TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.Hồ Chí Minh

**KHOA CÔNG NGHỆ THÔNG TIN**



Đề tài: Website giới thiệu, bán xe đạp trực tuyến

**BÁO CÁO BÀI TẬP LỚN LẬP TRÌNH WWW (JAVA)**

SVTH: Nguyễn Văn Trung - 17071691

Nguyễn Hưng Phát - 17071261

Nguyễn Minh Hiếu 17040391

GV HD: THS. PHẠM QUẢNG TRI

TP.Hồ Chí Minh - Năm 2021

# MỤC LỤC

[MỤC LỤC 2](#_Toc73469111)

[CHƯƠNG 1 : GIỚI THIỆU 5](#_Toc73469112)

[1.1 Tổng quan 5](#_Toc73469113)

[1.2 Mục tiêu đề tài 5](#_Toc73469114)

[1.3 Phạm vi đề tài 6](#_Toc73469115)

[1.4 Mô tả yêu cầu chức năng 6](#_Toc73469116)

[CHƯƠNG 2 : CƠ SỞ LÝ THUYẾT 8](#_Toc73469117)

[2.1 Ngôn ngữ Spring MVC 8](#_Toc73469118)

[2.1.1 Khái niệm về mô hình MVC 8](#_Toc73469119)

[2.1.2 Khái niệm về Spring 9](#_Toc73469120)

[2.1.3 Khái niệm về Spring MVC 9](#_Toc73469121)

[2.1.4 Giới thiệu về ngôn ngữ Spring MVC 11](#_Toc73469122)

[2.1.5 Spring security 15](#_Toc73469123)

[2.1.6 Cơ sở dữ liệu SQL Server 17](#_Toc73469124)

[CHƯƠNG 3 : PHÂN TÍCH 19](#_Toc73469125)

[3.1 Phân tích yêu cầu bằng UML 19](#_Toc73469126)

[3.1.1 Usecase tổng quát 19](#_Toc73469127)

[3.1.2 Danh sách tác nhân và mô tả 19](#_Toc73469128)

[3.1.3 Danh sách các tình huống hoạt động (Use cases) 20](#_Toc73469129)

[3.1.4 Đặc tả các yêu cầu của chức năng 20](#_Toc73469130)

[3.2 Class diagram 47](#_Toc73469131)

[CHƯƠNG 4 : THIẾT KẾ VÀ HIỆN THỰC 49](#_Toc73469132)

[4.1 Hiện thực giao diện 49](#_Toc73469133)

[4.1.1 Giao diện Admin 49](#_Toc73469134)

[4.1.2 Giao diện Khách hàng. 50](#_Toc73469135)

[CHƯƠNG 5 : KẾT LUẬN 55](#_Toc73469136)

[5.1 Kết quả đạt được 55](#_Toc73469137)

[5.2 Hạn chế của đồ án 55](#_Toc73469138)

[5.3 Hướng phát triển 55](#_Toc73469139)

[TÀI LIỆU THAM KHẢO 55](#_Toc73469140)

[PHỤ LỤC 57](#_Toc73469141)

# : GIỚI THIỆU

## Tổng quan

Ngày này trong nhịp sống hối hả của con người thì việc giành thời gian để ra ngoài để mua sắm trở nên là 1 điều quá xa sỉ. Những lo lắng về giao thông không an toàn và hạn chế trong việc mua hàng truyền thống có thể tránh được trong khi mua sắm trực tuyến. Với mua sắm trực tuyến(online), bạn cũng không cần phải lo lắng về điều kiện thời tiết. Người tiêu dùng và các khách hàng là những tổ chức, công ty,… đang dần chuyển sang mua sắm trực tuyến nhiều hơn nhằm tiết kiệm thời gian

Chính vì thế việc mua sắm online càng trở nên quan trọng và cần thiết,chỉ cần 1 cú click chuột thì họ có thể có được sản phẩm mà mình mong muốn.Việc mua sắm online có nhiều ưu điểm là có thể sở hữu mọi thứ thông qua các cú click chuột chứ không cần phải đến tận nơi để mua hàng. Sau khi vào [website bán hàn](http://thietkewebsitehcm.org/thiet-ke-website-ban-hang)g, chọn sản phẩm, chỉ cần đặt hàng (order) người bán sẽ mang sản phẩm đến tận nhà bạn. Mua sắm online cho phép mua hàng bất cứ khi nào bạn muốn. Các cửa hang trên mạng không bao giờ đóng cửa, có thể mua sắm 24/24 giờ và 7 ngày trong tuần. Khi mua hàng online, bạn dễ dàng so sánh và đưa ra lựa chọn sản phẩm phù hợp nhất. Đôi khi bạn gặp phải những người bán hàng khó tính tại một số địa điểm bán hàng. Mua sắm online thì khách hàng chẳng phải để ý đến chuyện ấy nữa.

Vì những lý do trên nên nhóm em quyết định làm đề tài: Website giới thiệu, bán xe đạp trực tuyến.

## Mục tiêu đề tài

* Khách hàng:
  + Có thể đăng ký tài khoản của website.
  + Xem danh sách sản phẩm
  + Xem chi tiết của từng sản phẩm
  + Chọn mua từng sản phẩm
  + Xem giỏ hàng
  + Thanh toán đơn hàng.
* Người quản lý website:
  + Tìm kiếm thông tin về sản phẩm/loại sản phẩm, các đơn đặt sản phẩm
  + Quản lý thông tin sản phẩm/loại sản phẩm.
  + Quản lý thông tin đơn hàng trực tuyến.

## Phạm vi đề tài

* Đặc điểm ứng dụng:
  + Giúp khách hàng dể dàng tìm kiếm sản phẩm, đặt hàng và thanh toán.
  + Giúp người quản lý dễ dàng quản lý sản phẩm hoặc đơn hàng mà khách hàng đã đặt.
* Phạm vi ứng dụng:
  + Được sự dụng trong quy mô nhỏ.
  + Phát triển trên PC.
* Đối tượng phục vụ:
  + Khách hàng.
  + Người quản lý website.

## Mô tả yêu cầu chức năng

* Chức năng chính của trang web:
  + Người dùng không có tài khoản (guest) có các chức năng:
    - Có thể đăng ký tài khoản của website với các thông tin cần thiết (email không trùng với tài khoản khác), sau khi đăng ký thành công với thông tin hợp lệ, lưu trữ CSDL
    - Xem danh sách sản phẩm ... tùy theo đề tài, danh sách này lấy từ CSDL).
    - Xem chi tiết của từng sản phẩm từ danh sách sản phẩm.
    - Chọn mua từng sản phẩm (có thể chọn mua từ trang Web danh sách sản phẩm hay từ trang Web chi tiết của từng sản phẩm), sản phẩm sau khi chọn mua sẽ được đưa vào trong giỏ hang, nếu chọn sản phẩm đã có trong giỏ hàng, cập nhật số lượng.
    - Xem giỏ hàng (danh sách sản phẩm đã chọn mua, thông tin này lưu trong biến Session, không cần cập nhật CSDL).
    - Khi xem giỏ hàng, có thể chỉnh sửa số lượng của từng sản phẩm trong giỏ hàng (nếu chỉnh sửa số lượng là 0 → bỏ sản phẩm đó ra khỏi giỏ hàng).
* Người dùng có tài khoản (customer) có thể thực hiện các chức năng của Người dùng không có tài khoản (guest), ngoài ra người dùng có tài khoản (customer) còn có thể:
  + Xử lý thanh toán (chức năng này thực hiện khi giỏ hàng đã có sản phẩm và người dùng đăng nhập thành công vào hệ thống): cập nhật thông tin vào CSDL + thông báo đăng ký đặt hàng thành công với các thông tin kèm theo. Sau khi xử lý thành công, Session được xóa về null.
* Người quản trị hệ thống (admin) có thể thực hiện được chức năng như một người dùng có tài khoản (customer). Ngoài ra, chức năng khác dành cho người quản trị hệ thống (admin) - Phần Back-End:
  + Tìm kiếm thông tin về sản phẩm/loại sản phẩm, các đơn đặt sản phẩm.
  + Quản lý thông tin sản phẩm/loại sản phẩm:
    - Xem danh sách sản phẩm/loại sản phẩm.
* Xem chi tiết từng sản phẩm/loại sản phẩm.
* Xóa sản phẩm/loại sản phẩm trong trường hợp sản phẩm chưa có trong đơn hàng nào hoặc loại sản phẩm chưa có sản phẩm nào.
* Thêm mới, cập nhật thông tin sản phẩm/loại sản phẩm.
  + Quản lý thông tin đơn hàng trực tuyến:
    - Xem danh sách các đơn hàng (sắp xếp theo ngày mua).
    - Xem chi tiết đơn hàng.
    - Cập nhật số lượng của mặt hàng trong đơn hàng trực tuyến
* Phi chức năng
* Thời gian : Phải đáp ứng đúng thời gian đã quy định
* Ngôn ngữ lập trình : Phù hợp với nhiều loại máy .
* Giao diện website: dễ nhìn, có độ tương phản cao, thân thiện với người dùng.
* Đội tin cậy:
* Tương thích về phần cứng : đáp ứng tương thích với đa số hệ điều hành, tránh ra yều cầu về cấu hình và những yêu cầu khác.
* Yêu cầu về lưu trữ :

# : CƠ SỞ LÝ THUYẾT

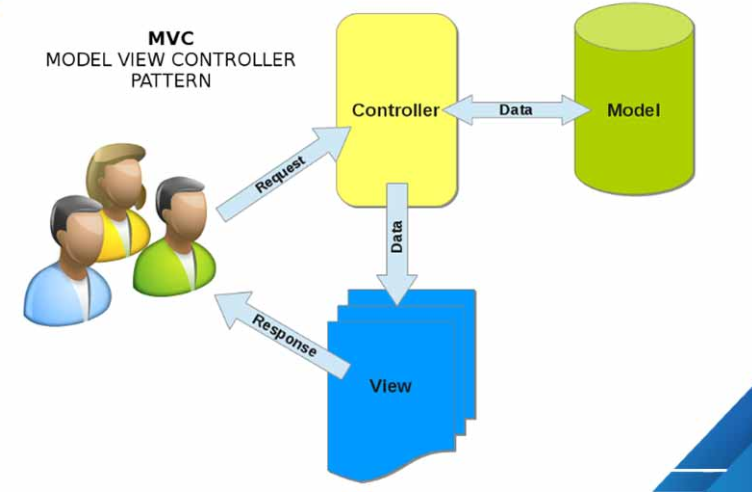
## Ngôn ngữ Spring MVC

### Khái niệm về mô hình MVC

Mô hình MVC là một chuẩn mô hình và đóng vai trò quan trọng trong quá trình xây dựng – phát triển – vận hành và bảo trì một hệ thống hay một ứng dụng – phần mềm.

Nó tạo ra một mô hình 3 lớp Model – View – Controller tách biệt và tương tác nhau, giúp các chuyên gia có thể dễ dàng dựa vào mô hình để trao đổi và xử lý những nghiệp vụ một cách nhanh chóng.

Đây là một mô hình đã xuất hiện từ những năm 70 của thế kỷ 20 tại phòng thí nghiệm Xerox PARC ở Palo Alto, nó không phụ thuộc vào môi trường, nền tảng xây dựng hay ngôn ngữ phát triển. Chúng ta có thể áp dụng mô hình MVC vào các dự án trong môi trường Windows, Linux… và sử dụng bất kỳ ngôn ngữ nào như PHP, ASP, JSP….



**Hình ảnh mô hình MVC**

**Mô hình MVC được chia làm 3 lớp xử lý gồm Model – View – Controller :**

**Model**: là nơi chứa những nghiệp vụ tương tác với dữ liệu hoặc hệ quản trị cơ sở dữ liệu (mysql, mssql… ); nó sẽ bao gồm các class/function xử lý nhiều nghiệp vụ như kết nối database, truy vấn dữ liệu, thêm – xóa – sửa dữ liệu…

**View**: là nới chứa những giao diện như một nút bấm, khung nhập, menu, hình ảnh… nó đảm nhiệm nhiệm vụ hiển thị dữ liệu và giúp người dùng tương tác với hệ thống.

**Controller**: là nới tiếp nhận những yêu cầu xử lý được gửi từ người dùng, nó sẽ gồm những class/ function xử lý nhiều nghiệp vụ logic giúp lấy đúng dữ liệu thông tin cần thiết nhờ các nghiệp vụ lớp Model cung cấp và hiển thị dữ liệu đó ra cho người dùng nhờ lớp View

### Khái niệm về Spring

Spring là một Framework phát triển các ứng dụng Java được sử dụng bởi hàng triệu lập trình viên. Nó giúp tạo các ứng dụng có hiệu năng cao, dễ kiểm thử, sử dụng lại code…

Spring nhẹ và trong suốt (nhẹ: kích thước nhỏ, version cơ bản chỉ khoảng 2MB; trong suốt: hoạt động một cách trong suốt với lập trình viên)

Spring là một mã nguồn mở, được phát triển, chia sẻ và có cộng đồng người dùng rất lơn.

### Khái niệm về Spring MVC

Spring MVC là một Framework / 1 Project mã nguồn mở của Spring.

Spring MVC Framewrok cung cấp kiến truc MVC (Model-View-Controller) và các component được sử dụng để phát triển các ứng dụng web một cách linh hoạt

***2.1.4 Tại sao nên dùng Spring MVC***

Thể hiện được tính chuyên nghiệp trong việc lập trình, phân tích thiết kế

Do được chia thành các thành phần độc lập nên hỗ trợ phát triển ứng dụng nhanh, đơn giản, dễ nâng cấp và bảo trì…

Tách logic ứng dụng và thiết kế web thông qua mẫu MVC.

Các tầng trong Spring MVC độc lập nên việc unit test dễ dàng hơn.

Phần view có thể tích hợp với nhiều Framework về UI như JSF, Freemarker, Themeleaf…

Spring MVC base trên các POJO class nên các hành động của nó khá đơn giản

Hỗ trợ cả Annotation và XML config giúp việc phát triển nhanh hơn và sạch hơn.

Cung cấp việc phân chia một cách rõ ràng, linh hoạt giữa controller, service, data acces layer.

Tất cả các framework đã được tích hợp rất tốt vào Spring.

Sử dụng cơ chế plug-in.

Kết hợp rất tốt với các O/R (object-relational) Mapping frameworks như là Hibernate.

Ít phức tạp hơn so với các framework khác.

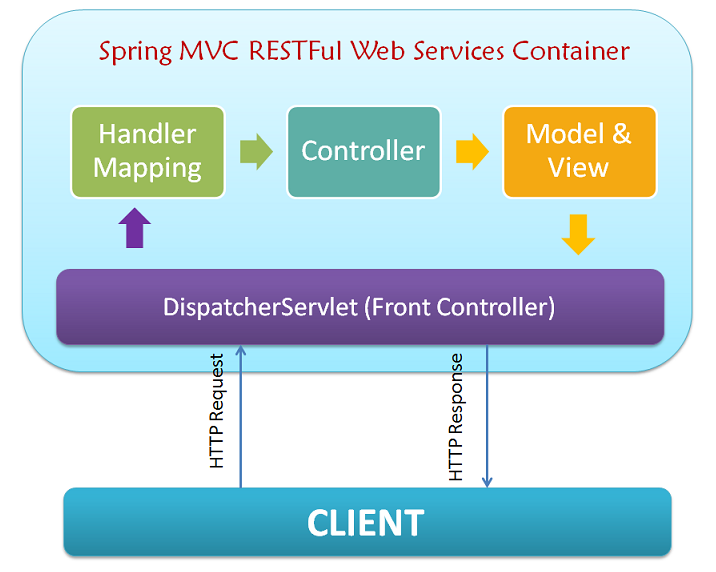
Tích hợp IDE.

Tích hợp với các ngôn ngữ mẫu

Cộng đồng người sử dụng rất nhiều, nhiều sách mới được xuất bản.

### Giới thiệu về ngôn ngữ Spring MVC

#### Mô hình MVC chia một ứng dụng thành ba phần:

****

**Hình ảnh minh họa Mô hình MVC**

**Model(Mô hình)**

Mô hình gồm các lớp java có nhiệm vụ:

Biểu diễn data và cho phép truy cập tới để get và set data trong (JAVABEAN), Thường thì phần layer này mô phỏng 1 cách đầy đủ đối tượng từ thế giới thực.

Nhận các yêu cầu từ khung nhìn

Thi hành các yêu cầu đó (tính toán, kết nối CSDL …)

Trả về các giá trị tính toán theo yêu cầu của Controller

**View(Khung nhìn):**

Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images...Hiểu một cách đơn giản, nó là tập hợp các form hoặc các file JSP, HTML, CSS, XML, Javascript,JSON…để hiển thị giao diện người dùng, các dữ liệu trả về từ Model thông qua Controller…

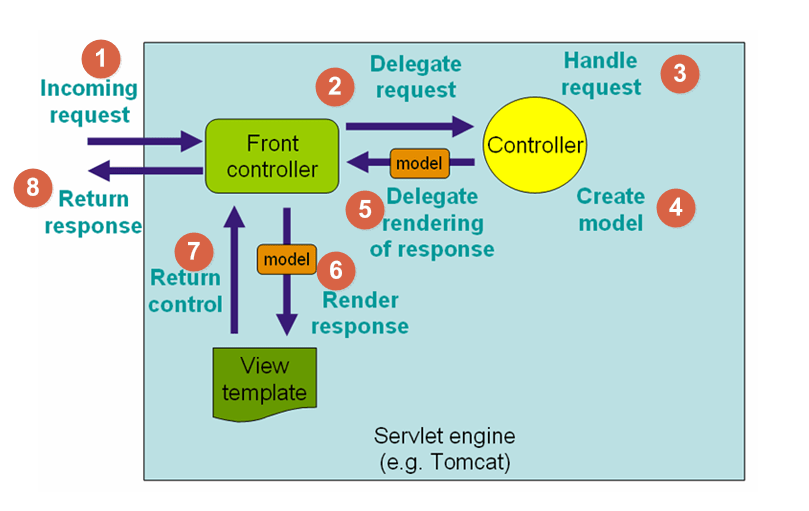
**Controller(Bộ điều khiển):**

Đồng bộ hoá giữa Khung nhìn và Mô hình. Tức là với một trang JSP này thì sẽ tương ứng với lớp java nào để xử lý nó và ngược lại, kết quả sẽ trả về trang jsp nào.Nó đóng vai trò điều tiết giữa View và Model.

Như vậy, chúng ta có thể tách biệt được các mã java ra khỏi mã html.. Người thiết kế giao diện và người lập trình java có thể mang tính chất độc lập tương đối.

Việc debug hay bảo trì sẽ dễ dàng hơn, việc thay đổi các theme của trang web cũng dễ dàng hơn …

#### **Hoạt động của mô hình MVC**



**Hình ảnh:Hoạt động của mô hình MVC**

#### Mô tả luồng sự kiện được xử lý trong MVC:

Bất kỳ request nào tới ứng dụng web đều sẽ được gửi tới Front Controller (Dispatcher Servlet)

Front Controller sẽ sử dụng Handler Mapping để biết được controller nào sẽ xử lý request đó

Controller nhận request, gọi tới các class service thích hợp để xử lý yêu cầu.

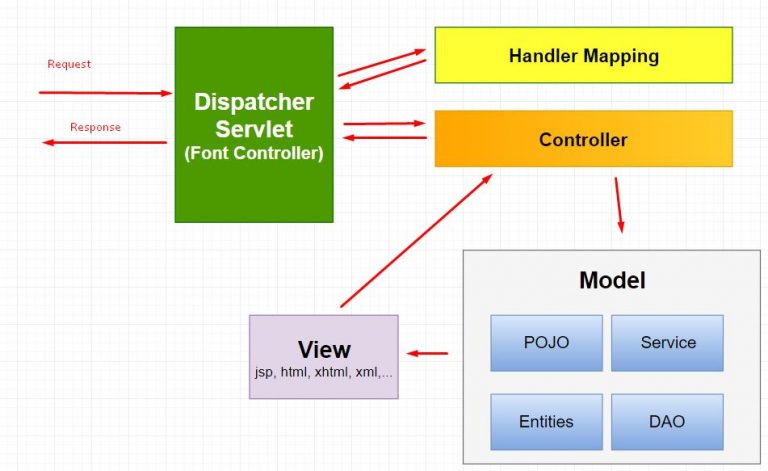
Sau khi xử lý xong, Controller sẽ nhận được model từ tầng Service hoặc tầng DAO.

Controller gửi model vừa nhận được tới Front Controller (Dispatcher Servlet)

Dispatcher Servlet sẽ tìm các mẫu view, sử dụng view resolver và truyền model vào nó.

View template, model, view page được build và gửi trả lại Front Controller

Front Controller gửi một page view tới trình duyệt để hiển thị nó cho người dùng.



**Trong Mô hình MVC thì:**

**Model**: là các file POJO, Service, DAO thực hiện truy cập database, xử lý business

**View**: là các file JSP, html.

**Control**: là Dispatcher Controller, Handler Mapping, Controller – thực hiện điều hướn các request.

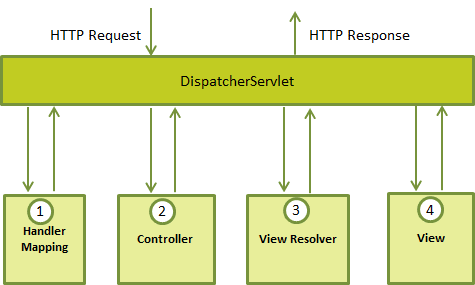
**Cơ chế xử lí Request và Response trong SpringMVC:**

**Ví dụ:**

Một ngày nọ bạn lướt web xem phim, bạn vô tình nhập vào đường link trong trình duyệt với nội dung là [tuisach.com](http://webgirlxinh.com/), mục đích của bạn chỉ để xem mẫu những túi sách. Đồng thời với đó, bạn đã yêu cầu server trả về trang danh sách những túi sách cho bạn. Thao tác này gọi là request.

Phía bên server tiếp nhận, nó xử lí,rồi trả về ngay cho bạn trang web túi sách mà bạn cần. Việc trả về như vậy là response.

\*Vậy Server tiếp nhận và xử lí như thế nào ?



Hình ảnh minh họa

Khi server nó nhận yêu cầu của bạn. Phía server có DispatcherServlet.( Spring Web MVC được xây dựng trên Servlet chung được gọi là lớp DispatcherServlet (Bộ điều khiển phía trước)

Trong SpringMVC thì nói gọi là lớp, tầng tiếp nhận và xử lí mọi yêu cầu. Nó có thể khóa, đóng mở tùy mình. Nó được cấu hình trong file web.xml.

DispatcherServlet này tiếp nhận yêu cầu rồi thì gọi đến Handler Mapping.Handler biết người dùng gửi yêu cầu vậy là phải hỏi (ánh xạ) đúng người này.

Controller là phần xử lí logic của ứng dụng. Ví dụ khi người dùng muốn đặt hàng ABC thì Controller sẽ làm gì.

DispatcherServlet nhận được yêu cầu vậy thì Controller nó liên lạc trực tiếp xuống Model, kéo dữ liệu ra và trả về dưới dạng ModelAndView object cho DispatcherServlet.

Cuối cùng DisatcherServlet trả về response cho người dùng và hiển thị lên View.

Tất cả mọi hoạt động đều thông qua DispatcherServlet.

**Sử dụng các ngôn ngữ ,thư viện để thiết website:**

HTML(Hyper Text Markup Language): Giúp thêm nội dung cho trang web,CSS(Cascading Style Sheet): Định dạng thiết kế, bố cục, phong cách, canh lề của trang web.,JS: Cải thiện cách hoạt động của trang web.

Thư viện Bootstrap: Bootstrap là một thư viện html, css và js. Bootstrap sử dụng html, css và js tạo ra những mẫu cơ bản như: form, button, table, navigation… giúp nhà phát triển web dễ dàng tạo ra những giao diện web tương thích với màn hình điện thoại.

### Spring security

#### Giới thiệu

Spring Security là một dự án nổi bật trong hệ sinh thái Spring. Spring Security cung cấp các dịch vụ bảo mật toàn diện cho các ứng dụng doanh nghiệp có nền tảng Java EE.

Spring Security cung cấp 2 cơ chế cơ bản:

• Authentication (xác thực): là tiến trình thiết lập một principal. Principal có thể hiểu là một người, hoặc một thiết bị, hoặc một hệ thống nào đó có thể thực hiện một hành động trong ứng dụng của bạn.

• Authorization (phân quyền) hay Access-control: là tiến trình quyết định xem một principal có được phép thực hiện một hành động trong ứng dụng của bạn hay không. Trước khi diễn tiến tới Authorization, principal cần phải được thiết lập bởi Authentication.

Ta có thể thấy đây là 2 cơ chế khá phổ biến trong các dịch vụ bảo mật, không chỉ riêng Spring Security.

Các thành phần cốt lõi: Security, SecurityContext và Authentication

o SecurityContext là interface cốt lõi của Spring Security, lưu trữ tất cả các chi tiết liên quan đến bảo mật trong ứng dụng. Khi chúng ta kích hoạt Spring Security trong ứng dụng thì SecurityContext cũng sẽ được kích hoạt theo.

o Chúng ta sẽ không truy cập trực tiếp vào SecurityContext, thay vào đó sẽ sử dụng lớp SecurityContextHolder. Lớp này lưu trữ security context hiện tại của ứng dụng, bao gồm chi tiết của principal đang tương tác với ứng dụng. Spring Security sẽ dùng một đối tượng Authentication để biểu diễn thông tin này.

UserDetails và UserDetailsService

- UserDetails là một interface cốt lõi của Spring Security. Nó đại diện cho một principal nhưng theo một cách mở rộng và cụ thể hơn. Vậy UserDetails cung cấp cho ta những thông tin gì? UserDetails bao gồm các method sau:

* getAuthorities(): trả về danh sách các quyền của người dùng
* getPassword(): trả về password đã dùng trong qúa trình xác thực
* getUsername(): trả về username đã dùng trong qúa trình xác thực
* isAccountNonExpired(): trả về true nếu tài khoản của người dùng chưa hết hạn
* isAccountNonLocked(): trả về true nếu người dùng chưa bị khóa
* isCredentialsNonExpired(): trả về true nếu chứng thực (mật khẩu) của người dùng chưa hết hạn
* isEnabled(): trả về true nếu người dùng đã được kích hoạt

GrantedAuthority

- Phương thức getAuthorities() sẽ trả về một tập hợp các đối tượng GrantedAuthority. Một GrantedAuthority là một quyền được ban cho principal. Các quyền đều có tiền tố là ROLE\_, ví dụ như ROLE\_ADMIN, ROLE\_MEMBER ...

### Cơ sở dữ liệu SQL Server

#### Giới thiệu cơ sở dữ liệu

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS) ) sử dụng câu lệnh SQL (**Transact-SQL)**để trao đổi dữ liệu giữa máy Client và máy cài SQL Server.

Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server được phát triển và tiếp thị bởi Microsoft.

SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server có thể kết hợp “ăn ý” với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server….

#### Mục đích sử dụng cơ sở dữ liệu

Chung quy lại chúng ta sử dụng bất kì một hệ quản trị CSDL nào cũng để lưu trữ dữ liệu và SQL cũng không ngoại lệ. Tuy nhiên điều mà ta mong đợi ở nó là các tính năng giúp việc sử dụng hiêu quả hơn như:

Cho phép tạo nhiều cơ sở dữ liệu

Duy trì lưu trữ bền vững

Bảo mật cao

Phân tích dữ liệu bằng SSAS – SQL Server Analysis Services

Tạo được báo cáo bằng SSRS – SQL Server Reporting Services

Thực hiện quá trình ETL (Extract-Transform-Load) bằng SSIS – SQL Server Integration Services.

#### ***Ưu và nhược điểm của SQL Server***

**Ưu điểm :**

* + - Có thể cài nhiều phiên bản MS SQL khác nhau trên cùng một máy tính.
    - Duy trì riêng biệt các môi trường sản xuất, phát triển, thử nghiệm.
    - Giảm thiểu các vấn đề tạm thời trên cơ sở dữ liệu.
    - Tách biệt các đặc quyền bảo mật.
    - Duy trì máy chủ dự phòng.

**Nhược điểm :**

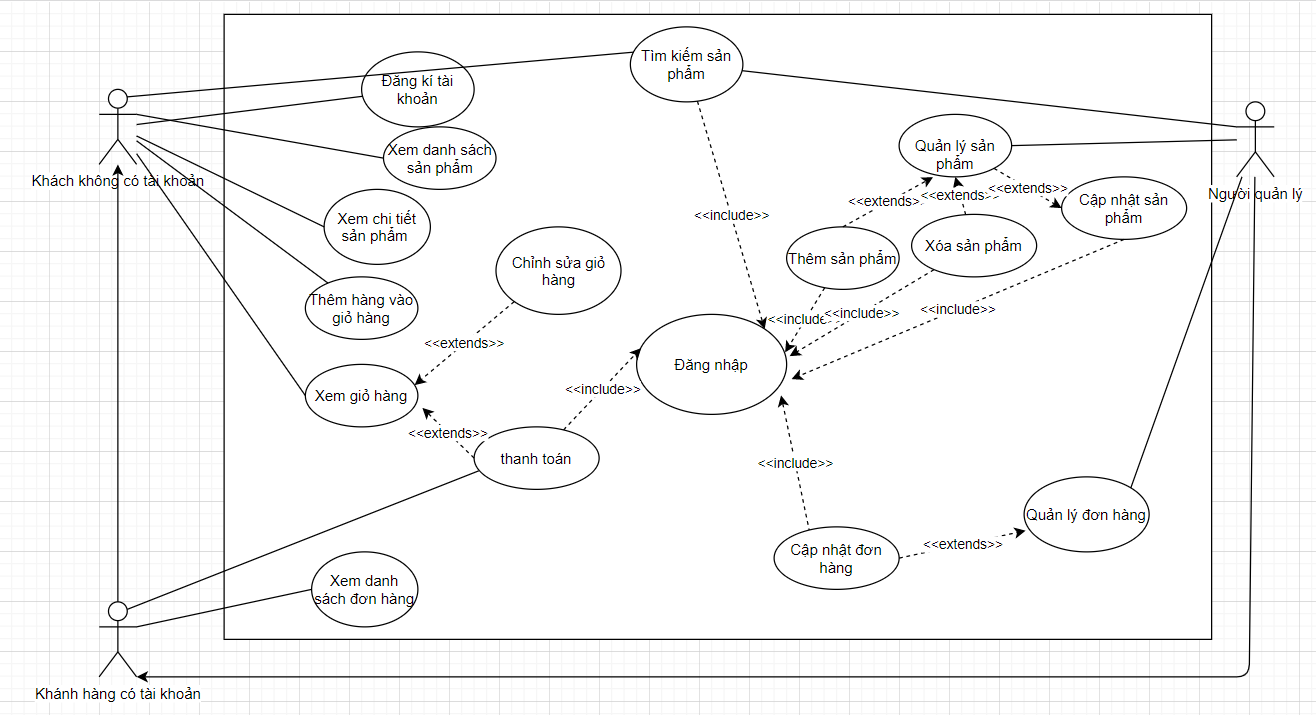
* SQL Server chỉ chạy trên hệ điều hành Windows.
* Cần thanh toán phí license để chạy nhiều CSDL (database)

# : PHÂN TÍCH

Chương này sinh viên trình bày quá trình phân tích bài toán hoặc hệ thống từ việc thu thập thông tin, thu nhận yêu cầu ban đầu. Sinh viên có thể trình bày các lược đồ (UML, ERD, …)

## Phân tích yêu cầu bằng UML

### Usecase tổng quát



### Danh sách tác nhân và mô tả

Danh sách tác nhân và mô tả xem ở Bảng 3-1.

|  |  |
| --- | --- |
| Tác nhân | Mô tả tác nhân |
| Khách hàng không tài khoản. | Khách hàng không có tài khoản có những chức năng như đăng kí tài khoản, xem danh sách sản phẩm, xem chi tiết sản phẩm, thêm hàng vào giỏ hàng, xem giỏ hàng. |
| Khách hàng có tài khoản. | Khách hàng có tài khoản có thể thực hiện các chức năng của Khách hàng không có tài khoản, ngoài ra Khách hàng có tài khoản còn có thể thanh toán đơn hàng và xem danh sách đơn hàng. |
| Người quản lý | Người quản lý có thể thực hiện được chức năng như một khách hàng có tài khoản. Ngoài ra, chức năng khác dành cho người quản lý như: tìm kiếm sản phẩm, quản lý sản phẩm, quản lý đơn hàng. |

Bảng 3‑1 Danh sách tác nhân và mô tả

### Danh sách các tình huống hoạt động (Use cases)

|  |  |
| --- | --- |
| ID | Tên Use case |
| UC01 | Đăng kí tài khoản. |
| UC02 | Xem danh sách sản phẩm. |
| UC03 | Xem chi tiết sản phẩm. |
| UC04 | Tìm kiếm. |
| UC05 | Xem giỏ hàng. |
| UC06 | Chỉnh sửa giỏ hàng. |
| UC07 | Thanh toán. |
| UC08 | Thêm hàng vào giỏ hàng. |
| UC09 | Thêm sản phẩm. |
| UC10 | Xóa sản phẩm. |
| UC11 | Xem danh sách đơn hàng. |
| UC12 | Cập nhật sản phẩm. |
| UC13 | Cập nhật đơn hàng. |
| UC14 | Đăng nhập. |

### Đặc tả các yêu cầu của chức năng

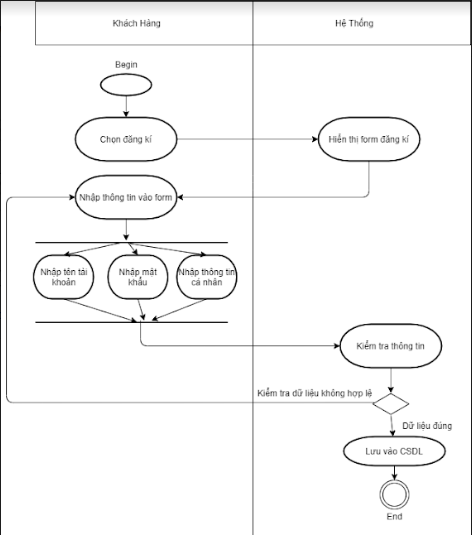
#### UC001\_Đăng kí tài khoản

##### Mô tả use case UC001

|  |  |  |
| --- | --- | --- |
| Use case: UC001\_ Đăng kí tài khoản | | |
| Mục đích | Cho phép đăng kí tài khoản vào hệ thống để có thể sử dụng các chức năng mua sản phẩm. | |
| Mô tả | Khi người dùng bấm vào nút đăng kí tài khoản trên hệ thống, hệ thống sẽ hiển thị màn hình đăng kí và yêu cầu người dùng điền đầy đủ các thông tin đăng nhập. Nếu thông tin hợp lệ thì hệ thống sẽ lưu thông tin tài khoản vào dữ liệu và người dùng có thể sử dụng lại tài khoản này đăng nhập vào sau này. | |
| Tác nhân | Khách hàng | |
| Điều kiện trước | Khách hàng chưa có tài khoản. | |
| Điều kiện sau | Tài khoản của khách hàng được lưu vào hệ thống. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người sử dụng chọn vào đăng kí tài khoản. |  |
|  | 2.Hệ thống hiển thị ra màn hình đăng kí,yêu cầu điền thông tin |
| 3.Người dùng nhập vào các thông tin tài khoản |  |
|  | 4.Hệ thống kiểm tra thông tin nhập vào có hợp lệ hay không. |
|  | 5.Hệ thống lưu thông tin tài khoản vào trong hệ thống. |
|  |  |
| Luồng sự kiện phụ  (Alternative Flows) | 4.1 Hệ thống kiểm tra nếu thông tin nhập sai sẽ thông báo kết quả và yêu cầu người dùng phải nhập lại thông tin. | |

##### Biểu đồ.

###### Activity



###### Sequence:

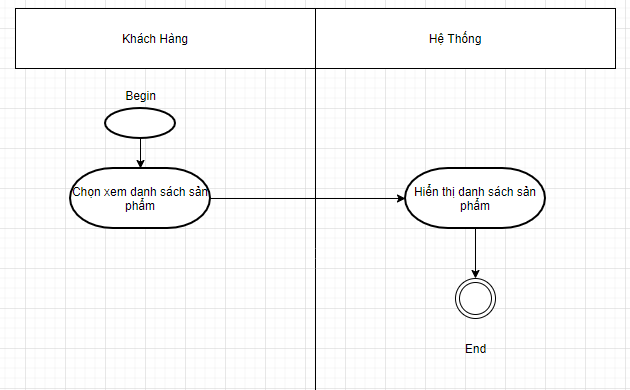
#### UC002\_Xem danh sách sản phẩm

##### Mô tả use case UC002

|  |  |  |
| --- | --- | --- |
| Use case: UC002\_Xem danh sách sản phẩm | | |
| Mục đích | Cho phép người dùng có thể xem danh sách các sản phẩm có trong website | |
| Mô tả | Khi người dùng bấm vào chức năng xem danh sách sản phẩm hệ thống sé hiển thị ra danh sách thông tin các sản phẩm có trong hệ thống. | |
| Tác nhân | Khách hàng, người quản lý | |
| Điều kiện trước | Các sản phẩm đã có trong hệ thống. | |
| Điều kiện sau | Danh sách các sản phẩm được hiện thị ra màn hình. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người sử dụng chọn vào xem danh sách sản phẩm. |  |
|  | 2.Hệ thống hiển thị ra danh sách các sản phẩm mà người dùng yêu cầu. |
|  | |
| Luồng sự kiện phụ  (Alternative Flows) |  | |

##### Biểu đồ

###### Activity



**Sequence:**

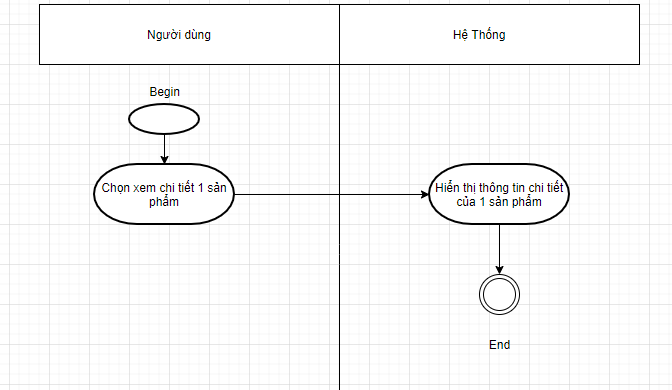
#### UC003\_Xem chi tiết sản phẩm.

##### Mô tả use case UC002

|  |  |  |
| --- | --- | --- |
| Use case: UC003\_ Xem chi tiết sản phẩm | | |
| Mục đích | Cho phép người dùng có thể xem thông tin chi tiết của sản phẩm. | |
| Mô tả | Khi người dùng bấm vào nút xem thông tin chi tiết hệ thống sẽ hiện thị ra thông tin chi tiết của sản phẩm cần xem chi tiết theo yêu cầu. | |
| Tác nhân | Khách hàng,người quản lý | |
| Điều kiện trước | Sản phẩm đã có trước trong hệ thống. | |
| Điều kiện sau | Thông tin chi tiết sản phẩm được hiển thị ra màn hình. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người sử dụng chọn vào xem thông tin chi tiết của 1 sản phẩm. |  |
|  | 2.Hệ thống hiển thị ra thông tin chi tiết của 1 sản phẩm đã chọn. |
|  | |
| Luồng sự kiện phụ  (Alternative Flows) |  | |

##### Biểu đồ

###### Activity



###### Sequence:

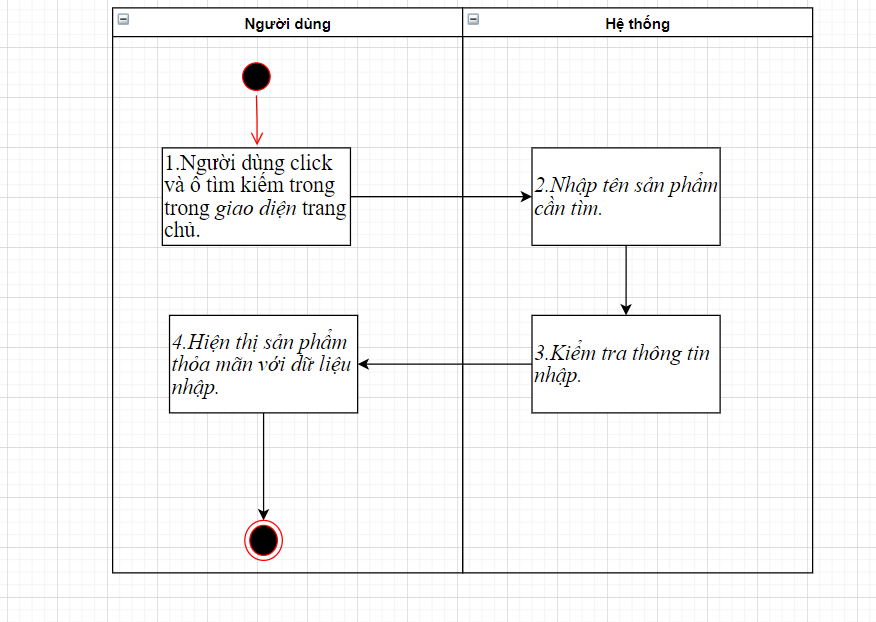
#### UC004 Tìm kiếm

##### Mô tả use case UC004

|  |  |  |
| --- | --- | --- |
| Use case: UC004\_Tìm Kiếm | | |
| Mục đích | Giúp người dùng dễ dàng tìm kiếm sản phẩm mà mình cần. | |
| Mô tả | Người dùng muốn tìm sản phẩm mà mình thích chỉ cần nhập thông tin thì hê thống sẽ hiện ra sản phẩm mà có tên giống với tên nhập. | |
| Tác nhân | Người dùng | |
| Điều kiện trước | Tác nhân đã vào trang chủ của website. | |
| Điều kiện sau | Hiện danh sách sản phẩm giống với tên nhập. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người dùng click và ô tìm kiếm trong trong giao diện trang chủ. |  |
| 2.Nhập tên sản phẩm cần tìm. |  |
|  | 3.Kiểm tra thông tin nhập. |
|  | 4.Hiện thị sản phẩm thỏa mãn với dữ liệu nhập. |
| Luồng sự kiện phụ  (Alternative Flows) |  | |

##### Biểu đồ

###### Activity



###### Senquence

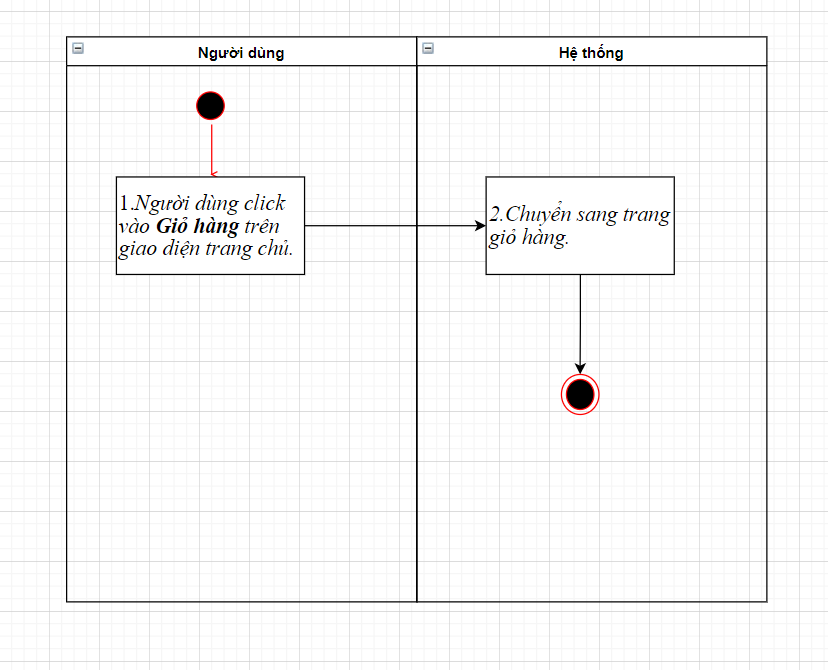
#### UC005\_Xem giỏ hàng

##### Mô tả use case UC005

|  |  |  |
| --- | --- | --- |
| Use case: UC005\_Xem giỏ hàng | | |
| Mục đích | Giúp người dùng xem danh sách sản phẩm mà mình đã chọn. | |
| Mô tả | Sau khi khách hàng đã chọn những sản phẩm mà mình cần thì khách hàng có thể vào giỏ hàng để xem danh sách sản phẩm mà mình đã chọn. | |
| Tác nhân | Người dùng | |
| Điều kiện trước | Tác nhân đã vào trang chủ của website. | |
| Điều kiện sau | Hiện danh sách sản phẩm trong giỏ hàng. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người dùng click vào **Giỏ hàng** trên giao diện trang chủ. |  |
|  | 2.Chuyển sang trang giỏ hàng. |
| Luồng sự kiện phụ  (Alternative Flows) |  | |

##### Biểu đồ

###### Activity



###### Sequence :

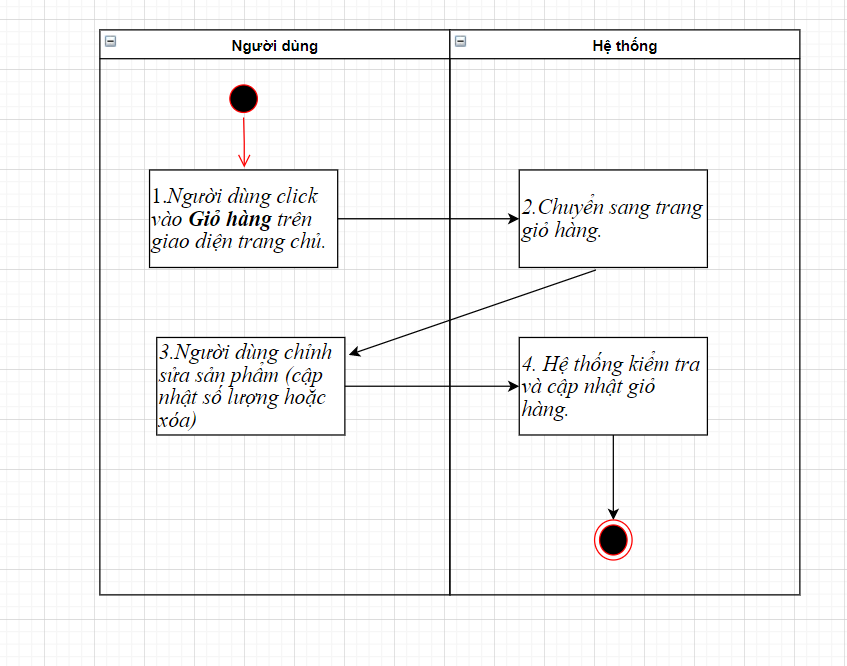
#### UC006\_Chỉnh sửa giỏ hàng

##### Mô tả use case UC006

|  |  |  |
| --- | --- | --- |
| Use case: UC006\_Chỉnh sửa giỏ hàng | | |
| Mục đích | Giúp người dùng chỉnh sửa sản phẩm mà mình đã chọn trong giỏ hàng. | |
| Mô tả | Khách hàng có thể chỉnh sửa số lượng hoặc xóa sản phẩm có trong giỏ hàng. | |
| Tác nhân | Người dùng | |
| Điều kiện trước | Tác nhân đã vào trang chủ của website. | |
| Điều kiện sau | Cập nhật danh sách sách sản phẩm trong giỏ hàng thành công. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người dùng click vào **Giỏ hàng** trên giao diện trang chủ. |  |
|  | 2.Chuyển sang trang giỏ hàng. |
|  | 3.Người dùng chỉnh sửa sản phẩm (cập nhật số lượng hoặc xóa) |  |
|  |  | 4. Hệ thống kiểm tra và cập nhật giỏ hàng. |
| Luồng sự kiện phụ  (Alternative Flows) |  | |

##### Biểu đồ

###### Activity



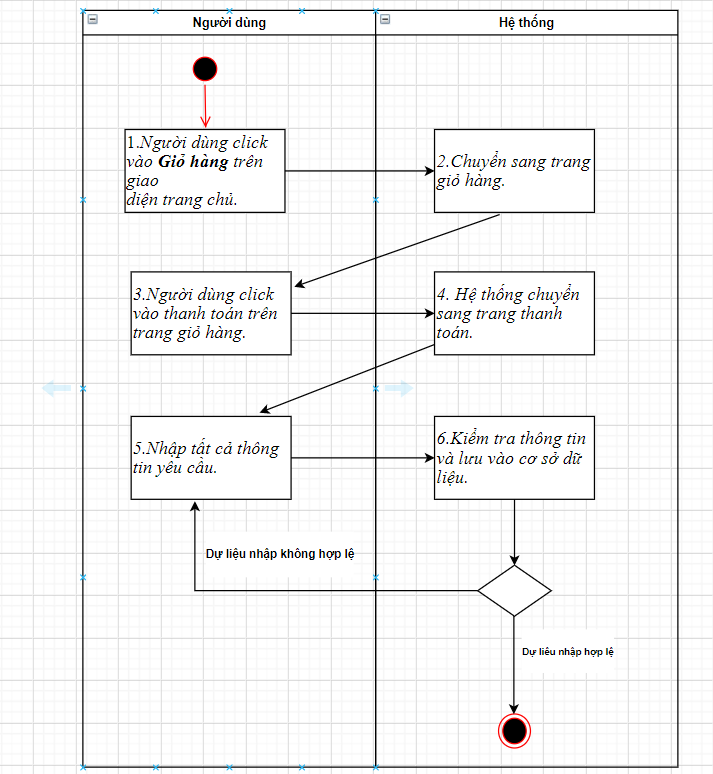
#### UC007\_Thanh toán

##### Mô tả use case UC007

|  |  |  |
| --- | --- | --- |
| Use case: UC007\_Thanh toán | | |
| Mục đích | Giúp người dùng thanh toán sản phẩm mà mình đã chọn. | |
| Mô tả | Người dùng sau khi đã chọn xong những sản phẩm cần thì có thể thanh toán để có thể hoàn tất thủ tục đặt hàng. | |
| Tác nhân | Người dùng | |
| Điều kiện trước | Tác nhân đã vào trang chủ của website. | |
| Điều kiện sau | Lưu dữ liệu đơn hàng vào Cơ sở dữ liệu. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người dùng click vào **Giỏ hàng** trên giao diện trang chủ. |  |
|  | 2.Chuyển sang trang giỏ hàng. |
| 3.Người dùng click vào thanh toán trên trang giỏ hàng. |  |
|  | 4. Hệ thống chuyển sang trang thanh toán. |
| 5.Nhập tất cả thông tin yêu cầu. |  |
|  | 6.Kiểm tra thông tin và lưu vào cơ sở dữ liệu. |
| Luồng sự kiện phụ  (Alternative Flows) |  | 6.1.Thông báo nhập sai thông tin yêu cầu nhập lại. |
| 6.2 Quay lại bước 5. |  |

##### Biểu đồ

###### Activity



###### Sequence:

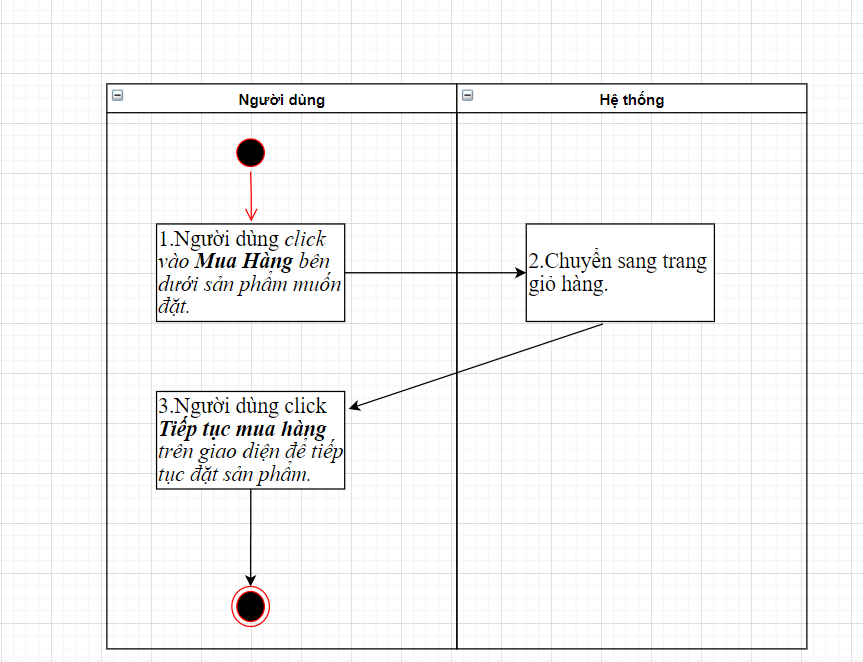
#### UC008\_Thêm hàng vào giỏ hàng

##### Mô tả use case UC008

|  |  |  |
| --- | --- | --- |
| Use case: UC008\_ Thêm hàng vào giỏ hàng | | |
| Mục đích | Giúp người dùng thêm sản phẩm vào giỏ hàng. | |
| Mô tả | Sau khi người dùng xem danh sách sản phảm và muốn đặt mua nó thì có thể sử dụng chức năng này có thể thêm sản phẩm mà mình thích vào trong giỏ hàng. | |
| Tác nhân | Người dùng | |
| Điều kiện trước | Tác nhân đã vào trang chủ của website. | |
| Điều kiện sau | Thêm sản phẩm vào giỏ hàng. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người dùng click vào **Mua Hàng** bên dưới sản phẩm muốn đặt. |  |
|  | 2.Chuyển sang trang giỏ hàng. |
| 3.Người dùng click **Tiếp tục mua hàng** trên giao diện để tiếp tục đặt sản phẩm. |  |
| Luồng sự kiện phụ  (Alternative Flows) |  |  |

##### Biểu đồ

###### Activity



###### Sequence:

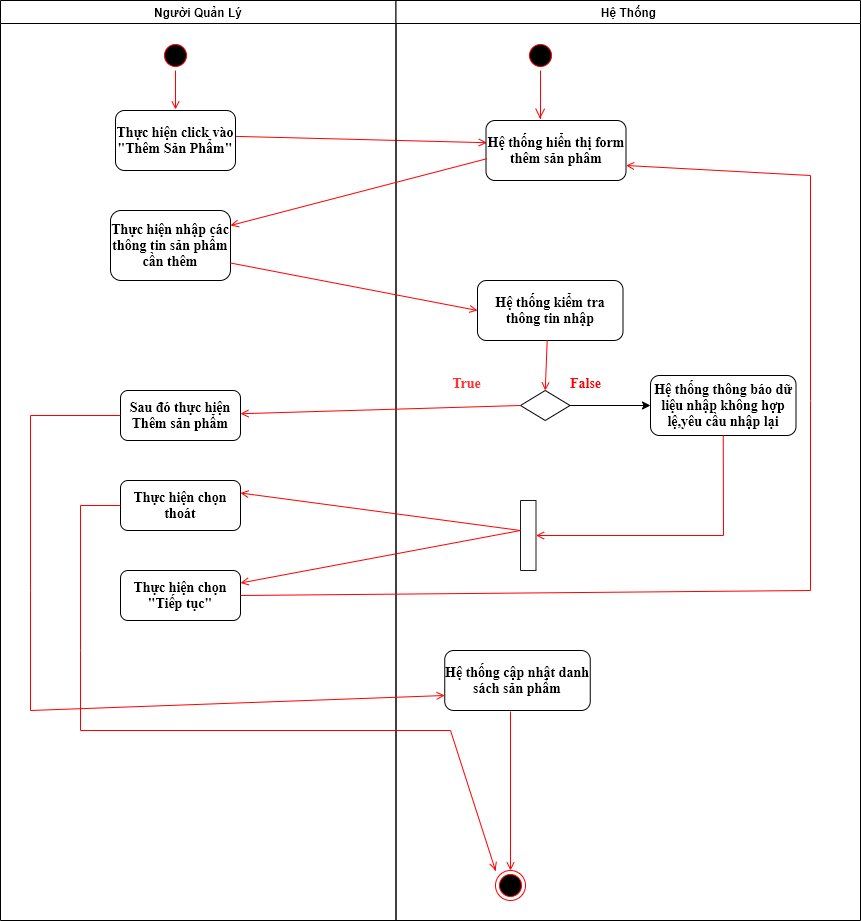
#### UC009\_Thêm sản phẩm

##### Mô tả use case UC009

|  |  |  |
| --- | --- | --- |
| Tên Use case | Thêm sản phẩm | UC009 |
| Mục đích | Người quản lí sẽ nhập thông tin sản phẩm vào trong danh sách | |
| Mô tả | Khi người quản lý muốn thêm sản phẩm, người quản lí sẽ vào giao diện quản lí và chọn chức năng thêm sản phẩm. Sau đó người người quản lí nhập thông tin sản phẩm cần thêm và chọn lưu. Hệ thống sẽ xử lý và thêm thông tin sản phẩm mới vào cơ sở dữ liệu. | |
| Tác nhân | Người quản lí | |
| Điều kiện trước | Người quản lí đăng nhập thành công vào hệ thống.Sau đó người quản lí chọn vào chức năng Thêm sản phẩm và người lý phải có thông tin sản phẩm cần thêm | |
| Điều kiện sau | Thông tin của chiếc sản phẩm mới sẽ được lưu vào cơ dữ liệu | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người quản lí đăng nhập giao diện người quản lí |  |
|  | 2.Hệ thống sẽ hiển thị giao diện về sản phẩm |
| 3.Người quản lí chọn chức năng thêm sản phẩm. |  |
|  | 4.Hiển thị form them sản phẩm |
| 5.Người quản lí nhập các dữ liệu thông tin về xe đạp mới muốn thêm |  |
|  | 6.Hệ thống sẽ kiểm tra thông tin nhập |
| 7.Thực hiện thêm sản phẩm |  |
|  | 8.Hệ thống sẽ cập nhật thông tin và lưu vào CSDL và thông báo kết quả. |
|  | 9.Người dùng đóng giao diện để kết thúc. |  |
| Luồng sự kiện phụ(Alternative Flows) | 6.1. Nếu dữ liệu rỗng hay dử liệu sai định dạng thì hệ thống sẽ thông báo lỗi và quay lại bước 4. | |
|  |  | |

##### Biểu đồ

###### Activity



###### Sequence:

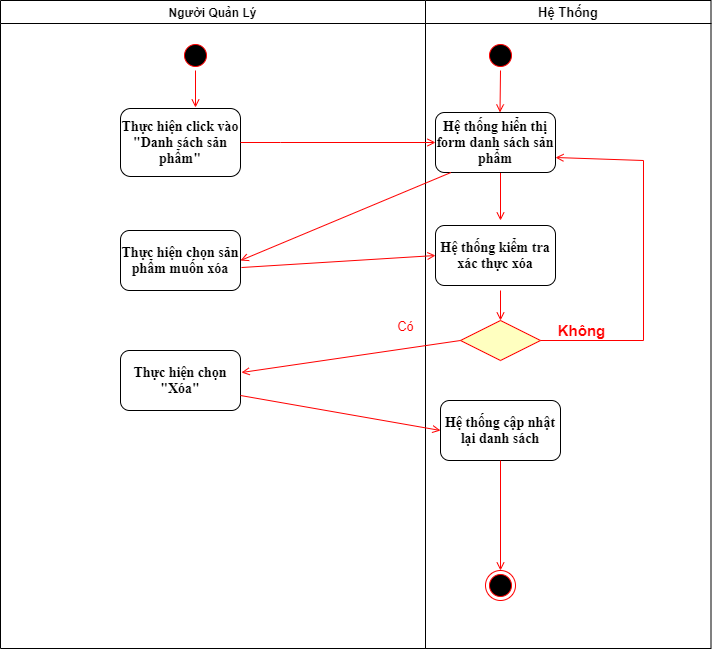
#### UC010\_Xóa sản phẩm

##### Mô tả use case UC010

|  |  |  |
| --- | --- | --- |
| Tên Use case | Xóa sản phẩm | UC010 |
| Mục đích | Người quản lí xóa thông tin sản phẩm trong danh sách sản phẩm | |
| Mô tả | Khi người dùng muốn xóa sản phẩm thì trước tiên người dùng vào giao diện sản phẩm chọn sản phẩm cần xóa và người quản lí chọn xóa. Khi đó hệ thống sẽ xử lý và thông báo kết quả xử lý. | |
| Tác nhân | Người quản lí | |
| Điều kiện trước | Người quản lí vào giao diện của mình và biết được thông tin sản phẩm muốn xóa. | |
| Điều kiện sau | Sản phẩm đã được xóa | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1. Người quản lí đăng nhập thành công và vào giao diện sản phẩm |  |
|  | 2. Hệ thống sẽ hiển thị giao diện về mục danh sách sản phẩm |
| 4.Người quản lí click vào danh sách sản phẩm |  |
|  | 5.Hệ thống hiển thị form danh sách sản phẩm |
| 6.Thực hiện chọn sản phẩm muốn xóa. |  |
|  | 7.Hệ thống kiểm tra về xác thực việc xóa |
|  | 8.Thực hiện chọn xóa. |  |
|  |  | 9.Hệ thống sẽ cập nhật lại danh sách và lưu vào CSDL và thông báo kết quả |
|  | 10.Người quản lí đóng giao diện để kết thúc. |  |
| Luồng sự kiện phụ(Alternative Flows) |  | 7.1.Nếu muốn xác thực việc xóa thực hiện tiếp bước 6  7.2.Nếu không muốn xác thực việc xóa,quay lại bước 5 |
|  |  |

##### Biểu đồ

###### Activity



###### Sequence:

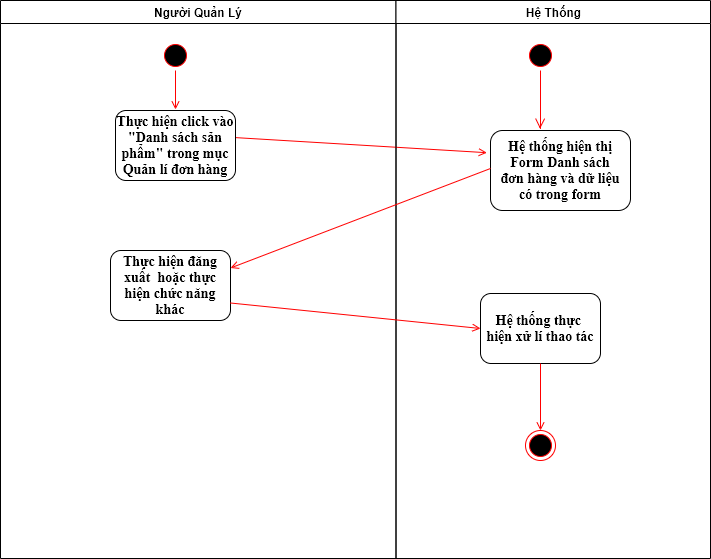
#### UC011\_Xem danh sách đơn hàng

##### Mô tả use case UC011

|  |  |  |
| --- | --- | --- |
| Tên Use case | Xem danh sách đơn hàng | UC011 |
| Mục đích | Người quản lí xem được danh sách những đơn hàng | |
| Mô tả | Khi người quản lí đăng nhập vào hệ thống thì chọn chức xem Danh sách đơn hàng ,hệ thống sẽ hiển thị thông tin danh sách những đơn hàng. | |
| Tác nhân | Người quản lí | |
| Điều kiện trước | Người quản lí đăng nhập thành công vào hệ thống.Tiếp theo người quản lí chọn chức năng Xem Danh sách đơn hàng | |
| Điều kiện sau | Người quản lí xem được Danh sách đơn hàng | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người quản lí chọn chức năng Xem Danh sách đơn hàng |  |
|  | 2.Hệ thống hiển thị form danh sách những đơn hàng |
| 3.Người quản lí thực hiện đăng xuất hoặc đóng rồi kết thúc. |  |
| Luồng sự kiện phụ(Alternative Flows) |  | |

##### Biểu đồ

###### Activity



###### Sequence:

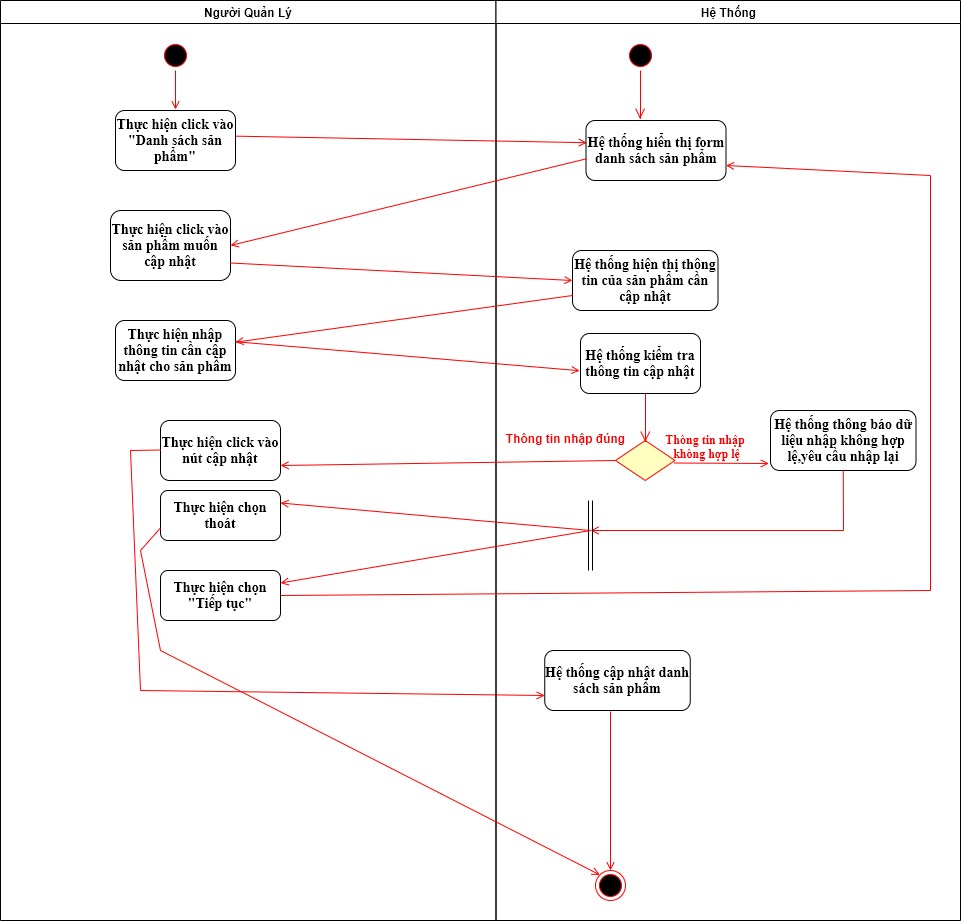
#### UC012\_Cập nhật sản phẩm

##### Mô tả use case UC012

|  |  |  |
| --- | --- | --- |
| Tên Use case | Cập nhật sản phẩm | UC012 |
| Mục đích | Người quản lí cập nhật thông tin sản phẩm vào hệ thống | |
| Mô tả | Khi người quản lí muốn cập nhật sản phẩm thì trước tiên cần xác định được t sản phẩm cần được cập nhậtsau đó chức năng sửa và điền các thông tin cần cập nhật. Cuối cùng chọn nút lưu khí đó hệ thống sẽ xử lý và thông báo kết quả xử lý cho người quản lí biết. | |
| Tác nhân | Người quản lí | |
| Điều kiện trước | Người quản lí đăng nhập thành công vào giao diện chính và có dữ sản phẩm muốn cập nhật. | |
| Điều kiện sau | Thông tin sản phẩm cập nhật được lưu vào cơ sở dữ liệu | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người quản lí chọn vào chức năng xem Danh sách sản phẩm |  |
|  | 2.Hệ thống sẽ hiển thị form Danh sách sản phẩm. |
| 3.Người quản lí chọn sản phẩm cần cập nhật |  |
|  | 4.Hệ thống hiển thị thông tin của sản phẩm cần cập nhật. |
| 5.Người quản lí nhập các thông tin sản phẩm cần cập nhật. |  |
|  | 6.Hệ thống kiểm tra thông tin vừa nhập |
| 7.Thực hiện chọn nút Lưu |  |
|  |  | 8.Hệ thống cập nhật thông tin vào CSDL |
|  | 9..Người quản lí đóng giao diện để kết thúc. |  |
| Luồng sự kiện phụ(Alternative Flows) |  | 6.1 Hệ thống kiểm tra đúng thì thì hiện tiếp bước 7  6.2 Hệ thống kiểm tra thông tin nhập sai,yêu cầu nhập lại |
| 6.2.1 Thực hiện chọn thoát  6.2.2 Chọn tiếp tục |  |
|  |  | 6.2.3 Quay lại bước 2 |

##### Biểu đồ

###### Activity



###### Senquence

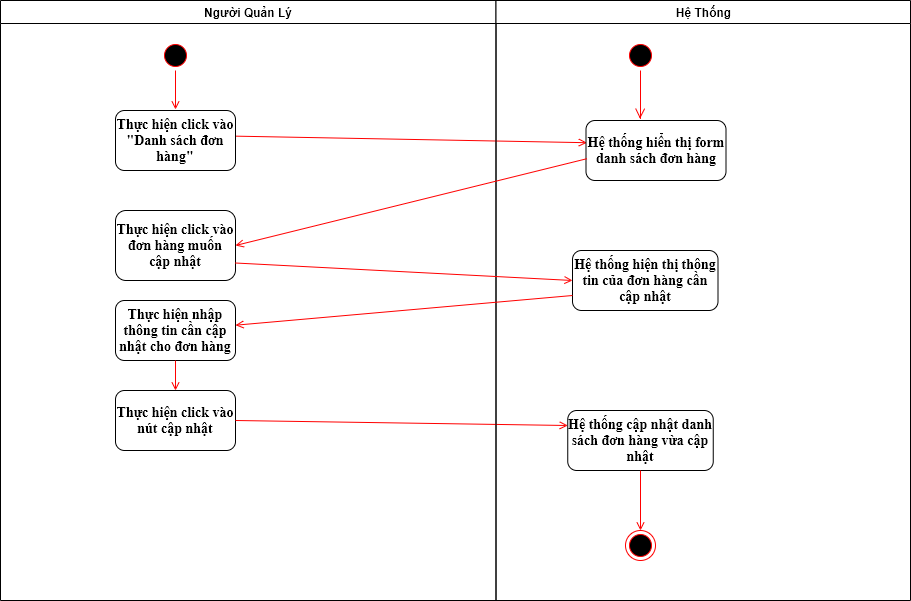
#### UC013\_Cập nhật đơn hàng

##### Mô tả use case UC013

|  |  |  |
| --- | --- | --- |
| Tên Use case | Cập nhật đơn hàng | UC013 |
| Mục đích | Người quản lí cập nhật được thông tin của đơn hàng | |
| Mô tả | Người quản lí sẽ chọn vào danh sách đơn hàng.Người quản lí phải xác định được đơn hàng cần cập nhật.Sau đó chọn đơn hàng và thực hiện nhập các thông tin cập nhật cho đơn hàng. Cuối cùng chọn nút lưu khí đó hệ thống sẽ xử lý và thông báo kết quả xử lý cho người quản lí biết. | |
| Tác nhân | Người quản lí | |
| Điều kiện trước | Người quản lí đăng nhập thành công vào hệ thống và có thông tin đơn hàng cần cập nhật. | |
| Điều kiện sau | Đơn hàng được cập nhật và lưu vào cơ sở dữ liệu | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người quản lý chọn vào mục Danh sách đơn hàng. |  |
|  | 2.Hê thống hiển thị form danh sách đơn hàng |
| 3.Người quản lý chọn vào đơn hàng cần cập nhật |  |
|  | 4.Hệ thống hiển thị thông tin của đơn hàng cần cập nhật. |
| 5.Thực hiện nhập các thông tin cần cập nhật của đơn hàng. |  |
| 6.Thực hiện chọn nút lưu. |  |
|  | 7.Hệ thống cập nhật thông tin đơn hàng xuống CSDL. |
|  | 8.Người quản lí đóng giao diện để kết thúc. |  |
| Luồng sự kiện phụ(Alternative Flows) |  |  |
|  |  |

##### Biểu đồ

###### Activity



###### Sequence:

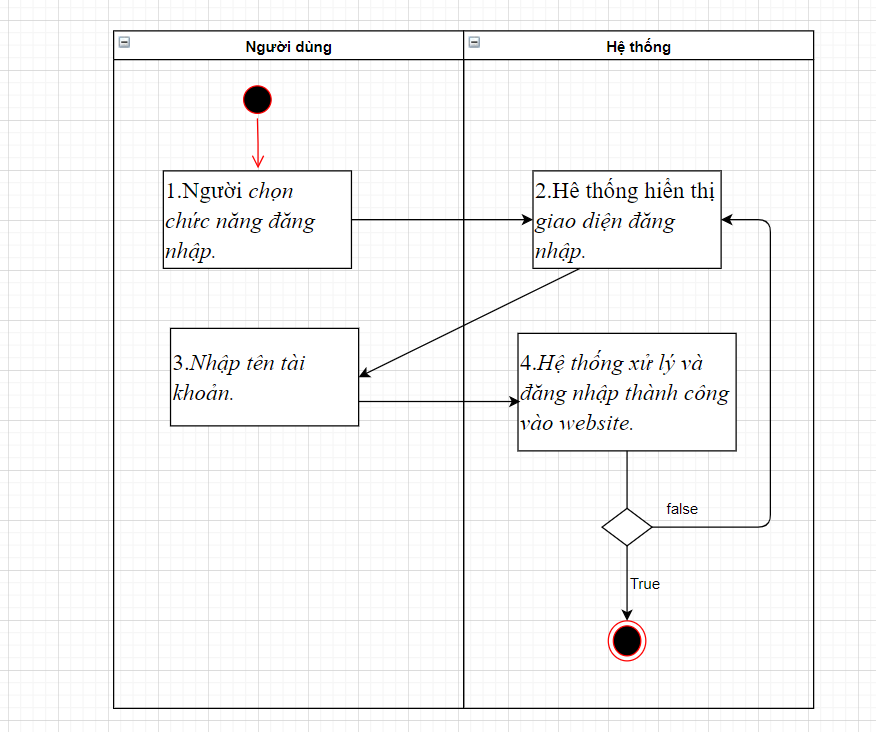
#### UC014\_Đăng nhập

##### Mô tả use case UC014

|  |  |  |
| --- | --- | --- |
| Tên Use case | Đăng nhập | UC014 |
| Mục đích | Người dùng đăng nhập vào hệ thống | |
| Mô tả | Người dùng muốn thanh toán đơn hàng hay xem thông tin cá nhân hoặc xem danh sách đơn hàng đã mua thì người dùng bắt buộc phải đăng nhập vào website. | |
| Tác nhân | Người dùng | |
| Điều kiện trước | Người dùng vào được website. | |
| Điều kiện sau | Đăng nhập thành công. | |
| Luồng sự kiện chính(Basic flows) | Tác nhân | Hệ thống |
| 1.Người chọn chức năng đăng nhập. |  |
|  | 2.Hê thống hiển thị giao diện đang nhập. |
| 3.Nhập tên tài khoản. |  |
|  | 4.Hệ thống xử lý và đăng nhập thành công vào website. |
| Luồng sự kiện phụ(Alternative Flows) |  | 4.1 Nếu sai tên đăng nhập thì quay lại bước 2 và thông báo sai tên đang nhập. |
|  |  |

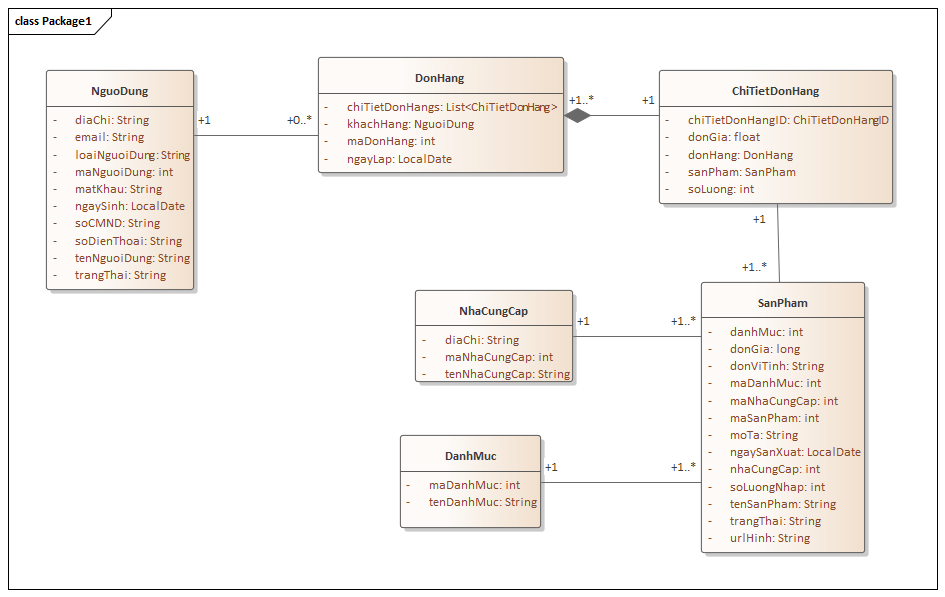
##### Biểu đồ

###### Activity:



###### Sequence:

## Class diagram



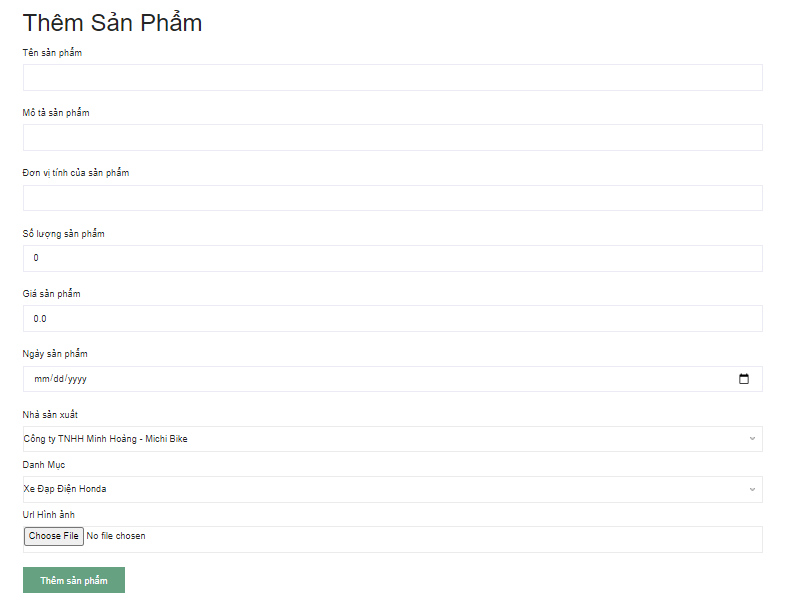
# : THIẾT KẾ VÀ HIỆN THỰC

Chương này sinh viên trình bày quá trình thiết kế, thử nghiệm và hiện thực hệ thống hoặc bài toán.

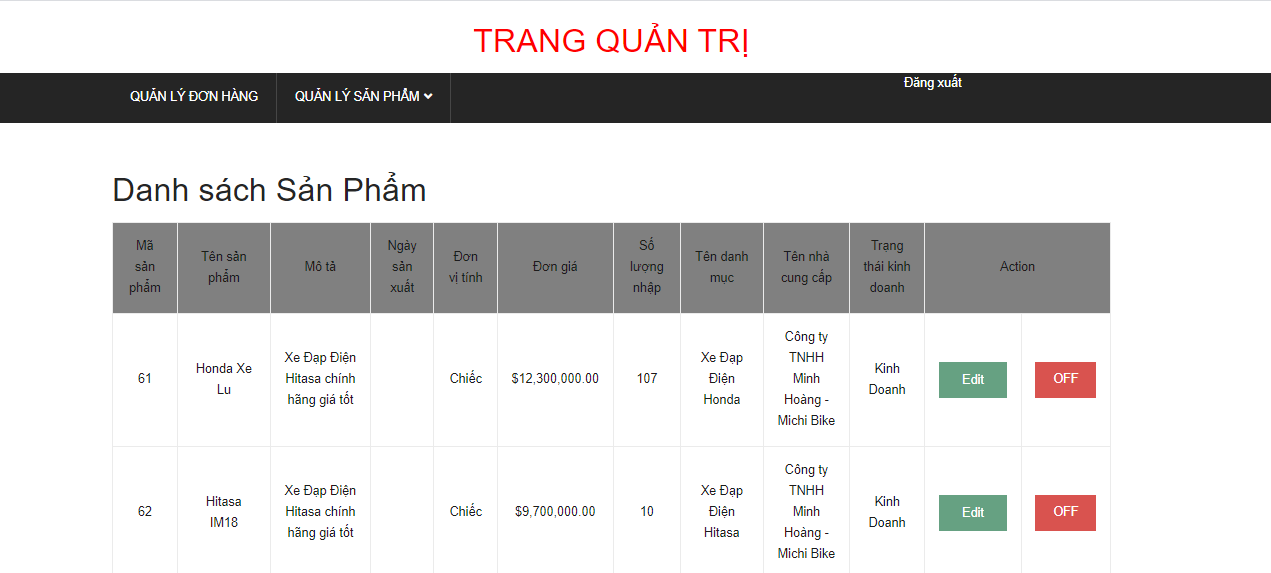
## Hiện thực giao diện

### Giao diện Admin

#### Giao diện thêm sản phẩm



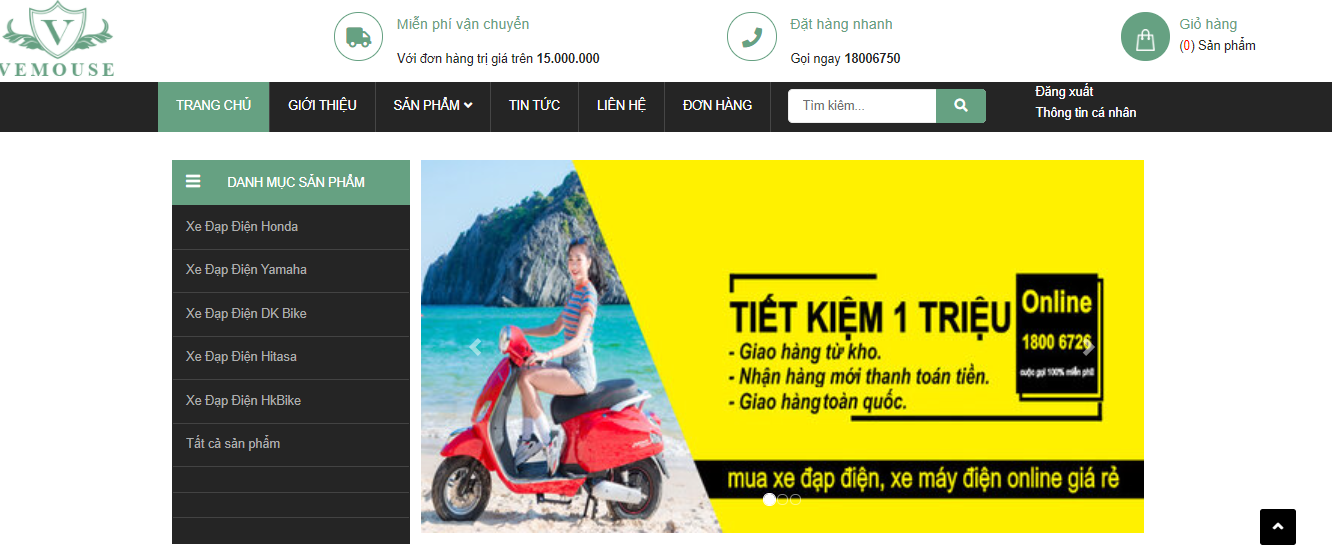
#### Giao diện danh sách sản phẩm.

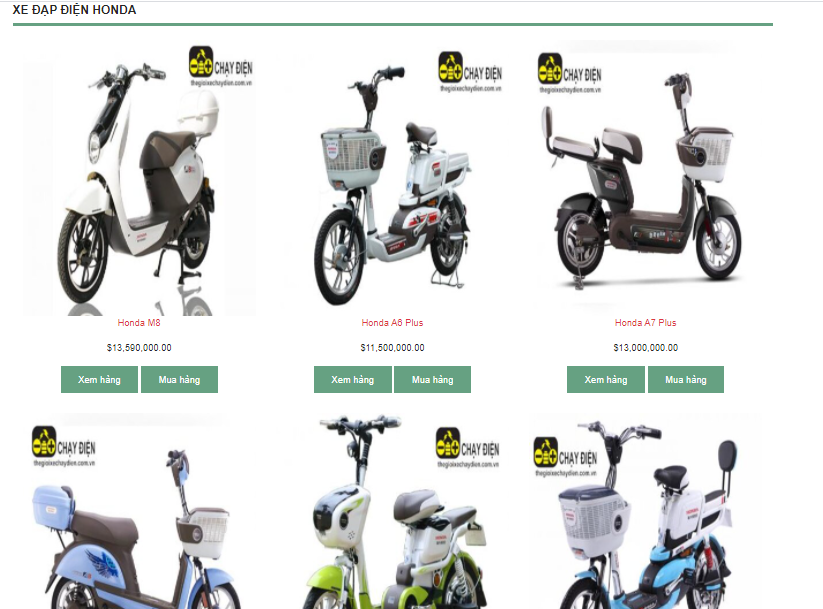


### Giao diện Khách hàng.

#### Giao diện trang chủ.

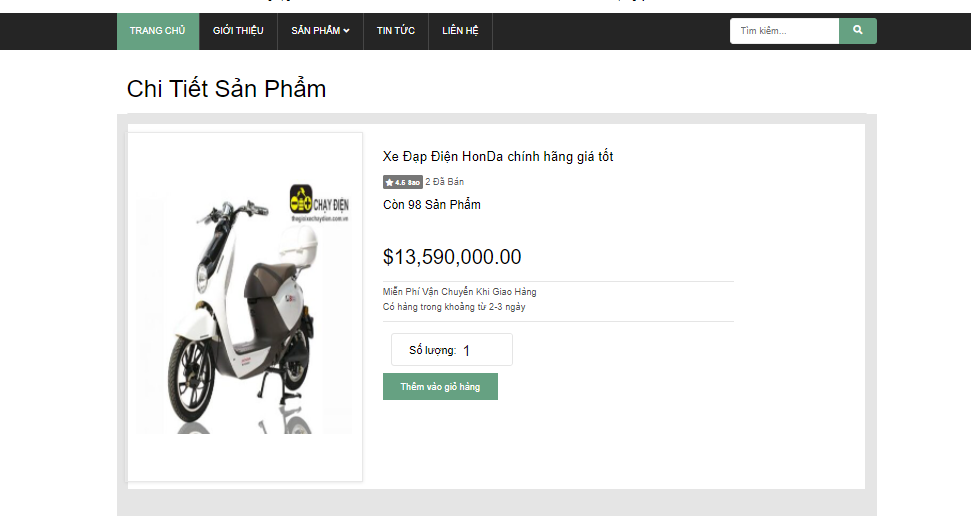
* Mô tả: là giao diện chính của website có chức năng hiện danh sách các sản phẩm mà hệ thống có giúp người dụng chọn sản phẩm dễ dàng.





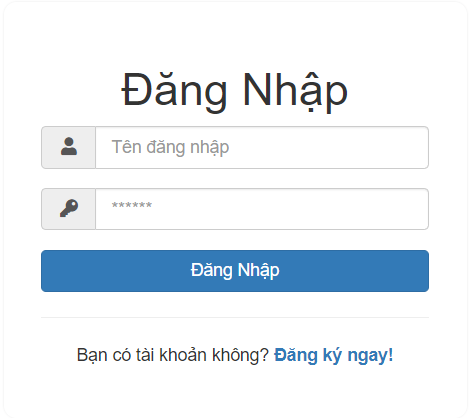
#### Giao diện Xem chi tiết sản phẩm:

* Mô tả: Xem chi tiết thông tin của một sản phẩm:



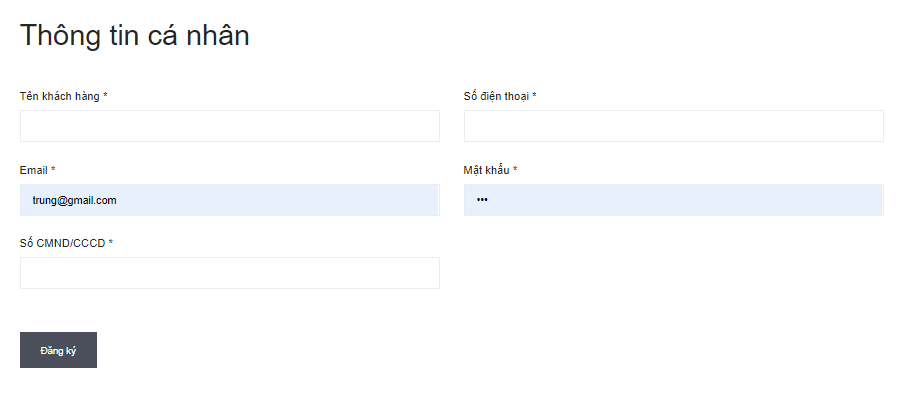
#### Giao diện đăng nhập.

* Mô tả: Thực hiện chức năng đăng nhập của khách hàng và người quản lý.



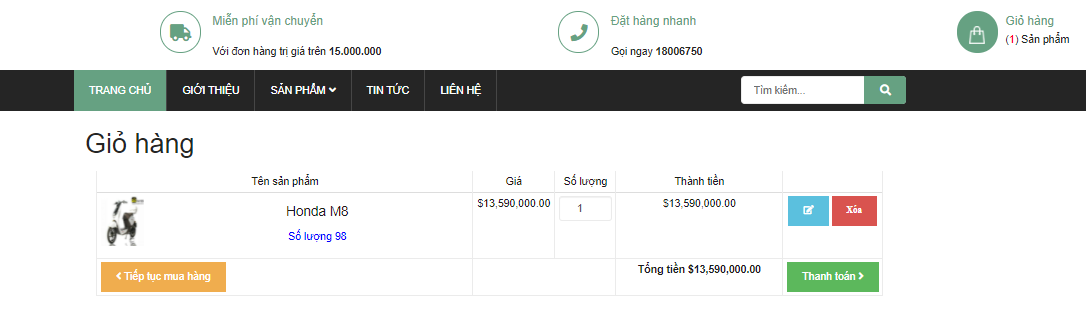
#### Giao diện đăng kí tài khoản.

* Mô tả: Thực hiện chức năng đăng kí tài khoản của khách hàng.



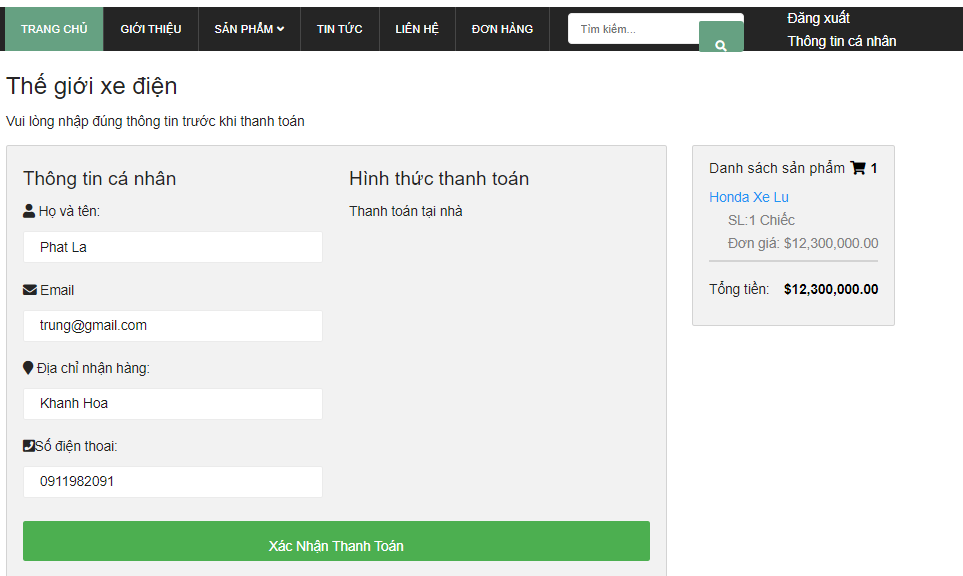
#### Giao diện xem giỏ hàng.

* Mô tả: Hiện thị danh sách sản phẩm có trong giỏ hàng.



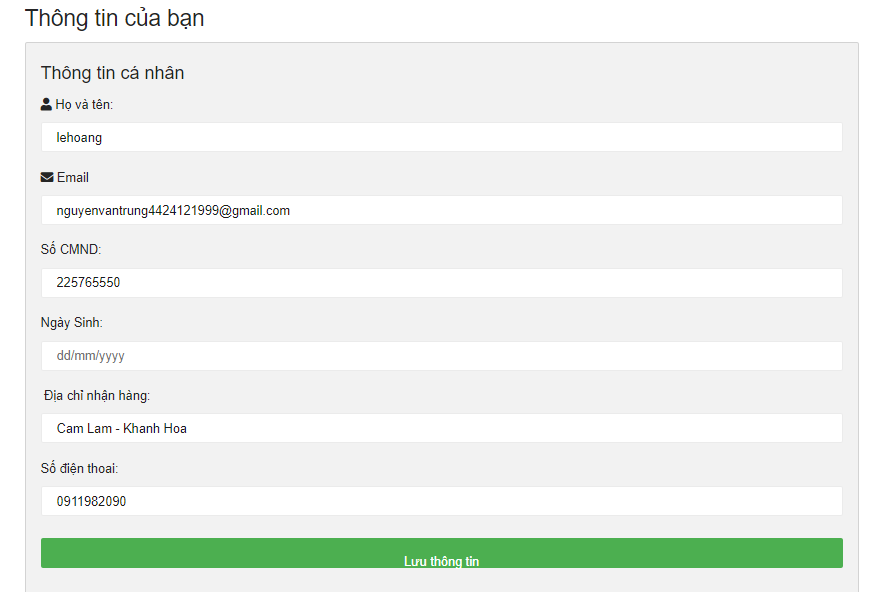
#### Giao diện thanh toán.

* Mô tả: Hiện thị form điền thông tin hộ trợ quá trình thanh toán.



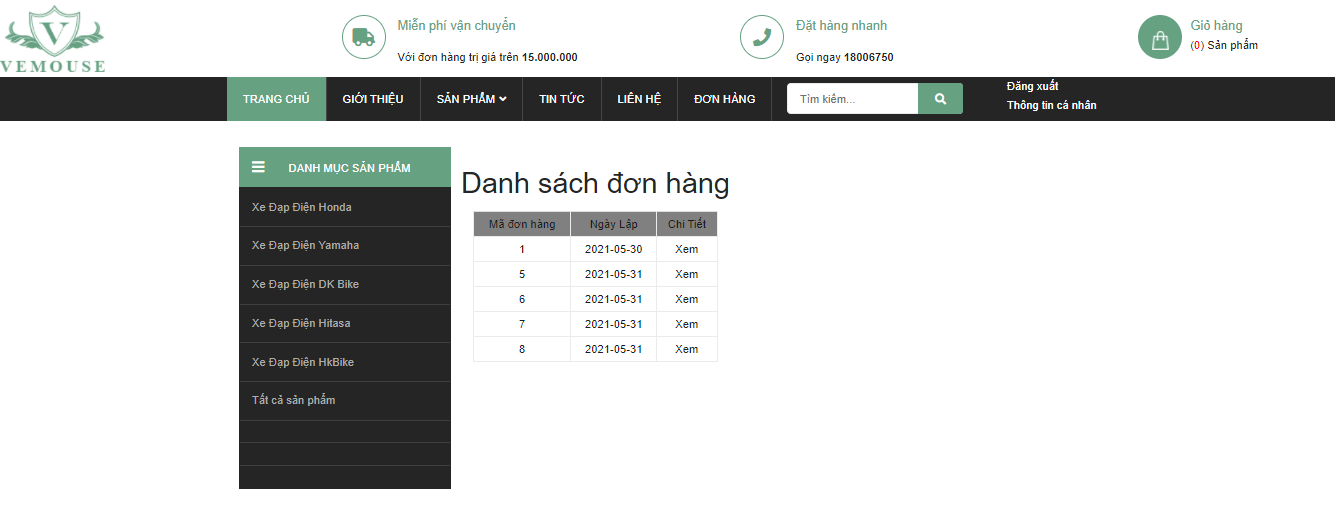
#### Giao diện xem thông tin các nhân của khách hàng.

* Mô tả: hiện thị thông tin của khách hàng.



#### Giao diện xem danh đơn hàng của khách hàng.

* Mô tả: Danh sách đơn hàng mà khách hàng đã đặt.



# : KẾT LUẬN

## Kết quả đạt được

Trình bày các chức năng đã hoàn thành trong đồ án hoặc kết quả nghiên cứu:

* Về Khách hàng:
  + Đăng nhập
  + Đăng Ký
  + Xem danh sách sản phẩm trên giao diện web site.
  + Tìm kiếm sản phẩm theo mong muốn.
  + Thêm hàng vào giỏ hàng.
  + Xem danh sách sản phẩm trong giỏ hàng.
  + Cập nhật số lượng sản phẩm hoặc xóa sản phẩm khỏi giỏ hàng.
  + Thanh toán đơn hàng.
  + Cập nhật thông tin cá nhân
  + Bảo mật tài khoản người dùng

## Hạn chế của đồ án

* Tìm kiếm chưa được tối ưu.
* Còn thiếu sót một số chức năng mà một website bán hàng phải có như bộ lọc, danh mục về khuyến mãi, ...
* Chưa gửi mail xác thực tài khoản

## Hướng phát triển

* Mở rộng phạm vị của đề tài.
* Thêm hình thức thanh toán online.
* Thêm các chức nâng cao để website trở nên chuyên nghiệp.
* Triển khai mở rộng phạm vi của đề tài.
* Áp dung nhiều công nghệ tối ưu hơn.

- Thiếu có chức năng cho admin.

- Xác thực email khách hàng.

# 

# TÀI LIỆU THAM KHẢO

Các tài liệu Tiếng Việt

1. <https://drive.google.com/drive/folders/14Dhh0kEtUzraH9daL0QpCFYqa0P_5FmK>

Các tài liệu từ Internet

1. <https://www.journaldev.com/3531/spring-mvc-hibernate-mysql-integration-crud-example-tutorial#spring-bean-configuration>
2. <https://www.journaldev.com/2715/spring-security-example-tutorial>

# PHỤ LỤC