được Website trên Internet.
* Quản lý tài liệu, quản lý phiên bản bằng các công cụ như Github, Gitlab, Google Drive.

*Sản phẩm:*

* Cuốn báo cáo trình bày đúng theo mẫu và yêu cầu.
* Website giới thiệu và đặt hàng phụ kiện điện tử được triển khai thử nghiệm trên môi trường Internet.

1.3. Giới hạn và phạm vi của đề tài

1.3.1 Đối tượng nghiên cứu

* Đối tượng nghiên cứu của đề tài: nhu cầu tìm hiểu và đặt mua phụ kiện qua mạng.
* Khách thể nghiên cứu: mọi người tiêu dùng trên toàn quốc có nhu cầu đặt mua phụ kiện điện tử, chủ cửa hàng phụ kiện điện tử.

1.3.2 Phạm vi nghiên cứu

* Phạm vi không gian: tại cửa hàng bán phụ kiện điện tử và các trang Web bán phụ kiện điện tử.
* Quy trình khách hàng đặt hàng, mua hàng, nhận hàng, căn cứ vào thông tin thực tế để thiết kế hệ thống thích hợp phục vụ đúng tuần tự nhu cầu sử dụng
* Ý nghĩa khoa học và thực tiễn của đề tài: nghiên cứu này góp phần nâng cao việc quản lý các mặt hàng cũng như các hoạt động đặt mua phụ kiện qua mạng.

1.4. Nội dung thực hiện

* *Xây dựng và khảo sát thực trạng của đề tài*

Khảo sát thực trạng về nhu cầu mua sắm phụ kiện điện tử phụ kiện của con người ngày nay ta xác định và lựa chọn đề tài “Xây dựng website giới thiệu và đặt hàng phụ kiện điện tử”.

* *Xác định các yêu cầu chức năng của người dùng*

Giai đoạn này ta cần thu thập các thông tin về đối tượng của đề tài ở đây là hệ thống bán phụ kiện điện tử. Xác định những yêu cầu cơ bản người dùng đặt ra cho hệ thống để tiến hành xây dựng một hệ thống đúng theo yêu cầu và tiêu chuẩn đề ra.

* *Phân tích các yêu cầu chức năng của người dùng*

Dựa vào những yêu cầu đã xác định ở trên, phân tích cụ thể từng yêu cầu chức năng của hệ thống. Xây dựng các ca sử dụng tương ứng với từng chức năng cụ thể bằng biểu đồ UML.

* *Xây dựng hệ cơ sở dữ liệu cho hệ thống*

Xây dựng cơ sở dữ liệu cho hệ thống. Sau khi đã có những thông tin về đối tượng thì xây dựng các mô hình thực thể và các mô hình chung liên quan.

* *Xây dựng chương trình website bằng công nghệ Angular và Web API .NET CORE*

Sau khi có một hệ thống hoàn chỉnh trên lý thuyết và đảm bảo các yêu cầu đã đề ra sẽ tiến hành xây dựng website cụ thể dựa trên framework Angular kết hợp với Web API NET CORE và hệ quản trị cơ sở dữ liệu SQL Server. Đây là bước khó khăn trong quá trình tiến hành thực hiện vì nó đòi hỏi những hiểu biết về framework Angular, Web API NET CORE và hệ quản trị cơ sở dữ liệu SQL server và trong quá trình xây dựng sẽ phát sinh rất nhiều lỗi. Đây là bước rất quan trọng trong quá trình thực hiện đồ án vì nếu bước này không được thực hiện thì quá trình xây dựng chỉ được tiến hành trên cơ sở lý thuyết.

* *Hoàn thiện chương trình, hướng dẫn cài đặt và sử dụng*

Đây là bước cuối cùng trong quá trình thực hiện đồ án, đưa sản phẩm vào thực nghiệm để đưa ra những đánh giá nhận xét về hệ thống, nhận xét ưu, nhược điểm để đưa ra các biện pháp nhằm chỉnh sửa, nâng cấp các chức năng làm sao cho hệ thống hoàn thiện hơn.

1.5. Phương pháp tiếp cận

* Nghiên cứu tài liệu về quy trình phát triển phần mềm, các phương pháp thiết kế hướng đối tượng. Nghiên cứu các kiến thức và kỹ năng lập trình phần front end (Angular, HTML, CSS, JavaScript, Jquery, Bootstrap…) và phần back end (Web API NET CORE). Tìm hiểu các tài liệu về hệ quản trị cơ sở dữ liệu SQL Server.
* Khảo sát một số website bán phụ kiện điện tử và các hệ thống bán phụ kiện điện tử trên thực tế.

# CƠ SỞ LÝ THUYẾT

2.1. Quy trình phát triển phần mềm

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

*Các giai đoạn của chu trình phát triển phần mềm:*

Phân tích yêu cầu (Analysis)

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

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

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

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

*Thiết kế hệ thống (Design of the System)*

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

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

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

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

*Xây dựng phần mềm (Software Construction)*

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

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

*Thử nghiệm đơn vị:*

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

*Thử nghiệm đơn vị độc lập:*

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

*Thử nghiệm hệ thống (System Testing)*

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

*Thực hiện, triển khai (System Implementation)*

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

*Bảo trì, nâng cấp (System Maintenance)*

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

2.2. Phương pháp phân tích thiết kế hướng đối tượng

*Khái niệm về Phân tích và thiết kế hướng đối tượng (Object Oriented Analysis and Design: OOAD)*

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

Để phân tích và thiết kế một phần mềm thì có nhiều cách làm, một trong những cách làm đó là xem hệ thống gồm những đối tượng sống trong đó và tương tác với nhau. Việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ hệ thống và cài đặt được nó. Phương thức này gọi là Phân tích thiết kế hướng đối tượng (OOAD).

*Khái niệm về UML (Unified Modeling Language)*

UML là ngôn ngữ mô hình hóa hợp nhất dùng để biểu diễn hệ thống. Nói một cách đơn giản là nó dùng để tạo ra các bản vẽ nhằm mô tả thiết kế hệ thống. Các bản vẽ này được sử dụng để các nhóm thiết kế trao đổi với nhau cũng như dùng để thi công hệ thống (phát triển), thuyết phục khách hàng, các nhà đầu tư.

OOAD cần các bản vẽ để mô tả hệ thống được thiết kế, còn UML là ngôn ngữ mô tả các bản vẽ nên cần nội dung thể hiện.  Do vậy, chúng ta phân tích và thiết kế theo hướng đối tượng và sử dụng UML để biểu diễn các thiết kế đó nên chúng thường đi đôi với nhau.

**Các biểu đồ trong UML**

Các biểu đồ mô tả khía cạnh tĩnh:

* Biểu đồ đối tượng (object diagram)
* Biểu đồ lớp (class diagram)
* Biểu đồ use case (use case diagram)
* Biểu đồ thành phần (component diagram)
* Biểu đồ triển khai (deployment diagram)
* Các sơ đồ mô tả khía cạnh động:
* Các biểu đồ tương tác: biểu đồ tuần tự (sequence diagram), biểu đồ hợp tác (collaboration diagram)
* Biểu đồ hoạt động (activity diagram)
* Biểu đồ chuyển dịch trạng thái (state transition diagram)

*Biểu đồ lớp và đối tượng*: được sử dụng để mô hình hóa cấu trúc tĩnh của hệ thống trong quá trình phát triển. Mỗi biểu đồ chứa các lớp và các mối quan hệ giữa chúng (quan hệ kế thừa (heritage), quan hệ kết hợp (association), quan hệ tập hợp (aggregation), quan hệ thành phần (coposition)). Chúng ta cũng có thể mô tả các hoạt động của lớp.

Biểu đồ đối tượng là một thể hiện của sơ đồ lớp. Nó mô tả trạng thái chi tiết của hệ thống tại một thời điểm cụ thể, do đó, biểu đồ đối tượng được dùng để minh họa một trường hợp thực tế của biểu đồ lớp.

*Biểu đồ use case*: nó mô tả giao diện với một hệ thống từ quan điểm và cách nhìn của người sử dụng. Một biểu đồ use case mô tả các tình huống tiêu biểu của việc sử dụng một hệ thống. Nó biểu thị các trường hợp sử dụng (trong việc mô hình hóa các tính năng hệ thống) và các tác nhân (trong việc mô hình hóa các vai trò tham gia bởi các cá nhân tương tác với hệ thống), và mối quan hệ giữa các use case và các tác nhân.

*Biểu đồ thành phần*: được sử dụng để biểu thị cái nhìn tĩnh trong việc cài đặt một hệ thống. Mỗi sơ đồ bao gồm các thành phần (component) và các mối quan hệ phụ thuộc giữa chúng trong môi trường cài đặt. Một thành phần đại diện cho một yếu tố cài đặt vật lý của môi trường (mã nguồn, mã thực thi, tập tin, cơ sở dữ liệu…)

*Biểu đồ triển khai*: mô tả cách bố trí vật lý các phụ kiện điện tử và sự phân phối các thành phần trú ngụ tại các phụ kiện điện tử này. Một sơ đồ triển khai bao gồm các nút (node) đại diện cho các tài nguyên phụ kiện điện tử và các thành phần được cài đặt trong phụ kiện điện tử, các liên kết trong sơ đồ dùng để mô tả của một hệ thống và kiến trúc về bố trí phụ kiện điện tử của nó.

*Biểu đồ tuần tự và biểu đồ hợp tác*: trình bày các cách nhìn động về tương tác giữa các đối tượng của hệ thống trong quá trình phát triển. biểu đồ hợp mô tả sự hợp tác giữa một nhóm các đối tượng trong hoạt động để đạt một mục tiêu cụ thể. Biểu đồ tuần tự thêm vào chiều thời gian nhằm thể hiện trực quan thứ tự trao đổi của các thông điệp (message).

*Biểu đồ chuyển đổi trạng thái*: mỗi biểu đồ được dùng có liên quan đến một lớp để biểu thị các trạng thái khác nhau của đối tượng của lớp và các biến cố kích hoạt sự dịch chuyển giữa các trạng thái.

*Biểu đồ hoạt động*: dùng để mô hình hóa các dòng hoạt động liên kết tới các lớp như là trong trường hợp của một nhóm các lớp hợp tác cùng thực hiện trong một loại tiến trình. Mỗi lớp sẽ đảm nhiệm các hoạt động và các chuyển dịch như được mô tả trong biểu đồ trạng thái. Một biểu đồ hoạt động có thể liên quan đến nhiều lớp hơn là một lớp. Mặt khác, nó mô tả tiến trình tuần tự các hoạt động, sự đồng bộ hóa các dòng điều khiển song song, các điều kiện và quyết định, điểm bắt đầu và điểm kết thúc tiến trình.

2.3. Giới thiệu các xu hướng lập trình web mới

*Xu hướng 1: Voice search – Tìm kiếm bằng giọng nói*

Tương lai phát triển web dường như sẽ nhiều tiếng nói hơn và nó không chỉ nói về các trợ lý gia đình đa chức năng ảo như Siri, Google Home hay Amazon Alexa.

Vào năm 2020, hơn một nửa trong số tất cả các phụ kiện điện tử thông minh và IoT sẽ nghe người dùng và thực hiện các lệnh được đưa ra bằng giọng nói.

*Xu hướng 2: WebAssembly*

WebAssembly ra đời nhằm mục đích cho hiệu suất giữa các ứng dụng web là như nhau trên mọi trình duyệt và nền tảng.

Với WebAssembly, mã trong bất kỳ ngôn ngữ lập trình nào cũng có thể được biên dịch thành mã byte chạy trong trình duyệt.

*Xu hướng 3: Cá nhân hoá nội dung thông qua học máy*

Trí tuệ nhân tạo, bao gồm cả học máy, ảnh hưởng đến các hoạt động hàng ngày của chúng ta trên internet mà chúng ta thậm chí không nhận ra. Học máy là khả năng của phần mềm để cải thiện hiệu suất mà không cần đầu vào trực tiếp từ các nhà phát triển.

*Xu hướng 4: Data security – Bảo mật dữ liệu*

Các mối đe dọa và hoạt động độc hại dễ dàng phát hiện hơn với phần mềm bảo mật do AI cung cấp. Khi sự đa dạng của phần mềm độc hại phát triển, nó trở nên mạnh hơn và nguy hiểm hơn.

Vì vậy, các công ty lớn hiện đào tạo các hệ thống AI của họ để phân tích các mẫu hành vi trong mạng và phản ứng với mọi hoạt động đáng ngờ ngay lập tức.

*Xu hướng 5: PWA và AMP*

PWA (Progressive Web Apps) hoặc AMP là những công nghệ độc đáo giúp giảm thời gian tải trang web.

*Xu hướng 6: trí tuệ nhân tạo và Bots*

Sự gia tăng của Bot cũng sẽ ảnh hưởng đến xu hướng thiết kế web, đặc biệt là UX. Sự tiến bộ đáng kể của AI và công nghệ máy học dẫn đến nhu cầu thiết kế trợ lý ảo, không chỉ về trải nghiệm hình ảnh, hoặc thuật toán phục vụ sản phẩm, mà còn về thiết kế âm thanh hữu ích.

*Xu hướng 7: Motion UI*

Thiết kế web Motion UI là một xu hướng trong thế giới kỹ thuật số.  Các trang web và ứng dụng đẹp mắt có nhiều cơ hội được người dùng tiềm năng chú ý và trở nên viral nhanh chóng. Trong đó, thiết kế Motion UI không phải là câu hỏi của kỹ thuật, mà là một trong những chiến lược tiếp thị.

*Xu hướng 8: Automation Testing – kiểm tra tự động*

Tự động hóa các giai đoạn phát triển là công cụ để đạt được mục tiêu đầu tiên. Phương pháp học máy và tiếp cận AI cho phép chúng xây dựng các dự án phức tạp với một nhóm nhỏ hoặc có ít chuyên gia phát triển hơn, trong khi tự động hóa thử nghiệm sản phẩm giúp chúng ta kiểm tra xem sản phẩm của mình đã sẵn sàng để triển khai hay chưa.

*Xu hướng 9: JavaScript Frameworks*

Các JavaScript Framework đang trong giai đoạn chuẩn hoá và tạo ra một hệ sinh thuận tiện, dễ dàng sử dụng. Trong năm 2020, các JavaScript Framework sẽ liên quan đến các khía cạnh UX, UI, thử nghiệm và quản lý sản phẩm. Về mặt kỹ thuật, các JavaScript Framework vẫn là Framework ứng dụng được tạo bằng ngôn ngữ lập trình JavaScript và dựa trên tích hợp các thư viện (cả trực quan và chức năng).

*Xu hướng 10: Internet of Things (IoT) – Internet vạn vật*

Internet of Things (hay IoT) là một mạng lưới các phụ kiện điện tử được kết nối với nhau, không yêu cầu hoặc gần như không có hành động nào từ phía người dùng để thực hiện các hoạt động và thực hiện các kết quả hữu ích. IoT rất có tiềm năng tồn tại và phát triển - là khả năng giúp cuộc sống hàng ngày của con người trở nên dễ dàng hoặc thuận tiện hơn.

Với tự động hóa thanh toán, nhà thông minh và công nghệ thành phố thông minh, phát triển môi trường y tế điện tử, chúng ta thường có thể trở nên ít quá tải hơn với các nhiệm vụ thông thường và đầu tư nguồn lực nhận thức của chúng ta vào những công việc phức tạp hơn.

2.4. Tổng quan về Angular

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

Các phiên bản của Angular:

* AngularJS: Phiên bản đầu tiền của angular là AngularJS được bắt đầu từ năm 2009 và đc ra mắt vào 20/10/2010. Lúc đó angular js được viết theo mô hình MVC (Model-View-Controller)
* Angular 2: Tháng 3 năm 2015 phiên bản bản angular 2 ra đời. Angular 2 thay đổi hoàn toàn so với angular js bằng việc thay Controllers và $scope (Angular js) bằng components và directives. Angular 2 hoàn toàn được viết bằng Typescript, nhanh hơn angular js, hỗ trợ đa nền tảng đa trình duyệt, cấu trúc cdoe được tổ chức đơn giản và dễ sử dụng hơn.
* Angular 4: Ra mắt vào tháng 3/2017 đây là một phiên bản nâng cấp từ Angular 2 nên kiến trúc không thay đổi nhiều ngoài việc giảm thiểu code được tạo ra từ đó giảm kích thước tệp được đóng gói xuống 60%, đẩy nhanh quá trình phát triển ứng dụng.
* Angular 5: Đã được phát hành vào ngày 1 tháng 11 năm 2017 với mục tiêu thay đổi về tốc độ và kích thước nên nó nhanh hơn và nhỏ hơn angular 4. Có các tính năng mới so với angular 4.
* Angular 6: Cập nhật CLI, command line interface, thêm 1 số lệnh mới như ng-update, ng-add. Angular Element, cho phép các component của Angular được triển khai dưới dạng component web, sau đó có thể được sử dụng trong bất kỳ trang HTML nào một cách dễ dàng. Multiple Validators, cho phép nhiều Validators được áp dụng trên form builder.
* Angular 7: Được phát hành vào 18 tháng 10 năm 2018 với những thay đổi như: ScrollingModule để scroll load dữ liệu, Drag and Drop có thể dễ dàng thêm tính năng kéo và thả vào một mục, Angular 7.0 đã cập nhật RxJS 6.3.
* Angular 8: Ra mắt 28 tháng 5 năm 2019 với CLI workflow improvements, Dynamic imports for lazy routes ….
* Angular 9, Angular 10.

2.5. Lập trình Web API NET CORE

**Khái niệm Web API**

API là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác. Nó là viết tắt của Application Programming Interface – giao diện lập trình ứng dụng. API cung cấp khả năng cung cấp khả năng truy xuất đến một tập các hàm hay dùng. Và từ đó có thể trao đổi dữ liệu giữa các ứng dụng.

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

**Những điểm nổi bật của Web API**

Web API hỗ trợ restful đầy đủ các phương thức: Get/Post/put/delete dữ liệu. Nó giúp bạn xây dựng các HTTP service một cách rất đơn giản và nhanh chóng. Nó cũng có khả năng hỗ trợ đầy đủ các thành phần HTTP: URI, request/response headers, caching, versioning, content format.

*Tự động hóa sản phẩm*

Với Web API chúng ta sẽ tự động hóa quản lý công việc, cập nhật luồng công việc, giúp tăng năng suất và tạo hiệu quả công việc cao hơn.

*Khả năng tích hợp linh động*

API cho phép lấy nội dung từ bất kỳ website hoặc ứng dụng nào một cách dễ dàng nếu được cho phép, tăng trải nghiệm người dùng. API hoạt động như một chiếc cổng, cho phép các công ty chia sẻ thông tin được chọn nhưng vẫn tránh được những yêu cầu không mong muốn.

*Cập nhật thông tin thời gian thực*

API có chức năng thay đổi và cập nhật thay đổi theo thời gian thực. Với công nghệ này, dữ liệu sẽ được truyền đi tốt hơn, thông tin chính xác hơn, dịch vụ cung cấp linh hoạt hơn.

Có tiêu chuẩn chung dễ sử dụng

Bất kỳ người dùng, công ty nào sử dụng cũng có thể điều chỉnh nội dung, dịch vụ mà họ sử dụng.

Hỗ trợ đầy đủ các thành phần MVC như: routing, controller, action result, filter, model binder, IoC container, unit test.

**Khái niệm ASP.NET**

ASP.NET Core là một open-source mới và framework đa nền tảng (cross-platform) cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây, giống như web apps, IoT và backend cho mobile. Do là một framework đa nền tảng nên .NET hỗ trợ hầu hết các hệ

**Đặc điểm của .NET Core**

Đa nền tảng: Chạy trên các hệ điều hành Windows, macOS và Linux.

Nhất quán trên các kiến ​​trúc: có thể chạy mã nguồn của bạn với cùng một hành vi trên nhiều kiến ​​trúc hệ thống, bao gồm x64, x86 và ARM.

Các công cụ dòng lệnh: Bao gồm các công cụ dòng lệnh dễ sử dụng, có thể được sử dụng để phát triển cục bộ và trong các tình huống tích hợp liên tục.

Triển khai linh hoạt: có thể cài đặt song song (cài đặt toàn người dùng hoặc toàn hệ thống). Có thể được sử dụng với các container Docker

Tương thích: .NET Core tương thích với .NET Framework, Xamarin và Mono, thông qua .NET Standard.

Mã nguồn mở: Nền tảng .NET Core là nguồn mở, sử dụng giấy phép MIT và Apache 2. .NET Core là một dự án .NET Foundation.

Được hỗ trợ bởi Microsoft: .NET Core được Microsoft hỗ trợ, theo Hỗ trợ .NET Core.iều hành như Windows, Linux, macOS.

ASP.NET Web Api là một framework giúp cho việc xây dựng HTTP service một cách dễ dàng. Chúng có thể phát triển cho nhiều clients khác nhau như trình duyệt, mobile app. Web api là một nền tảng để phát triển các ứng dụng dựa trên Restfull service trong .Net.

# PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

* 1. Đặc tả yêu cầu phần mềm

Để đáp ứng yêu cầu người dùng, hệ thống website bán phụ kiện điện tử online bao gồm 2 phân hệ chính: phân hệ cho người quản trị để quản trị nội dung và quản trị hoạt động kinh doanh online; phân hệ cho người dùng để cho khách hàng có thể xem thông tin về các mặt hàng kinh doanh và đặt hàng online. Các yêu cầu chi tiết của hai phân hệ như sau:

*Yêu cầu trang quản trị:*

Cho phép quản trị viên sẽ quản lý thông tin về loại sản phẩm, sản phẩm. Khi các thông tin về phụ kiện, loại phụ kiện có sai xót hệ thống cho phép quản trị viên có thể sửa hoặc xóa thông tin của sản phẩm, loại sản phẩm.

Cho phép quản trị viên quản lý các đơn hàng chưa xác thực, xác thực đơn hàng, đơn hàng đã xác thực.

Cho phép quản trị viên quản lý các đơn hàng đã giao: ghi nhận thanh toán, ghi nhận giao hàng không thành công.

*Yêu cầu trang người dùng:*

Khi khách hàng truy cập vào website giới thiệu và đặt hàng phụ kiện, website sẽ hiển thị các sản phẩm và các loại sản phẩm giúp người dùng xem tất cả sản phẩm một cách dễ dàng nhất. Thông tin về sản phẩm bao gồm: Mã sản phẩm, tên sản phẩm, loại sản phẩm, giá sản phẩm, hình ảnh.

Khi người dùng xem thông tin sản phẩm hoặc xem thông tin chi tiết của phụ kiện người dùng chọn mua sản phẩm , thông tin sản phẩm chọn mua sẽ được cập nhật vào giỏ hàng.

Khi người dùng muốn mua hàng, người dùng có thể chọn chức năng quản lý giỏ hàng. Người dùng có thể sửa, xóa thông tin của giỏ hàng. Nếu người dùng muốn mua thêm hàng sẽ vào trang xem thông tin của sản phẩm để chọn mua tiếp. Khi người dùng đặt hàng thì sẽ yêu cầu chức năng đặt hàng từ chức năng quản lý giỏ hàng.

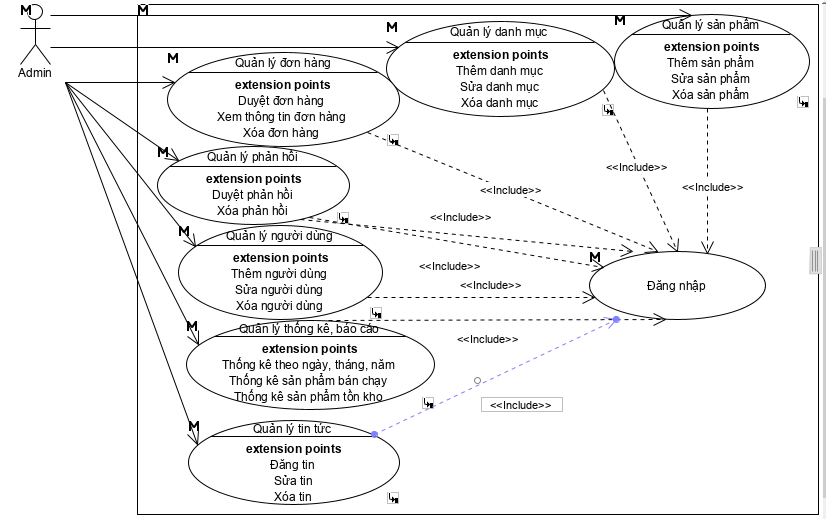
Khi người dùng xem thông tin của sản phẩm có thể thực hiện chức năng tìm kiếm phụ kiện theo tên, theo loại sản phẩm.

* + 1. Các yêu cầu chức năng
       1. Chức năng của phân hệ quản trị nội dung

Bảng . Chức năng của phân hệ quản trị nội dung

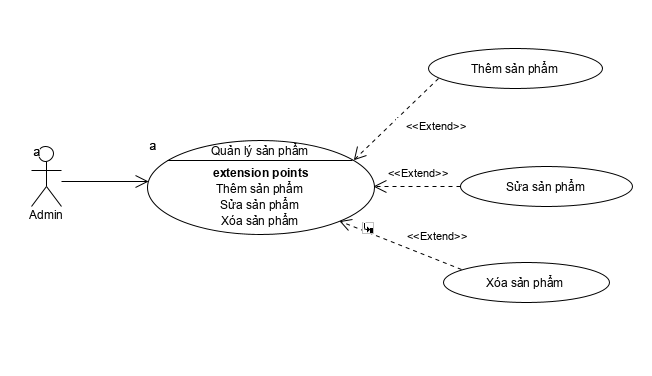
|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả** |
| 1 | Quản lý danh mục sản phẩm | Quản lý thông tin danh mục sản phẩm gồm các chức năng thêm, sửa, xóa danh mục sản phẩm. |
| 2 | Quản lý mặt hàng | Quản lý thông tin sản phẩm, người quản trị có thể upload hình ảnh sản phẩm, nhập thông tin mô tả tính năng sản phẩm, giá cả, danh mục, số lượng… |
| 3 | Quản lý đơn đặt hàng | Quản lý đơn đặt hàng bao gồm: tên khách hàng, địa chỉ giao, tổng đơn, trạng thái đơn hàng. |
| 4 | Quản lý chi tiết đơn đặt hàng | Hiển thị thông tin chi tiết đơn hàng bao gồm sản phẩm, giá, số lượng, mã đơn hàng, ghi chú của người quản trị. Hệ thống cho phép lọc những đơn hàng chưa duyệt, đã duyệt, xác nhận giao dịch, xác nhận không giao dịch. |
| 5 | Quản lý người dùng | Thêm mới, chỉnh sửa, xóa thông tin người dùng. |
| 6 | Thống kê báo cáo doanh thu | Chức năng này cho phép người quản trị xem các báo cáo doanh thu (theo ngày, theo tháng, theo quý), sản phẩm bán chạy, sản phẩm tồn kho |
| 7 | Quản lý phản hồi | Lưu trữ các thông tin phản hồi của khách hàng |
| 8 | Quản lý tin tức | Quản lý bao gồm tên tin tức, nội dung, ảnh, ngày đăng, và người đăng. |

* Biểu đồ use case tổng quát phân hệ trang quản trị



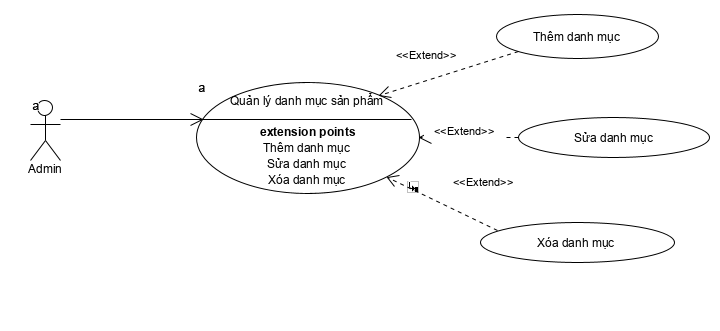
Hình: . Biểu đồ UseCase tổng quát trang quản trị

* Biểu đồ use case phân rã của trang người dùng
* Usecase quản lý sản phẩm



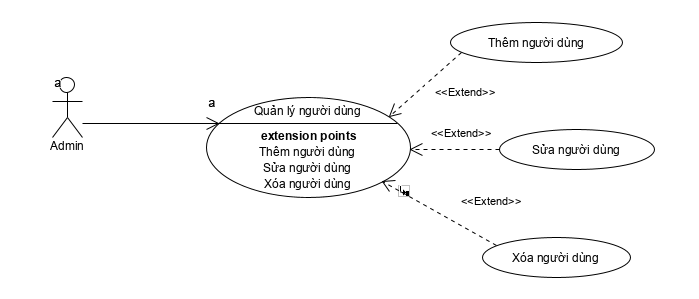
Hình: . Use Case quản lý sản phẩm

* Usecase quản lý danh mục sản phẩm



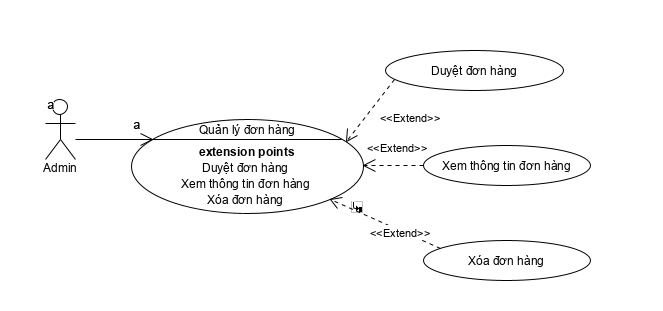
Hình: . UserCase quản lý danh mục sản phẩm

* Usecase quản lý người dùng



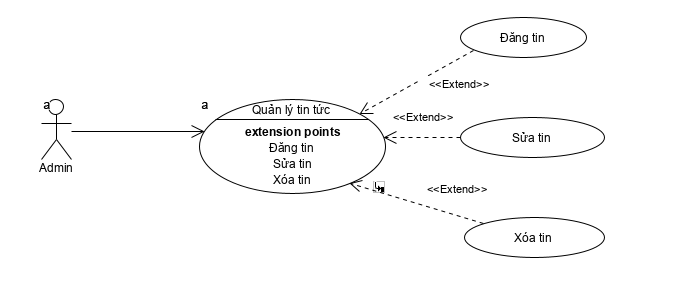
Hình: . Use Case quản lý người dùng

* Usecase quản lý đơn hàng



Hình: . Use Case quản lý đơn hàng

* Usecase quản lý tin tức



Hình: . Use Case quản lý tin tức

* + - 1. Chức năng của phân hệ người dùng

Bảng . Chức năng của phân hệ người dùng

|  |  |  |
| --- | --- | --- |
| STT | Chức năng | Mô tả |
| 1 | Xem hàng | Người xem có thể xem thông tin về sản phẩm bao gồm: tên sản phẩm, giá sản phẩm, giới thiệu và hình ảnh minh họa. |
| 2 | Bình luận | Người xem có thể bình luận về sản phẩm, nội dung và thông tin người bình luận cần lưu trong database. |
| 3 | Đặt hàng | Chức năng cho phép khách hàng duyệt sản phẩm trên trang web và nhấn nút đặt hàng sản phẩm đã chọn. |
| 4 | Quản lý giỏ hàng | Cho phép khách hàng xem và điều chỉnh các mặt hàng đã chọn mua. Bao gồm số các sản phẩm và tổng giá tiền các sản phẩm đã chọn mua. |
| 5 | Tìm kiếm sản phẩm | Khách hàng nhập thông tin sản phẩm vào thanh tìm kiếm để tìm kiếm sản phẩm. |
| 6 | Xem quảng cáo | Trên website có banner quảng cáo, khi click vào hình ảnh đó thì dẫn đến trang nguồn. |
| 7 | Xem tin tức liên quan | Trên website có các hình ảnh và thông tin liên quan, khi click vào thì dẫn đến trang chi tiết tin tức. |
| 8 | Xem sản phẩm mới nhập | Hiển thị thông tin sản phẩm mới nhập cho người xem. |
| 9 | Xem sản phẩm bán chạy | Hiển thị thông tin các sản phẩm bán chạy nhất cho người xem. |
| 10 | Theo dõi đơn hàng | Khi khách hàng đăng nhập Website thì có thể theo dõi được các đơn hàng đã đặt mua trên Website. |

* **Mô tả kịch bản của một số ca sử dụng**

*Mô tả luồng sự kiện của ca sử dụng thêm thiết bị điện tử*

|  |
| --- |
| **Use case name:** Thêm sản phẩm  **Brief description:** Admin thực hiện chức năng thêm sản phẩm  **Actors:** Admin  **Precondition:** Admin đăng nhập vào hệ thống thành công  **Post – condition:**  Nếu thêm sản phẩm thành công, một sản phẩm sản phẩm sẽ được lưu và hiển thị. Trong trường hợp khác, hệ thống yêu cầu hiển thị lỗi  **Trigger:** Admin yêu cầu chức năng thêm sản phẩm |
| **Basic flow** |
| Người quản trị chọn chức năng thêm sản phẩm  Người quản trị chọn loại sản phẩm cần thêm  2. SYSTEM Hệ thống hiển thị form nhập thông tin sản phẩm  3. Người quản lý nhập thông tin sản phẩm  4. Người quản lý thực hiện chức năng lưu thông tin, nếu không lưu thì chuyển sang luồng phụ 4a  5. SYSTEM Hệ thống kiểm tra thông tin sản phẩm  6. SYSTEM Hệ thống lưu thông tin về sản phẩm vào hệ thống |
| **Alternate flows** |
| 4a. Người dùng không lưu sản phẩm  1. Người quản trị không muốn thêm sản phẩm mới, chọn nút hủy  2. SYSTEM Thoát khỏi form nhập thêm thông tin sản phẩm  5a. Thông tin sản phẩm không hợp lệ  SYSTEM Hệ thống thông báo thông tin sản phẩm cần thêm không hợp lệ, yêu cầu người quản trị nhập lại thông tin quay về bước 5. |

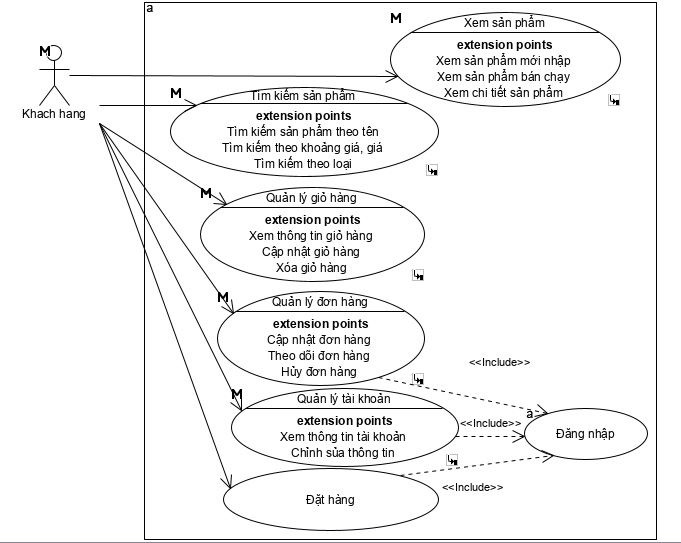
*Mô tả luồng sự kiện của ca sử dụng sửa sản phẩm*

|  |
| --- |
| **Use case name:** Sửa sản phẩm  **Brief description:** Admin thực hiện chức năng sửa sản phẩm  **Actors:** Admin  **Precondition:** Admin đăng nhập vào hệ thống thành công  **Post – condition:**  Nếu sửa sản phẩm thành công, một sản phẩm sản phẩm sẽ được cập nhật và hiển thị. Trong trường hợp khác, hệ thống yêu cầu hiển thị lỗi  **Trigger:** Admin yêu cầu chức năng sửa sản phẩm |
| **Basic flow** |
| 1. Người quản trị chọn chức năng sửa sản phẩm  2. SYSTEM Hệ thống hiển thị form sửa thông tin sản phẩm  3. Người quản lý nhập thông tin sản phẩm cần sửa  4. Người quản lý thực hiện chức năng lưu thông tin, nếu không lưu thì chuyển sang luồng phụ 4a  5. SYSTEM Hệ thống kiểm tra thông tin sản phẩm  6. SYSTEM Hệ thống lưu thông tin về sản phẩm vừa cập nhật vào hệ thống |
| **Alternate flows** |
| 4a. Người dùng không lưu sản phẩm  1. Người quản trị không muốn sửa sản phẩm đó, chọn nút hủy  2. SYSTEM Thoát khỏi form sửa thông tin sản phẩm  5a. Thông tin sản phẩm không hợp lệ  SYSTEM Hệ thống thông báo thông tin sản phẩm cần sửa không hợp lệ, yêu cầu người quản trị nhập lại thông tin quay về bước 5. |

*Mô tả luồng sự kiện của ca sử dụng xóa sản phẩm*

|  |
| --- |
| **Use case name:** Xóa sản phẩm  **Brief description:** Admin thực hiện chức năng xóa sản phẩm  **Actors:** Admin  **Precondition:** Admin đăng nhập vào hệ thống thành công  **Post – condition:**  Nếu xóa sản phẩm thành công, một sản phẩm sản phẩm sẽ được xóa. Trong trường hợp khác, hệ thống yêu cầu hiển thị lỗi  **Trigger:** Admin yêu cầu chức năng xóa sản phẩm |
| **Basic flow** |
| 1. Người quản trị chọn chức năng xóa sản phẩm  2. Người quản lý yêu cầu xóa sản phẩm  3. SYSTEM Hệ thống hiển thị thông báo xác nhận xóa bỏ, xóa sản phẩm đó khỏi hệ thống |
| **Alternate flows** |
| 4a. Người dùng không xóa sản phẩm  1. Người quản trị không muốn sửa sản phẩm đó, chọn nút hủy  2. SYSTEM Thoát khỏi form xóa thông tin sản phẩm |

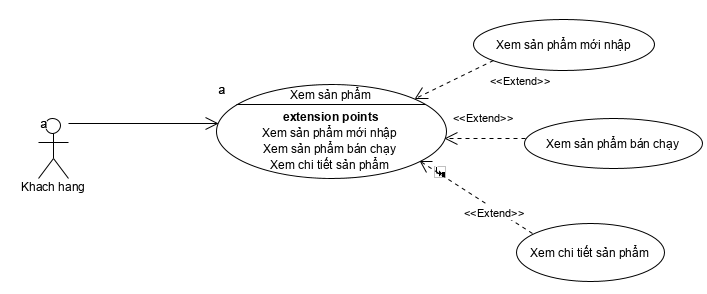
* Biểu đồ use case tổng quát phân hệ trang người dùng



Hình: . Use Case tổng quát phân hệ trang người dùng

Hình 3- . Biểu đồ use case tổng quát phân hệ trang người dùng

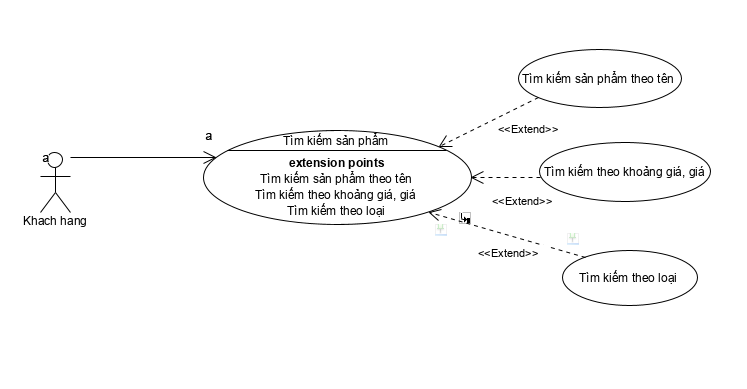
* Biểu đồ use case phân rã của trang người dùng
* Use case xem sản phẩm



Hình: . Use Case xem sản phẩm

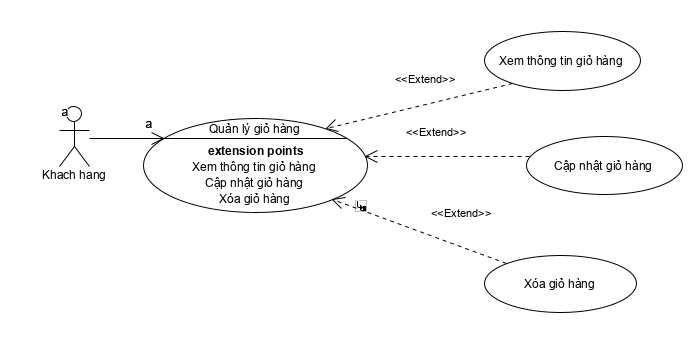
Hình 3- . Use case xem sản phẩm

* Use case tìm kiếm sản phẩm



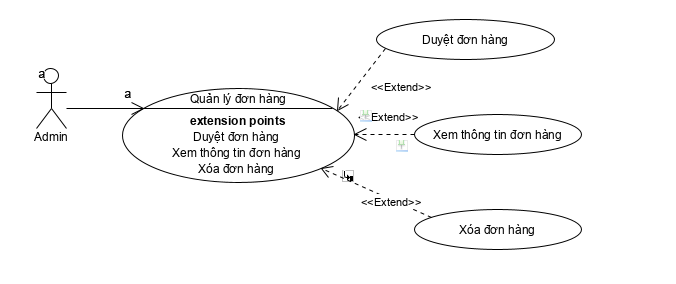
Hình: . Use Case tìm kiếm sản phẩm

* Use case giỏ hàng



Hình: . Use Case quản lý giỏ hàng

* Usecase quản lý đơn hàng



Hình: . Use Case quản lý đơn hàng

* Mô tả luồng sự kiện cho cho từng ca sử dụng

Use case xem sản phẩm

*Mô tả luồng sự kiện của ca sử dụng xem sản phẩm theo loại*

|  |
| --- |
| **Use case name:** Xem sản phẩm theo loại  **Mô tả:** Khách hàng chọn chức năng xem sản phẩm theo loại  **Actors:** Khách hàng  **Điều kiện tiền quyết:** Không có yêu cầu đặc biệt nào  **Hậu điều kiện:**  Hệ thống hiển thị danh sách các sản phẩm theo loại cần xem  **Trigger:** Khách hàng yêu cầu xem sản phẩm theo loại |
| **Basic flow** |
| 1. Người dùng yêu cầu xem sản phẩm theo loại từ giao diện của trang Web  2. SYSTEM Hệ thống lấy về sản phẩm theo loại hiển thị lên trang danh sách các sản phẩm theo loại. |
| **Extension** |
|  |

*Mô tả luồng sự kiện của ca sử dụng xem chi tiết sản phẩm*

|  |
| --- |
| **Use case name:** Xem chi tiết sản phẩm  **Mô tả:** Khách hàng chọn chức năng xem chi tiết sản phẩm  **Actors:** Khách hàng  **Điều kiện tiền quyết:** Không có yêu cầu đặc biệt nào  **Hậu điều kiện:**  Hệ thống hiển thị thông tin chi tiết của sản phẩm cần xem  **Trigger:** Khách hàng yêu cầu xem thông tin chi tiết sản phẩm |
| **Basic flow** |
| 1. Khách hàng yêu cầu xem thông tin chi tiết sản phẩm từ giao diện của trang Web  2. SYSTEM Hệ thống lấy về thông tin chi tiết sản phẩm hiển thị lên trang chi tiết sản phẩm |
| **Extension** |
|  |

Use case tìm kiếm sản phẩm

*Mô tả luồng sự kiện của ca sử dụng tìm kiếm theo tên/giá*

|  |
| --- |
| **Use case name:** Tìm kiếm  **Mô tả:** Khách hàng yêu cầu tìm kiếm theo tên/giá  **Actors:** Khách hàng  **Điều kiện tiền quyết:**  **Hậu điều kiện:** Hệ thống hiển thị danh sách sản phẩm theo tên/giá mà khách hàng yêu cầu |
| **Basic flow** |
| 1. Người dùng yêu cầu tìm kiếm từ giao diện của trang Web.  2. SYSTEM Hệ thống lấy về danh sách sản phẩm có tên/giá mà khách hàng yêu cầu, hiển thị lên giao diện của trang web |
| **Extension** |
|  |

Use case giỏ hàng

*Mô tả luồng sự kiện của ca sử dụng xem giỏ hàng*

|  |
| --- |
| **Use case:** Xem giỏ hàng  **Mô tả:** Khách hàng thực hiện chức năng xem giỏ hàng  **Actors:** Khách hàng  **Điều kiện tiền quyết:** Không có yêu cầu đặc biệt nào  **Hậu điều kiện:**  Hiển thị danh sách sản phẩm có trong giỏ hàng |
| **Basic flow** |
| 1. Khách hàng chọn chức năng xem giỏ hàng  2. SYSTEM Hệ thống hiển thị danh sách các sản phẩm mà khách hàng đã thêm vào giỏ |
| **Extension** |
|  |

*Mô tả luồng sự kiện của ca sử dụng cập nhật giỏ hàng*

|  |
| --- |
| **Use case:** Cập nhật giỏ hàng  **Mô tả:** Khách hàng thực hiện chức năng cập nhật giỏ hàng  **Actors:** Khách hàng  **Điều kiện tiền quyết:** Không có yêu cầu đặc biệt nào  **Hậu điều kiện:**  Nếu cập nhật giỏ hàng thành công, thông tin mới của giỏ hàng sẽ được cập nhật và hiển thị. Trong trường hợp khác, hệ thống yêu cầu hiển thị lỗi |
| **Basic flow** |
| 1. Khách hàng chọn chức năng cập nhật giỏ hàng  2. SYSTEM Hệ thống hiển thị giao diện sửa giỏ hàng  3. Khách hàng cập nhật lại thông tin giỏ hàng cần sửa  4. Khách hàng thực hiện chức năng lưu thông tin, nếu không lưu thì chuyển sang luồng phụ 4a  5. SYSTEM Hệ thống kiểm tra thông tin giỏ hàng, thông tin giỏ hàng không hợp lệ thì chuyển sang luồng phụ 5a  6. SYSTEM Hệ thống lưu thông tin giỏ hàng vừa cập nhật vào hệ thống |
| **Extension** |
| 4a. Người dùng không lưu thông tin cập nhật của giỏ hàng  1. Khách hàng không muốn sửa nhà cung cấp đó, chọn nút hủy  2. SYSTEM Thoát khỏi giao diện sửa giỏ hàng  5a. Thông tin giỏ hàng không hợp lệ  1. SYSTEM Hệ thống thông báo thông tin giỏ hàng cần sửa không hợp lệ, yêu cầu khách hàng nhập lại thông tin quay về bước 5. |

Use case đặt hàng

Mô tả luồng sự kiện của ca sử dụng đặt hàng

|  |
| --- |
| **Use case name:** Đặt hàng  **Brief description:** Người dùng yêu cầu đặt hàng  **Actors:** Khách hàng  **Điều kiện tiền quyết:** Khách hàng đăng nhập thành công vào hệ thống  **Hậu điều kiện:** Nếu đặt hàng thành công, một đơn hàng sẽ được lưu vào hệ thống. Trong trường hợp khác, hệ thống yêu cầu hiển thị lỗi  **Trigger:** |
| **Basic flow** |
| 1. Người dùng yêu cầu đặt hàng  2. SYSTEM Hệ thống hiển thị giao diện đặt hàng để khách hàng nhập thông tin  3. Khách hàng đặt hàng, nếu không thì chuyển sang luồng phụ 3a  4. SYSTEM Hệ thống kiểm tra thông tin đặt hàng, thông tin không hợp lệ thì chuyển sang luồng phụ 4a  5. SYSTEM Hệ thống lưu thông tin khách hàng  6. SYSTEM Hệ thống sinh mã đơn hàng tự động  7. SYSTEM Hệ thống lưu thông tin đơn hàng và danh sách các mặt hàng đã đặt vào chi tiết đơn hàng  8. SYSTEM Hệ thống hiển thị đặt hàng thành công. |
| **Extension** |
| 3a. Người dùng không đặt hàng  1. Người dùng muốn ngưng giao dịch, chọn nút hủy  2. SYSTEM Hệ thống hiển thị giao diện giỏ hàng  4a. Thông tin đặt hàng không hợp lệ  1. SYSTEM Hệ thống hiển thị thông báo việc tạo đơn hàng không thành công  2. SYSTEM Hệ thống hiển thị form nhập thông tin và các trường nhập sai, quay lại bước 3  3. Khách hàng nhập lại thông tin |

* + 1. Biểu đồ lớp thực thể
* Bảng danh mục sản phẩm

Bảng . Bảng danh mục sản phẩm

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | Category\_id | Mỗi danh mục sẽ có mã duy nhất và khác nhau |
| 2 | Category\_name | Tên của danh mục |
| 3 | url | Đường dẫn |
| 4 | Parent\_id | Danh mục cha |
| 5 | Created\_at | Ngày tạo |

* Bảng sản phẩm

Bảng . Bảng sản phẩm

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | product\_id | Mỗi sản phẩm sẽ có một mã duy |
| 2 | product\_name | Tên của sản phẩm |
| 3 | image | Hình ảnh |
| 4 | description | Mô tả sản phẩm |
| 5 | detail | Chi tiết sản phẩm |
| 6 | quantity | Số lượng của sản phẩm |
| 7 | price | Giá sản phẩm |
| 8 | Promotion\_price | Giá khuyến mãi |
| 9 | status | Trạng thái |
| 10 | Created\_at | Ngày tạo |

* Bảng khách hàng

Bảng . Bảng khách hàng

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | customer\_id | Mỗi khách hàng sẽ có một mã duy nhất |
| 2 | customer\_name | Tên của khách hàng |
| 3 | address | Địa chỉ |
| 4 | email | email |
| 5 | password | Mật khẩu |
| 6 | phone | Điện thoại |
| 7 | Created\_at | Ngày tạo |

* Bảng hóa đơn

Bảng . Bảng hóa đơn

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | Order\_id | Mỗi hóa đơn sẽ có một mã duy nhất |
| 2 | Customer\_id | Mã khách hàng |
| 3 | Name | Tên người nhận |
| 4 | Address | Địa chỉ |
| 5 | Created\_at | Ngày tạo |
| 6 | Phone | Điện thoại |
| 7 | Total | Tổng tiền |
| 8 | Status | Trạng thái |

* Bảng chi tiết hóa đơn

Bảng . Bảng chi tiết hóa đơn

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | id | Mỗi chi tiết hóa đơn sẽ có một mã duy nhất |
| 2 | Order\_id | Mã hóa đơn |
| 3 | Product\_id | Mã sản phẩm |
| 4 | quantity | Số lượng |
| 5 | price | Giá sản phẩm |

* Bảng người dùng

Bảng . Bảng người dùng

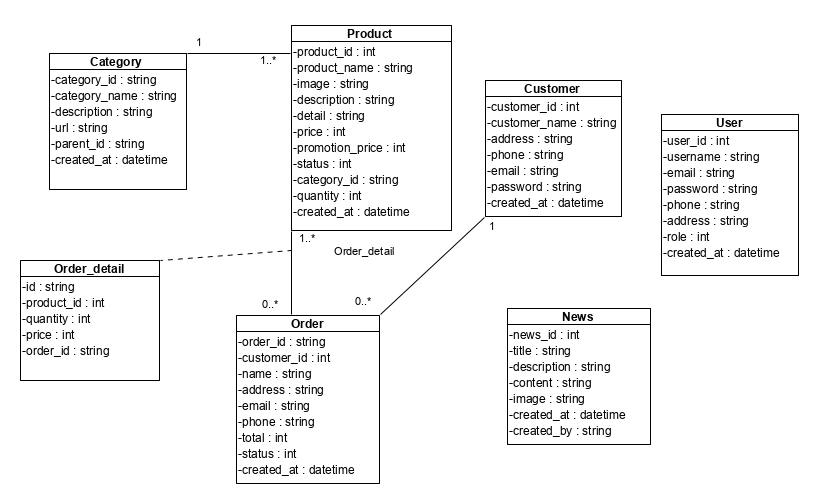
|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | user\_id | Mỗi người dùng sẽ có một mã duy nhất |
| 2 | name | Tên của người dùng |
| 3 | address | Địa chỉ |
| 4 | email | email |
| 5 | password | Mật khẩu |
| 6 | phone | Điện thoại |
| 7 | Created\_at | Ngày tạo |
| 8 | Role | Quyền |

* Bảng tin tức

Bảng . Bảng tin tức

|  |  |  |
| --- | --- | --- |
| STT | Tên thuộc tính | Mô tả |
| 1 | new\_id | Mỗi tin tức sẽ có một mã duy nhất và khác nhau |
| 2 | title | Tiêu đề tin |
| 3 | description | Mô tả |
| 4 | content | Nội dung |
| 5 | image | Hình ảnh |
| 6 | Created\_by | Người tạo |
| 7 | Created\_at | Ngày tạo |

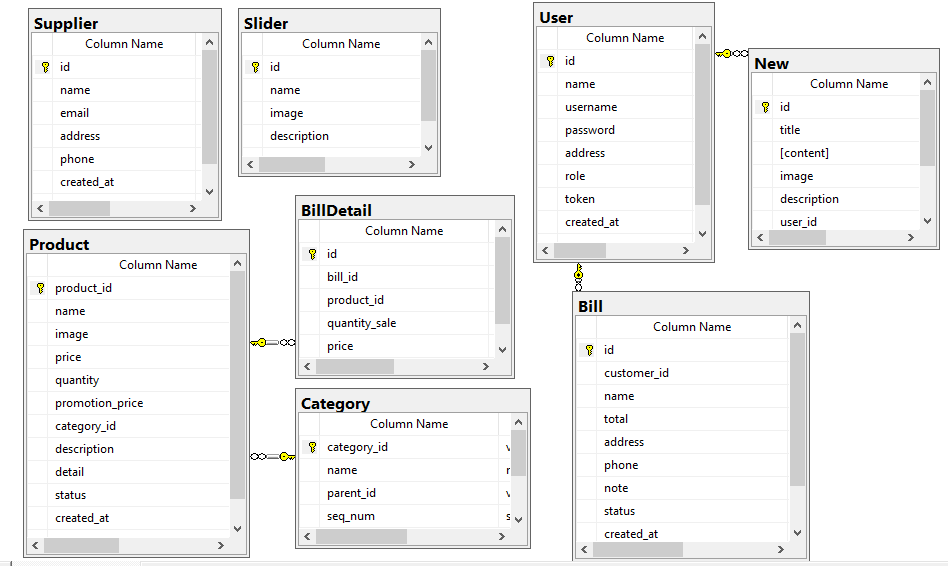
* Biểu đồ lớp thực thể của hệ thống



Hình: . Biểu đồ lớp thực thể hệ thống

* + 1. Các yêu cầu phi chức năng
* Yêu cầu về giao diện: giao diện đẹp, thân thiện và dễ thao tác
* Yêu cầu về sự logic của cơ sở dữ liệu
* Yêu cầu ràng buộc về thiết kế: hệ thống được thiết kế theo kiến trúc MVC
* Độ tin cậy cao
* Tính khả dụng
* Bảo mật: Thông tin người dùng phải được bảo mật và được mã hoá bằng MD5 Hash
* Bảo trì: Hệ thống có thể dễ dàng bảo trì được khi yêu cầu người dùng thay đổi.
* Tính khả chuyển: Hệ thống chạy được trên các môi trường khác nhau như mobile và destop.
  1. Thiết kế hệ thống
     1. Thiết kế cơ sở dữ liệu

*Mô hình cơ sở dữ liệu quan hệ*



Hình: .: Mô hình cơ sở dữ liệu quan hệ

*Mô tả cấu trúc bảng*

Bảng danh mục sản phẩm

Bảng . Cấu trúc bảng danh mục sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | Category\_id | Varchar (50) | Primary key |
| 2 | Category\_name | Nvarchar (50) | Not null |
| 3 | url | Nvarchar (250) | Null |
| 4 | Parent\_id | Varchar (50) | Null |
| 5 | Created\_at | date | Not null |

Bảng sản phẩm

Bảng . Cấu trúc bảng sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiễu dữ liệu | Mô tả |
| 1 | product\_id | int | Primary key |
| 2 | product\_name | Nvarchar (150) | Not null |
| 3 | image | Varchar (250) | Not null |
| 4 | description | text | Not null |
| 5 | detail | text | Null |
| 6 | quantity | int | Not null |
| 7 | price | int | Not null |
| 8 | Promotion\_price | int | Null |
| 9 | status | int | Not null |
| 10 | category\_id | Varchar (50) | Foreign key |
| 11 | created\_at | date | Not null |

Bảng khách hàng

Bảng . Cấu trúc bảng khách hàng

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | customer\_id | int | Primary key |
| 2 | customer\_name | Nvarchar (50) | Not null |
| 3 | address | Nvarchar (50) | Null |
| 4 | email | Varchar (150) | Not null |
| 5 | password | Varchar (150) | Not null |
| 6 | phone | Varchar (20) | Null |
| 7 | Created\_at | Date | Null |

Bảng hóa đơn

Bảng . Cấu trúc bảng hóa đơn

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | Order\_id | Varchar (50) | Primary key |
| 2 | Customer\_id | int | Foreign key |
| 3 | Name | Nvarchar (50) | Not null |
| 4 | Address | Nvarchar (250) | Not null |
| 5 | Created\_at | Datetime | Not null |
| 6 | Phone | Varchar (20) | Not null |
| 7 | Total | int | Not null |
| 8 | Status | int | Not null |

Bảng chi tiết hóa đơn

Bảng . Cấu trúc bảng chi tiết hóa đơn

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | id | Varchar (50) | Primary key |
| 2 | Order\_id | Varchar (50) | Foreign key |
| 3 | Product\_id | int | Foreign key |
| 4 | quantity | int | Not null |
| 5 | price | int | Not null |

Bảng người dùng

Bảng . Cấu trúc bảng người dùng

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | user\_id | int | Primay key |
| 2 | username | Nvarchar (50) | Not null |
| 3 | address | Nvarchar (50) | Not null |
| 4 | email | Nvarchar (150) | Not null |
| 5 | password | Nvarchar (150) | Not null |
| 6 | phone | Varchar (20) | Null |
| 7 | Role | Varchar (150) | Not null |
| 8 | Created\_at | date | Not null |

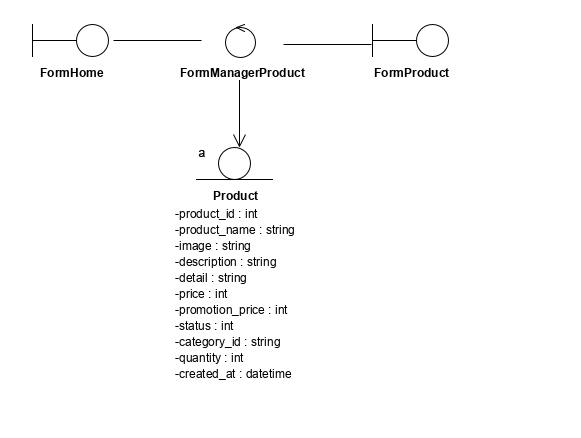
Bảng tin tức

Bảng . Cấu trúc bảng tin tức

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên trường | Kiểu dữ liệu | Mô tả |
| 1 | new\_id | int | Primary key |
| 2 | title | Nvarchar (150) | Not null |
| 3 | description | Nvarchar (250) | Not null |
| 4 | content | text | Not null |
| 5 | image | Varchar (250) | Not null |
| 6 | created\_by | Nvarchar (50) | Not null |
| 7 | created\_at | date | Not null |

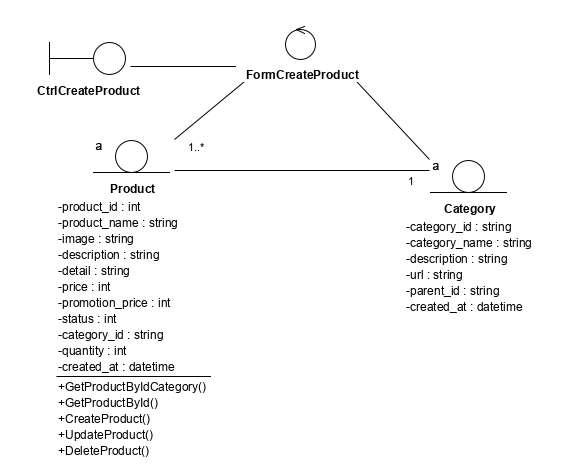
* + 1. Thiết kế lớp đối tượng
       1. Biểu đồ lớp VOPC của các ca sử dụng

Biều đồ lớp VOPC của UseCase quản lý sản phẩm



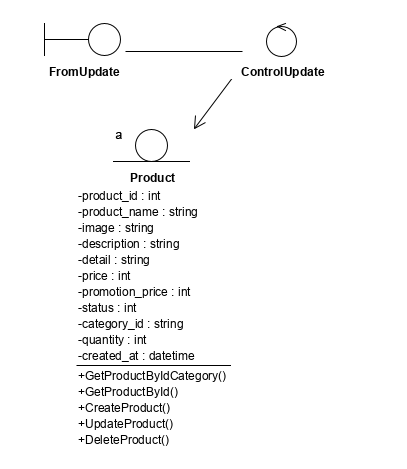
Hình: . Biểu đồ lớp VOPC của Use Case quản lý sản phẩm

Biều đồ lớp VOPC của UseCase thêm sản phẩm



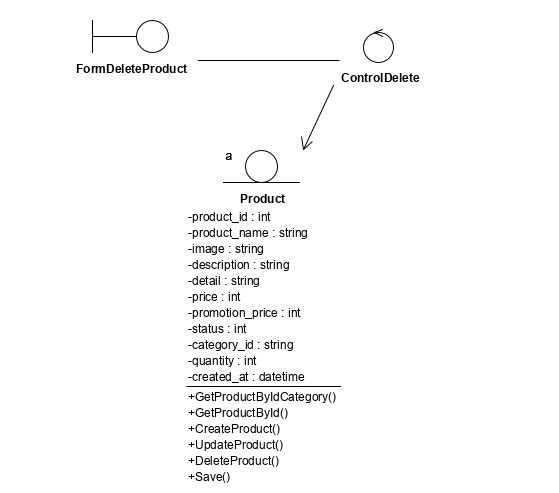
Hình: . Biều đồ lớp VOPC của UseCase thêm sản phẩm

Biều đồ lớp VOPC của UseCase sửa sản phẩm



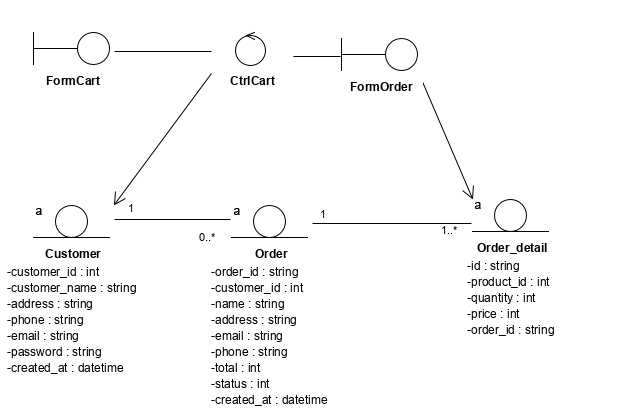
Hình: . Biều đồ lớp VOPC của UseCase sửa sản phẩm

Biều đồ lớp VOPC của UseCase xóa sản phẩm



Hình: . Biều đồ lớp VOPC của UseCase xóa sản phẩm

Biều đồ lớp VOPC của UseCase đặt hàng

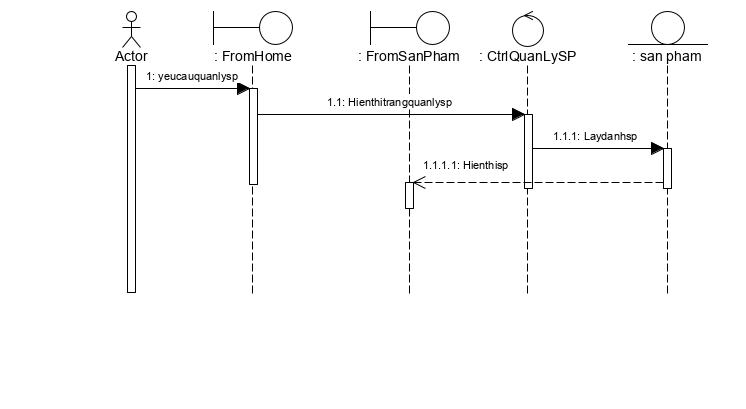


Hình: . Biều đồ lớp VOPC của UseCase đặt hàng

* + - 1. Biểu đồ tuần tự

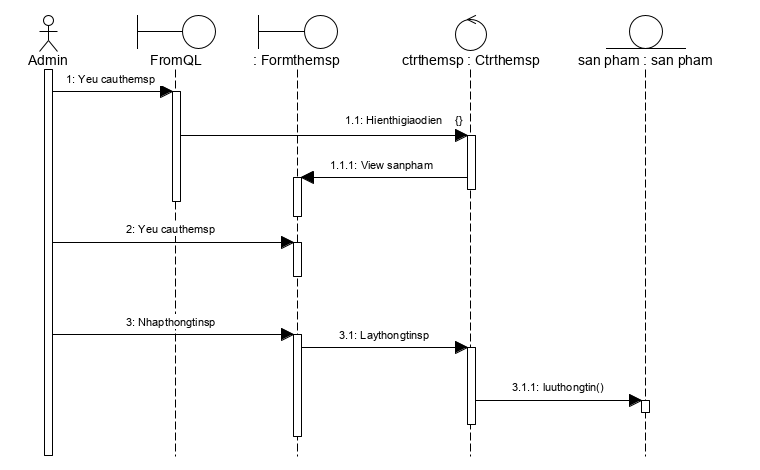
*Biểu đồ tuần tự trang quản trị viên*

Biểu đồ tuần tự của UseCase quản lý thiết sản phẩm



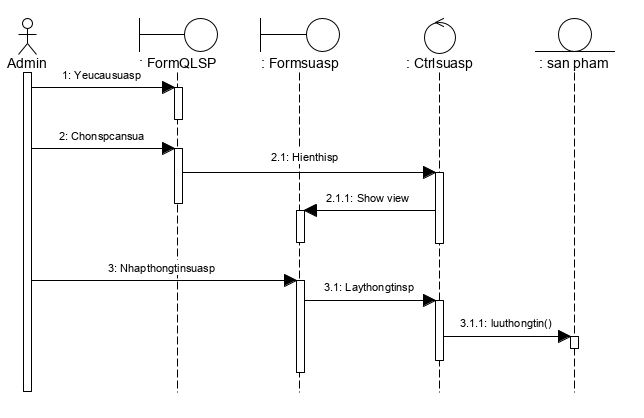
Hình: . Biểu đồ tuần tự của UseCase quản lý thiết sản phẩm

Biểu đồ tuần tự của Use case thêm sản phẩm



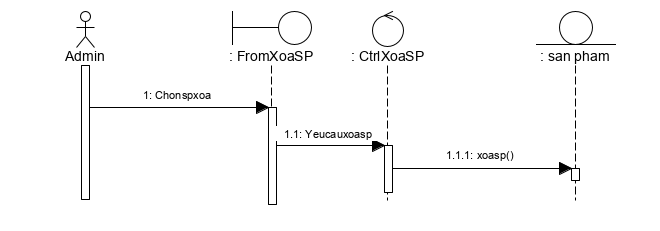
Hình: . Biểu đồ tuần tự của Use case thêm sản phẩm

Biểu đồ tuần tự của Use case sửa sản phẩm



Hình: . Biểu đồ tuần tự của Use case sửa sản phẩm

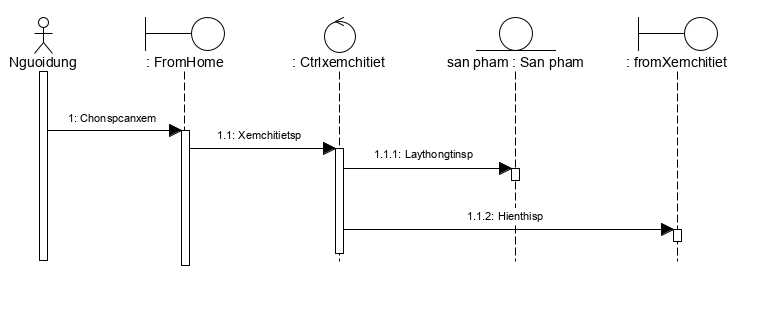
Biểu đồ tuần tự của Use case xóa thiết sản phẩm



Hình: 3.22 Biểu đồ tuần tự của Use case xóa thiết sản phẩm

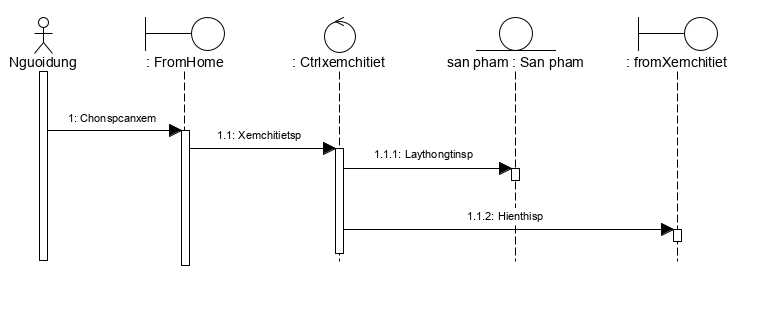
*Biểu đồ tuần tự trang người dùng*

Biểu đồ tuần tự của Use case xem sản phẩm theo loại



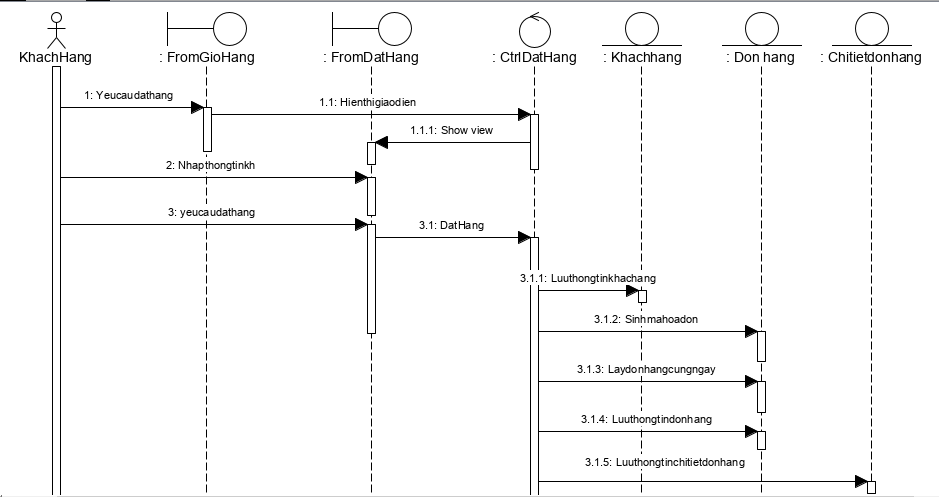
Hình: . Biểu đồ tuần tự của Use case xem sản phẩm theo loại

Biểu đồ tuần tự của Use case xem chi tiết sản phẩm



Hình: . Biểu đồ tuần tự của Use case xem chi tiết sản phẩm

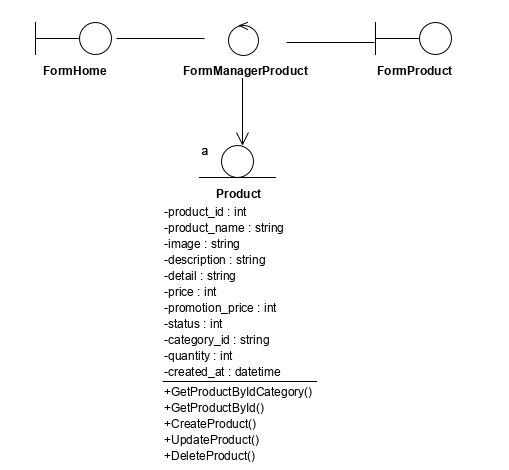
Biểu đồ tuần tự của Use case đặt hàng



Hình: . Biểu đồ tuần tự của Use case đặt hàng

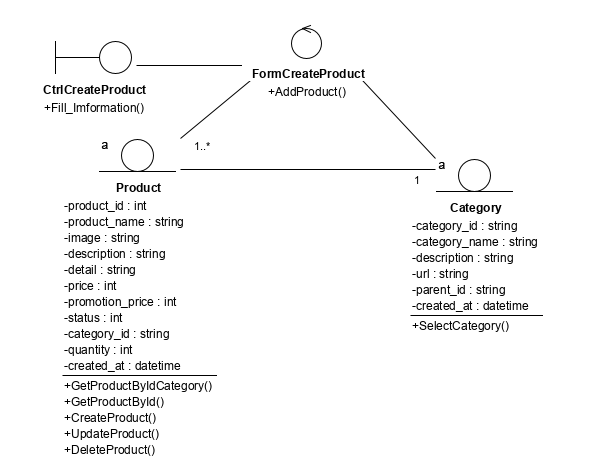
* + - 1. Biểu đồ lớp chi tiết

Biểu đồ lớp chi tiết VOPC quản lý sản phẩm



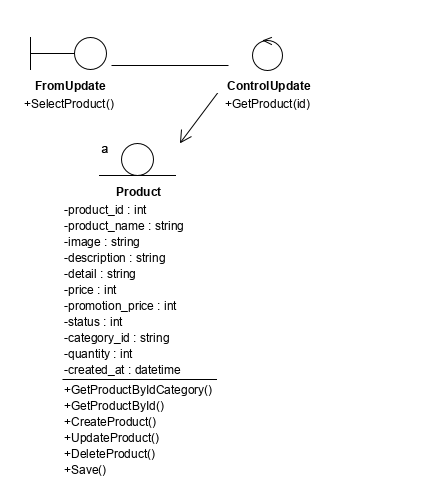
Hình: . Biểu đồ lớp chi tiết VOPC quản lý sản phẩm

Biểu đồ lớp chi tiết VOPC thêm sản phẩm



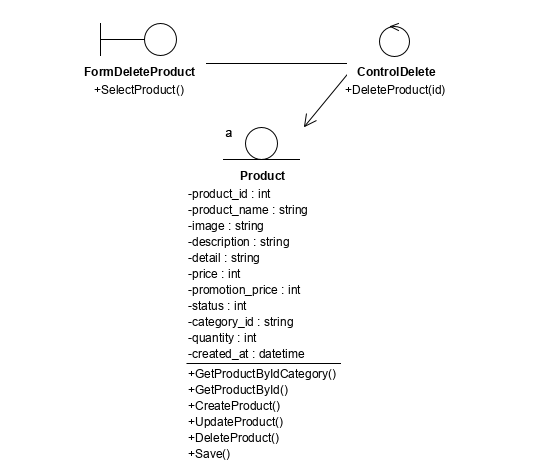
Hình: . Biểu đồ lớp chi tiết VOPC thêm sản phẩm

Biểu đồ lớp chi tiết VOPC sửa sản phẩm



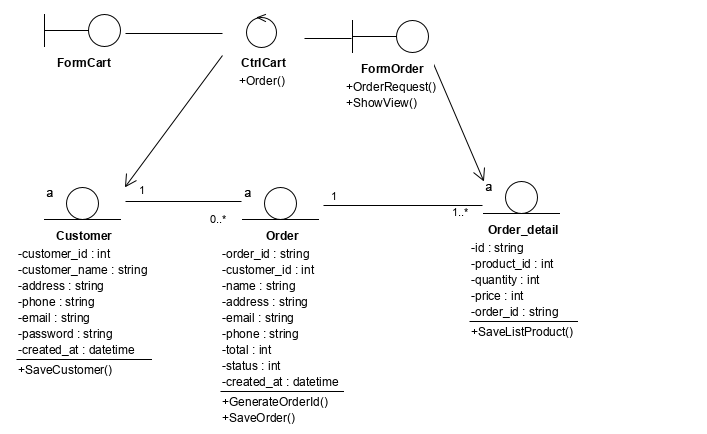
Hình: . Biểu đồ lớp chi tiết VOPC sửa sản phẩm

Biểu đồ lớp chi tiết VOPC xóa sản phẩm



Hình: . Biểu đồ lớp chi tiết VOPC xóa sản phẩm

Biểu đồ lớp chi tiết VOPC đặt hàng

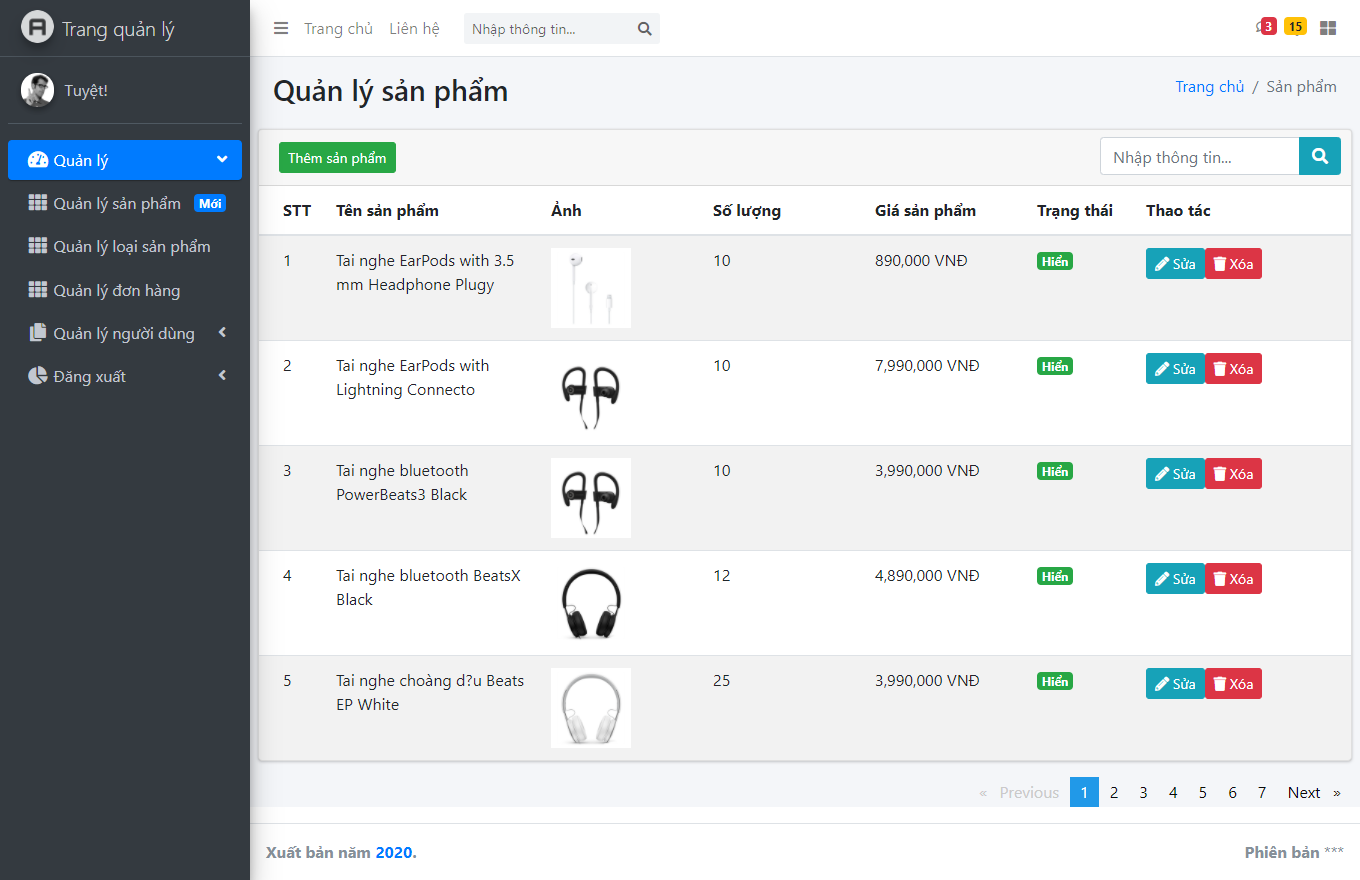


Hình: . Biểu đồ lớp chi tiết VOPC đặt hàng

* + 1. **Thiết kế giao diện**

**Giao diện của trang quản trị**

*Giao diện của trang quản lý sản phẩm*

**

Hình: . Giao diện trang quản lý sản phẩm

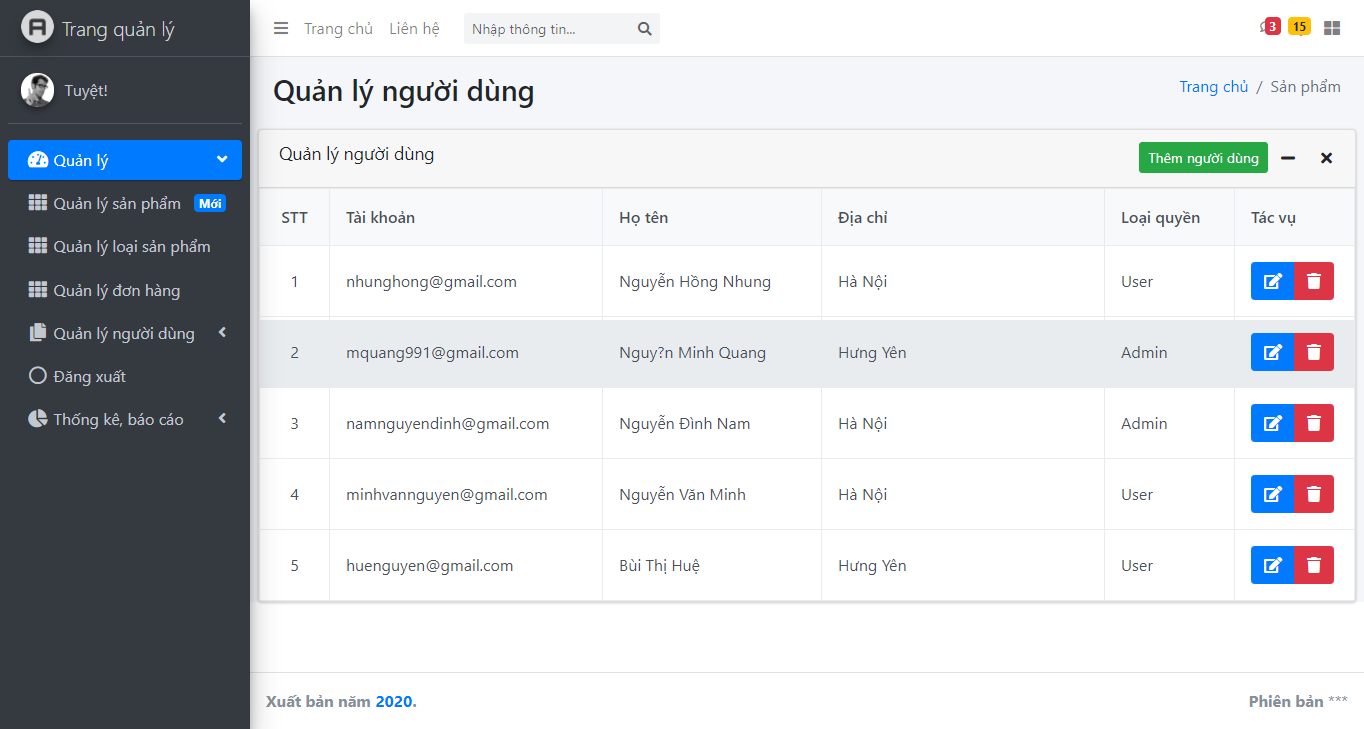
Chức năng: Hiển thị danh sách sản phẩm và các chức năng cho phép người quản trị thêm, sửa, xoá sản phẩm.

Sự kiện: Đăng nhập vào trang quản trị

Thành công thì có thể thực thi các quyền quản lý thêm sửa xóa các thông tin có thẩm quyền.

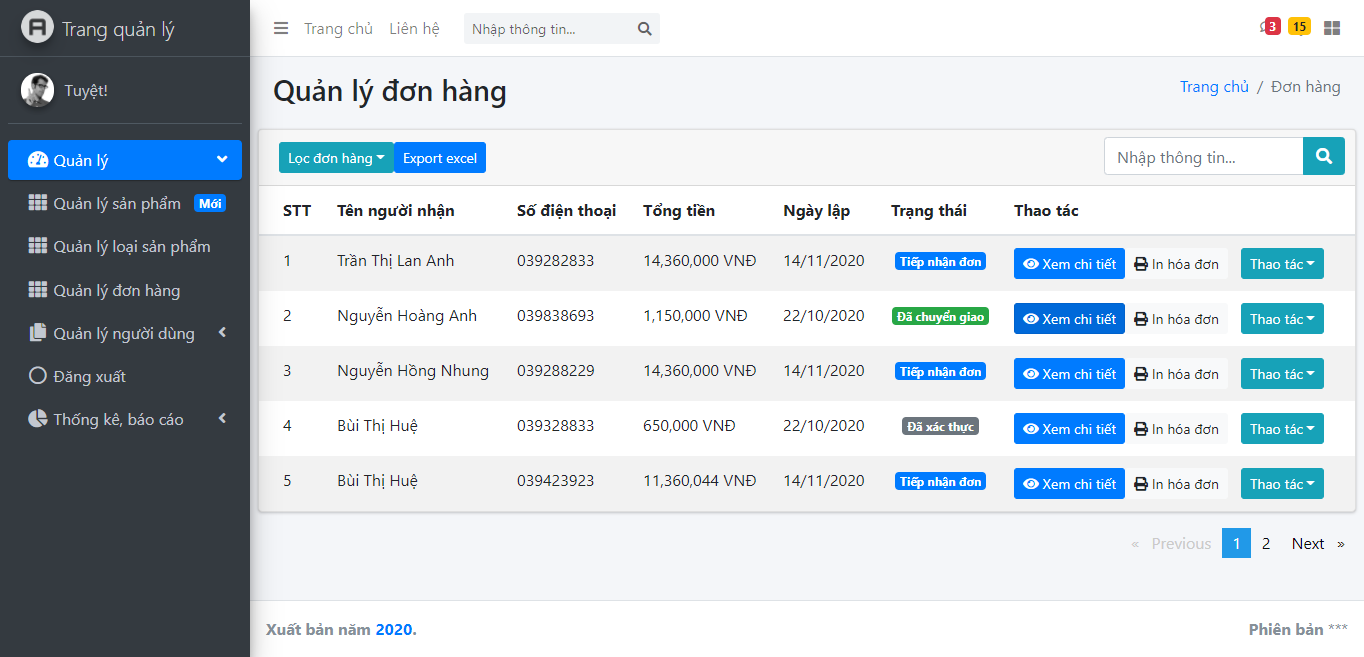
Sai thông báo không đúng, nhập lại.

*Giao diện quản lý người dùng*



Hình: . Giao diện trang quản lý người dùng

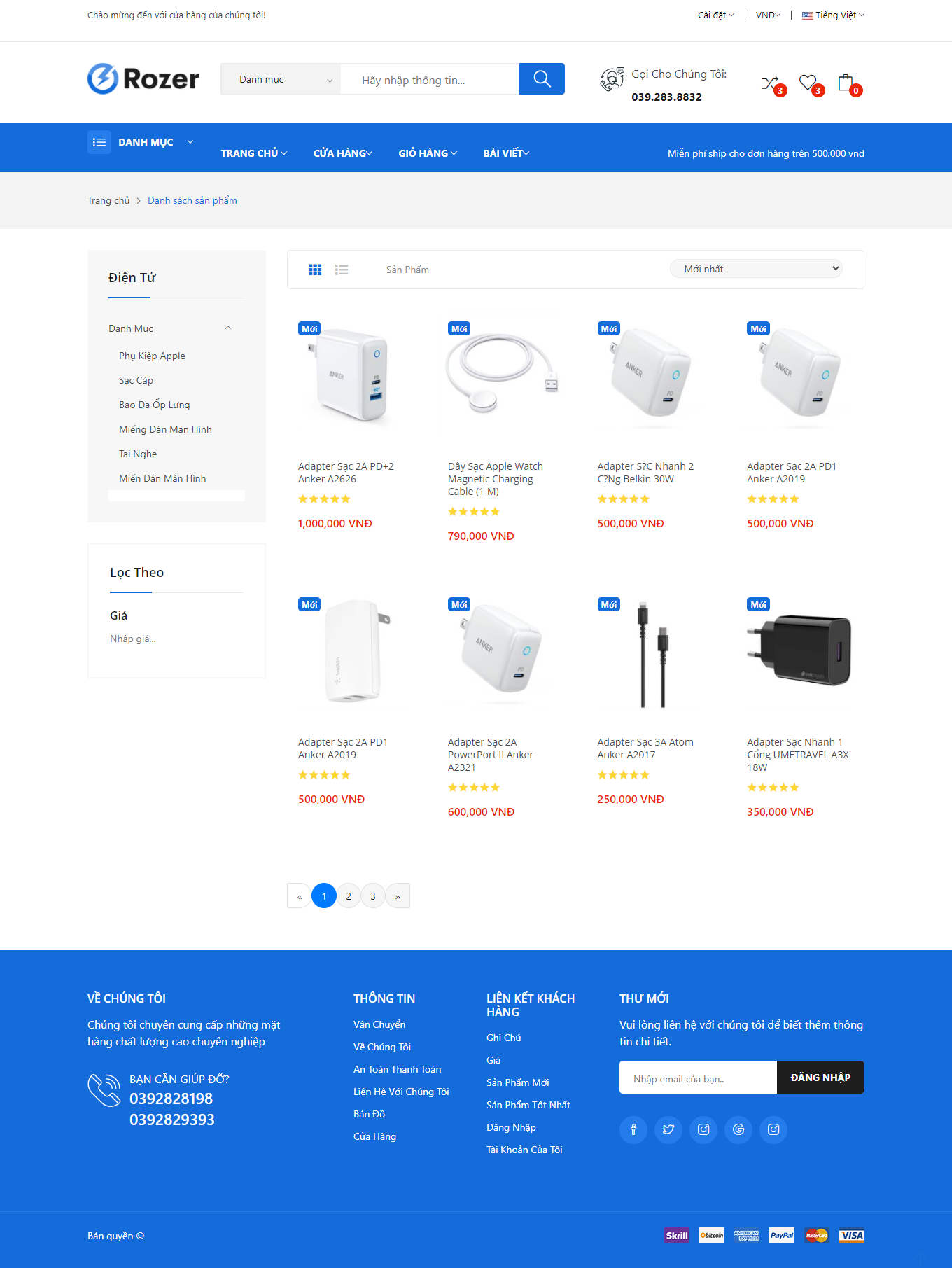
*Giao diện quản lý đơn hàng*

**

Hình: . Giao diện trang quản lý đơn hàng

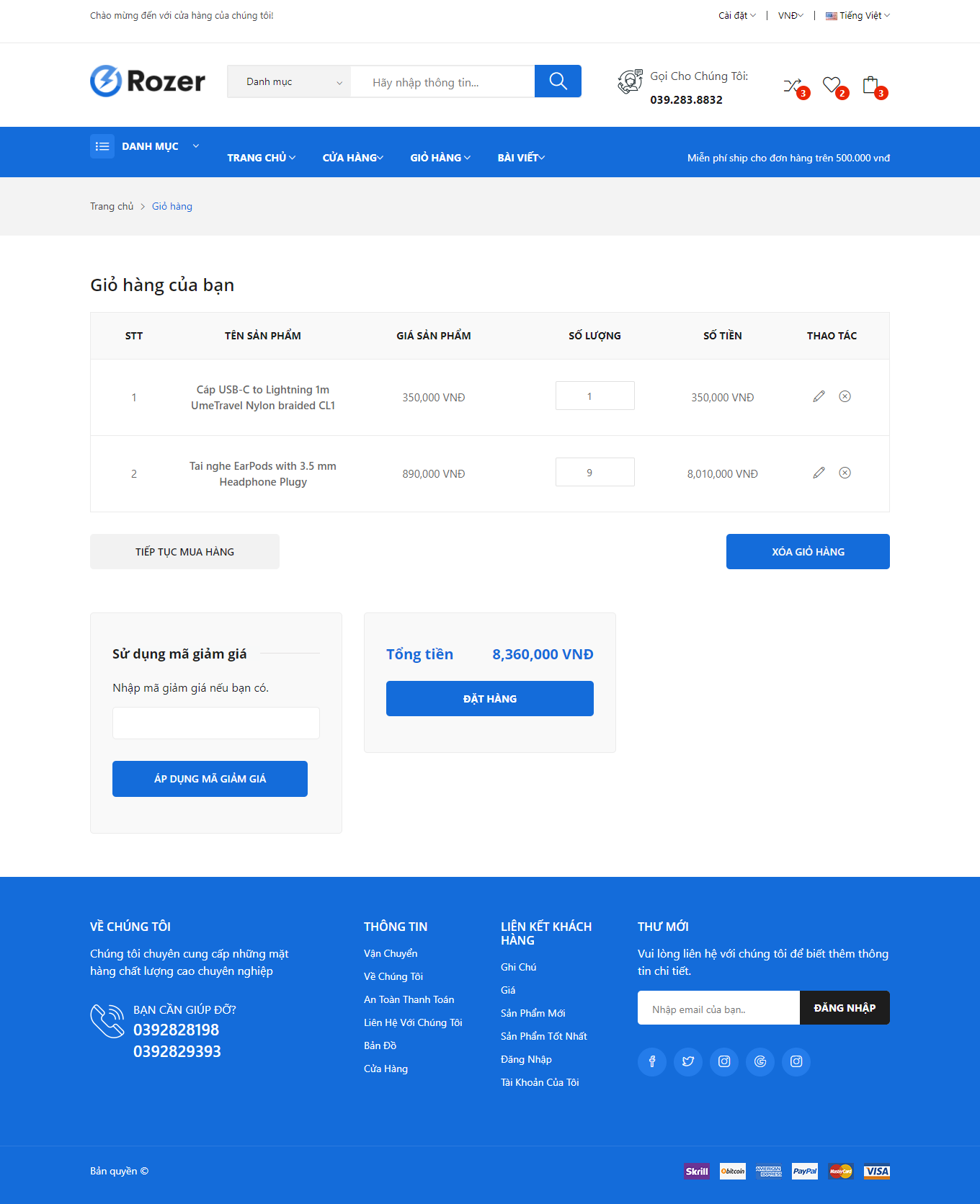
**Giao diện của trang người dùng**

*Giao diện của trang xem theo loại*



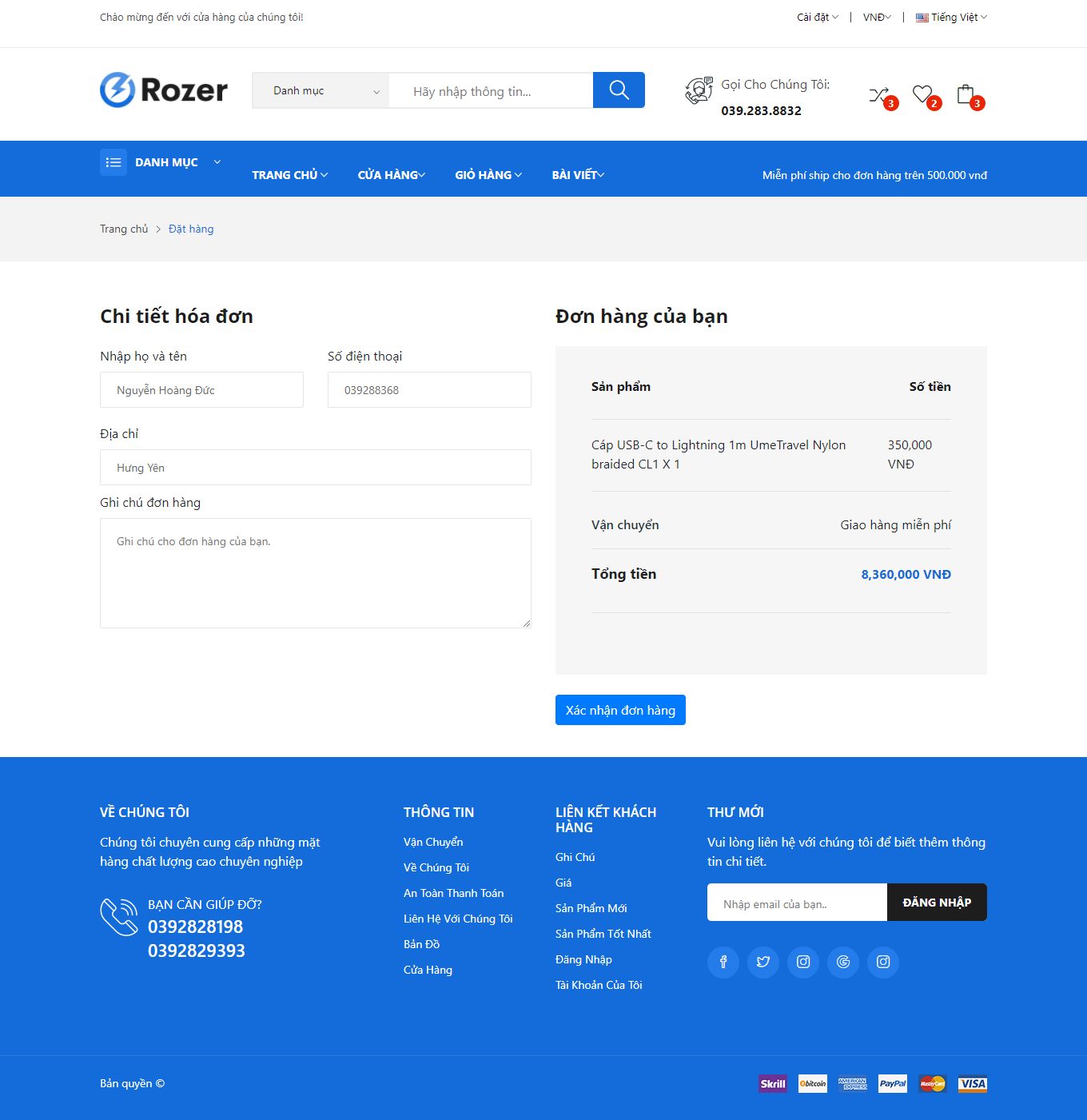
Hình: . Giao diện trang loại sản phẩm

*Giao diện giỏ hàng*



Hình: . Giao diện trang giỏ hàng

Giao diện trang đặt hàng



Hình: . Giao diện trang đặt hàng

# TRIỂN KHAI WEBSITE

* 1. Triển khai các chức năng cho phân hệ người dùng
* Thực hiện được các chức năng cơ bản của website đáp ứng được mọi nhu cầu khách hàng:

Giao diện tối ưu, tư vấn đầy đủ, giới thiệu về website và sản phẩm thực tế…

Xem sản phẩm, xem chi tiết thông tin…

* Khách hàng có thể tìm kiếm các sản phẩm cần qua giá hoặc tên sản phẩm.
* Khách hàng có thể đăng nhập và đăng ký tài khoản cho mình.
* Khách hàng xem được giỏ hàng của mình vào chỉnh sửa nó một cách đầy đủ và chính xác để có thể thực hiện đặt hàng.
  + 1. Trang chủ

a) Phía font end

* Xây dựng bố cục trang Home bằng các thẻ HTML

Trang chủ chia làm 3 phần là đầu trang, thân trang và cuối trang.

* Kỹ thuật định dạng bằng CSS

Đặt tất cả các file css vào 1 tệp sau đó dùng thì gọi ra

* Sử dụng Javascript, Jquery, Angular để lập trình các chức năng
* Cần thực hiện từng bước từ viết giao diện bằng html
* Sau đó dùng các hàm bên file app.module.ts, app-routing.module.ts và các hàm api.service để gọi được csdl từ backend ra.

b) Phía backend

* Triển khai các lớp tầng DataAccess

Lớp này có chức năng giao tiếp với hệ quản trị cơ sở dữ liệu như thực hiện các công việc liên quan đến lưu trữ và truy vấn dữ liệu (thêm, sửa, xoá, tìm kiếm…)

* Triển khai lớp tầng Bussiness

Phần này trình bày về cách thức cài đặt lớp ở tầng Bussiness để đáp ứng được các xử lý được gọi ở tầng Controller

* Triển khai lớp tầng Controller

Phần này trình bày về cách thức cài đặt các Action để đáp ứng được các xử lý được gọi ở tầng View cho trang chủ

* + 1. Trang Xem Sản phẩm

a) Phía font end

* Xây dựng bố cục trang Xem Sản Phẩm bằng các thẻ HTML:
* Trang chủ chia làm 3 phần là đầu trang, thân trang và cuối trang
* Kỹ thuật định dạng bằng CSS và
* Đặt tất cả các file css vào 1 tệp sau đó dùng thì gọi ra
* Sử dụng Javascript, Jquery, Angular để lập trình các chức năng
* Cần thực hiện từng bước từ viết giao diện bằng html sau đó dùng các hàm bên file app.module.ts, app-routing.module.ts và các hàm api.service để gọi được csdl từ backend ra.

b) Phía backend

* Triển khai các lớp tầng DataAccess

Phần này nhằm lấy csdl dữ liệu từ Model của sản phẩm, loại sản phẩm…

* Triển khai lớp tầng Bussines

Phần này giúp sử lý các dữ liệu đã lấy được từ DataAccess đẩy lên controller

* Triển khai lớp tầng Controller

Phần này giúp điều khiển các hoạt động và đưa dữ liệu có thể view được trên giao diện xem sản phẩm

* + 1. Trang Quản lý giỏ hàng

a) Phía font end

* Xây dựng bố cục trang Giỏ Hàng bằng các thẻ HTML:
* Trang chủ chia làm 3 phần là đầu trang, thân trang và cuối trang
* Kỹ thuật định dạng bằng CSS và
* Đặt tất cả các file css vào 1 tệp sau đó dùng thì gọi ra
* Sử dụng Javascript, Jquery, Angular để lập trình các chức năng
* Cần thực hiện từng bước từ viết giao diện bằng html sau đó dùng các hàm bên file app.module.ts, app-routing.module.ts và các hàm api.service để gọi được csdl từ backend ra.

b) Phía backend

* Triển khai các lớp tầng DataAccess

Phần này nhằm lấy csdl dữ liệu từ Model của sản phẩm, loại sản phẩm…

* Triển khai lớp tầng Bussiness

Phần này giúp sử lý các dữ liệu đã lấy được từ DataAccess đẩy lên controller

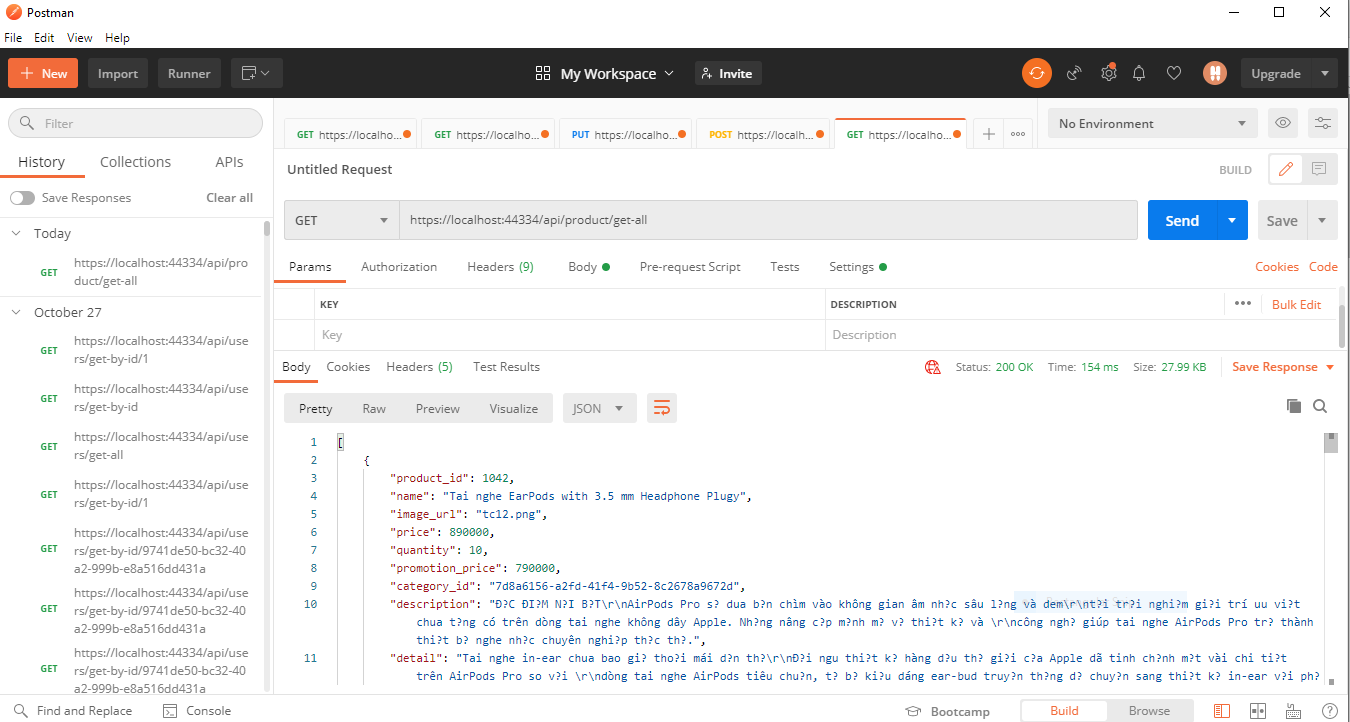
* Triển khai lớp tầng Controller

Phần này giúp điều khiển các hoạt động và đưa dữ liệu có thể view được trên giao diện giỏ hàng

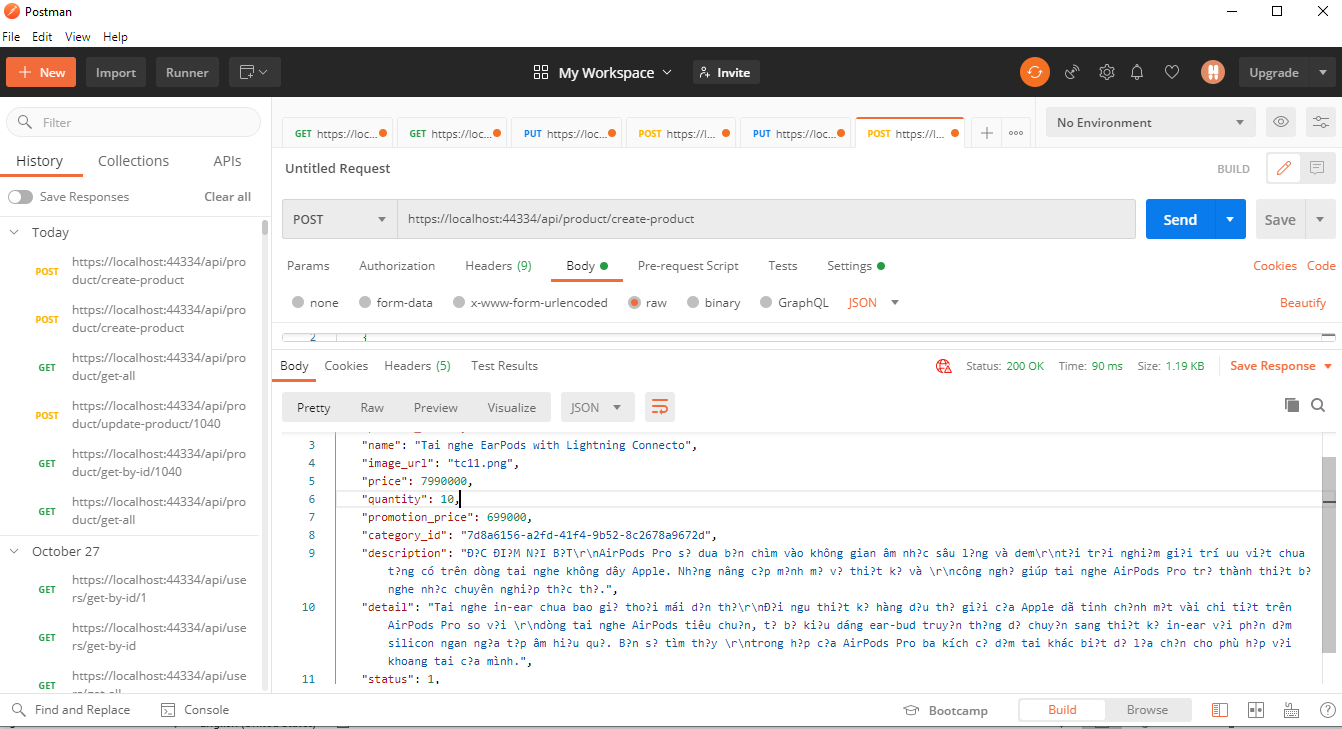
* 1. Triển khai các chức năng cho phân hệ quản trị nội dung
* Sau thời gian thực hiện nhóm chúng em đã hoàn thiện tương đối hệ thống theo đúng tiến độ và yêu cầu đặt ra cho phân hệ quản trị:
* Tìm hiểu hệ thống kỹ càng chính xác. Phân tích rõ được:
* Yêu cầu nghiệp vụ bài toán
* Hiện trạng hệ thống và yêu cầu người sử dụng.
* Các chức năng cần có của hệ thống.
* Phân tích thiết kế với UML.
* Thiết kế các chức năng của hệ thống đáp ứng yêu cầu của ban quản lý.
* Giao diện thân thiện, dễ sử dụng.
* Đưa hệ thống vào thử nghiệm và bàn giao hệ thống thành công.
  1. Kiểm thử và triển khai ứng dụng
     1. Kiểm thử

Xây dựng và mô tả dữ liệu/hành động cho input và kết quả cho output

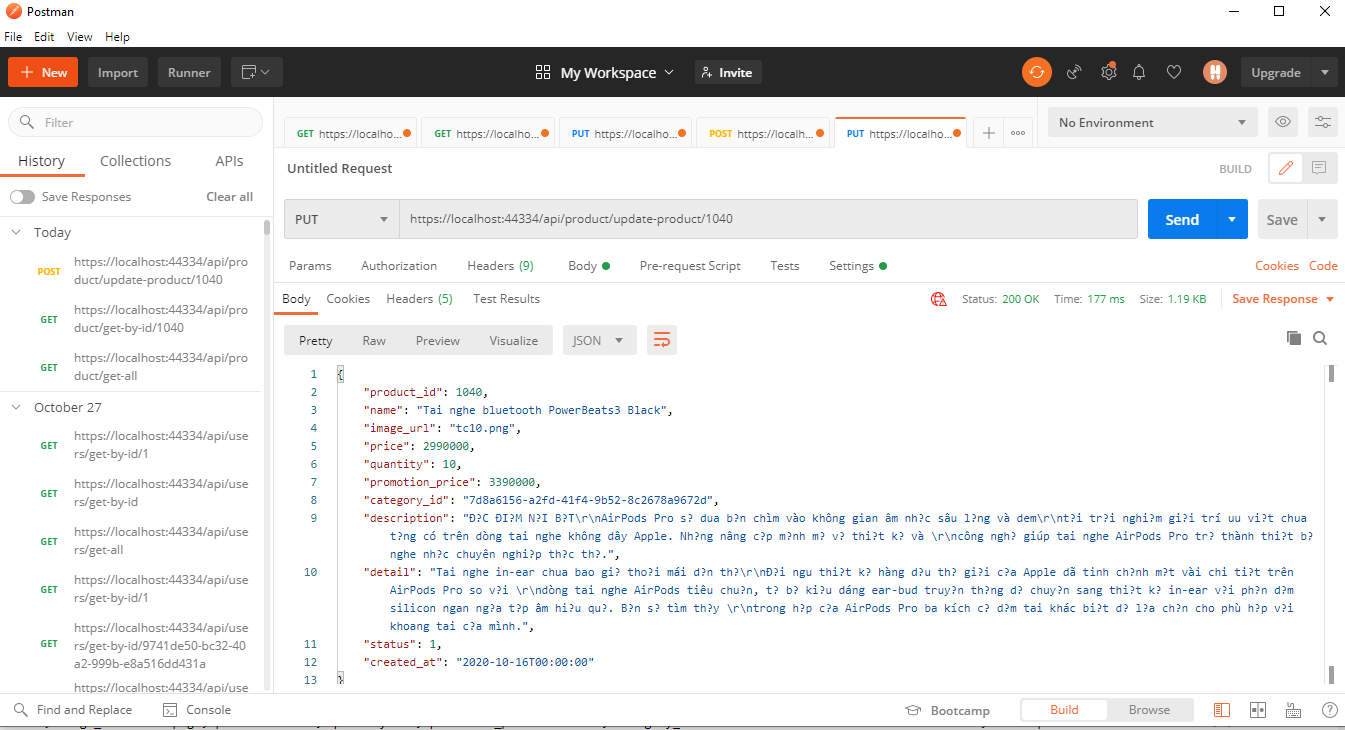
* Lấy tất cả sản phẩm



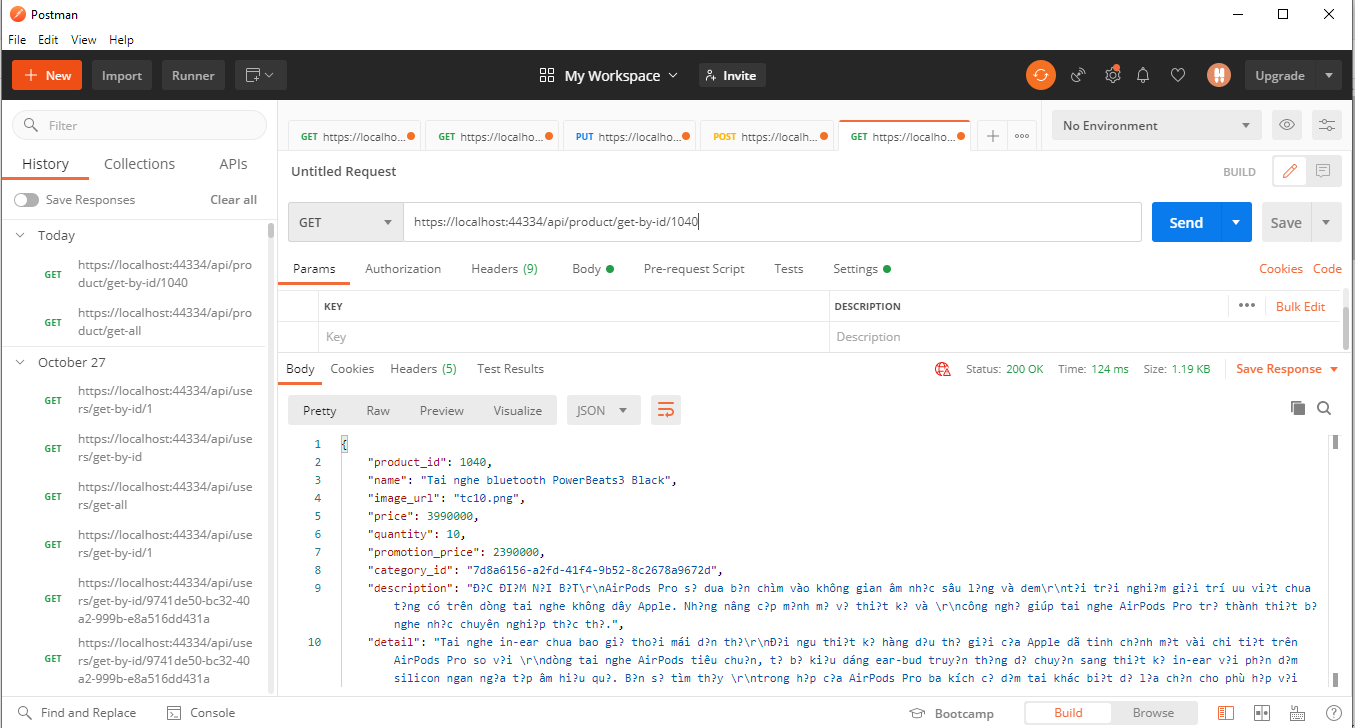
* Thêm sản phẩm



* Sửa sản phẩm



* Lấy sản phẩm theo mã



**Kiểm thử chức năng**

Việc kiểm thử chức năng yêu cầu chúng ta thực hiện test tất cả các link trong trang web, định dạng được sử dụng trong các trang web để gửi và nhận các thông tin cần thiết từ người dùng. Ngoài ra còn có kết nối cơ sở dữ liệu, kiểm tra cookies và xác minh HTML/CSS.

Kiểm tra các liên kết (links) bao gồm kiểm tra liên kết ngoài trang web, liên kết nội bộ, liên kết tới các vị trí trong cùng trang, liên kết sử dụng để gửi email tới admin hoặc người dùng khác trong trang…

Kiểm tra form của các trang: form là phần cơ bản của các trang web, nên cần được kiểm tra một cách kỹ càng, bao gồm các yêu cầu sau:

* Không nhập gì vào các trường bắt buộc thì sao?
* Giá trị mặc định của các trường là gì?
* Nhập đầu vào không đúng validate của các trường thì sao?
* Thao tác trên các trường: xem, nhập, lưu, sửa, xóa…

Kết nối cơ sở dữ liệu: yêu cầu kiểm tra tính toàn vẹn của dữ liệu khi tạo, sửa, xóa form…hoặc bất cứ chức năng nào có liên quan tới cơ sở dữ liệu. Khi truy vấn cơ sở dữ liệu thì kết quả trả về có tương ứng với kết quả nhìn thấy được trên giao diện không, dữ liệu có được lấy và cập nhật chính xác không?...

Kiểm thử cookies: cookies là các tệp được tạo bởi trang web đã truy cập để lưu trữ thông tin duyệt web, như các tùy chọn trang web hoặc thông tin đăng nhập của người dùng. Người dùng có thể tùy chỉnh trên trình duyệt nhằm quản lý cookies, thực hiện các thao tác cho phép lưu, hoặc xóa, hoặc chặn…để kiểm tra các tính năng lưu hoặc không lưu trạng thái đăng nhập, tính năng bảo mật của ứng dụng web.

* + - 1. Kiểm thử tính khả dụng

Tính khả dụng của trang web được định nghĩa là trang web dễ sử dụng, có hướng dẫn sử dụng rõ ràng, rành mạch, mỗi trang đều có menu chính và menu này phải nhất quán.

Ngoài ra, khi kiểm thử tính khả dụng, chúng ta còn cần thực hiện kiểm thử các điều khiển chuyển hướng (như button, text box, text link, bread crum…), nội dung của trang web phải dễ hiểu và thân thiện với người sử dụng. Không có lỗi chính tả, không khó để đọc chữ, hình ảnh được sắp xếp gọn gàng, hợp lý.

* + - 1. Kiểm thử giao diện

Các giao diện chính bao gồm:

Giao diện web server và server ứng dụng

Giao diện server ứng dụng và giao diện server dữ liệu

Kiểm tra tất cả các tương tác giữa các server. Nếu server dữ liệu hoặc web server trả lại bất kỳ báo lỗi nào cho bất kỳ truy vấn nào từ server ứng dụng thì ngay lập tức server ứng dụng phải nhận được và cho hiển thị cảnh báo tới người dùng. Kiểm tra các trường hợp giao dịch bị ngắt đột ngột do người dùng, hoặc kết nối tới server bị gián đoạn, bị khởi động lại…

* + - 1. Kiểm thử khả năng tương thích

Yêu cầu thực hiện các kiểm tra như sau:

* Tương thích với trình duyệt (trên máy tính và điện thoại di động): Người dùng khác nhau có thể sử dụng trình duyệt khác nhau tùy theo nhu cầu, thói quen…của họ. Cần phải kiểm tra ứng dụng web trên càng nhiều trình duyệt càng tốt (IE, Firefox, Chrome, Safari, Opera…) để kiểm tra sự tương thích. Kiểm tra trên cả các phiên bản khác nhau của trình duyệt. Kiểm tra trên cả trình duyệt của thiết bị điện thoại thông minh. Nếu ứng dụng chạy tốt hơn, hoặc có ưu tiên tương thích hơn với trình duyệt nào đó thì cần có thông báo tới người dùng.
* Tương thích với hệ điều hành: một số chức năng của ứng dụng có thể không tương thích với một số hệ điều hành, hoặc có những lưu ý khác khi sử dụng, điều này cần phải được kiểm tra kỹ và thông báo cho người dùng được biết.
* Tương thích với các thiết bị ngoại vi (máy in…): khi người dùng có lệnh in trang thì phải đảm bảo tính chính xác của fonts, cỡ chữ, cỡ giấy…mà người dùng đã chọn.
  + - 1. Kiểm thử hiệu năng

Bao gồm các yêu cầu:

*Kiểm thử chịu tải*: kiểm thử hiệu năng của ứng dụng với các tốc độ kết nối mạng khác nhau. Kiểm thử khi có nhiều người dùng cùng truy cập hoặc cùng yêu cầu một trang xem hệ thống có thể duy trì hoạt động được không? Hoặc kiểm thử khi người dùng tải lên hoặc tải xuống một số lượng dữ liệu đặc biệt lớn…

*Kiểm thử áp lực*: tức là việc đẩy hệ thống ra ngoài giới hạn của nó, thử làm gián đoạn trang web bằng cách tăng lượng tải cao hơn và kiểm tra xem hệ thống phản ứng như thế nào và phục hồi như thế nào.

* + - 1. Kiểm thử bảo mật

Một số case cho kiểm thử bảo mật web như sau:

Gõ trực tiếp url vào thanh địa chỉ của trình duyệt mà không qua đăng nhập. Các trang nội bộ phải không được mở.

Sau khi đăng nhập và mở các trang nội bộ, thay đổi url trực tiếp bằng cách đổi tham số ID của trang tới trang thuộc quyền người dùng đã đăng nhập khác. Truy cập phải bị từ chối bởi người dùng này không thể xem trang thống kê của người dùng khác.

Thử các giá trị đầu vào không hợp lệ trong các trường username, password. Hệ thống phải báo lỗi.

Các thư mục web, các tệp tin không được truy nhập trực tiếp mà không có tùy chọn “Download”.

Kiểm tra CAPTCHA cho các đăng nhập tự động

Tất cả các phiên giao dịch, các thông báo lỗi, các hành vi cố gắng xâm phạm an ninh phải ghi trong log và lưu tại web server.

* + 1. Đóng gói ứng dụng

 Khi chúng ta đã hoàn thành xong chương trình để tránh bị mất dư liệu đột ngột thì chúng ta nên đóng gói ứng dụng. Và cụ thể ở chương trình của em thì việc được coi như đóng gói ứng dụng chính là tải dữ liệu lên github nếu khi mất chương trình thì chỉ việc nên và tải file trên github là có thể lấy lại dữ liệu được cập nhật gần nhất.

* + 1. Triển khai ứng dụng
* Để xây dựng được dự án bằng angular chúng ta cần cài đặt:

**Nodejs**: version phải lớn hơn 8.11

**Npm**: version phải lớn hơn 5.6

Node.js và npm là nền tảng để phát triển web hiện đại sử dụng Angular và các nền tảng khác. Node cho phép khách hàng phát triển và xây dựng các công cụ. Chúng tôi sẽ sử dụng trình quản lý gói nút (npm) để cài đặt tất cả các phụ thuộc thư viện JavaScript.

* **Angular CLI**: dùng những câu lệnh giúp tạo nhanh các components, services và modules.

Ứng dụng Angular được tạo và phát triển chủ yếu thông qua Angular CLI (công cụ giao diện dòng lệnh) giúp tạo dự án, thêm tệp và thực hiện nhiều tác vụ phát triển liên tục như kiểm tra, đóng gói và triển khai.

Angular CLI chăm sóc cấu hình và khởi tạo các thư viện khác nhau. Nó cũng giúp chúng tôi thêm các thành phần, chỉ thị, dịch vụ, v.v. vào các ứng dụng Angular đã có. Cũng cần nhắc lại rằng CLI sử dụng Typecript và Webpack để đóng gói mô-đun, Karma để kiểm tra đơn vị và Thước đo góc để kiểm tra đầu cuối. Nó bao gồm mọi thứ bạn cần để bắt đầu viết ứng dụng Angular ngay lập tức.

* **IDE**: Visual Studio Code hoặc Sublime Text để lập trình trên đó.

Ứng dụng Angular được tạo và phát triển chủ yếu thông qua Angular CLI (công cụ giao diện dòng lệnh) giúp tạo dự án, thêm tệp và thực hiện nhiều tác vụ phát triển liên tục như kiểm tra, đóng gói và triển khai.

Angular CLI chăm sóc cấu hình và khởi tạo các thư viện khác nhau. Nó cũng giúp chúng tôi thêm các thành phần, chỉ thị, dịch vụ, v.v. vào các ứng dụng Angular đã có. Cũng cần nhắc lại rằng CLI sử dụng Typecript và Webpack để đóng gói mô-đun, Karma để kiểm tra đơn vị và Thước đo góc để kiểm tra đầu cuối. Nó bao gồm mọi thứ bạn cần để bắt đầu viết ứng dụng Angular ngay lập tức.

# KẾT LUẬN

**Kết quả đạt được**

* Sau thời gian thực hiện em đã hoàn thiện hệ thống theo đúng tiến độ và yêu cầu đặt ra. Tìm hiểu hệ thống kỹ càng chính xác, phân tích rõ được:
* Hiện trạng website và yêu cầu người sử dụng.
* Các chức năng cần có của website.
* Thiết kế các chức năng của hệ thống đáp ứng yêu cầu của người dùng
* Sau khi hoàn thành xong được được Website bán hàng điện tử online, em có thêm những kỹ năng nền tảng thiết kế và lập trình web phía front end (HTML, JavaScript, Bootstrap, Angular…) và back end (Net Core) để xây dựng website theo thiết kế và xây dựng giao diện lập trình API.
* Bên cạnh đó, em cũng được mở rộng thêm các kiến thức chuyên ngành mới nhờ việc tìm hiểu một số kiến thức lập trình để áp dụng vào việc phát triển ứng dụng này. Nhờ vậy, kỹ năng tự học và vận dụng các kỹ năng mới của em được nâng cao hơn.
* Khi thực hiện và hoàn thành dự án, em cũng đã biết thêm được nhiều kinh nghiệm quý giá khi xây dựng website nói riêng và phát triển phần mềm. Những kỹ năng này sẽ nền tảng để giúp em nâng cao trình độ bản thân, kinh nghiệm trong thực tế để có thể làm việc ngoài các doanh nghiệp sau này.

**Hạn chế đề tài**

* Dù đã toàn tâm cố gắng, song do khả năng và trình độ nhất định đề tài còn giới hạn về mặt quy mô ứng dụng.
* Một số chức năng còn bị hạn chế chưa giải quyết được tối ưu.

TÀI LIỆU THAM KHẢO

x

|  |  |
| --- | --- |
| [1] | Nguyễn Hữu Đông, Công nghệ Web và Ứng dụng (2020), Đại học SPKT Hưng Yên. |
| [2] | Nguyễn Hoàng Điệp, Lập trình Web API (2020), Đại học SPKT Hưng Yên |
| [3] | <https://angular.io/start> |
| [4] | <https://dotnet.microsoft.com/apps/aspnet/apis> |
| [5] | <https://dotnet.microsoft.com/download/dotnet-core> |