TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP

Trong thời đại ngày nay CNTT đóng vai trò quan trọng hầu như trong tất cả các lĩnh vực. Do vậy con người phải không ngừng học tập để nâng cao trình độ hiểu biết nếu không nâng cao trình độ hiểu biết con người sẽ bị tục hậu trong thời đại CNTT phát triển một cách nhanh chóng như hiện nay. Nhất là từ khi xuất hiện internet, nhu cầu trao đổi thông tin ngày càng cao, nhu cầu trao đổi mua bán của con người ngày càng đa dạng và phong phú việc mua bán hàng trên mạng ngày càng được chú trọng đến.

LaRose chuyên nhập hàng hóa từ những nước như Pháp, Mỹ, Đức…với đa dạng mặt hàng như mỹ phẩm, đồ gia dụng, thời trang. Chất lượng sản phẩm của LaRose luôn được đảm bảo. Tuy nhiên, LaRose vẫn chưa được nhiều khách hàng biết đến một cách rộng rãi.

Từ vấn đề phát sinh đó, với ý tưởng mang LaRose đến với nhiều người hơn, em đã thực hiện đề tài ”Xây dựng website bán hàng LaRose”. Trong đồ án này, em đã xây dựng website bằng ngôn ngữ PHP và Laravel Framework.

LỜI CẢM ƠN

Lời đầu tiên, em xin gửi lời cảm ơn đến tất cả thầy cô Viện Công nghệ thông tin và truyền thông, đặc biệt là thầy cô Bộ môn Công nghệ phần mềm đã truyền đạt vốn kiến thức quý báu cho chúng em trong thời gian học tập tại trường.

Đặc biệt, em xin cảm ơn cô Nguyễn Thị Thu Trang, người đã trực tiếp hướng dẫn, giúp đỡ và góp ý kiến cho em hoàn thành đồ án tốt nghiệp này.

Em xin gửi lời cảm ơn đến hai bạn cùng nhóm Nguyễn Quang Thái và Nguyễn Văn Cương đã động viên và giúp đỡ em tiếp tục nghiên cứu và học tập.

Cuối cùng, em xin kính chúc các thầy cô dồi dào sức khỏe và ngày càng thành công trong sự nghiệp cao quý.

**Trân trọng cảm ơn**

Hà Nội, tháng 5 năm 2017

Sinh viên

**Hồ Thị Vinh**

MỤC LỤC

[TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP 1](#_Toc483206986)

[LỜI CẢM ƠN 2](#_Toc483206987)

[MỤC LỤC 3](#_Toc483206988)

[DANH MỤC CÁC BẢNG 5](#_Toc483206989)

[DANH MỤC HÌNH ẢNH 6](#_Toc483206990)

[DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ 7](#_Toc483206991)

[MỞ ĐẦU 8](#_Toc483206992)

[1. Lý do chọn đề tài 8](#_Toc483206993)

[2. Mục đích của đề tài 8](#_Toc483206994)

[CHƯƠNG 1: ĐẶT VẤN ĐỀ VÀ ĐỊNH HƯỚNG GIẢI PHÁP 9](#_Toc483206995)

[1.1 Mô tả bài toán 9](#_Toc483206996)

[1.2 Bố cục chung của đồ án 10](#_Toc483206997)

[CHƯƠNG 2: CÁC NỀN TẢNG CÔNG NGHỆ 12](#_Toc483206998)

[2.1 Hệ quản trị cơ sở dữ liệu MySQL 12](#_Toc483206999)

[2.1 jQuery Ajax 13](#_Toc483207000)

[CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG 14](#_Toc483207001)

[3.1 Mô hình hóa yêu cầu 14](#_Toc483207002)

[3.1.1 Khảo sát yêu cầu người dùng 14](#_Toc483207003)

[3.1.2 Chức năng tổng quan của hệ thống 14](#_Toc483207004)

[3.1.3 Đặc tả yêu cầu 17](#_Toc483207005)

[3.1.4 Quy trình nghiệp vụ 26](#_Toc483207006)

[3.2 Thiết kế kiến trúc 29](#_Toc483207007)

[3.2.1 Mô hình MVC 29](#_Toc483207008)

[3.2.2 Thiết kế kiến trúc 30](#_Toc483207009)

[3.3 Thiết kế chi tiết 31](#_Toc483207010)

[3.3.1 Thiết kế giao diện 31](#_Toc483207011)

[3.3.2 Thiết kế lớp 38](#_Toc483207012)

[3.3.3 Thiết kế cơ sở dữ liệu 40](#_Toc483207013)

[CHƯƠNG 4: NHỮNG VẤN ĐỀ KHÓ KHĂN VÀ GIẢI PHÁP 44](#_Toc483207014)

[4.1 Công nghệ sử dụng – Laravel Framewok 44](#_Toc483207015)

[4.2 Tìm kiếm bằng Full Text Search 46](#_Toc483207016)

[CHƯƠNG 5: XÂY DỰNG PHẦN MỀM VÀ TRIỂN KHAI 49](#_Toc483207017)

[5.1 Xây dựng hệ thống 49](#_Toc483207018)

[5.1.1 Công cụ lập trình 49](#_Toc483207019)

[5.1.2 Xây dựng hệ thống bằng Laravel Framework 50](#_Toc483207020)

[5.2 Triển khai và chạy thử 50](#_Toc483207021)

[5.2.1 Triển khai 50](#_Toc483207022)

[5.2.2 Chạy thử 51](#_Toc483207023)

[CHƯƠNG 6: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 54](#_Toc483207024)

[6.1 Kết luận 54](#_Toc483207025)

[6.2 Hướng phát triển 54](#_Toc483207026)

[TÀI LIỆU THAM KHẢO 55](#_Toc483207027)

DANH MỤC CÁC BẢNG

[Bảng 3.1 Danh sách các Usecase chính 19](#_Toc483134376)

[Bảng 3.2 Mô tả UC Tìm kiếm sản phẩm 19](#_Toc483134377)

[Bảng 3.3 Luồng chính UC Tìm kiếm sản phẩm 20](#_Toc483134378)

[Bảng 3.4 Mô tả UC Thêm sản phầm vào giỏ hàng 20](#_Toc483134379)

[Bảng 3.5 Luồng chính UC Thêm sản phẩm vào giỏ hàng 20](#_Toc483134380)

[Bảng 3.6 Luồng phụ UC Thêm sản phẩm vào giỏ hàng 21](#_Toc483134381)

[Bảng 3.7 Mô tả UC Xóa sản phẩm trong giỏ hàng 21](#_Toc483134382)

[Bảng 3.8 Luồng chính UC Xóa sản phẩm trong giỏ hàng 22](#_Toc483134383)

[Bảng 3.9 Mô tả UC chỉnh sửa mặt hàng trong giỏ hàng 22](#_Toc483134384)

[Bảng 3.10 Luồng chính UC Chỉnh sửa mặt hàng trong giỏ hàng 22](#_Toc483134385)

[Bảng 3.11 Luồng phụ UC Chỉnh sửa mặt hàng trong giỏ hàng 23](#_Toc483134386)

[Bảng 3.12 Mô tả UC Đặt hàng 23](#_Toc483134387)

[Bảng 3.13 Luồng chính UC Đặt hàng 24](#_Toc483134388)

[Bảng 3.14 Luồng phụ UC Đặt hàng 25](#_Toc483134389)

[Bảng 3.15 Mô tả UC xem thông tin cá nhân 25](#_Toc483134390)

[Bảng 3.16 Luồng chính UC Xem thông tin cá nhân 25](#_Toc483134391)

[Bảng 3.17 Luồng phụ UC Xem thông tin cá nhân 26](#_Toc483134392)

[Bảng 3.18 Mô tả UC Chỉnh sửa thông tin cá nhân 26](#_Toc483134393)

[Bảng 3.19 Luồng chính UC chỉnh sửa thông tin cá nhân 27](#_Toc483134394)

[Bảng 3.20 Luồng phụ UC Chỉnh sửa thông tin cá nhân 27](#_Toc483134395)

[Bảng 3.21 Mô tả UC Bình luận cho từng sản phẩm 27](#_Toc483134396)

[Bảng 3.22 Luồng chính UC Bình luận cho từng sản phẩm 28](#_Toc483134397)

[Bảng 3.23 Danh sách một số giao diện 34](#_Toc483134398)

[Bảng 3.24 Đặc tả màn hình giỏ hàng 38](#_Toc483134399)

[Bảng 3.25 Đặc tả màn hình chi tiết sản phẩm 39](#_Toc483134400)

[Bảng 3.26 Đặc tả màn hình thanh toán 40](#_Toc483134401)

DANH MỤC HÌNH ẢNH

[Hình 3.1 Biểu đồ Usecase Tổng quan 15](#_Toc483207163)

[Hình 3.2 Biểu đồ Usecase Quản lý giỏ hàng 16](#_Toc483207164)

[Hình 3.3 Biểu đồ Usecase Quản lý thông tin cá nhân 16](#_Toc483207165)

[Hình 3.4 Biểu đồ hoạt động UC Thêm sản phẩm vào giỏ hàng 27](#_Toc483207166)

[Hình 3.5 Biểu đồ hoạt động UC Xem giỏ hàng 27](#_Toc483207167)

[Hình 3.6 Biểu đồ hoạt động UC Đặt hàng 28](#_Toc483207168)

[Hình 3.7 Biểu đồ hoạt động UC Quản lý thông tin cá nhân 29](#_Toc483207169)

[Hình 3.8: Mô hình MVC trong Laravel 30](#_Toc483207170)

[Hình 3.9 Biểu đồ dịch chuyển màn hình 33](#_Toc483207171)

[Hình 3.10 Màn hình Mockup Trang chủ 33](#_Toc483207172)

[Hình 3.11 Màn hình Mockup Thanh toán 34](#_Toc483207173)

[Hình 3.12 Màn hình Mockup Giỏ hàng 34](#_Toc483207174)

[Hình 3.13 Màn hình Mockup Chi tiết sản phẩm 35](#_Toc483207175)

[Hình 3.14 Biểu đồ lớp chức năng quản lý giỏ hàng 38](#_Toc483207176)

[Hình 3.15 Biểu đò lớp chức năng thanh toán 39](#_Toc483207177)

[Hình 3.16 Biểu đồ lớp chức năng quản lý thông tin cá nhân 40](#_Toc483207178)

[Hình 3.17 Cơ sở dữ liệu phần khách hàng 41](#_Toc483207179)

[Hình 3.18 Cơ sở dữ liệu phần sản phẩm 42](#_Toc483207180)

[Hình 3.19 Cơ sở dữ liệu phần đặt hàng 43](#_Toc483207181)

[Hình 4.1 Cấu trúc thư mục Laravel 44](#_Toc483207182)

[Hình 5.1 Xây dựng hệ thống bằng Laravel 50](#_Toc483207183)

[Hình 5.2: Màn hình trang chủ 51](#_Toc483207184)

[Hình 5.3: Màn hình chi tiết sản phẩm 52](#_Toc483207185)

[Hình 5.4: Màn hình giỏ hàng 52](#_Toc483207186)

[Hình 5.5: Màn hình liên hệ 53](#_Toc483207187)

DANH MỤC CÁC TỪ VIẾT TẮT VÀ THUẬT NGỮ

|  |  |  |
| --- | --- | --- |
| STT | Chữ viết tắt | Chữ viết đầy đủ |
| 1 | UC | Usecase |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

MỞ ĐẦU

## Lý do chọn đề tài

Việc kinh doanh - mua bán là nhu cầu không thể thiểu đối với mỗi chúng ta. Trong thời đại cạnh tranh hiện nay việc giới thiệu sản phẩm kinh doanh đến từng cá nhân với chi phí thấp, hiệu quả cao là một vấn đề nan giải của doanh nghiệp cùng với nhu cầu mua sắm với những sản phẩm đa chủng loại, đạt chất lượng, và hợp túi tiền của người tiêu dùng vì vậy thương mại điện tử đã được ra đời và dần dần phát triển trên toàn thế giới.

Thương mại điện tử là một hình thức mua bán và trao đổi thông tin qua Internet. Công ty, doanh nghiệp có thể giới thiệu tất cả sản phẩm của mình bằng hình ảnh và thông tin trên trang web để khách hàng có thể xem, thao khảo và lựa chọn. Đây là hình thức kinh doanh tiện lợi, dễ dàng và đặc biệt đang phát triển, sẽ có cơ hội, thuận lợi lớn cho các công ty, doanh nghiệp đang trên đường phát triển tạo nên thành công và danh tiếng cho mình.

La Rose là một Website cung cấp cho người sử dụng những chức năng cần thiết để tiến hành giao dịch, quản lý sự hoạt động của cửa hàng. Đối với khách hàng, hệ thống cho phép xem thông tin về sản phẩm của cửa hàng, khách hàng có thể chọn sản phẩm trên website đưa và giỏ hàng và tiến hành giao dịch mua bán.

## Mục đích của đề tài

Website bán hàng là một showroom ảo mở cửa 24/24h có thể đón khách hàng vào bất cứ lúc nào. Dựa vào website, khách hàng có thể biết được cửa hàng kinh doanh mặt hàng gì? Giá bao nhiêu?...đó là những câu hỏi mà khách hàng muốn biết về cửa hàng bán đồ xách tay La Rose.

Bên cạnh đó, website còn giúp cửa hàng nhận được phản hồi của khách hàng một cách nhanh chóng và hiệu quả.

# ĐẶT VẤN ĐỀ VÀ ĐỊNH HƯỚNG GIẢI PHÁP

## Mô tả bài toán

“Website bán hàng xách tay La Rose” là hệ thống quảng bá hình ảnh của cửa hàng cũng như thể hiện việc chăm sóc khách hàng ở mọi lúc, mọi nơi. Với website này khách hàng không cần đến cửa hàng mà vẫn tham khảo được tác dụng, giá cả và thông tin về từng sản phẩm và sự đảm bảo về sự cung cấp. Tạo sự thuận tiện, thoải mái cho khách hàng khi mua sản phẩm.

Trang web cần đáp ứng các nghiệp vụ cụ thể như sau:

**Xem hàng**

Khách hàng truy cập vào địa chỉ của website trên thanh url, chọn vào các trang trên site để xem thông tin.

Khi khách hàng chọn một sản phẩm bất kì trên page, thông tin mà khách hàng gửi sẽ được xử lý, hệ thống sẽ lấy dữ liệu từ cơ sở dữ liệu, gửi lên cho người dùng.

**Bình luận**

Người xem có thể chia sẻ cảm nghĩ của họ về sản phẩm, thông tin bình luận của khách hàng sẽ được lưu vào database bao gồm tên, email, số điện thoại của khách hàng cùng nội dung mà khách hàng bình luận.

Hệ thống sẽ kiểm tra nội dung mà người dùng nhập vào, nếu hợp lệ thì lưu dữ liệu đó vào database, nếu không thì thông báo lỗi cho người dùng.

**Đặt hàng**

Cho phép khách hàng duyệt sản phẩm trên trang web. Khi khách hàng gửi thông tin đặt hàng thì mặt hàng đó sẽ được gửi đến giỏ hàng.

Khi khách hàng chọn mặt hàng muốn mua, ở phiên làm việc đó nếu khách hàng chọn lần đầu tiên thì ở giỏ hàng sẽ tạo mới sản phẩm số lượng bằng một. Nếu đã tồn tại mặt hàng đó trong giỏ hàng rồi thì số lượng của mặt hàng đó sẽ được tăng lên một.

**Giỏ hàng**

Cho phép khách hàng xem, điều chỉnh, thêm, xóa mặt hàng mà khách hàng đã đặt mua. Có chức năng tính tổng giá trị của đơn hàng đó.

Khách hàng có thể thay đổi số lượng mặt hàng muốn mua, khi khách hàng chọn lại số lượng và chọn cập nhật thì số lượng của mặt hàng đó cần được cập nhật lại, tổng giá cũng cần được tính toán lại. Khi khách hàng chọn xóa sản phẩm thì sản phẩm đó phải được xóa khỏi giỏ hàng. Khách hàng chọn xóa hết thì sẽ thông báo không có mặt hàng nào trong giỏ hàng. Khách hàng chọn mua tiếp thì sẽ chuyển về trang chủ, chọn đặt hàng thì sẽ chuyển đến trang mua hàng.

**Tìm kiếm sản phẩm**

Để không mất thời gian duyệt từng sản phẩm trên website, khách hàng có thể biết thông tin về sản phẩm mình muốn mua bằng cách nhập thông tin vào ô tìm kiếm.

Khi khách hàng nhập tên sản phẩm muốn tìm kiếm, hệ thống sẽ so sánh tên đó với dữ liệu có trong database. Nếu nhận được kết quả sẽ trả kết quả tìm kiếm về cho người dùng.

**Xem mặt hàng mới**:

Khách hàng có thể xem thông tin các điện thoại mới nhất của cửa hàng. Hệ thống sẽ lựa chọn những sản phẩm mới nhất mà người quản trị web nhập vào, trả về kết quả, hiển trị cho người dùng xem.

**Xem điện thoại bán chạy**

Hiển thị thông tin các điện thoại bán chạy nhất của của hàng. Hệ thống sẽ lựa chọn những mặt hàng có số lượng bán được nhiều nhất để trả về kết quả và hiển thị cho người dùng.

## Bố cục chung của đồ án

**Mở đầu**

* Lý do chọn đề tài
* Cơ sở lý thuyết

**Chương 1: Đặt vấn đề và định hướng giải pháp**

* Mô tả bài toán

**Chương 2: Các nên tảng công nghệ**

* Laravel Framework
* Hệ quản trị cơ sở dữ liệu MySQL
* jQuery Ajax

**Chương 3: Phân tích thiết kế hệ thống**

* Mô hình hóa yêu cầu người dùng bằng biểu đồ Usecase
* Thiết kế kiến trúc: mô hình MVC, biểu đồ giao tiếp cho từng usecase
* Thiết kế giao diện, lớp và cơ sở dữ liệu

**Chương 4: Những vấn đề khó khăn và giải pháp**

* Công nghệ sử dụng
* Tìm kiếm sản phẩm bằng Full Text Search

**Chương 5: Xây dựng phần mềm và triển khai**

* Công cụ lập trình
* Triển khai và chạy thử

**Chương 6: Kết luận và hướng phát triển**

* Những vấn đề được giải quyết
* Những vấn đề còn tồn tại
* Hướng phát triển cho hệ thống trong tương lai

# CÁC NỀN TẢNG CÔNG NGHỆ

## Hệ quản trị cơ sở dữ liệu MySQL

#### Khái niệm

MySQL là hệ quản trị cơ sở dữ liệu mã nguồn mở miễn phí, được tích hợp sử dụng chung với apache, PHP. MySQL quản lý dữ liệu thông qua các cơ sở dữ liệu, mỗi cơ sở dữ liệu có thể có nhiều bảng quan hệ chứa dữ liệu. MySQL có cơ chế phân quyền người sử dụng riêng, mỗi người dùng có thể được quản lý một hoặc nhiều cơ sở dữ liệu khác nhau, mỗi người dùng có một tên truy cập (user name) và mật khẩu tương ứng để truy xuất đến cơ sở dữ liệu. Khi ta truy vấn tới cơ sở dữ liệu MySQL, ta phải cung cấp tên truy cập và mật khẩu của tài khoản có quyền sử dụng cơ sở dữ liệu đó. Nếu không, chúng ta sẽ không làm được gì cả

#### Ưu điểm của MySQL

* Tốc độ: MySQL rất nhanh. Những nhà phát triển cho rằng MySQL là cơ sở dữ liệu nhanh nhất mà bạn có thể có.
* Dễ sử dụng: MySQL tuy có tính năng cao nhưng thực sự là một hệ thống cơ sở dữ liệu rất đơn giản và ít phức tạp khi cài đặt và quản trị hơn các hệ thống lớn. –
* Giá thành: MySQL là miễn phí cho hầu hết các việc sử dụng trong một tổ chức.
* Hỗ trợ ngôn ngữ truy vấn: MySQL hiểu SQL, là ngôn ngữ của sự chọn lựa cho tất cả các hệ thống cơ sở dữ liệu hiện đại. Bạn cũng có thể truy cập MySQL bằng cách sử dụng các ứng dụng mà hỗ trợ ODC (Open Database Connectivity - một giao thức giao tiếp cơ sở dữ liệu được phát triển bởi Microsoft)
* Năng lực: Nhiều client có thể truy cập đến server trong cùng một thời gian. Các client có thể sử dụng nhiều cơ sở dữ liệu một cách đồng thời. Bạn có thể truy cập MySQL tương tác với sử dụng một vài giao diện để bạn có thể đưa vào các truy vấn và xem các kết quả: các dòng yêu cầu của khách hàng, các trình duyệt Web…
* Kết nối và bảo mật: MySQL được nối mạng một cách đầy đủ, các cơ sở dữ liệu có thể được truy cập từ bất kỳ nơi nào trên Internet do đó bạn có thể chia sẻ dữ liệu của bạn với bất kỳ ai, bất kỳ nơi nào. Nhưng MySQL kiểm soát quyền truy cập cho nên người mà không nên nhìn thấy dữ liệu của bạn thì không thể nhìn được.
* Tính linh động: MySQL chạy trên nhiều hệ thống UNIX cũng như không phải UNIX chẳng hạn như Windows. MySQL chạy được các với mọi phần cứng từ các máy PC ở nhà cho đến các máy server.
* Sự phân phối rộng: MySQL rất dễ dàng đạt được, chỉ cần sử dụng trình duyệt web của bạn. Nếu bạn không hiểu làm thế nào mà nó làm việc hay tò mò về thuật toán, bạn có thể lấy mã nguồn và tìm tòi nó. Nếu bạn không thích một vài cái, bạn có thể thay đổi nó.
* Sự hỗ trợ: Bạn có thể tìm thấy các tài nguyên có sẵn mà MySQL hỗ trợ. Cộng đồng MySQL rất có trách nhiệm. Họ trả lời các câu hỏi trên mailing list thường chỉ trong vài phút. Khi lỗi được phát hiện, các nhà phát triển sẽ đưa ra cách khắc phục trong vài ngày, thậm chí có khi trong vài giờ và cách khắc phục đó sẽ ngay lập tức có sẵn trên Internet.

#### Tại sao ta sử dụng hệ cơ sở dữ liệu MySQL?

* Nhanh và mạnh: MySQL không có đầy đủ những cơ sở vật chất cho một hệ Quản trị CSDL chính tông, nhưng đối với công việc thường nhật của phần đông mọi người thì nó cung cấp cũng khá nhiều thứ. Nếu công việc của bạn là lưu trữ dữ liệu trên web hoặc làm một trang Thương mại điện tử thì MySQL có đủ những thứ bạn cần. Đối với những CSDL cỡ trung bình thì MySQL hỗ trợ tuyệt vời về tốc độ.
* Cải tiến liên tục: MySQL được cải thiện liên tục với một tần số không ngờ. Các nhà phát triển nó cập nhật thường xuyên, ngoài ra còn bổ sung các tính năng hữu ích cho nó.

## jQuery Ajax

#### Ajax (Asynchoronous JavaScript và XML)

Nói ngắn gọn , Ajax là một bộ công cụ cho phép load dữ liệu từ server mà không yêu cầu tải lại trang.Nó sử dụng chức năng sẵn có XMLHttpRequest(XHR) của trình duyệt để thực hiện một yêu cầu đến server và xử lý dữ liệu server trả về.

Ví dụ: khi một người dùng viết một nhận xét trên bài viết đăng trên trang Facebook. Sau khi người dùng gửi nhận xét thành công trang Facebook mà người đó đang truy cập cần phải được cập nhật để hiển thị nhận xét vừa mới được tạo ra này. Nếu load lại toàn bộ trang mà người dùng đang truy cập thì sẽ không hiệu quả do tất cả những gì chúng ta muốn là hiển thị nhận xét mới được tạo ra, Ajax được tạo ra để giải quyết vấn đề này, thay vì tải lại toàn bộ trang trình duyệt sẽ chỉ l những phần được thay đổi để tiết kiệm thời gian chờ đợi một lượng thông tin lớn về từ server .

Một số ứng dụng sử dụng Ajax như : Gmail , Google Maps , Youtube , Facebook …

#### b) JQuery Ajax

Jquery cung cấp một số phương thức để thực hiện các chức năng ajax. Chúng ta có thể yêu cầu các text, HTML, XML và JSON từ server sử dụng cả giao thức HTTP GET và HTTP POST, chúng ta cũng có thể lấy dữ liệu từ bên ngoài trực tiếp vào trong phần tử được chọn.

# PHÂN TÍCH THIẾT KẾ HỆ THỐNG

## Mô hình hóa yêu cầu

### Khảo sát yêu cầu người dùng

Cửa hàng LaRose chuyên cung cấp các mặt hàng xách tay từ các nước như Mỹ, Pháp, Đức… Với hơn một năm hoạt động, LaRose luôn là điểm đến ưa thích của nhiều khách hàng. Cửa hàng luôn cập nhật các thông tin, mặt hàng mới nhất trên thế giới.

Tiêu chí hoạt động của cửa hàng là làm sao mặt hàng đến tay người tiêu dùng giá rẻ nhất thị trường nhưng chất lượng phải được đảm bảo. Việc giữ uy tín với khách hàng là phương châm hoạt động, phát triển thương hiệu của cửa hàng.

Website bán hàng là một showroom ảo mở cửa 24/24h có thể đón khác hàng vào bất cứ luc nào. Dựa vào website, khách hàng có thể biết được cửa hàng đang kinh doanh mặt hàng gì? Giad báo nhiêu?...đó là những câu hỏi mà khách hàng muốn biết về cửa hàng và sản phẩm của cửa hàng. Bên cạnh đó, website bán hàng còn giúp cửa hàng nhận được phản hồi của khách hàng một cách nhanh chóng và hiệu quả.

* Yêu cầu trang Web

Khách hàng là những người có nhu cầu mua sắm hàng hóa, họ sẽ tìm kiếm các mặt hàng cần thiết từ hệ thống và đặt mua các mặt hàng này. Vì thế hệ thống phải có các chức năng sau:

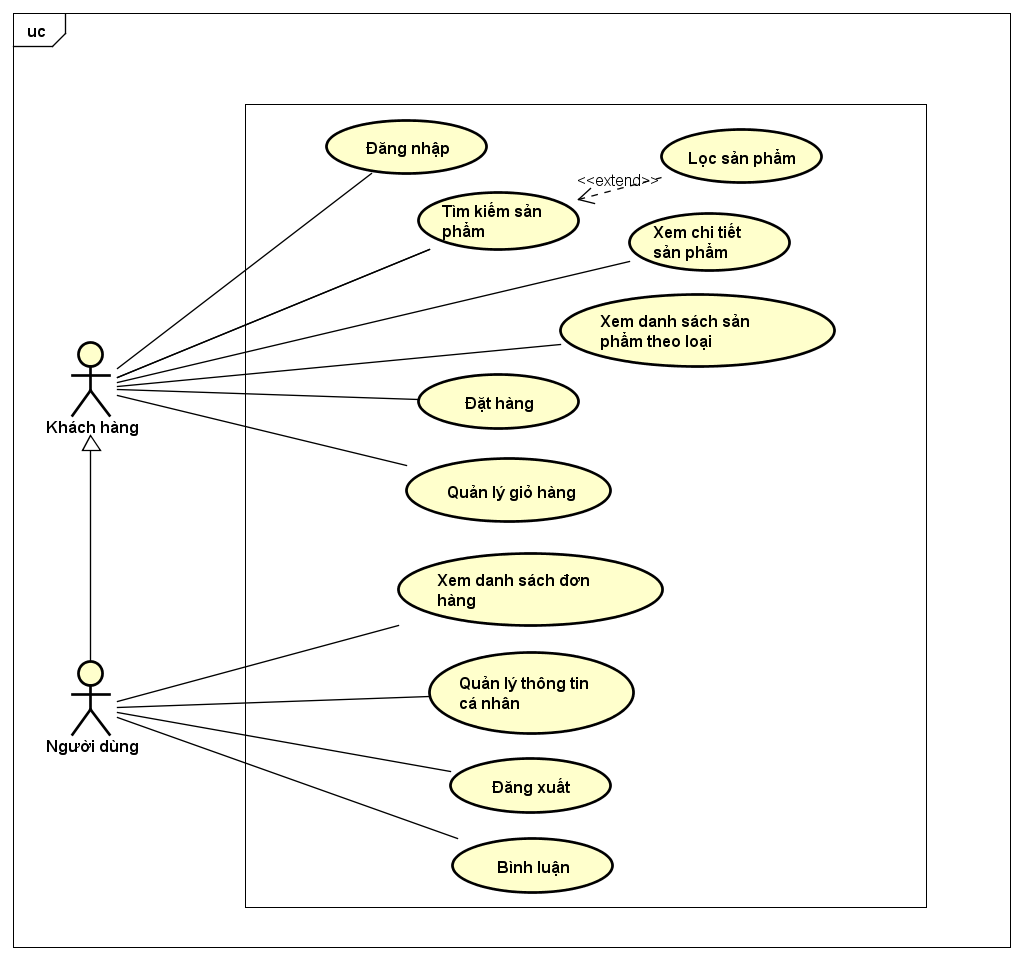
* Hiển thị danh sách các mặt hàng của cửa hàng để khách hàng xem, lựa chọn và mua
* Khách hàng xem các thông tin tin tức mới, khuyến mãi trên trang web.
* Sau khi khách hàng chọn và đặt hàng trực tiếp thì phải hiện lên giỏ hàng để khách hàng có thể nhập thông tin mua hàng và xem hóa đơn.

### Chức năng tổng quan của hệ thống

1. Biểu đồ usecase tổng quan

Hình 3.1 là biểu đồ usecase tổng quan, hệ thống gồm 2 nhóm người dùng: người dùng đã có tài khoản và đăng nhập vào hệ thống, khách hàng vãng lai.

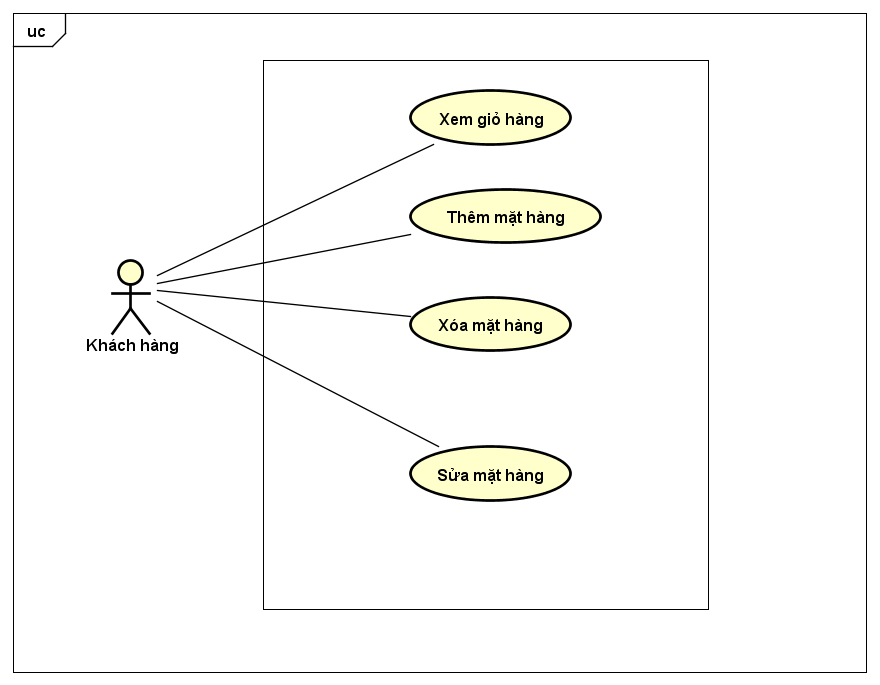
* Actor khách hàng: Khách hàng vào website có thể đăng nhập, tìm kiếm và xem sản phẩm. Ngoài ra khách hàng cũng có thể cho sản phẩm vào giỏ hàng và tiến hành mua hàng.
* Actor người dùng: Đã đăng ký tài khoản và đăng nhập vào hệ thống, người dùng có thể tìm kiếm, xem sản phẩm; đặt hàng và mua hàng. Ngoài những chức năng trên thì người dùng có thể xem được lịch sử mua hàng của mình cũng như bình luận về các sản phẩm trên website



Hình 3.1 Biểu đồ Usecase Tổng quan

1. Biểu đồ usecase quản lý giỏ hàng

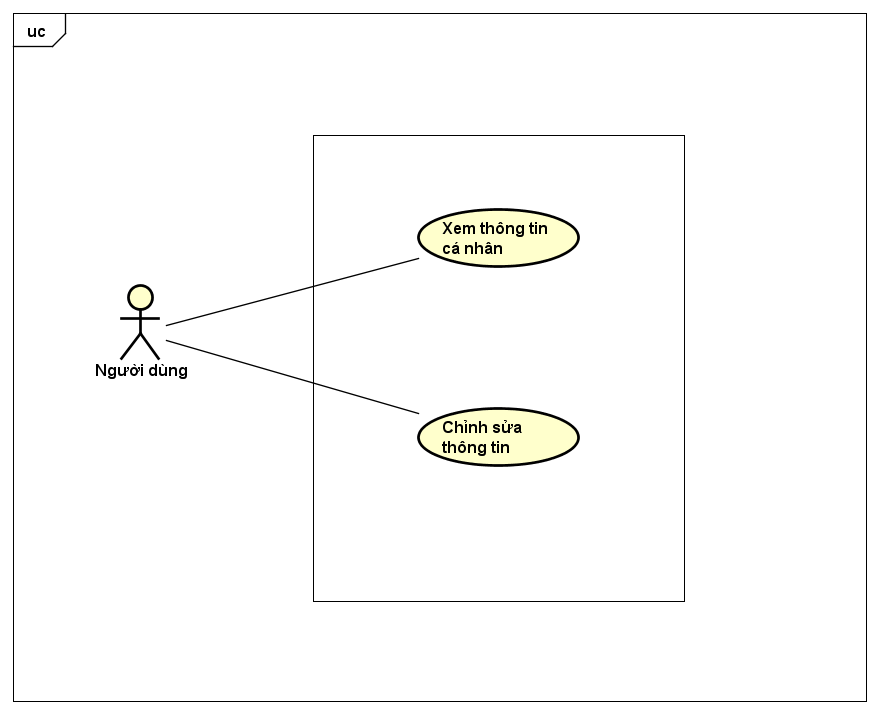
Trên hình 3.2 là biểu đồ usecase quản lý giỏ hàng, khách hàng có thể xem giỏ hàng, thêm sản phẩm vào giỏ, chỉnh sửa số lượng mỗi sản phẩm trong giỏ hàng, xóa những sản phẩm không muốn mua.



Hình 3.2 Biểu đồ Usecase Quản lý giỏ hàng

1. Biểu đồ usecase quản lý thông tin cá nhân

Hình 3.3 là biểu đồ usecase quản lý thông tin cá nhân. Sau khi người dùng đăng kí và đăng nhập vào hệ thống thì người dùng có thể xem thông tin đã đăng kí của mình trên website và có thể chỉnh sửa nếu muốn.



Hình 3.3 Biểu đồ Usecase Quản lý thông tin cá nhân

### Đặc tả yêu cầu

Danh sách các Usecase chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên chức năng | Mã Usecase | Tên Usecase |
| 1 | Tìm kiếm sản phẩm | UC01 | Tìm kiếm sản phẩm |
| 2 | Quản lý giỏ hàng | UC02 | Thêm sản phẩm vào giỏ hàng |
| UC03 | Xóa sản phẩm trong giỏ hàng |
| UC04 | Chỉnh sửa mặt hàng trong giỏ |
| 3 | Thanh toán | UC05 | Đặt hàng |
| 4 | Quản lý thông tin cá nhân | UC06 | Xem thông tin cá nhân |
| UC07 | Chỉnh sửa thông tin cá nhân |
| 5 | Bình luận | UC08 | Phản hồi cho từng sản phẩm |

Bảng 3.1 Danh sách các Usecase chính

* 1. UC01: Tìm kiếm sản phẩm

|  |  |
| --- | --- |
| Tên Usecase | Tìm kiếm sản phẩm |
| Tác nhân | Khách hàng, người dùng |
| Điều kiện đầu vào | Tên, giá, thông tin sản phẩm. |
| Kết quả đầu ra | Hiển thị danh sách kết quả |
| Mô tả | Khách hàng, người dùng muốn tìm kiếm thông tin sản phẩm. |

Bảng 3.2 Mô tả UC Tìm kiếm sản phẩm

* Luồng chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Khách hàng, người dùng | Nhập từ khóa và bấm tìm kiếm |  |
| 2 | Hệ thống | Nếu từ khóa có trong dữ liệu, hệ thống hiển thị danh sách kết quả. Nếu không thì trả về “Không tìm thấy sản phẩm nào” |  |

Bảng 3.3 Luồng chính UC Tìm kiếm sản phẩm

* 1. UC02: Thêm sản phẩm vào giỏ hàng

|  |  |
| --- | --- |
| Tên Usecase | Thêm sản phẩm vào giỏ hàng |
| Tác nhân | Khách hàng, người dùng |
| Điều kiện đầu vào | Không có |
| Kết quả đầu ra | Một sản phẩm được thêm vào giỏ hàng |
| Mô tả | Khách hàng, người dùng muốn thêm một sản phẩm vào giỏ hàng của mình |

Bảng 3.4 Mô tả UC Thêm sản phầm vào giỏ hàng

* Luồng chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Khách hàng, người dùng | Click vào button ”Thêm vào giỏ” của một sản phẩm cụ thể. |  |
| 2 | Hệ thống | Tạo giỏ hàng và lưu sản phẩm vào giỏ hàng |  |
| 3 | Hệ thống | Hiển thị giỏ hàng | Sau bước 2 |

Bảng 3.5 Luồng chính UC Thêm sản phẩm vào giỏ hàng

* Luồng phụ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Hệ thống | Nếu sản phẩm đã có trong giỏ hàng, hệ thống tự động cập nhật số lượng của sản phẩm đó. | Sau bước 1 |

Bảng 3.6 Luồng phụ UC Thêm sản phẩm vào giỏ hàng

* 1. UC03: Xóa sản phẩm trong giỏ hàng

|  |  |
| --- | --- |
| Tên Usecase | Xóa sản phẩm trong giỏ hàng |
| Tác nhân | Khách hàng, người dùng |
| Điều kiện đầu vào | Giỏ hàng đã có sản phẩm |
| Kết quả đầu ra | Sản phẩm được xóa khỏi giỏ hàng |
| Mô tả | Khách hàng, người dùng không muốn mua một sản phẩm bất kì trong giỏ hàng. |

Bảng 3.7 Mô tả UC Xóa sản phẩm trong giỏ hàng

* Luồng chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Khách hàng, người dùng | Click vào icon xóa |  |
| 2 | Hệ thống | Tự động xóa mặt hàng đó ra khỏi giỏ hàng |  |
| 3 | Hệ thống | Hiển thị giỏ hàng sau khi xóa |  |

Bảng 3.8 Luồng chính UC Xóa sản phẩm trong giỏ hàng

* 1. UC04: Chỉnh sửa mặt hàng trong giỏ hàng

|  |  |
| --- | --- |
| Tên Usecase | Chỉnh sửa mặt hàng trong giỏ hàng |
| Tác nhân | Khách hàng, người dùng |
| Điều kiện đầu vào | Giỏ hàng đã có sản phẩm |
| Kết quả đầu ra | Số lượng mỗi mặt hàng được sửa thành công |
| Mô tả | Khách hàng, người dùng muốn chỉnh sửa số lượng mỗi sản phẩm trong giỏ hàng. |

Bảng 3.9 Mô tả UC chỉnh sửa mặt hàng trong giỏ hàng

* Luồng chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Khách hàng, người dùng | Nhập số lượng muốn thay đổi trong mỗi mặt hàng và chọn icon chỉnh sửa |  |
| 2 | Hệ thống | Tự động cập nhật số lượng |  |
| 3 | Hệ thống | Hiển thị giỏ hàng sau khi cập nhật |  |

Bảng 3.10 Luồng chính UC Chỉnh sửa mặt hàng trong giỏ hàng

* Luồng phụ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Khách hàng, người dùng | Nhập số âm |  |
| 2 | Hệ thống | Thông báo lỗi, yêu cầu nhập số lớn hơn 0 |  |

Bảng 3.11 Luồng phụ UC Chỉnh sửa mặt hàng trong giỏ hàng

* 1. UC05: Đặt hàng

|  |  |
| --- | --- |
| Tên Usecase | Đặt hàng |
| Tác nhân | Khách hàng, người dùng |
| Điều kiện đầu vào | Giỏ hàng đã có sản phẩm |
| Kết quả đầu ra | Thông tin việc đặt hàng được lưu trong cơ sở dữ liệu |
| Mô tả | Khách hàng, người dùng muốn mua các sản phẩm đã được thêm vào giỏ hàng. |

Bảng 3.12 Mô tả UC Đặt hàng

* Luồng chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Khách hàng, người dùng | Click vào button thanh toán trong trang giỏ hàng |  |
| 2 | Hệ thống | Hiển thị trang thanh toán |  |
| 3 | Người dùng, khách hàng | * Người dùng: xem thông tin cá nhân trên trang thanh toán. * Khách hàng: Nhập thông tin giao hàng.   Nếu muốn giao hàng đến địa chỉ khác:   * Click vào ô textbox giao hàng đến địa chỉ khác. * Nhập thông tin địa chỉ giao hàng |  |
| 4 | Khách hàng, người dùng | Click vào button Đặt hàng |  |
| 5 | Hệ thống | * Lưu thông tin khách hàng, đơn hàng, chi tiết đơn hàng * Xóa giỏ hàng |  |

Bảng 3.13 Luồng chính UC Đặt hàng

* Luồng phụ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Hệ thống | Hiển thị lỗi khách chưa nhập đủ các trường bắt buộc | Sau bước 4 |

Bảng 3.14 Luồng phụ UC Đặt hàng

* 1. UC06: Xem thông tin cá nhân

|  |  |
| --- | --- |
| Tên Usecase | Xem thông tin cá nhân |
| Tác nhân | Người dùng |
| Điều kiện đầu vào | Người dùng đang ở trạng thái đăng nhập |
| Kết quả đầu ra | Thông tin cá nhân của người dùng đó |
| Mô tả | Người dùng muốn xem thông tin cá nhân của mình trên hệ thống |

Bảng 3.15 Mô tả UC xem thông tin cá nhân

* Luồng chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Người dùng | Chọn chức năng xem thông tin các nhân |  |
| 2 | Hệ thống | Hiển thị thông tin của người dùng đó |  |

Bảng 3.16 Luồng chính UC Xem thông tin cá nhân

* Luồng phụ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Hệ thống | Hiển thị lịch sử mua hàng của người dùng | Sau bước 2 |

Bảng 3.17 Luồng phụ UC Xem thông tin cá nhân

* 1. UC07: Chỉnh sửa thông tin cá nhân

|  |  |
| --- | --- |
| Tên Usecase | Chỉnh sửa thông tin cá nhân |
| Tác nhân | Người dùng |
| Điều kiện đầu vào | Người dùng đang ở trạng thái đăng nhập |
| Kết quả đầu ra | Thông tin cá nhân được chỉnh sửa theo yêu cầu người dùng |
| Mô tả | Người dùng muốn chỉnh sửa thông tin cá nhân của mình |

Bảng 3.18 Mô tả UC Chỉnh sửa thông tin cá nhân

* Luồng chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Người dùng | Nhập thông tin muốn chỉnh sửa trên trang thông tin cá nhân |  |
| 2 | Người dùng | Click vào button Thay đổi |  |
| 3 | Hệ thống | Tự động lưu vào cơ sở dữ liệu và thông báo thay đổi thành công. |  |
| 4 | Hệ thống | Hiển thị trang thông tin cá nhân của người dùng sau khi chỉnh sửa. |  |

Bảng 3.19 Luồng chính UC chỉnh sửa thông tin cá nhân

* Luồng phụ

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Hệ thống | Hiển thị thông báo lỗi người dùng nhập sai định dạng của các trường. |  |

Bảng 3.20 Luồng phụ UC Chỉnh sửa thông tin cá nhân

* 1. UC08: Phản hồi cho từng sản phẩm

|  |  |
| --- | --- |
| Tên Usecase | Bình luận cho từng sản phẩm |
| Tác nhân | Ngươi dùng |
| Điều kiện đầu vào | Người dùng đang ở trạng thái đăng nhập |
| Kết quả đầu ra | Hiển thi phản hồi trên sản phẩm |
| Mô tả | Người dùng muốn để lại phản hồi cho sản phẩm. |

Bảng 3.21 Mô tả UC Bình luận cho từng sản phẩm

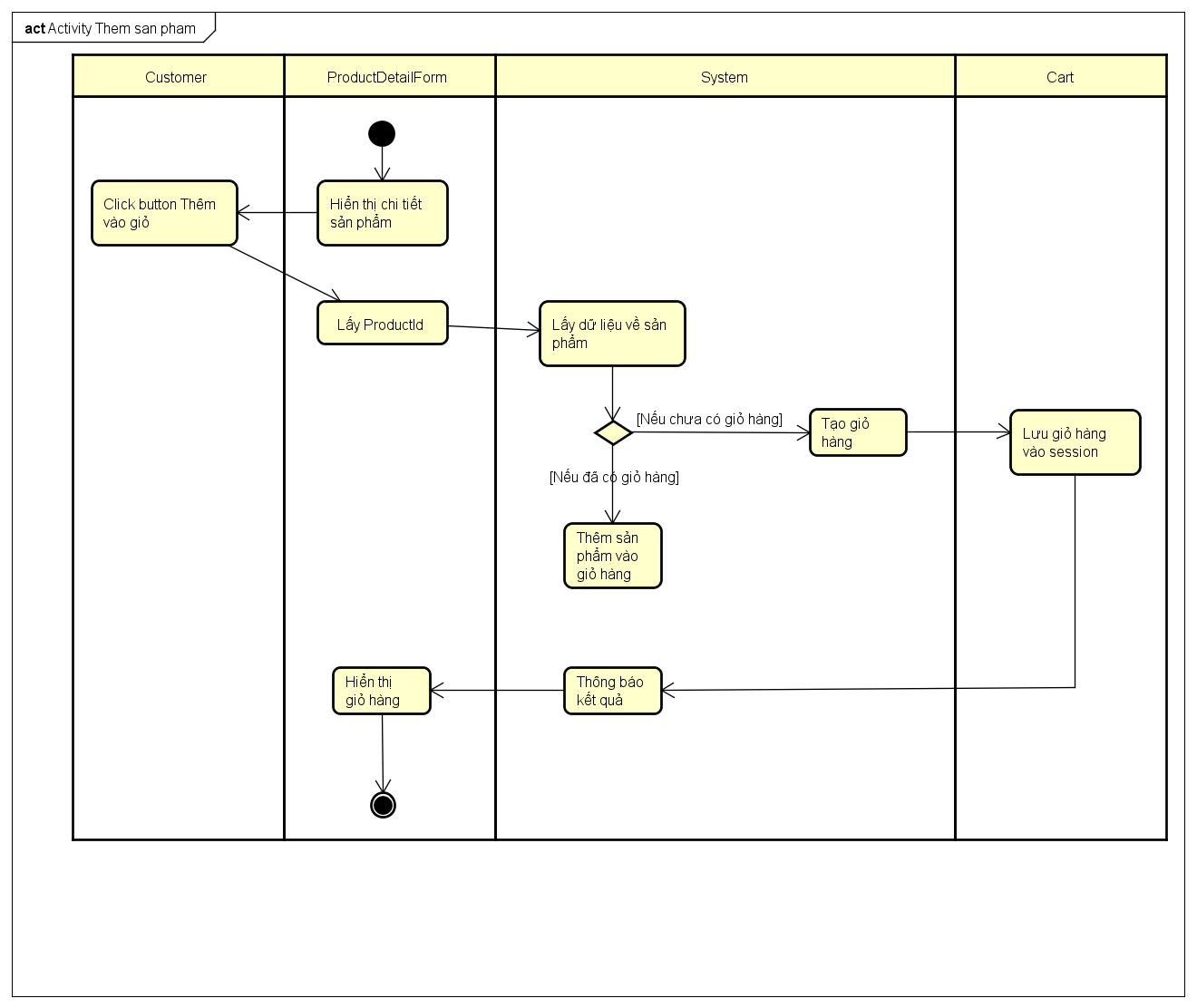
* Luồng chính

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Bên thực hiện | Hành động | Điều kiện thực hiện |
| 1 | Người dùng | Nhập phản hồi cho sản phẩm đang xem |  |
| 2 | Người dùng | Click vào button Gửi |  |
| 3 | Hệ thống | Lưu phản hổi và hiển thị phản hồi trên trang chi tiết sản phẩm. |  |

Bảng 3.22 Luồng chính UC Bình luận cho từng sản phẩm

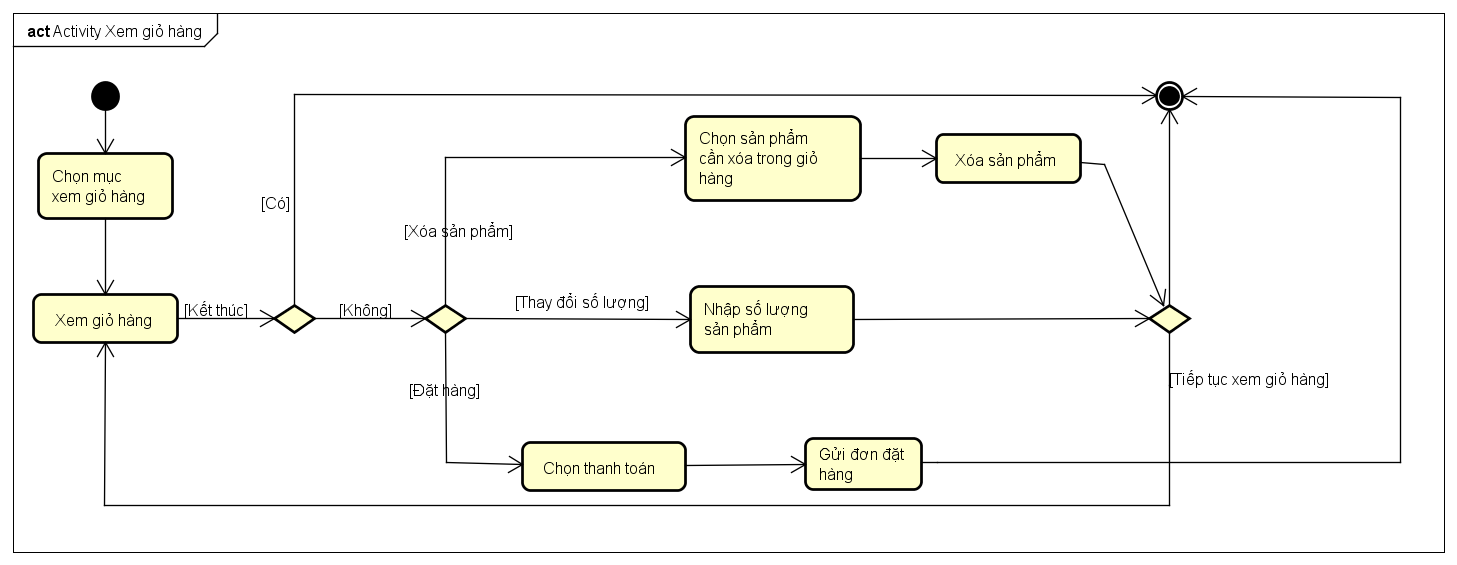
### Quy trình nghiệp vụ

1. Biểu đồ hoạt động Usecase Thêm sản phẩm vào giỏ hàng



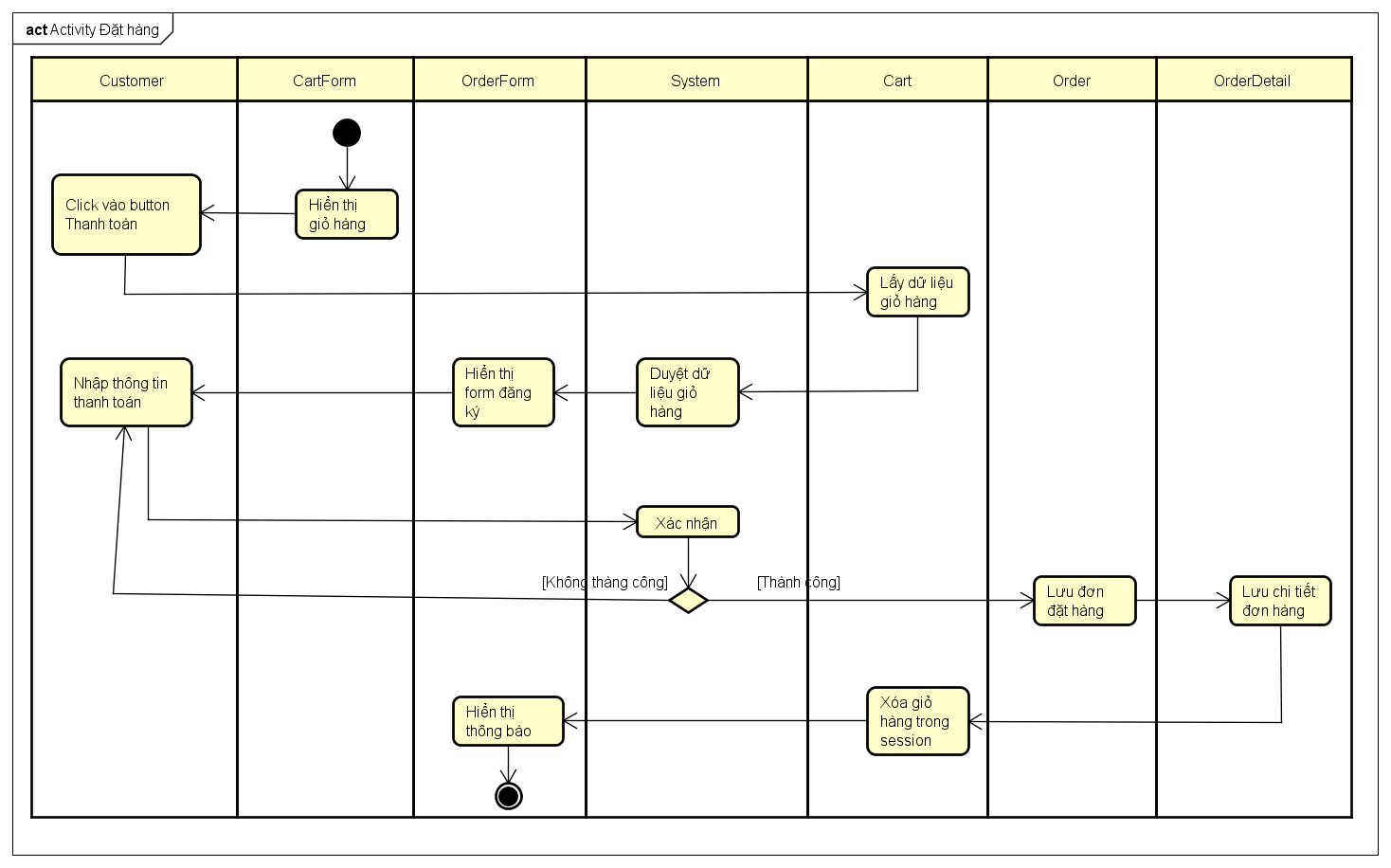
Hình 3.4 Biểu đồ hoạt động UC Thêm sản phẩm vào giỏ hàng

1. Biểu đồ hoạt động Usecase Xem giỏ hàng



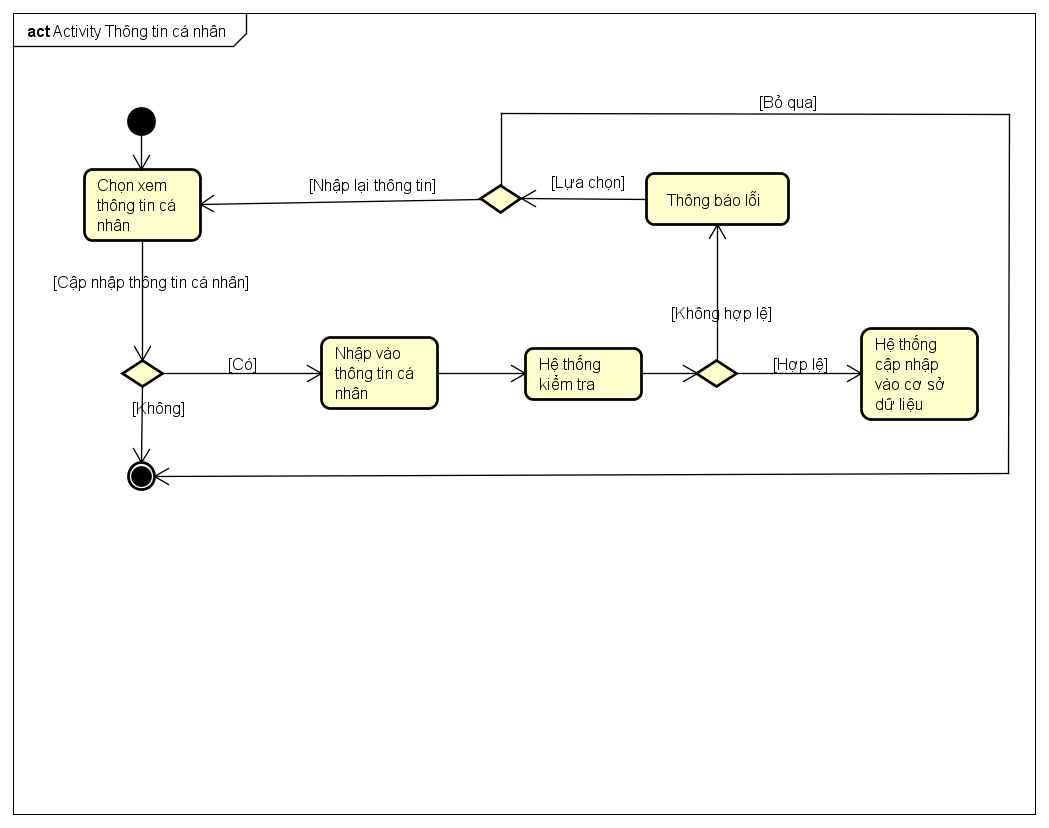
Hình 3.5 Biểu đồ hoạt động UC Xem giỏ hàng

1. Biểu đồ hoạt động Usecase Đặt hàng



Hình 3.6 Biểu đồ hoạt động UC Đặt hàng

1. Biểu đồ hoạt động Usecase Quản lý thông tin cá nhân



Hình 3.7 Biểu đồ hoạt động UC Quản lý thông tin cá nhân

## Thiết kế kiến trúc

### Mô hình MVC

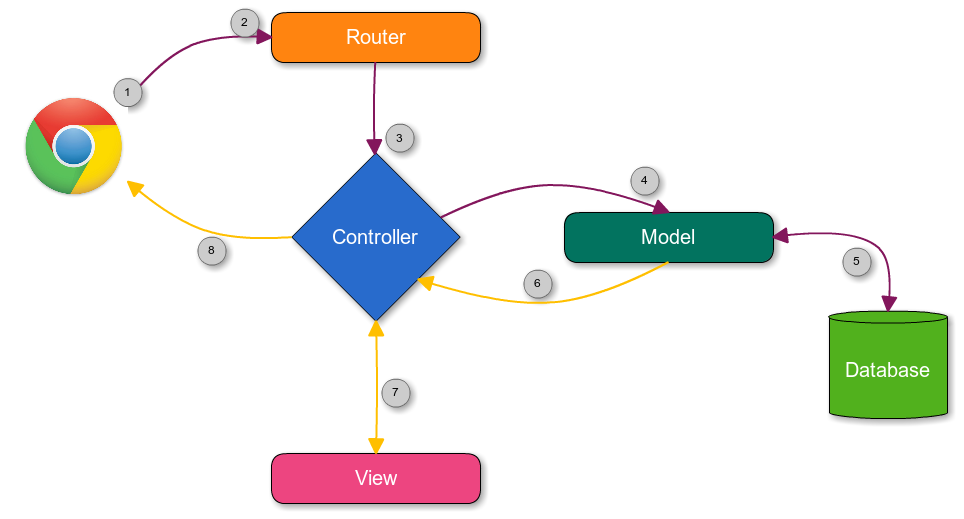
Mô hình MVC (model, controller, view) là mô hình chuẩn cho ứng dụng web được sử dụng nhiều nhất ngày nay. Mô hình MVC được sử dụng lần đầu tiên trong Smalltalk, sau đó được sử dụng phổ biến trong ngôn ngữ lập trình Java. Hiện nay, đã có hơn hàng chục PHP framework dựa trên mô hình này.

Mô hình MVC là viết tắt của 3 chữ Model, View, Controller. Mô hình này tách một ứng dụng web ra làm 3 thành phần đảm nhiệm chức năng tách biệt, thuận tiện cho việc xử lý và bảo trì

**Model** : Chịu trách nhiệm quản lý dữ liệu, nó lưu trữ và truy xuất các thực thể từ cơ sở dữ liệu như mysql, sql server, postresSQL,… đồng thời chưa các logic được thực thi bởi ứng dụng

**View** : Chịu trách nhiệm hiển thị dữ liệu đã được truy xuất từ model theo một format nào đó theo ý đồ của lập trình viên. Cách sử dụng của View tương tự như các module templates thường thấy trong các ứng dụng web phổ biến như WordPress, Joomla,…

**Controller** : trung gian, làm nhiệm vụ xử lý cho model và view tương tác với nhau. Controller nhận request từ client, sau đó gọi các model để thực hiện các hoạt động được yêu cầu và gửi ra ngoài View. View sẽ chịu trách nhiệm format lại data từ controller gửi ra và trình bày dữ liệu theo 1 định dạng đầu ra (html).



Hình 3.8: Mô hình MVC trong Laravel

Từ hình trên, có thể thấy luồng đi của Laravel:

* Người dùng sử dụng trình duyệt web để truy cập vào địa chỉ website.
* Router sẽ phân tích xem đường dẫn truy cập đến Controller nào.
* Controller sẽ truy cập vào Model để lấy dữ liệu từ Database, sau đó trả về Controller.
* Controller đổ dữ liệu đó ra view cho người dùng.

**Ưu điểm:**

* Hệ thống phân ra từng phần nên dễ dáng phát triển
* Chia thành nhiều modun nhỏ nên nhiều người có thể làm chung dự án
* Vấn đề bảo trì cũng tương đối tốt, dễ nâng cấp
* Dễ dàng rà soát lỗi trong quá trình xây dựng

**Nhược điểm:**

* Hệ thống sẽ chạy chậm hơn PHP thuần, tuy nhiên nó ko phải là vấn đề
* Xây dựng cầu kì và mất thời gian để xây dựng thư viện, cấu trúc

### Thiết kế kiến trúc

1. Biểu đồ trình tự cho Usecase Thêm sản phẩm vào giỏ hàng
2. Biểu đồ trình tự cho Usecase Đặt hàng

## Thiết kế chi tiết

### Thiết kế giao diện

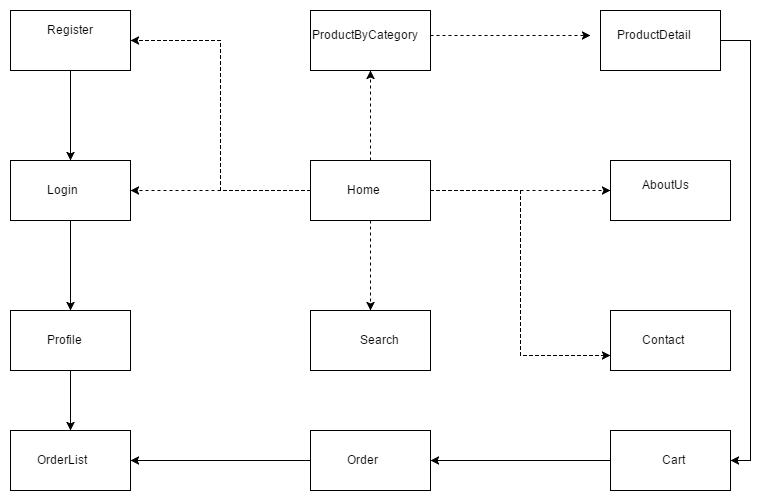
Hệ thống chỉ có tác nhân là người dùng nên hệ thống chỉ phân tích thiết kế giao diện đồ họa người dùng.

1. Danh sách một số giao diện

|  |  |  |
| --- | --- | --- |
| STT | Tên | Mô tả |
| 1 | Home | Màn hình trang chủ khi truy cập vào website |
| 2 | ProductByCategory | Danh sách sản phẩm theo thể loại |
| 3 | ProductDetail | Chi tiết sản phẩm |
| 4 | Contact | Màn hình thông tin liên hệ của cửa hàng, đồng thời cũng là khách hàng gửi tin nhắn phản hồi về cho cửa hàng |
| 5 | Search | Hiển thị kết quả tìm kiếm |
| 6 | Cart | Màn hình hiển thị giỏ hàng |
| 7 | Order | Hiển thị đơn đặt hàng và thông tin thanh toán |
| 8 | Login | Màn hình đăng nhập |
| 9 | Register | Màn hình đăng ký |
| 10 | Profile | Thông tin cá nhân của người dùng |
| 11 | OrderList | Lịch sử mua hàng |

Bảng 3.23 Danh sách một số giao diện

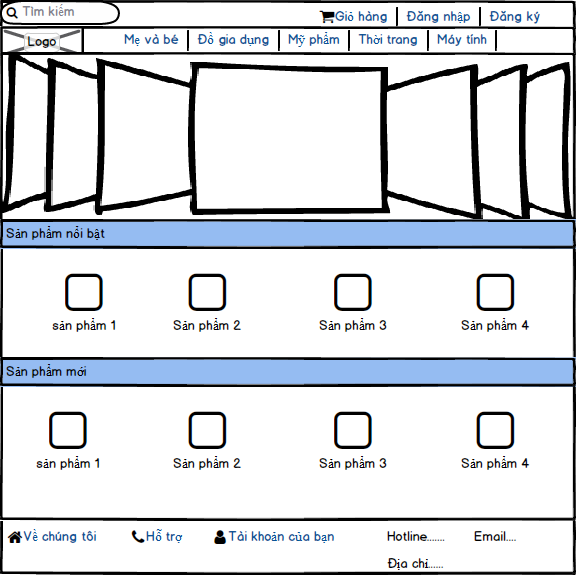
1. Biểu đồ dịch chuyển màn hình



Hình 3.9 Biểu đồ dịch chuyển màn hình

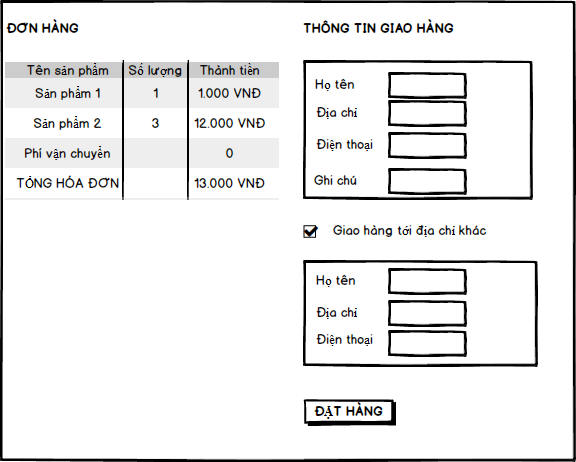
1. Mockup một số giao diện

* Màn hình trang chủ



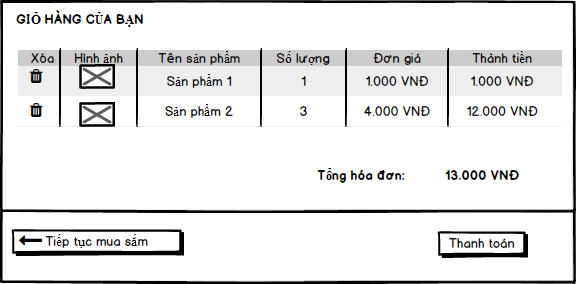
Hình 3.10 Màn hình Mockup Trang chủ

* Màn hình thanh toán



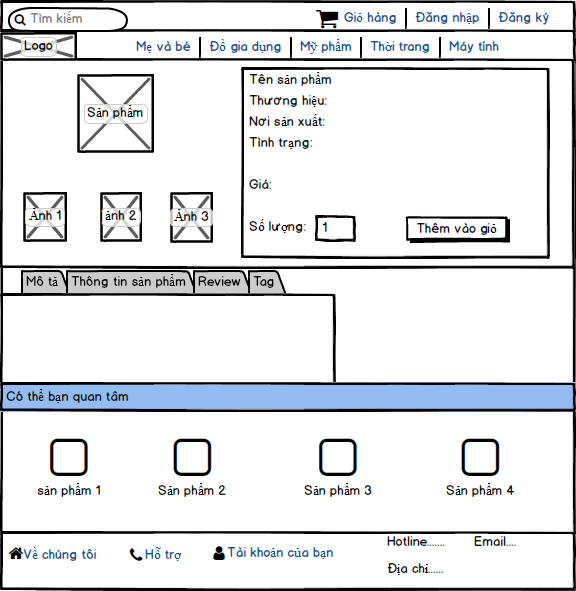
Hình 3.11 Màn hình Mockup Thanh toán

* Màn hình giỏ hàng



Hình 3.12 Màn hình Mockup Giỏ hàng

* Màn hình chi tiết sản phẩm



Hình 3.13 Màn hình Mockup Chi tiết sản phẩm

1. Đặc tả màn hình

* Giỏ hàng

|  |  |  |  |
| --- | --- | --- | --- |
| Mô tả màn hình | Màn hình hiển thị danh sách các sản phẩm khách hàng muốn mua | | |
|  | Các thành phần | Hoạt động | Chức năng |
| Khu vực hiển thị danh sách các sản phẩm và tổng hóa đơn |  | * Hiển thị thông tin những sản phẩm được khách hàng chọn mua * Button xóa, sửa thực hiện trên form * Hiển thị tổng hóa đơn hiện tại của giỏ hàng |
| Button “Thanh toán” | Click | Hiển thị màn hình thanh toán |
| Button “Tiếp tục mua sắm” | Click | Trở về màn hình trang chủ |

Bảng 3.24 Đặc tả màn hình giỏ hàng

* Chi tiết sản phẩm

|  |  |  |  |
| --- | --- | --- | --- |
| Mô tả màn hình | Hiển thị thông tin chi tiết một sản phẩm | | |
|  | Các thành phần | Hoạt động | Chức năng |
| Khu vực hiển thị tất cả thông tin của một sản phẩm |  | * Hiển thị tất cả ảnh và thông tin cơ bản của sản phẩm, * Tab bar: hiển thị các thông tin thêm (mô tả chi tiết, tag, review) về sản phẩm. |
| Khu vực các sản phẩm liên quan |  | Hiển thị các sản phẩm trong cùng thể loại với sản phẩm đang xem. |
| Ô nhập số lượng | Nhập số | Nhập số lượng muốn mua cho sản phẩm đó. |
| Button “Thêm vào giỏ” | Click | * Thêm sản phẩm vào giỏ hàng, * Hiển màn hình giỏ hàng. |

Bảng 3.25 Đặc tả màn hình chi tiết sản phẩm

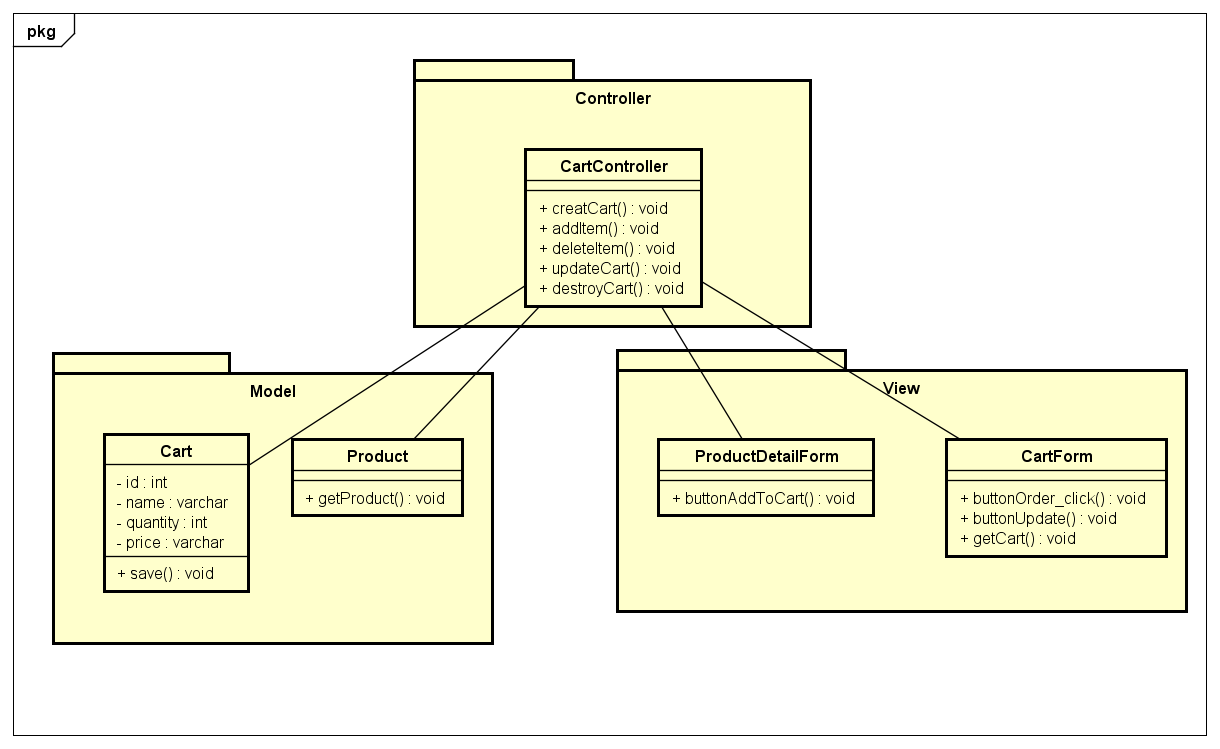
* Thanh toán

|  |  |  |  |
| --- | --- | --- | --- |
| Mô tả màn hình | Hiển thị đơn hàng, thông tin giao hàng của khách hàng | | |
|  | Các thành phần | Hoạt động | Chức năng |
| Đơn hàng |  | Hiển thị các sản phẩm đặt hàng, tổng giá trị hóa đơn. |
| Thông tin giao hàng |  | * Nếu khách hàng đang ở trạng thái đăng nhập thò toàn bộ thông tin của khách hàng sẽ được hiển thị ra, * Nếu khách hàng không đăng nhập hoặc chưa có tài khoản thì hệ thống sẽ tự động lưu như là một khách hàng mới, * Khách hàng muốn giao hàng tới địa chỉ khác với thông tin của bản thân thì nhập thông tin vào phần dưới ô checkbox. |
| Checkbox “Giao hàng tới địa chỉ khác” | Check | Hiển thị bảng input nhập thông tin giao hàng . |
| Button “Đặt hàng” | Click | * Đặt hàng * Hiển thị màn hình trang chủ |

Bảng 3.26 Đặc tả màn hình thanh toán

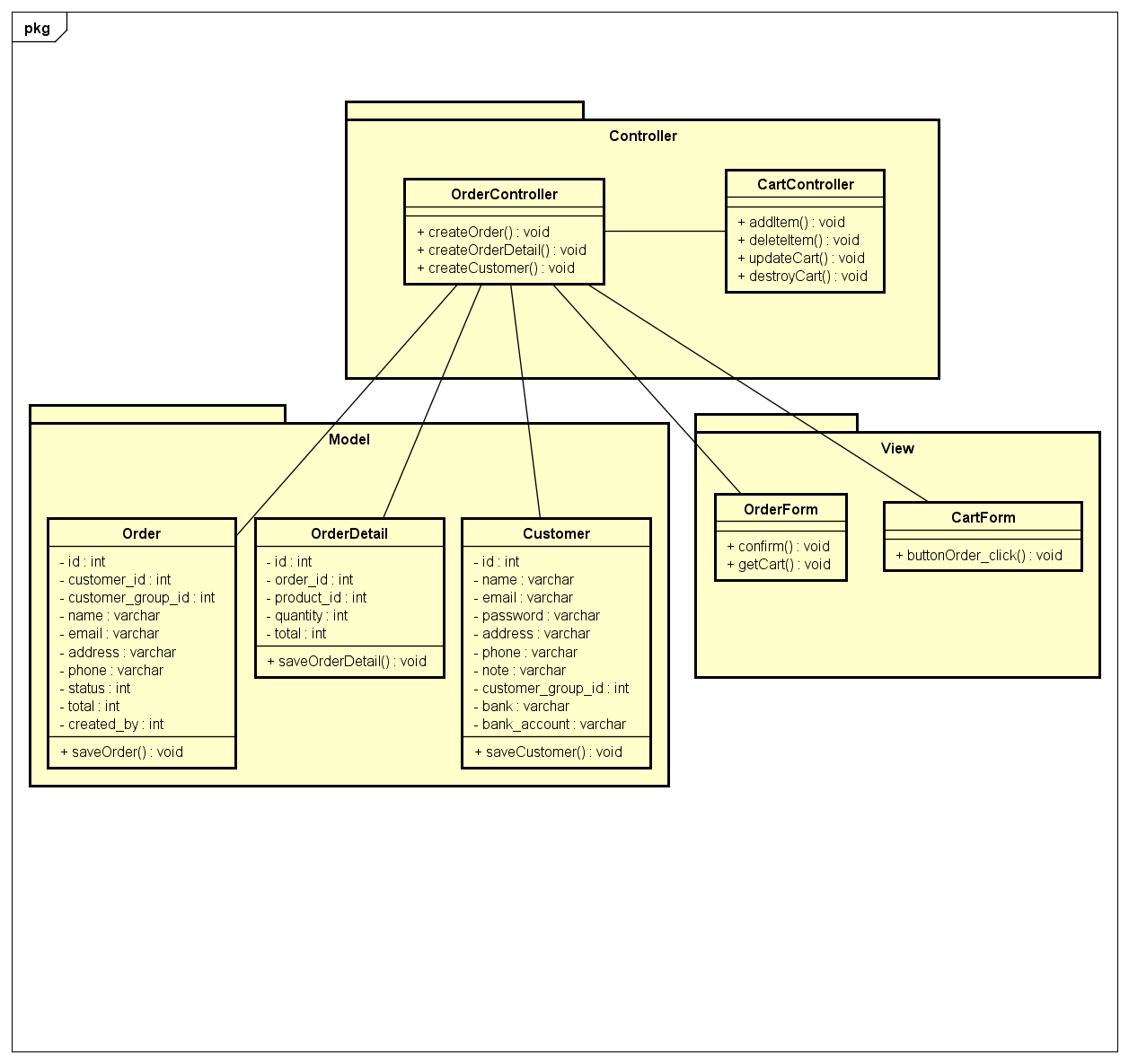
### Thiết kế lớp

a) Biểu đồ lớp chức năng quản lý giỏ hàng



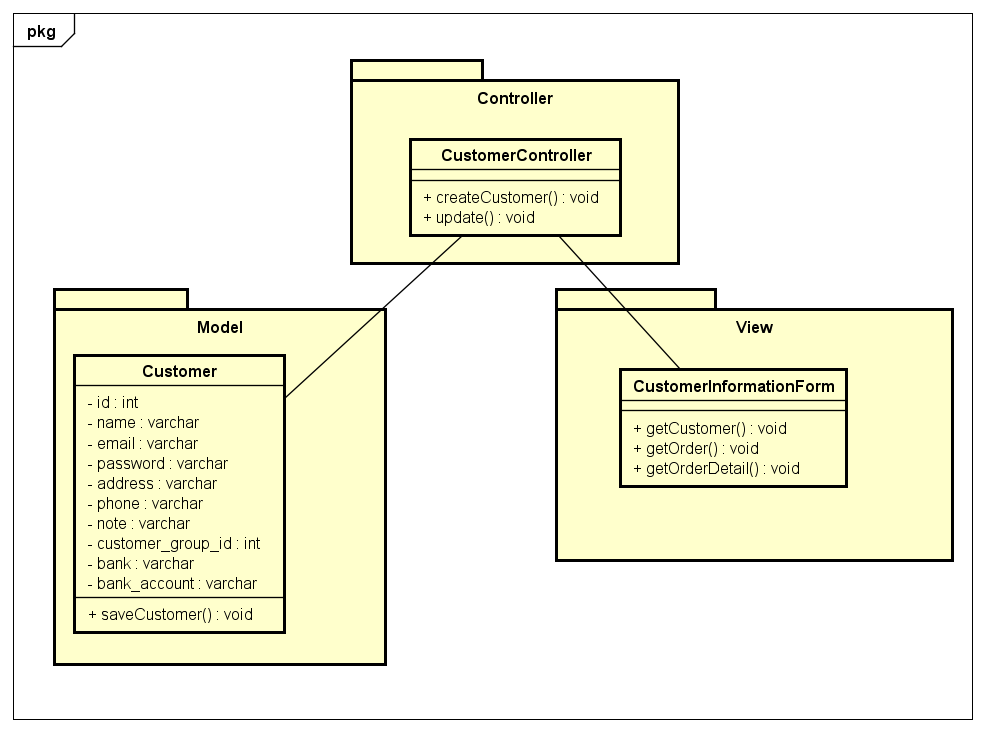
Hình 3.14 Biểu đồ lớp chức năng quản lý giỏ hàng

b) Biểu đồ lớp chức năng thanh toán



Hình 3.15 Biểu đò lớp chức năng thanh toán

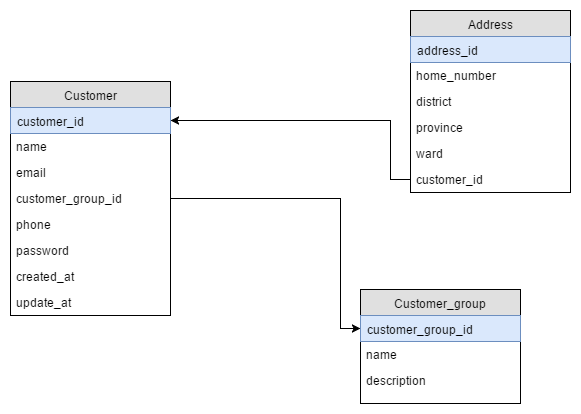
c) Biểu đồ lớp chức năng quản lý thông tin cá nhân



Hình 3.16 Biểu đồ lớp chức năng quản lý thông tin cá nhân

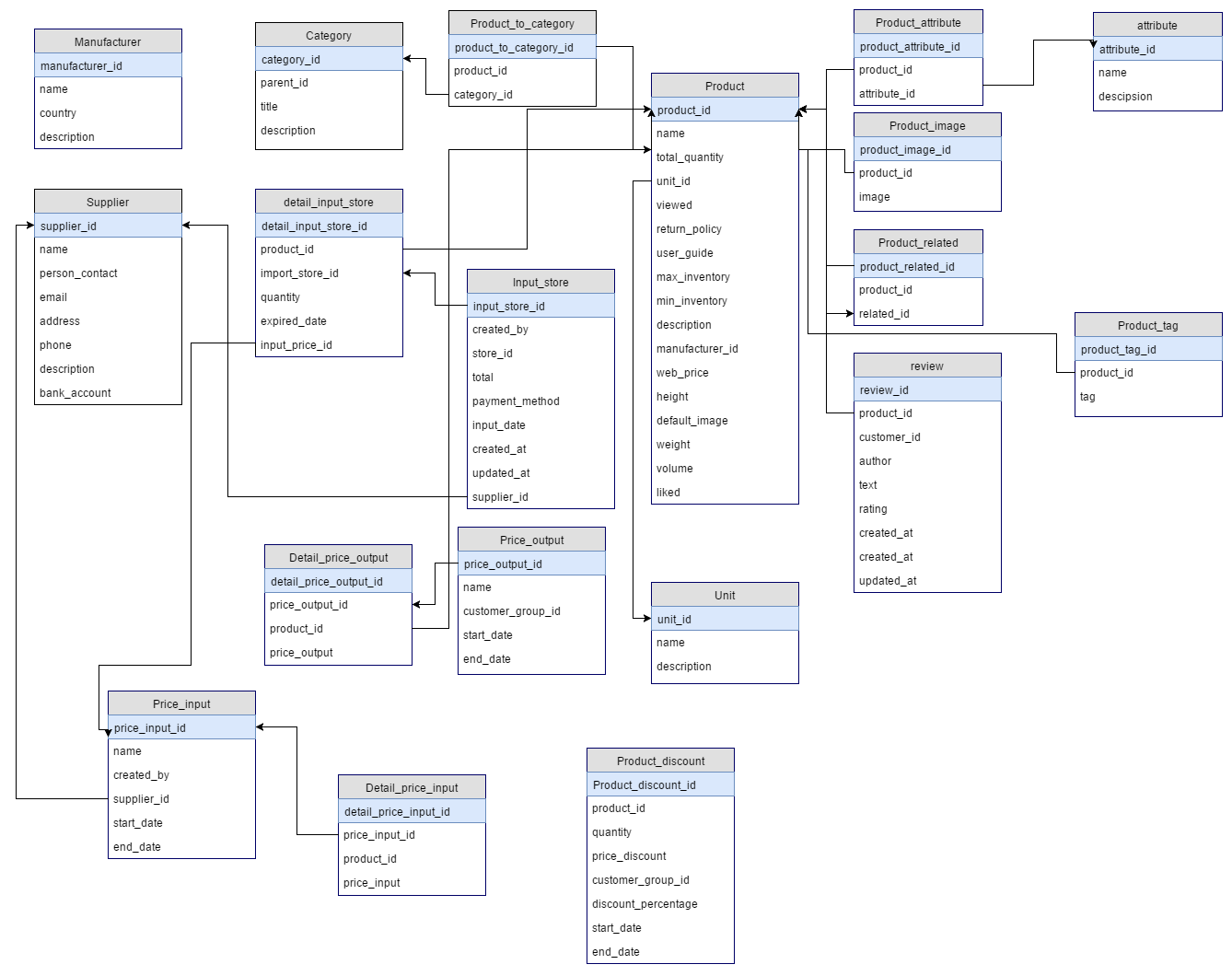
### Thiết kế cơ sở dữ liệu

1. Khách hàng: Customer



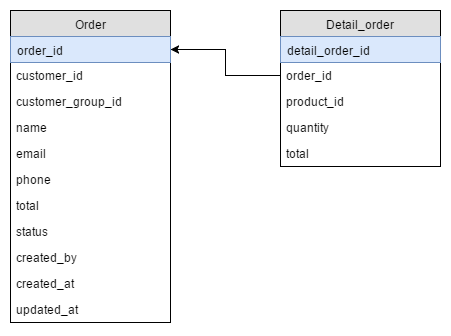
Hình 3.17 Cơ sở dữ liệu phần khách hàng

1. Sản phẩm: Product



Hình 3.18 Cơ sở dữ liệu phần sản phẩm

1. Đặt hàng: Order



Hình 3.19 Cơ sở dữ liệu phần đặt hàng

# NHỮNG VẤN ĐỀ KHÓ KHĂN VÀ GIẢI PHÁP

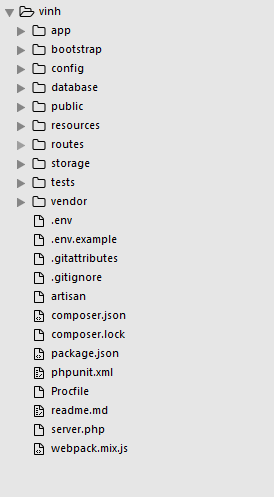
## Công nghệ sử dụng – Laravel Framewok

#### Khái niệm

Laravel là một Open source PHP Framework miễn phí được phát triển bởi Taylor Otwell ra mắt lần đầu vào tháng 6 năm 2011, được thiết kế dựa trên mô hình MVC (Model, Controller, View).

Hiện tại, Laravel là PHP Framework được sử dụng phổ biến nhất trên thế giới vì những ưu điểm vượt trội tận dụng các kỹ thuật Design Pattern, các công nghệ mới nhất của PHP và rất dễ dàng tiếp cận và sử dụng nó. Điểm yếu của Laravel là không hỗ trợ các phiên bản PHP cũ vì thế các website có nền tảng PHP version thấp hơn khá là khó khi có ý định chuyển sang Laravel và phải luôn cập nhật các thông tin mới nhất về PHP để áp dụng vào Laravel.

**Cấu trúc thư mục Laravel phiên bản 5.4**



Hình 4.1 Cấu trúc thư mục Laravel

* **app** là thư mục chứa tất cả các thư mục, các tập tin php, các class php, thư viện, models
  + **Console** thư mục chứa các tập tin định nghĩa các lệnh thực thi trên Artisan.
  + **Excerption** thư mục chứa các tập tin quản lý, điều hướng lỗi.
  + **Http**
    - **Controllers** là thư mục chứa các tập tin controllers
    - **Middleware** là thư mục chứa các tập tin lọc và ngăn chặn các requests.
    - **Kernel.php**là tập tin cấu hình, định nghĩa Middleware hoặc nhóm Middleware.
  + **Providers** chứa các providers mình sẽ nói rõ về nó trong các bài nâng cao.
* **bootstrap** thư mục chứa tập tin điều hướng khởi động hệ thống, thường thì chúng ta sẽ không làm gì đến nó.
* **config** chứa mọi tập tin cấu hình của Laravel
* **database** chứa các thư mục tập tin về CSDL
  + **migrations** chứa các tập tin định nghĩa khởi tạo và sử bảng.
  + **seeds** chứa các tập tin định nghĩa dữ liệu insert vào database.
  + **factories** chứa các tập tin định nghĩa các cột bảng dữ liệu để tạo ra các dữ liệu ảo phục vụ cho tests.
* **public** chính là webroot người dùng sẽ truy cập vào đây, đây cũng là nơi chứa các tập tin css, js, image.
* **resources** chứa các tập tin giao diện (js, css, less, sass, coffeescript,...), views, ngôn ngữ.
* **storage** chứa các tập tin hệ thống như upload, cache, session, cookie, log...
* **routes** là thư mục chứa các tập tin định nghĩa các router, xử lý router hoặc điều hướng router (tức là URL, laravel không tự đặt url theo kiểu example.com/controller/action/value mà chúng ta phải tự định nghĩa chúng) bao gồm 3 loại là web, api và console.
* **tests** chứa các tập tin định nghĩa tests.
* **vendor** thư mục của composer.
* **.env** và **.env.example** là 2 tập tin cấu hình chính của laravel như key app, tên app, url app, email,  env mode, CSDL hay bật tắt debug.
* **composer.json**, **composer.lock** tập tin của composer.
* **package.js** tập tin cấu hình của nodejs chứa các package cần thiết cho projects.
* **gulpfile.js** là tập tin gulp builder.
* **phpunit.xml** là tập tin xml của phpunit dùng để testing project.
* **server.php** là tập tin để artisan trỏ đến tạo server khi gõ lệnh php artisan server
* **artisan** tập tin thực thi lệnh của Laravel, cũng là tập tin mà chúng ta tương tác nhiều nhất.

#### Tại sao chọn Laravel Framework

**Emerging Quickly**: Hiện tay, có nhiều công ty lập trình đã quyết định chọn Laravel do các tính năng tuyệt vời của nó khi xử lý các dự án phát triển web lớn. Theo báo cáo xu hướng của Google, Laravel là PHP framework phổ biến nhất hiện nay.

**Open Source:** Nó mở và hoàn toàn miễn phí, không cần quan tâm đến việc trả phí khi ứng dụng web được phát triển lên. Điều cần làm là hiểu nó và code những tính năng tuyệt với mà thôi.

**Templete engine:** Laravel sử dụng Blade template và các công cụ như Elixir dễ dàng quản lý các assets( biên dịch, ghép nối, nén ...), cũng như chia nhỏ code HTML ra thành nhiều phần. Giúp cho việc thiết kế cũng như quản lý các assets trở nên vô cùng đơn giản.

**Modular:** Laravel được xây dựng dựa trên hơn 20 thư viện khác nhau. Hiểu được cách thiết kế framework khiến các nhà phát triển hoàn toàn có thể đóng góp cho framework cũng như mở rộng chúng một cách dễ dàng.

**MVC Architecture Support:** Laravel sử dụng MVC pattern để xây dựng framework. Nó đảm bảo rằng các code được trình bày logic và hợp lý. Dễ dàng để có thể hiểu được nó. Vì vậy nó làm cho dự án web được trở nên dễ quản lý cũng như phát triển rất rất nhiều.

**Libraries and configuration:** Nó hỗ trợ nhiều môi trường phát triển và tự điều chỉnh dựa theo nền tảng tương ứng mà ứng dụng đang chạy. Tất cả điều này có thể là do các thư viện được tích hợp bên trong. Các thư viện của Laravel được hỗ trợ bởi các tính năng hoàn toàn tự động mà ở các PHP framework thông dụng khác không có.

## Tìm kiếm bằng Full Text Search

#### a) Full Text Search là gì?

* Full-Text search (FTS) là cách tự nhiên nhất để tìm kiếm thông tin.
* Full-Text search đề cập đến một công nghệ tìm kiếm thông tin mới trong cơ sở dữ liệu.
* Full-Text search (FTS) cho phép tìm kiếm ký tự, chuỗi, cụm từ trong cơ sở dữ liệu thay vì tìm kiếm trên từng cột dữ liệu của bảng.
* Full-Text Search cho phép tạo chỉ mục một cách uyển chuyển và nhanh chóng dễ truy vấn với từ khóa trên chuỗi dữ liệu trong cơ sở dữ liệu.

#### b) Inverted Index

Nhắc đến Full Text Search ta không thể không nhắc đến Inverted Index, đây là một kĩ thuật index trong database, một kĩ thuật quan trọng trong Full Text Search.

Inverted Index là một cấu trúc dữ liệu, nhằm mục đích map giữa các từ hay chữ số và các document chứa chúng.

Ví dụ:

Document1 = "Vinh is Developer"  
Document2 = "Developer C#"  
Document3 = "Vinh is Developer and Student"

Và Inverted Index được lưu như sau:

"Vinh" => {Document1,Document3}  
"is" => {Document1, Document3}  
"Developer" => {Document1, Document2, Document3}  
"C#" => {Document2}  
"and" => {Document3}  
"Student" => {Document3}

#### Cách hoạt động

Vậy khi tìm kiếm “Vinh is Developer” thì hệ thống nó hoạt động ra sao?

Đầu tiên hệ thống nó sẽ tách chuỗi tìm kiếm ra thành các từ Vinh, is, Developer. Sau đó sẽ tìm trong inverted index và được các tập hợp như sau:

{Document1,Document3}  
{Document1,Document3}  
{Document1, Document2, Document3}

Sau đó union các tập hợp lại

{Document1,Document3} union {Document1,Document3} union {Document1, Document2, Document3} => {Document1}

Cũng vì nó tách chuỗi ra và tìm kiếm trên index như thế, nên Full Text Search rất linh hoạt trong tìm kiếm, vì nó không quan tâm đầu vào có thứ tự các từ như thế nào. Ví dụ “Vinh is Developer”, “Developer is Vinh”, “Vinh Developer is” đều cho kết quả như nhau và độ phức tạp cũng như nhau.

* Ưu điểm

Vì Full Text Search hoạt động dựa trên index và tránh tìm kiếm tuyến tính, vì thế tốc độ của nó tối ưu hơn việc dùng LIKE Query rất nhiều, nhưng nó ít được biết đến với những người mới. Bởi vì khi dùng nó thì đòi hỏi phải có index ở database

* Nhược điểm

Bất kì kĩ thuật nào cũng đều có nhược điểm của nó. Và với Full Text Search, nếu bỏ qua việc phải dùng index thì nhược điểm của nó nằm ở việc Index không phải lúc nào cũng tốt

#### Ứng dụng của Full text Search

Trong đồ án ứng dụng full text search để tìm kiếm sản phẩm dựa vào 2 trường name và description trong database.

# XÂY DỰNG PHẦN MỀM VÀ TRIỂN KHAI

## Xây dựng hệ thống

### Công cụ lập trình

#### Sublime Text

Sublime Text là một text editor được viết bằng ngôn ngữ lập trình Python và có thể được sử dụng trên nhiều nền tảng hệ điều hành khác nhau như Windows, Mac, Linux. Sublime Text hỗ trợ nhiều ngôn ngữ lập trình và là một phần mềm có thể được sử dụng miễn phí tuy nhiên cũng có phiên bản trả phí của Sublime Text.

Sublime Text khác với các IDE khác ở đặc điểm nhẹ, linh hoạt trong tùy biến và cung cấp nhiều tính năng rất tiện lợi khi lập trình.

Với các IDE khác thì thông thường sẽ cần phải chờ một khoảng thời gian vào khoảng 10 giây khi khởi động. Với Sublime Text chỉ phải đợi khoảng từ 1 tới 2 giây.

Ngoài ra Sublime Text có thể tùy biến thông qua việc thêm các plugin hoặc tạo các automation task bằng việc tùy biến các đoạn snippet.

Sublime Text hỗ trợ nhiều tính năng hữu ích khi lập trình như: multiple selection, goto anything, command pallete...

#### Git – Công cụ quản lý mã nguồn

Git là tên gọi của một Hệ thống quản lý phiên bản phân tán (Distributed Version Control System – DVCS) là một trong những hệ thống quản lý phiên bản phân tán phổ biến nhất hiện nay. DVCS nghĩa là hệ thống giúp mỗi máy tính có thể lưu trữ nhiều phiên bản khác nhau của một mã nguồn được nhân bản (clone) từ một kho chứa mã nguồn (repository), mỗi thay đổi vào mã nguồn trên máy tính sẽ có thể ủy thác (commit) rồi đưa lên máy chủ nơi đặt kho chứa chính. Và một máy tính khác (nếu họ có quyền truy cập) cũng có thể clone lại mã nguồn từ kho chứa hoặc clone lại một tập hợp các thay đổi mới nhất trên máy tính kia. Trong Git, thư mục làm việc trên máy tính gọi là Working Tree.

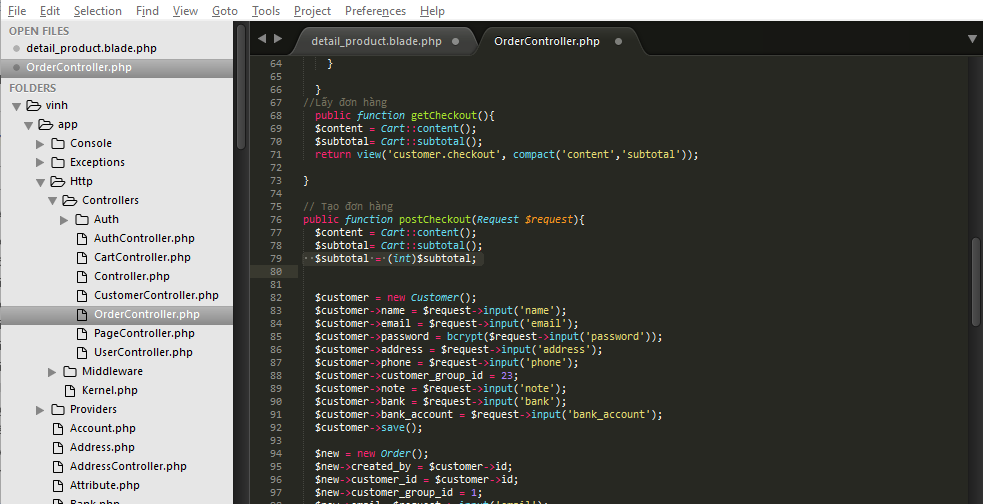
Ngoài ra, có một cách hiểu khác về Git đơn giản hơn đó là nó sẽ lưu lại các phiên bản của những lần thay đổi vào mã nguồn và có thể dễ dàng khôi phục lại dễ dàng mà không cần copy lại mã nguồn rồi cất vào đâu đó. Và một người khác có thể xem các thay đổi ở từng phiên bản, họ cũng có thể đối chiếu các thay đổi rồi gộp các phiên bản lại. Cuối cùng là tất cả có thể đưa các thay đổi vào mã nguồn của mình lên một kho chứa mã nguồn.

Cơ chế lưu trữ phiên bản của Git là nó sẽ tạo ra một “ảnh chụp” (snapshot) trên mỗi tập tin và thư mục sau khi commit, từ đó nó cho phép tái sử dụng lại một ảnh chụp nào đó mà có thể hiểu đó là một phiên bản. Đây cũng chính là lợi thế của Git so với các DVCS khác khi nó không “lưu cứng” dữ liệu mà sẽ lưu với dạng snapshot.

### Xây dựng hệ thống bằng Laravel Framework

Trên hình 5.1 là ví dụ khi áp dụng laravel framework để xây dựng hệ thống:

* Phương thức GET gửi dữ liệu từ client lên server thông qua các paramter( tham số) trên URL, từ đó server sẽ phân tích dữ liệu để thực thi một hành động nào đó
* Khác với phương thức GET phương thức POST không gửi dữ liệu thông qua paramter trên URL, nên chúng ta không thể nhìn thấy được dữ liệu đang được gửi là gì.



Hình 5.1 Xây dựng hệ thống bằng Laravel

## Triển khai và chạy thử

### Triển khai

Sau khi tạo xong một project PHP Laravel MVC. Muốn publish lên host chúng ta phải thực hiện theo các bước sau:

**Bước 1:** Truy cập remote máy từ xa với tài khoản

ssh thai8763@43.239.223.142 -p 50142

**Bước 2:** Cài các gói phần mềm cần thiết như

* Apache2
* Mysql
* Phpmyadmin
* git

Cấu hình Apache2

**Bước 3:** Tạo thư mục larose trong /var/www/html

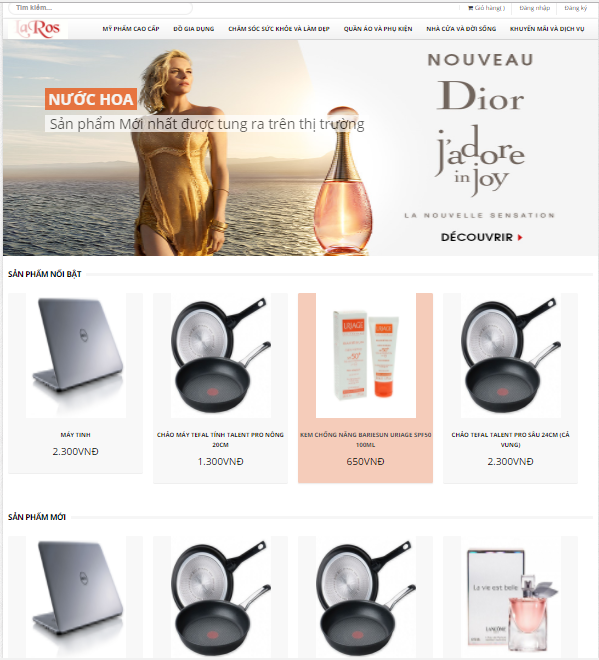
Clone project vào thư mục larose

**Bước 4:** Cấu hình port trên apache để vào được web larose

### Chạy thử

#### a) Màn hình trang chủ

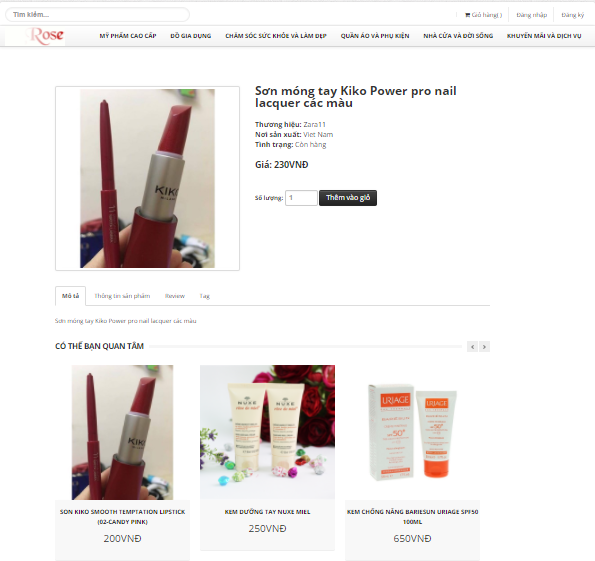
Hình 5.2 là màn hình trang chủ của website, màn hình trang chủ bao gồm danh sách các thể loại mặt hàng, slide giới thiệu những đợt khuyến mãi hay các sản phẩm bán chạy. Ngoài ra, trang chủ còn hiển thị những sản phẩm nổi bật và sản phẩm mới.



Hình 5.2: Màn hình trang chủ

#### b) Màn hình chi tiết một sản phẩm

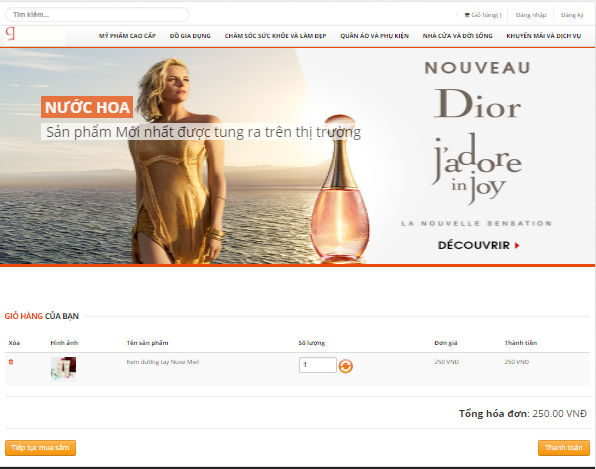
Trên hình 5.3 là giao diện màn hình chi tiết một sản phẩm. Ngoài các thuộc tính và hình ảnh của sản phẩm thì màn hình còn hiển thị những sản phẩm liên quan với sản phẩm đang xem.



Hình 5.3: Màn hình chi tiết sản phẩm

#### Màn hình giỏ hàng

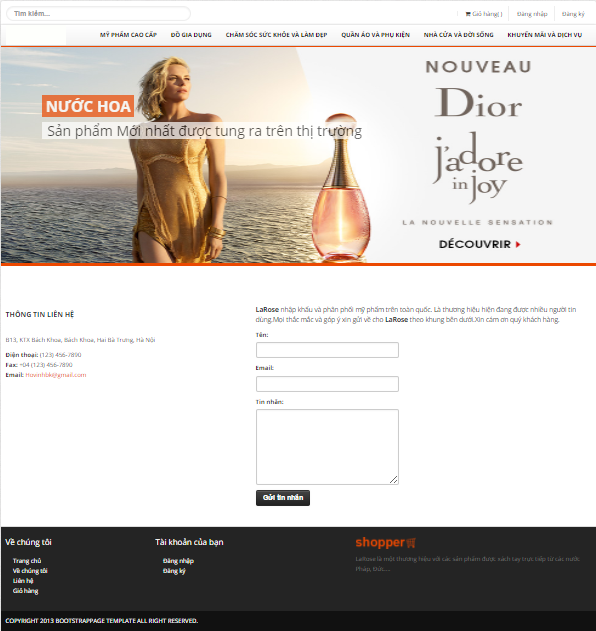
Hình 5.4 là giao diện màn hình giỏ hàng. Màn hình hiển thị các sản phẩm trong giỏ hàng, tổng hóa đơn tạm thời. Khách hàng có thể thanh toán đặt hàng bằng cách bấm vào button “Thanh toán”, hoặc nếu muốn tiếp tục mua sắm thì click vào button “Tiếp tục mua sắm”, màn hình trang chủ sẽ hiển thị.



Hình 5.4: Màn hình giỏ hàng

#### Màn hình liên hệ

Hình 5.5 dưới đây là màn hình liên hệ dành cho khách hàng nào muốn gửi phản hồi của mình về cửa hàng, khách hàng nhập tên, email và nội dung, sau khi khách hàng gửi đi thì nội dung được gửi đến mail đã được cấu hình trong hệ thống.



Hình 5.5: Màn hình liên hệ

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

## Kết luận

Xây dựng Website bán hàng không phải là một vấn đề mới mẻ nhưng nó cũng không phải là vấn đề quá cũ, nó mang tính thực tế cao, nhất là trong giai đoạn hiện nay – khi nhu cầu cảu khách hàng (đặc biệt là giới doanh nhân) muố tìm hiểu các sản phẩm, và tìm một nơi tin cậy để mua hàng. Dưới sự hướng dẫn của Cô Nguyễn Thị Thu Trang, em thực hiện đề tài này nhằm đưa ra một giải pháp phù hợp cho việc đưa hình ảnh cửa hàng đến với mội người rộng hơn. Do sự hạn hẹp về trình độ nên Website em xây dựng được trong đồ án này chỉ mới đáp ứng được những yêu cầu cơ bản, nếu đưa vào thực tế sử dụng thì chỉ đáp ứng được chức năng mua bán, đặt hàng cơ bản.

Với đồ án này, em đã hoàn thành các yêu cầu đặt ra, bao gồm:

* Giao diện: thiết kế trên Photoshop và ký thuật tạo hình ảnh động, tăng tính thẩm mỹ và tạo tính sống động, thu hút khách đến thăm trang web,
* Khách hàng có thể đăng ký thành viên, quản lý thông tin cá nhân và xem lịch sử mua hàng.
* Khách hàng có thể mua và đặt hàng mà không cần đăng nhập,
* Tạo chức năng quản lý giỏ hàng
* Khách hàng có thể liên hệ hoặc phản hồi tới cửa hàng,
* Tất cả các form nhập liệu đều được validate cho các trường.

## Hướng phát triển

Hướng phát triển của Website là hướng đến một thương mại điện tử với đầy đủ các chức năng thanh toán tiền qua mạng. Nhưng vì trong nước ta hiện nay vấn đề thanh toán tiền quan mạng còn gặp nhiều khó khăn và chưa phổ biến nhiều nên chức năng của Website để hỗ trợ còn hạn chế.

Xây dựng chức năng chat trực tiếp với khách hàng trên Website.

TÀI LIỆU THAM KHẢO

[1] <https://laravel.com/docs/5.4>

[2] <http://getbootstrap.com/getting-started/>

[3] Bài giảng môn “Thiết kế và xây dựng phần mềm’’– TS. Nguyễn Thị Thu Trang

[4] <https://www.w3schools.com/>

[5] <https://www.javascript.com/learn/javascript/strings>