|  |  |
| --- | --- |
|  |  |
| NGUYỄN NGỌC ANH | **BỘ CÔNG THƯƠNG**  **TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**  **---------------------------------------** |
|  |
| ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC NGÀNH KỸ THUẬT PHẦN MỀM |
|  |
| **XÂY DỰNG WEBSITE BÁN SÁCH CHO NHÀ SÁCH TRANG ANH SỬ DỤNG SPRING FRAMEWORK** |
|  |
|  |
| **CBHD :TS. Vũ Việt Thắng** |
| KỸ THUẬT PHẦN MỀM | **Sinh viên thực hiện : Nguyễn Ngọc Anh** |
| **Mã sinh viên : 2017605519** |
|  |
|  |
|  |
| Hà Nội – Năm 2022 |

# Lời cảm ơn

Em xin chân thành cảm ơn Khoa Công nghệ thông tin Trường Đại học Công nghiệp Hà Nội đã tạo điều kiện tốt cho em thực hiện tốt đề tài đồ án tốt nghiệp này.

Em xin chân thành cảm ơn quý thầy cô trong khoa Công nghệ thông tin đã tận tình giảng dạy, trang bị cho em những kiến thức quý báu trong những năm học qua, giúp em có một nền tảng kiến thức vững chắc để hoàn thành đề tài này. Đặc biệt em xin chân thành cảm ơn sự nhiệt tình hướng dẫn và đóng góp ý kiến của thầy *Vũ Việt Thắng* đã giúp em hoàn thành đề tài đồ án tốt nghiệp.

Mặc dù em đã cố gắng hoàn thành đề tài đồ án tốt nghiệp trong phạm vi và khả năng cho phép nhưng chắc chắn vẫn không tránh khỏi những thiếu sót. Em kính mong nhận được sự thông cảm và tận tình đóng góp ý kiến của quý thầy cô và các bạn.

*Em xin chân thành cảm ơn!*

Hà Nội, ngày ... tháng ... năm 2022

**Sinh viên thực hiện**

*Nguyễn Ngọc Anh*

MỤC LỤC

[Lời cảm ơn i](#_Toc102806035)

[Danh mục các thuật ngữ, ký hiệu và chữ viết tắt iv](#_Toc102806036)

[Danh mục hình ảnh iv](#_Toc102806037)

[Mở đầu 1](#_Toc102806038)

[Chương 1. Tổng quan 3](#_Toc102806039)

[1.1. Giới thiệu chung 3](#_Toc102806040)

[1.2. Những vẫn đề còn tồn tại 5](#_Toc102806041)

[Chương 2. Cơ Sở lý thuyết 6](#_Toc102806042)

[2.1. Giới thiệu về công cụ và công nghệ thực hiện 6](#_Toc102806043)

[2.1.1. Công cụ Intellij 6](#_Toc102806044)

[2.1.2. Công cụ MySQL 7](#_Toc102806045)

[2.3. Giới thiệu công nghệ Spring Framework 8](#_Toc102806046)

[2.3.1. Spring là gì 8](#_Toc102806047)

[Lợi ích của việc sử dụng Spring Framework 9](#_Toc102806048)

[2.3.2. Kiến trúc Spring 10](#_Toc102806049)

[Chương 3. Kết quả 14](#_Toc102806050)

[3.1 Phân tích thiết kế 14](#_Toc102806051)

[3.1.1 Mô hình hóa dữ liệu 14](#_Toc102806052)

[3.1.2 Mô hình hóa chức năng 16](#_Toc102806053)

[3.1.3 Biểu đồ trình tự 34](#_Toc102806054)

[3.2 Thiết kế giao diện 49](#_Toc102806055)

[3.2.1 Giao diện người dùng 49](#_Toc102806056)

[3.2.2 Giao diện quản trị 52](#_Toc102806057)

[3.3 Cài đặt hệ thống 55](#_Toc102806058)

[3.3.1 Chức năng User 55](#_Toc102806059)

[3.3.2 Chức năng Admin 55](#_Toc102806060)

[Kết luận 57](#_Toc102806061)

[Tài liệu tham khảo 58](#_Toc102806062)

# Danh mục các thuật ngữ, ký hiệu và chữ viết tắt

|  |  |  |
| --- | --- | --- |
| THUẬT NGỮ, KÝ HIỆU | Tên gọi đầy đủ | Dịch nghĩa |
| IDE | Integrated Development Environment | môi trường tích hợp dùng để viết code |
| POJO | Plain Old Java 0bject | Các đối tượng java thuần túy |
| ORM | Object Relational Mapping | ánh xạ các record dữ liệu trong hệ quản trị cơ sở dữ liệu sang dạng đối tượng mà mã nguồn đang định nghĩa trong class |
| MVC | Model View Controller | Là một mẫu kiến trúc phần mềm |
| API | Application programming interface | Giao diện lập trình ứng dụng |
| OXM | Object XML Mappers |  |
| JMS | Java Message Service |  |
| AOP | Aspect Oriented Programming | Lập trình hướng khía cạnh |
| CSDL |  | Cơ sở dữ liệu |
| JNDI | Java Naming Directory Interface |  |

# Danh mục hình ảnh

[*Hình 2. 1* *Spring Framework Runtime* 8](#_Toc102812185)

# Mở đầu

1. **Lý do chọn đề tài**

Ngày nay, ứng dụng về công nghệ và việc tin học hóa đã và đang được xem là một trong những yếu tố quyết định đối với mọi hoạt động của các doanh nghiệp, tổ chức, cũng như của các công ty lớn hay nhỏ, nó đóng vai trò hết sức quan trọng để có thể tạo ra những bước đột phá vô cùng mạnh mẽ.

Do đó, cùng với sự phát triển không ngừng về kỹ thuật máy tính điện tử, công nghệ thông tin cũng đã lần lượt chinh phục hết đỉnh cao này đến đỉnh cao khác. Internet là một trong những thứ có giá trị to lớn và ngày càng trở nên phát triển. Nó một công cụ không thể thiếu, là nền tảng chính cho sự truyền tải, trao đổi thông tin và kiểm soát dữ liệu trên toàn cầu.

Trong hoạt động sản xuất, kinh doanh nói chung và đối với thời trang nói riêng thương mại điện tử đã khẳng định được vai trò xúc tiến và thúc đẩy sự phát triển của các công ty hay doanh nghiệp. Đối với một cửa hàng hay những shop nhỏ lẻ, việc quảng bá và giới thiệu đến khách hàng các sản phẩm mới đáp ứng được nhu cầu của khác hàng sẽ là rất cần thiết.

Sách là kho tàng tri thức vô cùng quý giá của nhân loại, là nguồn kiến thức vô tận và không bao giờ cạn kiệt. Đọc sách không chỉ để tiếp nhận kiến thức mà còn để hoàn thiện nhân cách, phát triển con người. Rất nhiều người muốn tìm mua những cuốn sách để đọc, nhưng do công việc bận rộn, hay một lý do nào đó mà nhiều người chưa thể sở hữu cho mình cuốn sách muốn đọc. Vì lý do đó nên em quyết định xây dựng một website bán sách, để có thể đưa nguồn kiến thức phong phú này đến với nhiều người dễ dàng hơn.

Đề tài: Xây dựng ứng dụng web bán sách bằng ngôn ngữ java trên nền tảng Spring framework.

1. **Mục tiêu nghiên cứu**

Tìm kiếm thông tin và xây dựng chức năng của một website bán sách giúp các cửa hàng có thể dễ dàng hơn trong việc kinh doanh và khách hàng dễ dàng khi mua sản phẩm.

Xây dựng và đánh giá sản phẩm giúp cho việc bán hàng và việc mua hàng hiệu quả hơn đồng thời xây dựng cho người quản trị dễ dàng trong việc phân quyền cũng như quản lý thông tin của khách hàng và nhân viên.

Việc phân tích thiết kế cho website để đảm bảo thông tin của người dùng an toàn và được bảo mật tránh tình trạng lộ thông tin người dùng.

1. **Đối tượng nghiên cứu**

* Tìm hiểu về các trang web thực tiễn.
* Sử dụng ngôn ngữ lập trình Java trên nền tảng Spring framework.
* Sử dụng cơ sở dữ liệu My sql để lưu trữ dữ liệu.
* Công cụ lập trình Intellij.

1. **Ý nghĩa khoa học, thực tiễn của đề tài**

Tạo ra một website bán sách :

+ Thân thiện với người dùng, dễ sử dụng.

+ Bảo trì dễ dàng

+ Có các chức năng cơ bản đáp ứng cho quản trị viên và khách hàng.

1. **Cấu trúc báo cáo**

Cấu trúc báo cáo gồm 3 chương:

Chương 1: Tổng quan

Chương 2: Cơ sở lý thuyết

Chương 3: Kết quả

# Chương 1. Tổng quan

## 1.1. Giới thiệu chung

#### 1.1.1. Mô tả đề tài

Ứng dụng web bán sách và đánh giá sản phẩm là hệ thống quảng bá hình ảnh cửa hàng cùng với sự chăm sóc khách hàng mọi lúc mọi nơi. Với trang web này khách hàng không cần đến các cửa hàng hay nhà phân phối.

#### 1.1.2. Tiền đề xây dựng ứng dụng

Giờ đây mọi việc liên quan đến thông tin thật dễ dàng cho người sử dụng: chỉ cần có một máy tính kết nối internet và một dòng dữ liệu truy tìm thì gần như ngay lập tức cả thế giới vấn đề bạn đang quan tâm sẽ hiện ra, có đầy đủ thông tin, hình ảnh âm thanh, video….

Bằng internet chúng ta thực hiện được nhiều công việc với tốc độ nhanh hơn và chi phí thấp hơn nhiều so với cách thức truyền thống. Chính điều này đã thúc đẩy sự khai sinh và phát triển của thương mại điện tử và chính phủ điện tử trên khắp thế giới, làm thay đổi đáng kể bộ mặt văn hóa, nâng cao chất lượng cuộc sống con người.

Xây dựng các trang web bán hàng. Tạo không gian mua bán sản phẩm rộng lớn lớn. Tích kiệm nhiều chi phí mà còn đạt được doanh số bán hàng gấp rất nhiều lần với bán hàng thông thường.

#### 1.1.3. Các hình thức quản lí ứng dụng web bán sách và đánh giá sản phẩm

- Các hình thức quản lí người quản trị

* Quản lý khách hàng
* Quản lý sản phẩm
* Quản lý chủ đề
* Quản lý tác giả
* Quản lý nhà xuất bản
* Quản lý đơn hàng

- Các hình thức quản lí của khách hàng

* Quản lý giỏ hàng
* Quản lý tài khoản người dùng
* Quản lý thông tin cá nhân

#### 1.1.4. Ngôn ngữ hỗ trợ việc xây dựng website

**-** Sử dụng ngôn ngữ Java trên công cụ Intellij

* Back end : Xây dựng project trên nền tàng Spring Framework như Spring MVC, Spring data jpa, hibernate, Spring security.... Và sử dụng cơ sở dữ liệu My Sql đề lưu trữ dữ liệu.
* Front end : Sử dụng HTML, CSS, Bootstrap, Ajax, JavaScript và sử dụng template.

#### 1.1.5. Ưu điểm và nhược điểm khi sử dụng mua hàng qua ứng dụng web bán sách và đánh giá sản phẩm

* Ưu điểm

- Tiết kiệm thời gian và công sức: Thời gian là lợi ích lớn nhất khi nói đến mua sắm online. Bạn tiết kiệm thời gian tìm kiếm và đến một cửa hàng. Bạn cũng tiết kiệm được chi phí nhiên liệu. Có thể mua sắm online thoải mái tại nhà hoặc từ văn phòng. Nó làm cho việc mua sắm dễ dàng ở hầu hết mọi nơi chỉ cần có internet.

- Mua hàng online giá cả phải chăng, giá thành sản phẩm rẻ hơn: hầu hết các trang web bán hàng trực tuyến uy tín lớn và lâu đời hay các sàn thương mại điện tử theo hình thức bán hàng trên mạng tôi đang nói này đều có giá rẻ hơn giá bán trong các cửa hàng truyền thống rất nhiều. Bởi vì một điều dễ hiểu đó chính là họ không mất tiền thuê mặt bằng, thuê nhân viên và bảo vệ cũng sẽ ít đi, các phụ phí phát sinh khác trong quá trình sinh hoạt như tiền điện, nước, môi trường, chi phí này nọ… sẽ ở mức thấp nhất tối ưu nhất.

- Thỏa sức lựa chọn sản phẩm phù hợp với mục đích và túi tiền của khách hàng: người mua có thể xem và lựa chọn hàng trăm sản phẩm với vài bước tìm kiếm. Lựa chọn thỏa thích mà không làm ảnh hưởng đến người bán hàng. Tìm ra những sản phẩm phù hợp với từng đối tượng khách hàng.

* Nhược điểm

- Mất thời gian giao hàng : mỗi sản phẩm từ những khu vực khác nhau vì vậy cần một bên trung gian để vận chuyển đến tay người dùng .Điều đó phát sinh chi phí giao hàng và một khoảng thời gian trước khi khách hàng nhận được sản phẩm.

## 1.2. Những vẫn đề còn tồn tại

- Hệ thống vẫn còn thiếu sót nhiều chức năng như: thanh toán online…

- Hệ thống được thiết kế chỉ dành cho quy mô bán hàng nhỏ.

# Chương 2. Cơ Sở lý thuyết

## 2.1. Giới thiệu về công cụ và công nghệ thực hiện

### 2.1.1. Công cụ Intellij

#### 2.1.1.1 Khái niệm

**IntelliJ IDEA** là một [IDE](https://vi.wikipedia.org/wiki/IDE) [Java](https://vi.wikipedia.org/wiki/Java_(ng%C3%B4n_ng%E1%BB%AF_l%E1%BA%ADp_tr%C3%ACnh)) để phát triển các phần mềm máy tính. Nó được phát triển bởi JetBrains (trước đây gọi là IntelliJ), nó được cấp phép [Apache 2](https://vi.wikipedia.org/wiki/Gi%E1%BA%A5y_ph%C3%A9p_Apache) cho phiên bản cộng đồng,và một phiên bản [thương mại](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_th%C6%B0%C6%A1ng_m%E1%BA%A1i) [độc quyền](https://vi.wikipedia.org/wiki/Ph%E1%BA%A7n_m%E1%BB%81m_s%E1%BB%9F_h%E1%BB%AFu_%C4%91%E1%BB%99c_quy%E1%BB%81n).Cả hai có thể được sử dụng cho phát triển thương mại.

#### 2.1.1.2 Lịch sử phát triển

Phiên bản đầu tiên của IntelliJ IDEA được phát hành vào tháng 1/2001, và là một trong những IDE Java có sẵn đầu tiên với khả năng điều hướng mã và [cải tiến mã nguồn](https://vi.wikipedia.org/wiki/C%E1%BA%A3i_ti%E1%BA%BFn_m%C3%A3_ngu%E1%BB%93n) được tích hợp.

Tuy nhiên, năm 2010 *Infoworld* báo cáo, IntelliJ đã nhận được điểm trung bình kiểm tra thấp nhất vì tồn tại nhiều lỗi trong số bốn công cụ lập trình Java hàng đầu: [Eclipse](https://vi.wikipedia.org/wiki/Eclipse), IntelliJ IDEA, [NetBeans](https://vi.wikipedia.org/wiki/NetBeans) và [JDeveloper](https://vi.wikipedia.org/w/index.php?title=JDeveloper&action=edit&redlink=1).

Tháng 12/2014, Google giới thiêu phiên bản [Android Studio](https://vi.wikipedia.org/wiki/Android_Studio) 1.0, một IDE nguồn mở cho các ứng dụng [Android](https://vi.wikipedia.org/wiki/Android_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)), dựa trên phiên bản cộng đồng nguồn mở của IntelliJ IDEA.Các môi trường phát triển khác dựa trên framework của IntelliJ bao gồm [AppCode](https://vi.wikipedia.org/w/index.php?title=AppCode&action=edit&redlink=1), CLion, PhpStorm, PyCharm, RubyMine.

Hệ điều hành và ngôn ngữ

Hỗ trợ trên nhiều hệ điều hành như: Windows, MacOS, Linux.

Ngôn ngữ hỗ trợ: Java, JavaScript, HTML, CSS, Ruby…

#### 2.1.1.3 Lợi ích của công cụ IntelliJ IDEA

* Xử lý được các mã hỗn hợp của Java, Ruby,….
* Cho phép viết , debug một cách nhanh chóng.
* Được thiết kế theo từng quy mô riêng: cá nhân, doanh nghiệp.
* Hỗ trợ các framework và các ngôn ngữ chính.

#### 2.1.1.4 Các tính năng của IntelliJ IDEA

* Sửa lỗi nhanh.
* Tự động tạo ra các mã hỗ trợ cho việc viết code.
* Phát hiện việc lặp mã.
* Được tích hợp kiểm thử Junit và TestNG.
* Tương tác được với Eclipse và Maven.

### 2.1.2. Công cụ MySQL

MySQL là một mã nguồn mở phổ biến nhất. MySQL là hệ quản trị cơ sở dữ liệu ổn định, tốc độ tốt, rất dễ dàng sử dụng, tính bảo mật cao

MySQL được tích hợp trên rất nhiều hệ điều hành khác nhau ví dụ như : Windows, Linux, Mac OS X, Solaris, Unix, ….

MySQL sử dụng ngôn ngữ truy vấn có cấu trúc.

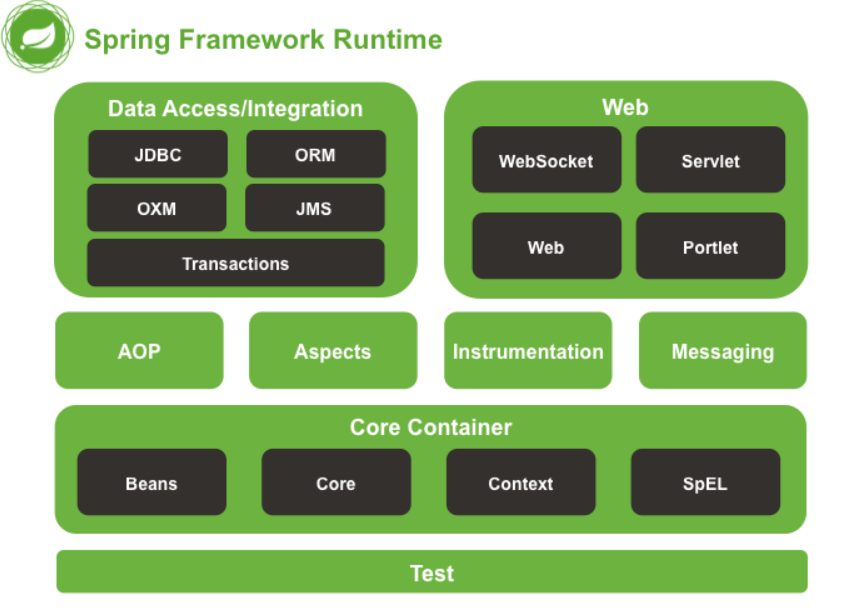
## 2.3. Giới thiệu công nghệ Spring Framework

### 2.3.1. Spring là gì

**Spring** là framework phát triển ứng dụng phổ biến nhất dành cho Java Enterprise. Ban đầu nó được viết bởi **Rod Johnson** và lần đầu tiên được phát hành theo giấy phép Apache 2.0 vào tháng 6 năm 2003. Spring có kích nhẹ, phiên bản cơ bản của Spring framework có kích thước khoảng 2MB.

**Spring framework** là một Java Platform mã nguồn mở, một giải pháp gọn nhẹ dành cho Java Enterprise. Với Spring Framework các nhà phát triển có thể tạo ra các mã có hiệu suất cao, dễ kiểm thử và có thể sử dụng lại được.

Các tính năng core của Spring Framework có thể được sử dụng trong việc phát triển bất kỳ ứng dụng Java nào. Bên cạnh đó, phần mở rộng được sử dụng để xây dựng các ứng dụng web trên nền tảng Java EE. Mục tiêu của Spring Framework là làm cho việc phát triển ứng dụng J2EE dễ dàng hơn và thúc đẩy việc lập trình tốt hơn bằng mô hình POJO-based.



*Hình 2. 1* *Spring Framework Runtime*

### Lợi ích của việc sử dụng Spring Framework

Dưới đây là danh sách các lợi ích tuyệt vời của việc sử dụng Spring Framework:

* Spring cho phép các nhà phát triển tạo các ứng dụng cấp Enterprise sử dụng các POJO. Lợi ích của việc sử dụng các POJO là bạn không cần một sản phẩm chứa EJB như một máy chủ ứng dụng, mà bạn chỉ có thể sử dụng một bộ chứa servlet mạnh mẽ như Tomcat hoặc một số sản phẩm thương mại khác.
* Spring được tổ chức theo kiểu mô đun. Mặc dù số lượng các gói và các lớp là khá nhiều, nhưng bạn chỉ cần quan tâm đến những gì bạn cần và không cần quan tâm đến phần còn lại.
* Spring sử dụng một số công nghệ hiện có như một số [ORM Framework](https://viettuts.vn/hibernate/orm-la-gi), logging frameworks, Quartz, JDK timers và các công nghệ View khác.
* Dễ dàng để kiểm thử một chương trình được viết bằng Spring.
* Web framework của Spring là một Web MVC framework có thiết kế tốt, nó là một thay thế tuyệt vời cho Struts và các công nghệ kém phổ biến khác.
* Spring cung cấp một API thuận tiện để dịch các ngoại lệ công nghệ cụ thể (ném bởi JDBC, Hibernate) vào các trường hợp ngoại lệ nhất quán, không được kiểm soát.
* IoC Container có trọng lượng nhẹ. Điều này có lợi cho việc phát triển và triển khai các ứng dụng trên các máy tính có bộ nhớ và tài nguyên CPU hạn chế.
* Spring cung cấp một giao diện quản lý transaction nhất quán có thể mở rộng đến một local transaction (ví dụ như sử dụng một cơ sở dữ liệu) và mở rộng lên các global transaction.

### 2.3.2. Kiến trúc Spring

#### 2.3.2.1 Dependency Injection và Inversion of Control

Dependency Injection (DI) là một design pattern để loại bỏ sự phụ thuộc vào mã lập trình, vì vậy nó giúp dễ dàng quản lý và test ứng dụng. Dependency Injection giúp mã lập trình liên kết lỏng với nhau.

Inversion of Control (IoC) Container có trách nhiệm khởi tạo, configure, gắn kết các đối tượng. Có hai kiểu của Io containers là BeanFactory và ApplicationContext.

#### 2.3.2.2 Các module của Spring Framework

Spring Framework bao gồm nhiều tính năng được tổ chức thành khoảng 20 module. Các module này được phân nhóm thành Core Container, Data Access/Integration, Web, AOP (Aspect Oriented Programming), Instrumentation, Messaging, và Test

**Core Container**

Core Container bao gồm các module: *spring-core*, *spring-beans*, *spring-context*, *springcontext-support*, và *spring-expression*(Spring Expression Language).

* Module **spring-core** và **spring-beans** cung cấp những phần cơ bản của framework, bao gồm Inversion of Control và Dependency Injection.
* Module **spring-context**xây dựng trên một nền tảng vững chắc được cung cấp bởi các module Core và Beans, được sử dụng để truy cập vào các đối tượng trong framework-style tương tự như việc đăng ký một JNDI. **spring-context-support**hỗ trợ tích hợp libraries của bên thứ 3 (third-party) vào ứng dụng Spring như caching (EhCache, Guava, JCache), mailing (JavaMail), schedule (CommonJ, Quatz) và các template engine (FreeMarker, JasperReports, Velocity)).
* Module **spring-expression** cung cấp một **Expression Language** mạnh mẽ cho việc truy vấn và tính toán một đồ thị đối tượng lúc runtime. Nó được kế thừa và mở rộng từ unified expression language (unified EL). Ngôn ngữ này hỗ trợ setting và getting giá trị các property, gọi phương thức, truy cập vào nội dung của mảng, tập hợp và chỉ mục, operator logic và tính toán, đặt tên biến, và truy xuất các đối tượng theo tên từ IoC container của Spring.

**Data Access/Integration**

Data Access/Integration bao gồm các module JDBC, ORM, OXM, JMS và Transaction.

* Module **spring-jdbc**cung cấp một lớp JDBC-abstraction để loại bỏ những code tẻ nhạt cả JDBC và phân tích những mã lỗi cụ thể của database-vendor.
* Module **spring-orm**cung cấp lớp tích hợp với các orm API phổ biến như JPA và Hibernate.
* Module **spring-oxm**cung cấp lớp abstraction hỗ trợ triển khai Object/XML mapping như AXB, Castor, XMLBeans, JiBX and XStream.
* Module **spring-transaction**hỗ trợ quản lý giao dịch theo chương trình và khai báo cho các lớp mà thực hiện các giao diện đặc biệt và cho tất cả các POJO của bạn.

**Web**

Tầng Web bao gồm các module *spring-web*, *spring-webmvc*, *spring-websocket* và *springwebmvc-portlet*.

* Module **spring-web**cung cấp tính năng tích hợp web theo định hướng cơ bản như chức năng tập tin tải lên nhiều phần dữ liệu và khởi tạo các container IoC sử dụng nghe servlet và một bối cảnh ứng dụng web theo định hướng.
* Module **spring-webmvc**bao gồm việc triển khai Model-View-Controller (MVC) của Spring cho các ứng dụng web.
* Module **spring-websocket**cung cấp hỗ trợ cho WebSocket-based, giao tiếp hai chiều giữa client và server trong các ứng dụng web.
* Module **springwebmvc-portlet**cung cấp việc triển khai MVC được sử dụng trong môi trường portlet và ánh xạ chức năng của module Web-Servlet.

**Các module khác**

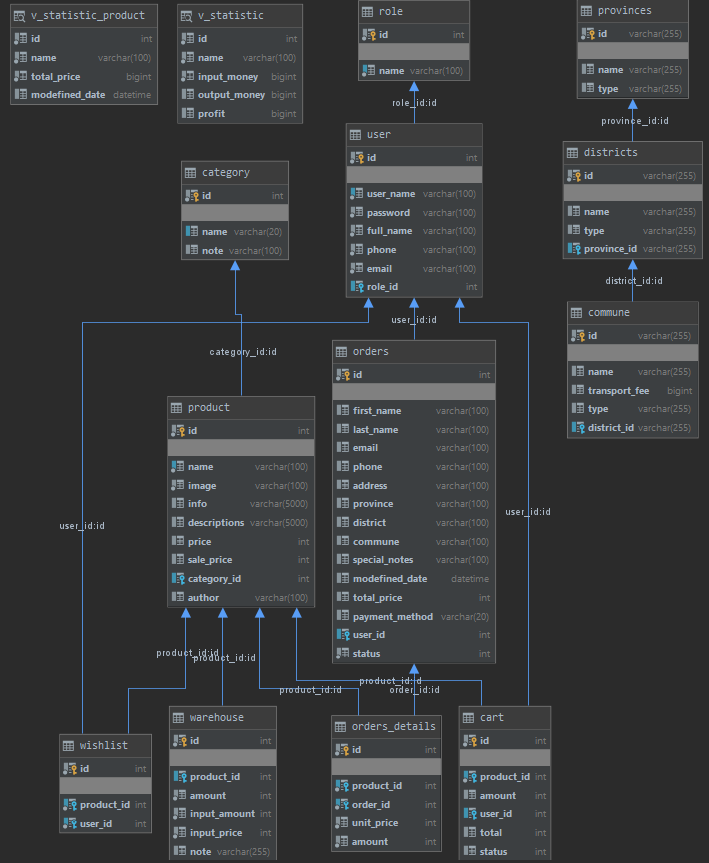
Có vài module quan trọng khác như module AOP, Aspects , Instrumentation, Web và Test mô tả chi tiết như sau:

* Module **AOP** cung cấp một thực hiện lập trình hướng khía cạnh cho phép bạn xác định phương pháp-chặn và pointcuts để sạch tách mã thực hiện chức năng đó nên được tách ra.
* Module **Aspects** cung cấp tích hợp với AspectJ, mà lại là một khuôn khổ AOP mạnh mẽ và trưởng thành.
* Module **Instrumentation** cung cấp thiết bị đo đạc lớp hỗ trợ và triển khai lớp bộ nạp được sử dụng trong các máy chủ ứng dụng nhất định.
* Module **Messaging** cung cấp hỗ trợ cho STOMP như WebSocket sub-protocol để sử dụng trong các ứng dụng. Nó cũng hỗ trợ một mô hình lập trình chú thích cho việc định tuyến và xử lý tin nhắn STOMP từ các máy khách WebSocket.
* Module **Test** hỗ trợ việc kiểm tra các thành phần spring với JUnit hoặc TestNG khuôn khổ.

# Chương 3. Kết quả

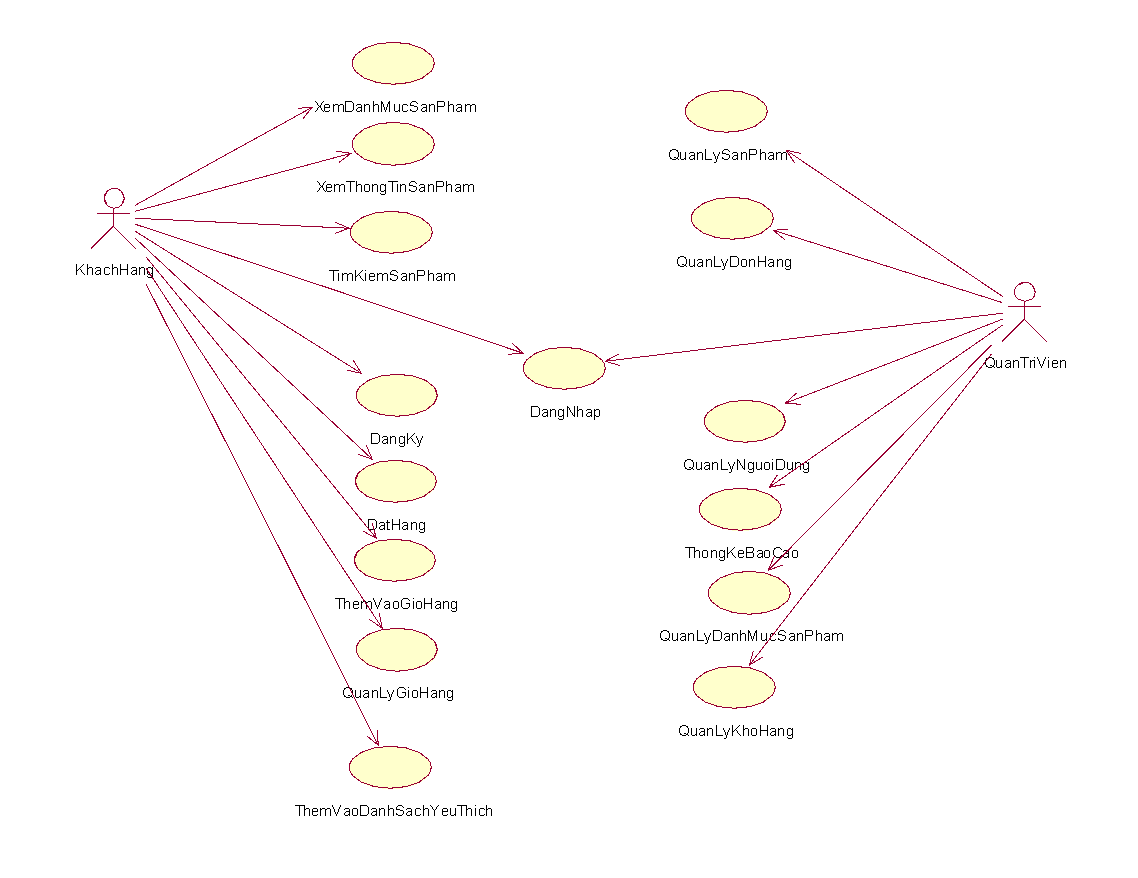
## 3.1 Phân tích thiết kế

### 3.1.1 Mô hình hóa dữ liệu

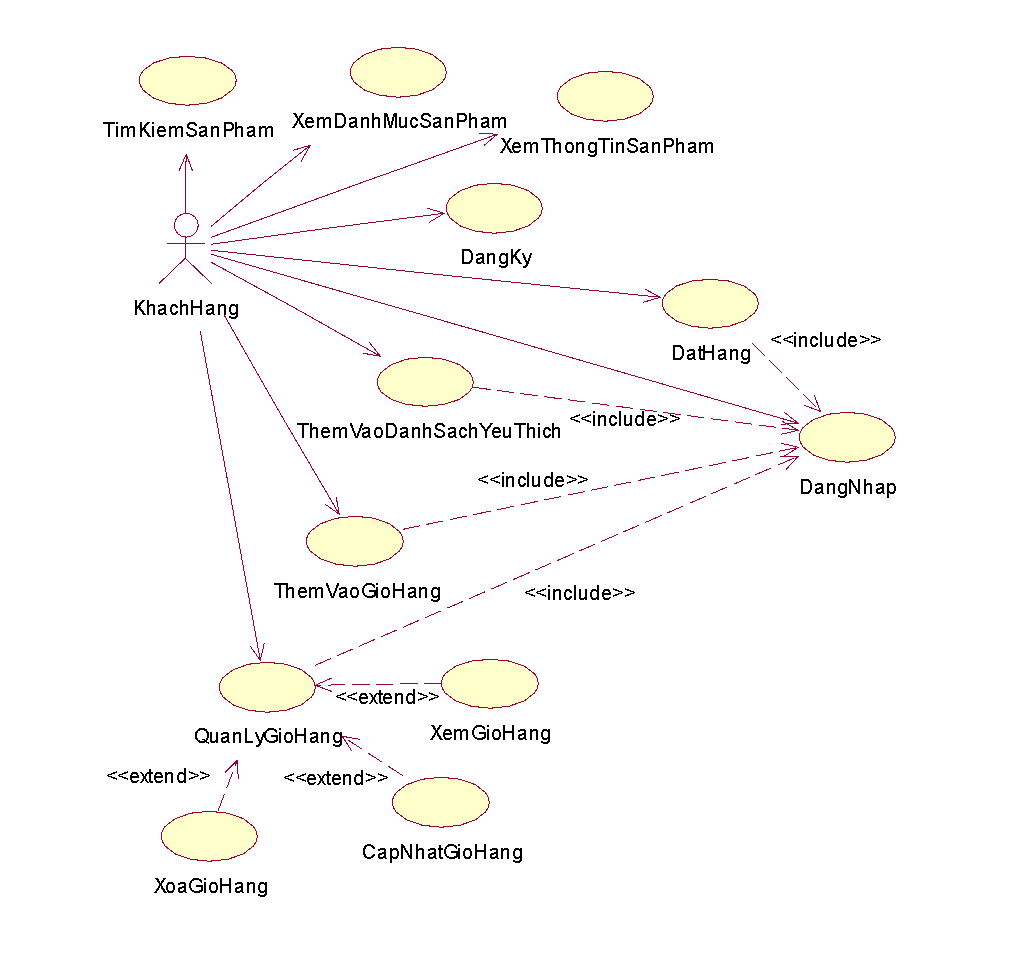


Hình 3. 1 Biểu đồ liên kết Database Diagrams

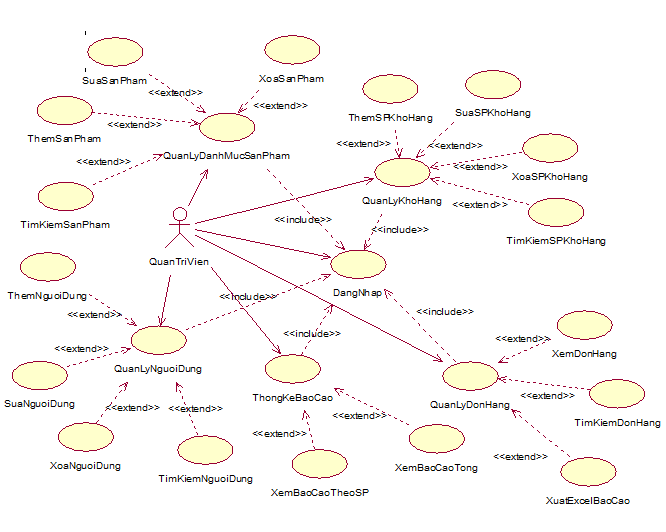
### 3.1.2 Mô hình hóa chức năng



Hình 3. 2 Biểu đồ use case



Hình 3. 3 Biểu đồ phân rã use case tác nhân khách hàng



Hình 3. 4 Biểu đồ phân rã use case tác nhân quản trị viên

* + - 1. **Mô tả Use Case XemDanhMucSanPham**
* **Tên use case:** Xem danh mục sản phẩm.
* **Mô tả vắn tắt:** Use case này cho phép người sử dụng có thể xem được danh mục sản phẩm có trong website.
* **Luồng sự kiện**
  + **Luồng cơ bản**

Use case bắt đầu khi người sử dụng kích vào các danh mục trên menu chính của website. Hệ thông lấy thông tin từ bảng CATEGORY trong cơ sở dữ liệu và hiển thị lên màn hình. Use case kết thúc.

* + **Luồng rẽ nhánh**

Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.

* **Các yêu cầu đặc biệt**

Không có.

* **Tiền điều kiện**

Không có.

* **Hậu điều kiện**

Không có.

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case TimKiemSanPham**
* **Tên use case:** Tìm kiếm sản phẩm.
* **Mô tả vắn tắt:** Use case này cho phép người dùng tìm kiếm các sản phẩm của cửa hàng trên website.
* **Luồng sự kiện**
  + **Luồng cơ bản**

Use case này bắt đầu khi người dùng chọn vào ô tìm kiếm sản phẩm tại trang chủ của website. Người dùng nhập tên sản phẩm vào ô tìm kiếm rồi ấn nút tìm kiếm. Hệ thống sẽ lấy thông tin của sản phẩm từ bảng PRODUCT và hiện lên màn hình. Use case kết thúc.

* + **Luồng rẽ nhánh**

1. Nếu người dùng không nhập tên sản phẩm mà vẫn ấn vào tìm kiếm thì hệ thống sẽ hiển thị lại màn hình tìm kiếm để người dùng nhập lại tên sản phẩm muốn tìm.
2. Nếu người dùng nhập sai tên sản phẩm mà vẫn ấn vào tìm kiếm thì hệ thống sẽ hiển thị màn hình không tìm kiếm được sản phẩm nào.
3. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ không làm việc và use case kết thúc.

* **Các yêu cầu đặc biệt**

Không có

* **Tiền điều kiện**

Không có

* **Hậu điều kiện**

Không có

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case XemThongTinSanPham**
* **Tên use case:** Xem thông tin sản phẩm.
* **Mô tả vắn tắt:** Use case này cho phép người dùng xem thông tin chi tiết của sản phẩm.
* **Luồng sự kiện**
  + **Luồng cơ bản**

Use case này bắt đầu khi người dùng click vào 1 sản phẩm, hệ thống sẽ lấy thông tin từ bảng PRODUCT trong cơ sở dữ liệu và hiển thị lên màn hình. Use case kết thúc.

* + **Luồng rẽ nhánh**

1. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case kết thúc.

* **Các yêu cầu đặc biệt**

Không có

* **Tiền điều kiện**

Không có

* **Hậu điều kiện**

Không có

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case DangKy**
* **Tên use case:** Đăng ký.
* **Mô tả vắn tắt:** Use case này cho phép người dùng đăng ký tài khoản để sử dụng thêm 1 số chức năng của trang web.
* **Luồng sự kiện**
  + **Luồng cơ bản**

Use case bắt đầu khi người dùng kích vào nút “Đăng ký” trên trang web. Hệ thống sẽ chuyển sang một trang mới yêu cầu người dùng nhập các thông tin đầy đủ của cá nhân. Khi người dùng nhập đầy đủ thông tin thì hệ thống sẽ lưu thông tin vào bảng USER trong cơ sở dữ liệu và hiển thị lại lên màn hình. Use case kết thúc.

* + **Luồng rẽ nhánh**

1. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.
2. Trong quá trình đăng kí, nếu người dùng nhập sai hoặc nhập thiếu thông tin thì hệ thống sẽ thông báo lỗi và yêu cầu nhập lại hoặc nhập đầy đủ thông tin.

* **Các yêu cầu đặc biệt**

Không có

* **Tiền điều kiện**

Không có

* **Hậu điều kiện**

Nếu use case kết thúc (thành công) thì thông tin về khách hàng phải được thêm vào trong cơ sở dữ liệu.

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case DangNhap**
* **Tên use case:** Đăng nhập.
* **Mô tả vắn tắt:** Use case này cho phép người dùng có thể đăng nhập vào trang web để sử dụng các dịch vụ.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Use case bắt đầu khi khách hàng kích vào nút “Đăng nhập”. Hệ thống sẽ chuyển hướng sang một trang mới yêu cầu nhập thông tin: tài khoản, mật khẩu.
2. Khách hàng nhập thông tin và kích vào nút “Đăng nhập”. Hệ thống so sánh thông tin tài khoản và mật khẩu do người dùng nhập so với hai trường tài khoản và mật khẩu trong các bản ghi thuộc bảng USER. Nếu tổn tại, hệ thống cho phép khách hàng đăng nhập vào hệ thống và chuyển hướng về trang chủ. Use case kết thúc.
   * **Luồng rẽ nhánh**
3. Tại bước 2 ở luồng cơ bản, nếu khách hàng nhập thiếu thông tin thì hệ thống sẽ thông báo yêu cầu nhập lại đầy đủ thông tin. Khách hàng phải nhập đầy đủ thông tin để tiếp tục hoặc có thể thoát ra khỏi giao diện đăng nhập.
4. Tại bước 2 ở luồng cơ bản, nếu khách hàng nhập thông tin không có trong các bản ghi thuộc bảng USER, hệ thống sẽ báo lỗi và yêu cầu nhập lại. Khách hàng phải nhập đúng thông tin đã đăng ký hoặc thoát ra khỏi giao diện đăng nhập.
5. Tại bất kỳ thời điểm nào trong quá trình thực hiện use case, nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ không làm việc và use case kết thúc.

* **Các yêu cầu đặc biệt**

Không có

* **Tiền điều kiện**

Không có

* **Hậu điều kiện**

Nếu use case kết thúc (thành công) thì người dùng phải được sử dụng các chức năng của hệ thống.

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case ThemVaoGioHang**
* **Tên use case:** Thêm vào giỏ hàng.
* **Mô tả vắn tắt:** Use case này cho phép người dùng đưa những sản phẩm mình cần mua vào không gian lưu trữ tạm thời trên web.
* **Luồng sự kiện**
  + **Luồng cơ bản**

Use case này bắt đầu khi người dùng click “Thêm vào giỏ hàng” trên giao diện màn hình chi tiết sản phẩm. Hệ thống sẽ lưu trữ thông tin sản phẩm mà khách hàng thêm vào giỏ. Use case kết thúc.

* + **Luồng rẽ nhánh**

Không có

* **Các yêu cầu đặc biệt**

Không có

* **Tiền điều kiện**

Cần phải đăng nhập vào hệ thống.

* **Hậu điều kiện**

Không có

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case ThemVaoDanhSachYeuThich**
* **Tên use case:** Thêm vào danh sách yêu thích.
* **Mô tả vắn tắt:** Use case này cho phép người dùng đưa những sản phẩm mình yêu thích vào bảng WISHLIST trong cơ sở dữ liệu.
* **Luồng sự kiện**
  + **Luồng cơ bản**

Use case này bắt đầu khi người dùng click “Thêm vào danh sách yêu thích” trên giao diện màn hình chi tiết sản phẩm. Hệ thống sẽ lưu trữ thông tin sản phẩm mà khách hàng thêm vào danh sách. Use case kết thúc.

* + **Luồng rẽ nhánh**

Không có

* **Các yêu cầu đặc biệt**

Không có

* **Tiền điều kiện**

Cần phải đăng nhập vào hệ thống.

* **Hậu điều kiện**

Không có

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case DatHang**
* **Tên use case:** Đặt hàng.
* **Mô tả vắn tắt:** Use case này cho phép người dùng đặt hàng để mua sản phẩm.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Use case này bắt đầu khi người dùng click vào “THANH TOÁN” trên giao diện đặt hàng. Hệ thống sẽ hiển thị giao yêu cầu cập nhật thông tin khách hàng và chọn phương thức thanh toán.
2. Sau khi điền đầy đủ thông tin thì khách hàng click vào “ĐẶT HÀNG”. Use case kết thúc.
   * **Luồng rẽ nhánh**
3. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.
4. Tại bước 2 ở luồng cơ bản, nếu khách hàng nhập thiếu thông tin thì hệ thống sẽ thông báo yêu cầu nhập lại đầy đủ thông tin. Khách hàng phải nhập đầy đủ thông tin để tiếp tục.

* **Các yêu cầu đặc biệt**

Không có

* **Tiền điều kiện**

Cần phải đăng nhập vào hệ thống.

* **Hậu điều kiện**

Không có.

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case QuanLyGioHang**
* **Tên use case:** Quản lý giỏ hàng.
* **Mô tả vắn tắt:** Use case này cho phép người dùng có thể xem, cập nhật, xoá giỏ hàng.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Xem giỏ hàng

Use case này bắt đầu khi người dùng nhấn chọn biểu tượng giỏ hàng trên giao diện trang web. Hệ thống lấy thông tin chi tiết của các sản phẩm (TenSP, HinhAnhSP, GiaBan) mà người dùng thêm vào trong cơ sở dữ liệu và hiển thị lên màn hình.

1. Xoá giỏ hàng

Use case này bắt đầu khi người dùng nhấn vào biểu tượng thùng rác trên một dòng của sản phẩm trong giỏ hàng. Hệ thống sẽ xóa sản phẩm đó ra khỏi giỏ hàng.

1. Cập nhật giỏ hàng

Use case này bắt đầu khi người dùng ấn thay đổi số lượng sản phẩm cần mua trong giao diện giỏ hàng và ấn nút “CẬP NHẬT” trên giao diện giỏ hàng. Hệ thống sẽ cập nhật số lượng sản phẩm cần mua.

* + **Luồng rẽ nhánh**

Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.

* **Các yêu cầu đặc biệt**

Không có

* **Tiền điều kiện**

Cần phải đăng nhập vào hệ thống.

* **Hậu điều kiện**

Không có.

* **Điểm mở rộng**

Không có

* + - 1. **Mô tả Use Case QuanLyDonHang**
* **Tên use case:** Quản lý đơn hàng.
* **Mô tả vắn tắt:** Use case này cho phép người quản trị có thể xem, xác nhận đơn hàng.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Xem đơn hàng

Use case này bắt đầu khi người quản trị nhấn “Quản lý đơn hàng” trên thanh menu của người quản trị. Hệ thống lấy thông tin chi tiết của các đơn hàng trong bảng ORDERS trong cơ sở dữ liệu và hiển thị lên màn hình.

1. Xác nhận đơn hàng

Use case này bắt đầu khi người dùng nhấn vào nút “Xác nhận” trên một dòng của hoá đơn. Hệ thống sẽ xác nhận đơn hàng đó để tiến hành giao sản phẩm cho khách hàng.

* + **Luồng rẽ nhánh**

Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.

* **Các yêu cầu đặc biệt**

Use này chỉ cho phép người quản trị viên thực hiện.

* **Tiền điều kiện**

Người quản trị cần đăng nhập với tài khoản quản trị viên.

* **Hậu điều kiện**

Không có.

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case QuanLyNguoiDung**
* **Tên use case:** Quản lý người dùng.
* **Mô tả vắn tắt:** Use case này cho phép người quản trị có thể xem, xoá tài khoản khách hàng.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Thêm người dùng
2. Use case này bắt đầu khi người quản trị nhấn “Thêm mới” trên giao diện của người quản trị. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho người dùng.
3. Người quản trị nhập thông tin sản phẩm và nhấn vào nút “Thêm mới”. Hệ thống sẽ tạo một bản ghi mới trong cơ sở dữ liệu và hiển thị danh sách các người dùng được cập nhật.
4. Sửa người dùng
5. Người quản trị nhấn vào nút “Sửa” trên một dòng người dùng. Hệ thống hiển thị một màn hình với các thông tin cũ của người dùng.
6. Người quản trị sửa các thông tin của người dùng và nhấn vào nút “Cập nhật”. Hệ thống sẽ cập nhật thông tin của người dùng trong cơ sở dữ liệu và hiển thị danh sách tin tức đã cập nhật.
7. Xoá người dùng
8. Use case này bắt đầu khi người dùng nhấn vào nút “Xóa” trên một dòng người dùng. Hệ thống sẽ hiển thị thông báo xác nhận xoá.
9. Quản trị viên nhấn xác nhận xoá. Hệ thống sẽ xoá người dùng trong cơ sở dữ liệu và hiển thị lại danh sách người dùng vừa cập nhật lên màn hình.
   * **Luồng rẽ nhánh**
10. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.

* **Các yêu cầu đặc biệt**

Use case này chỉ cho phép người quản trị thực hiện.

* **Tiền điều kiện**

Người quản trị phải đăng nhập với tài khoản quản trị viên.

* **Hậu điều kiện**

Không có

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case QuanLySanPham**
* **Tên use case:** Quản lý sản phẩm.
* **Mô tả vắn tắt:** Use case này cho phép người quản trị có thể thêm, sửa, xoá sản phẩm.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Thêm sản phẩm
2. Use case này bắt đầu khi người quản trị nhấn “Thêm mới” trên giao diện của người quản trị. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho sản phẩm.
3. Người quản trị nhập thông tin sản phẩm và nhấn vào nút “Thêm mới”. Hệ thống sẽ tạo một bản ghi mới trong cơ sở dữ liệu và hiển thị danh sách các sản phẩm được cập nhật.
4. Sửa sản phẩm
5. Người quản trị nhấn vào nút “Sửa” trên một dòng sản phẩm. Hệ thống hiển thị một màn hình với các thông tin cũ của sản phẩm.
6. Người quản trị sửa các thông tin của sản phẩm và nhấn vào nút “Cập nhật”. Hệ thống sẽ cập nhật thông tin của sản phẩm trong cơ sở dữ liệu và hiển thị danh sách tin tức đã cập nhật.
7. Xoá sản phẩm
8. Use case này bắt đầu khi người dùng nhấn vào nút “Xóa” trên một dòng sản phẩm. Hệ thống sẽ hiển thị thông báo xác nhận xoá.
9. Quản trị viên nhấn xác nhận xoá. Hệ thống sẽ xoá sản phẩm trong cơ sở dữ liệu và hiển thị lại danh sách sản phẩm vừa cập nhật lên màn hình.
   * **Luồng rẽ nhánh**
10. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.

* **Các yêu cầu đặc biệt**

Use case này chỉ cho phép người quản trị thực hiện.

* **Tiền điều kiện**

Người quản trị phải đăng nhập với tài khoản quản trị viên.

* **Hậu điều kiện**

Không có

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case QuanLyKhoHang**
* **Tên use case:** Quản lý kho hàng.
* **Mô tả vắn tắt:** Use case này cho phép người quản trị có thể thêm, sửa, xoá sản phẩm.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Thêm sản phẩm
2. Use case này bắt đầu khi người quản trị nhấn “Thêm mới” trên giao diện của người quản trị. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho kho hàng.
3. Người quản trị nhập thông tin kho hàng và nhấn vào nút “Thêm mới”. Hệ thống sẽ tạo một bản ghi mới trong cơ sở dữ liệu và hiển thị danh sách các thông tin kho hàng được cập nhật.
4. Sửa sản phẩm
5. Người quản trị nhấn vào nút “Sửa” trên một dòng thông tin kho hàng. Hệ thống hiển thị một màn hình với các thông tin cũ của kho hàng.
6. Người quản trị sửa các thông tin của kho hàng và nhấn vào nút “Cập nhật”. Hệ thống sẽ cập nhật thông tin của kho hàng trong cơ sở dữ liệu và hiển thị danh sách tin tức đã cập nhật.
7. Xoá sản phẩm
8. Use case này bắt đầu khi người dùng nhấn vào nút “Xóa” trên một dòng thông tin kho hàng. Hệ thống sẽ hiển thị thông báo xác nhận xoá.
9. Quản trị viên nhấn xác nhận xoá. Hệ thống sẽ xoá thông tin kho hàng trong cơ sở dữ liệu và hiển thị lại danh sách thông tin kho hàng vừa cập nhật lên màn hình.
   * **Luồng rẽ nhánh**
10. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.

* **Các yêu cầu đặc biệt**

Use case này chỉ cho phép người quản trị thực hiện.

* **Tiền điều kiện**

Người quản trị phải đăng nhập với tài khoản quản trị viên.

* **Hậu điều kiện**

Không có

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case QuanLyDanhMucSanPham**
* **Tên use case:** Quản lý danh mục sản phẩm.
* **Mô tả vắn tắt:** Use case này cho phép người quản trị có thể thêm, sửa, xoá danh mục sản phẩm.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Thêm danh mục sản phẩm
2. Use case này bắt đầu khi người quản trị nhấn “Thêm mới” trên giao diện của người quản trị. Hệ thống hiển thị màn hình yêu cầu nhập thông tin chi tiết cho danh mục sản phẩm.
3. Người quản trị nhập thông tin danh mục sản phẩm và nhấn vào nút “Thêm mới”. Hệ thống sẽ tạo một bản ghi mới trong cơ sở dữ liệu và hiển thị danh sách các danh mục sản phẩm được cập nhật.
4. Sửa danh mục sản phẩm
5. Người quản trị nhấn vào nút “Sửa” trên một dòng danh mục sản phẩm. Hệ thống hiển thị một màn hình với các thông tin cũ của danh mục sản phẩm.
6. Người quản trị sửa các thông tin của danh mục sản phẩm và nhấn vào nút “Cập nhật”. Hệ thống sẽ cập nhật thông tin của danh mục sản phẩm trong cơ sở dữ liệu và hiển thị danh sách tin tức đã cập nhật.
7. Xoá sản phẩm
8. Use case này bắt đầu khi người dùng nhấn vào nút “Xóa” trên một dòng danh mục sản phẩm. Hệ thống sẽ hiển thị thông báo xác nhận xoá.
9. Quản trị viên nhấn xác nhận xoá. Hệ thống sẽ xoá danh mục sản phẩm trong cơ sở dữ liệu và hiển thị lại danh sách danh mục sản phẩm vừa cập nhật lên màn hình.
   * **Luồng rẽ nhánh**
10. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.

* **Các yêu cầu đặc biệt**

Use case này chỉ cho phép người quản trị thực hiện.

* **Tiền điều kiện**

Người quản trị phải đăng nhập với tài khoản quản trị viên.

* **Hậu điều kiện**

Không có

* **Điểm mở rộng**

Không có.

* + - 1. **Mô tả Use Case ThongKeBaoCao**
* **Tên use case:** Thống kê báo cáo.
* **Mô tả vắn tắt:** Use case này cho phép người quản trị xem thống kê doanh thu theo sản phẩm và tổng doanh thu.
* **Luồng sự kiện**
  + **Luồng cơ bản**

1. Thêm danh mục sản phẩm
2. Use case này bắt đầu khi người quản trị nhấn “Thống kê doanh thu” và ấn vào loại thống kê trên giao diện của người quản trị. Hệ thống hiển thị màn hình thông tin chi tiết về thống kê doanh thu theo sản phẩm hoặc theo tổng doanh thu.
   * **Luồng rẽ nhánh**
3. Tại bất kì thời điểm nào trong quá trình thực hiện use case nếu không kết nối được với cơ sở dữ liệu thì hệ thống sẽ thông báo lỗi và use case sẽ kết thúc.

* **Các yêu cầu đặc biệt**

Use case này chỉ cho phép người quản trị thực hiện.

* **Tiền điều kiện**

Người quản trị phải đăng nhập với tài khoản quản trị viên.

* **Hậu điều kiện**

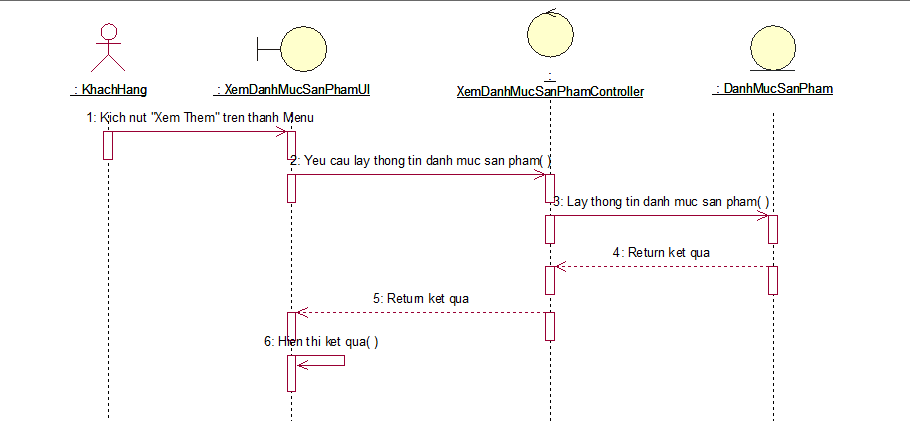
Không có

* **Điểm mở rộng**

Không có.

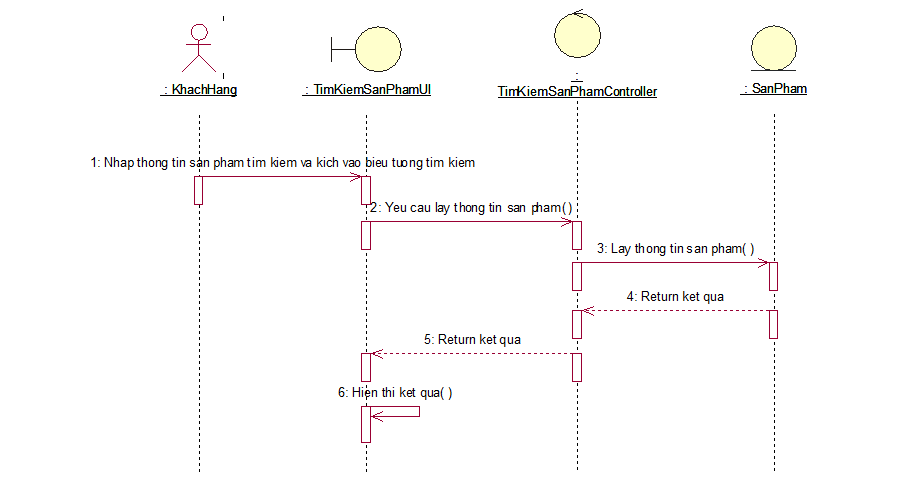
### 3.1.3 Biểu đồ trình tự

1. **Biểu đồ trình tự XemDanhMucSanPham**



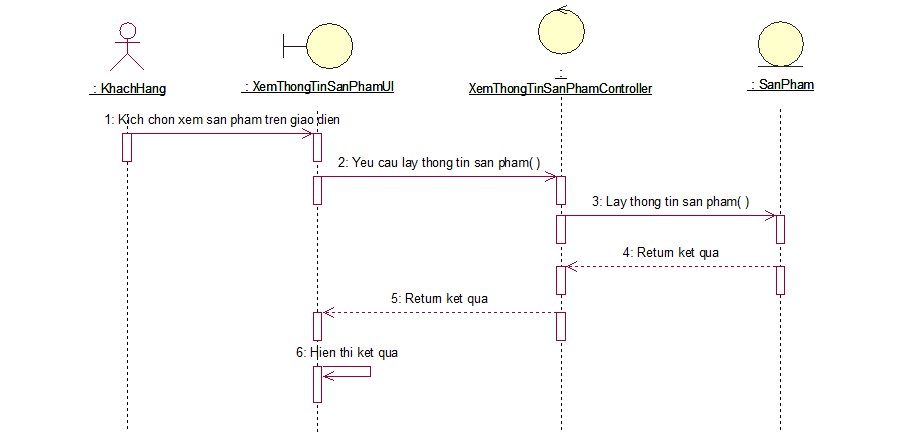
Hình 3. 5 Biểu đồ trình tự XemDanhMucSanPham

1. **Biểu đồ trình tự TimKiemSanPham**



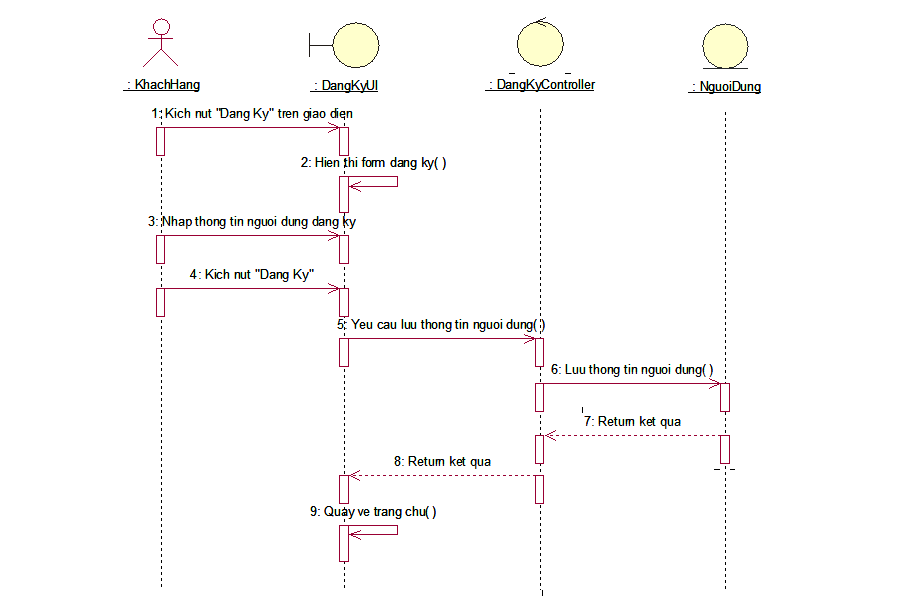
Hình 3. 6 Biểu đồ trình tự TimKiemSanPham

1. **Biểu đồ trình tự XemThongTinSanPham**



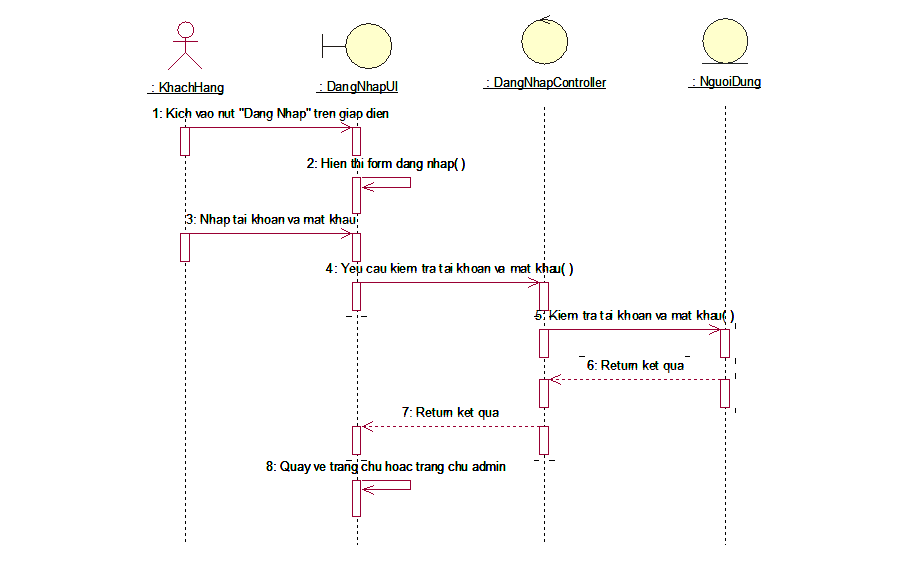
Hình 3. 7 Biểu đồ trình tự XemThongTinSanPham

1. **Biểu đồ trình tự DangKy**



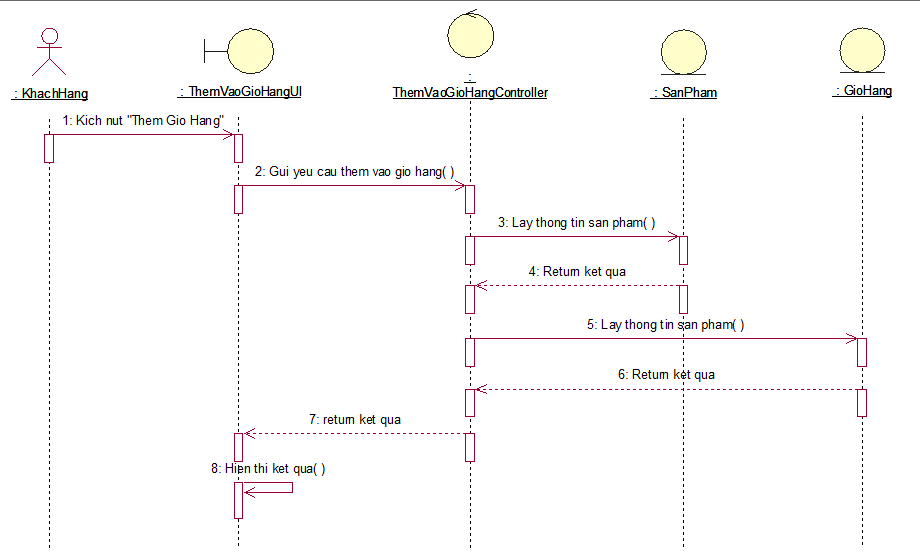
Hình 3. 8 Biểu đồ trình tự DangKy

1. **Biểu đồ trình tự DangNhap**



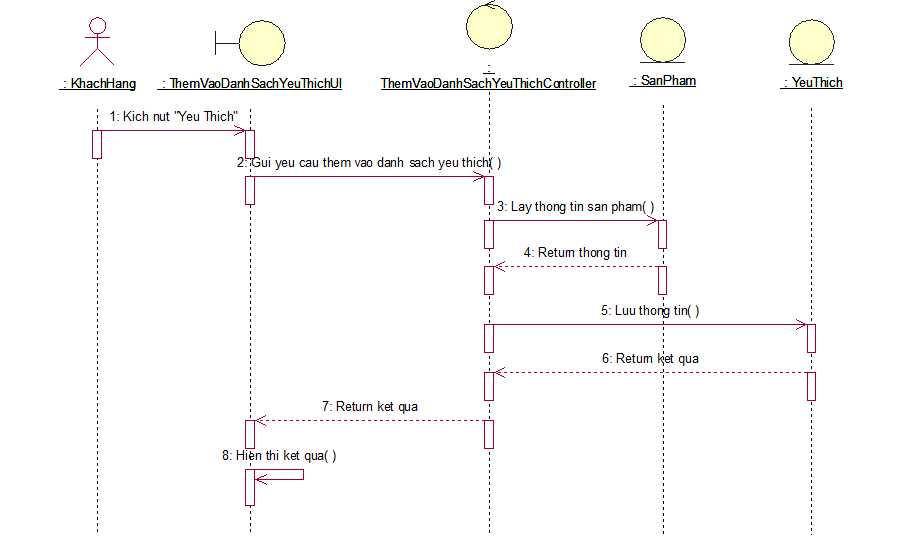
Hình 3. 9 Biểu đồ trình tự DangNhap

1. **Biểu đồ trình tự ThemVaoGioHang**



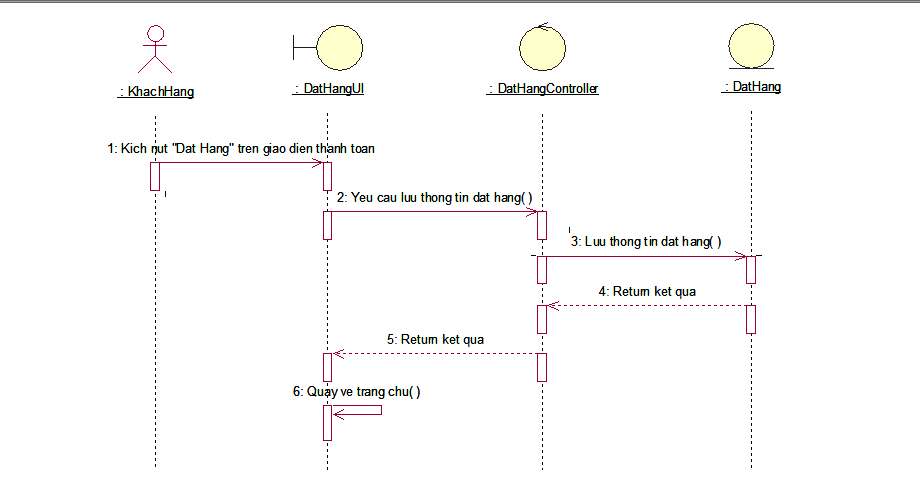
Hình 3. 10 Biểu đồ trình tự ThemVaoGioHang

1. **Biểu đồ trình tự ThemVaoDanhSachYeuThich**



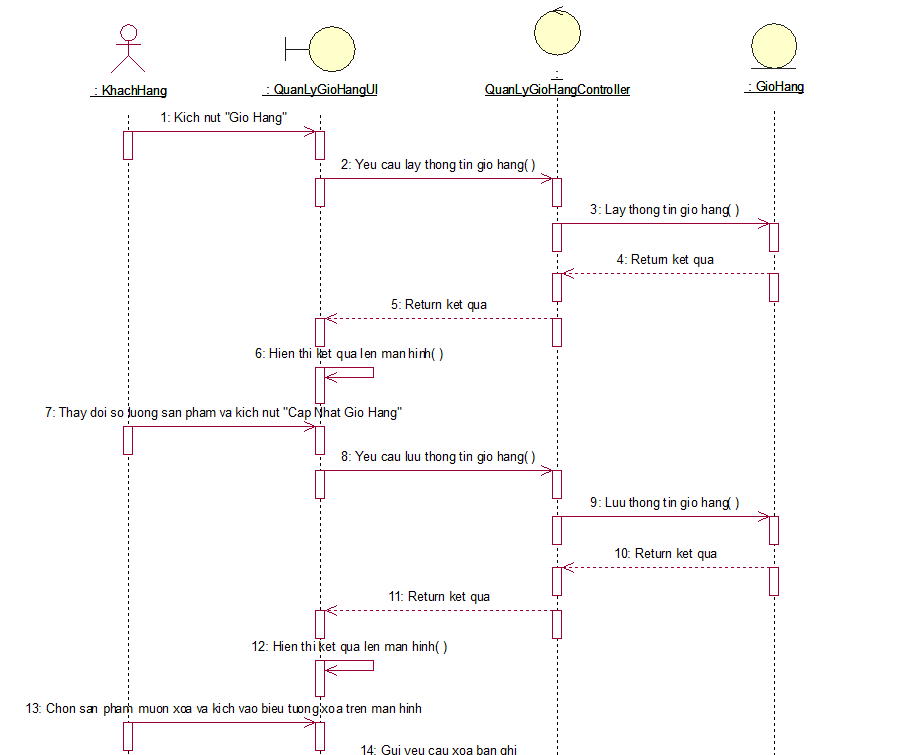
Hình 3. 11 Biểu đồ trình tự ThemVaoDanhSachYeuThich

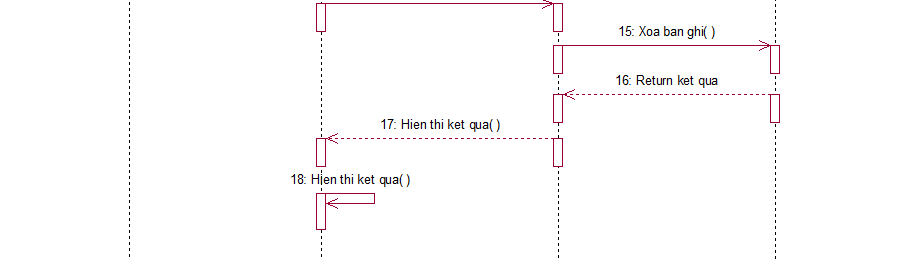
1. **Biểu đồ trình tự DatHang**



Hình 3. 12 Biểu đồ trình tự DatHang

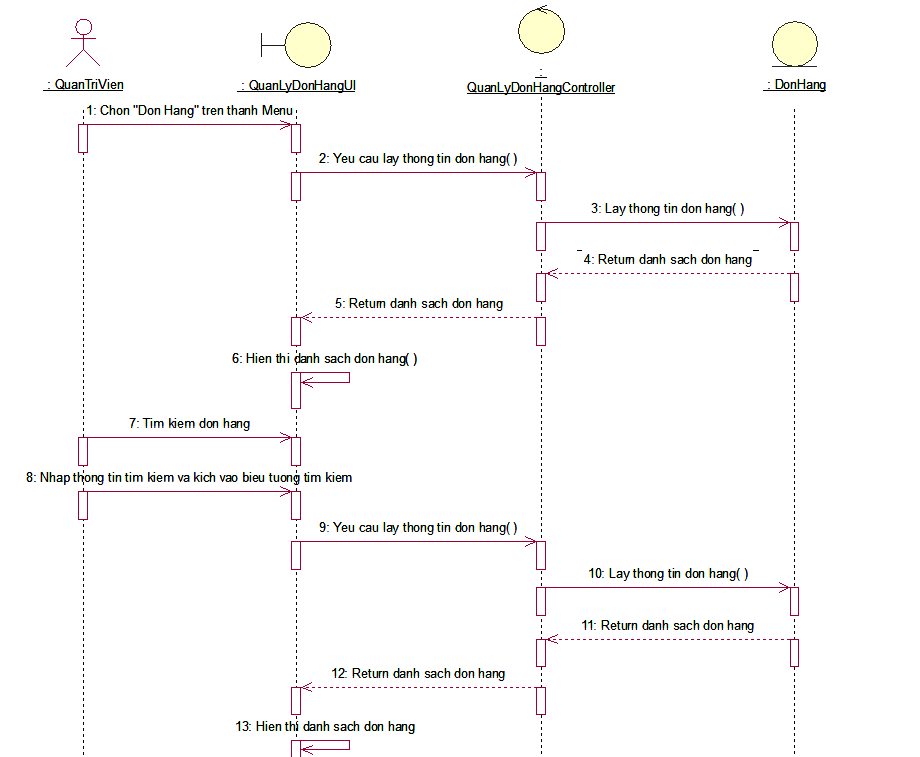
1. **Biểu đồ trình tự QuanLyGioHang**

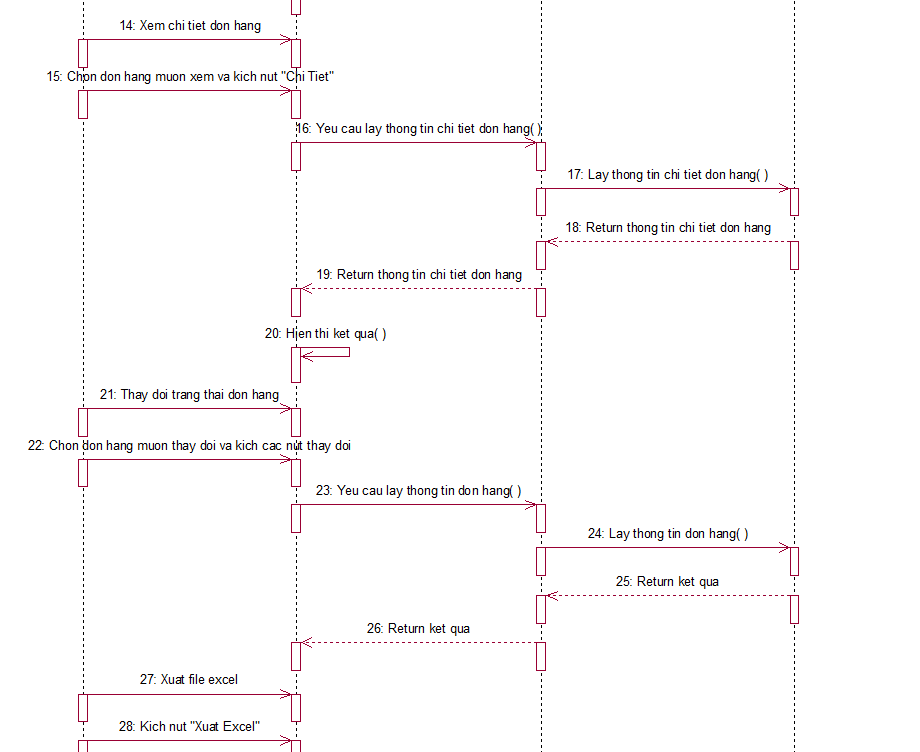


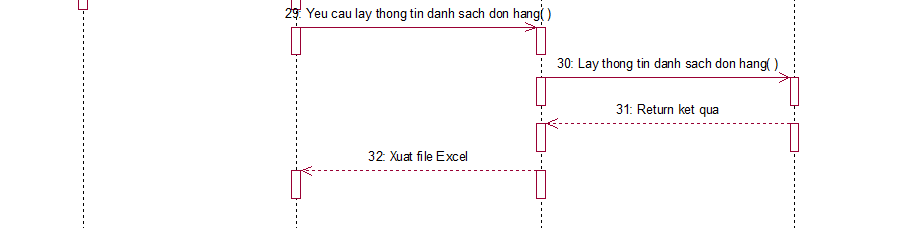


Hình 3. 13 Biểu đồ trình tự QuanLyGioHang

1. **Biểu đồ trình tự QuanLyDonHang**

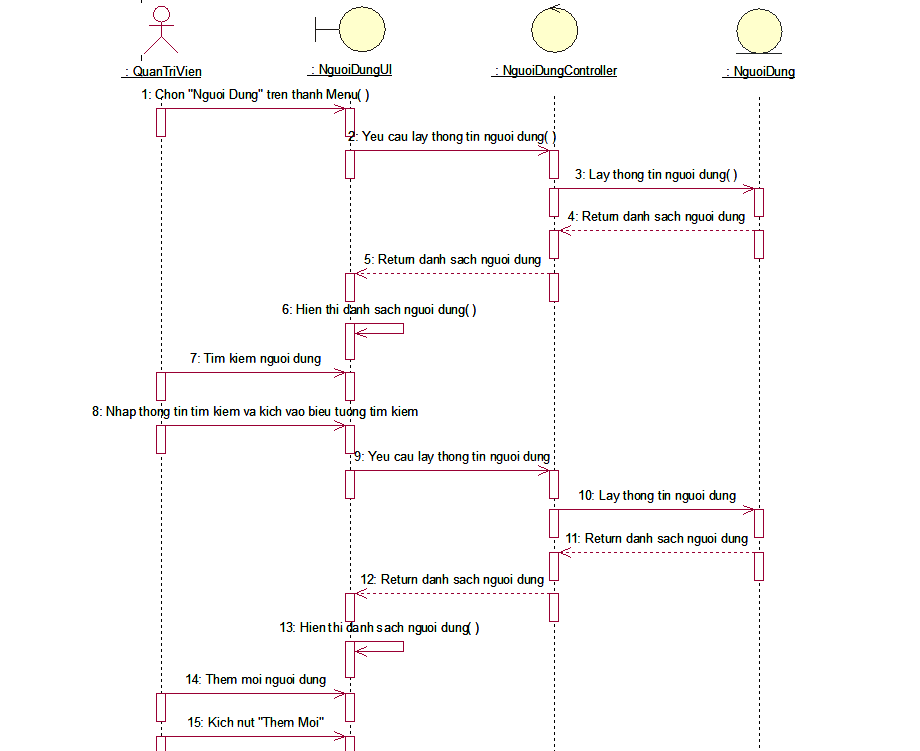


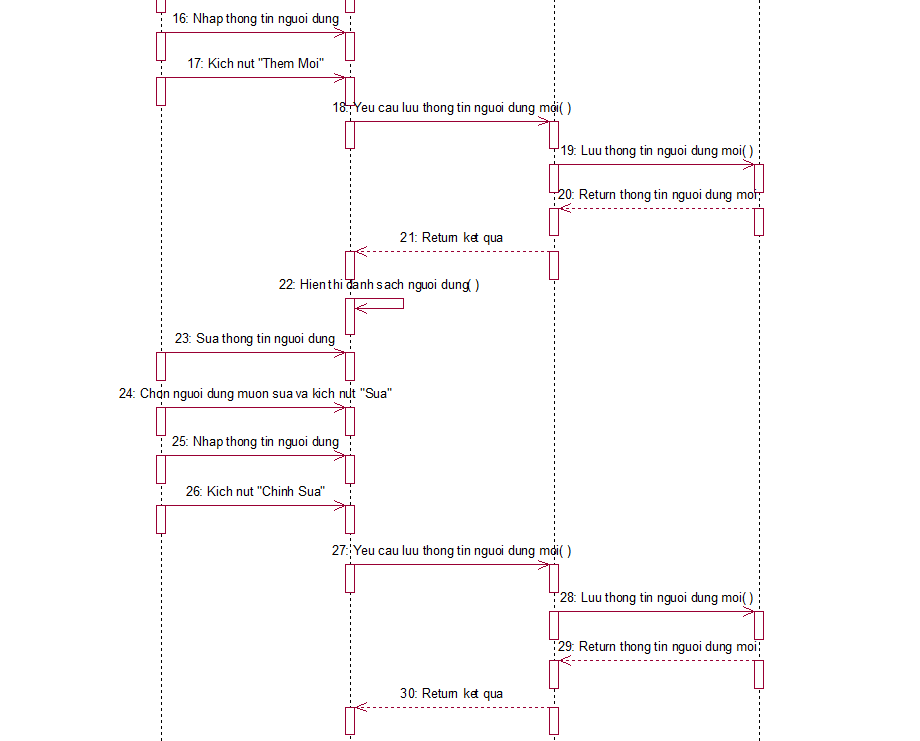


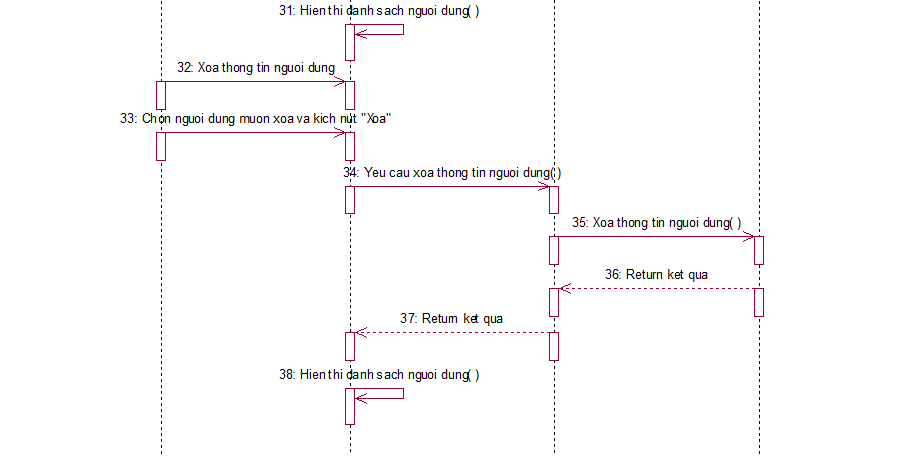


Hình 3. 14 Biểu đồ trình tự QuanLyDonHang

1. **Biểu đồ trình tự QuanLyNguoiDung**

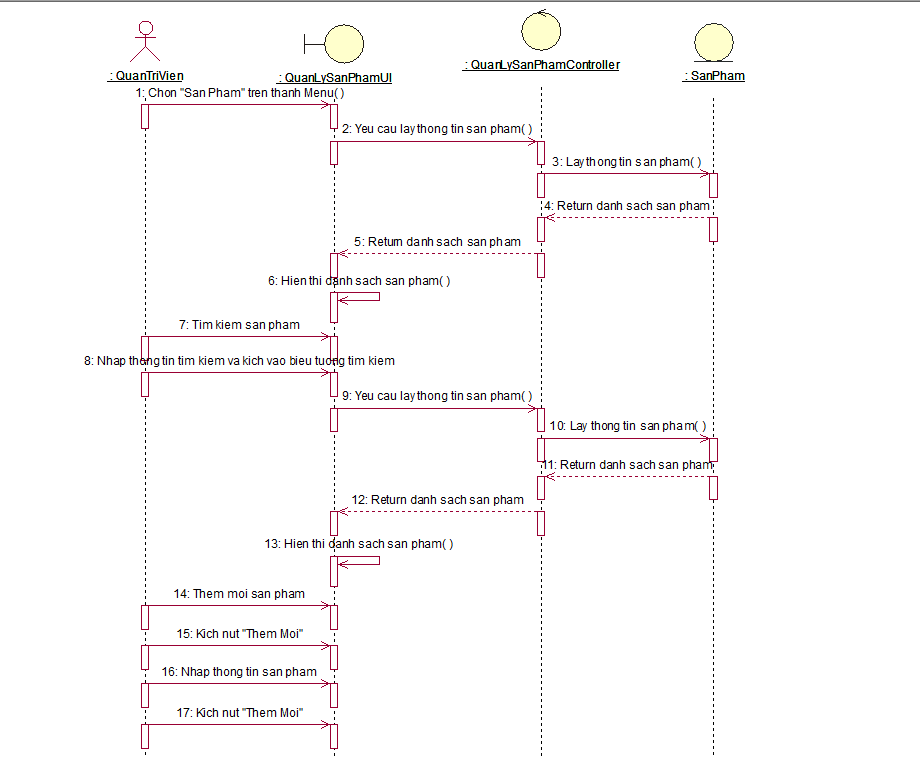


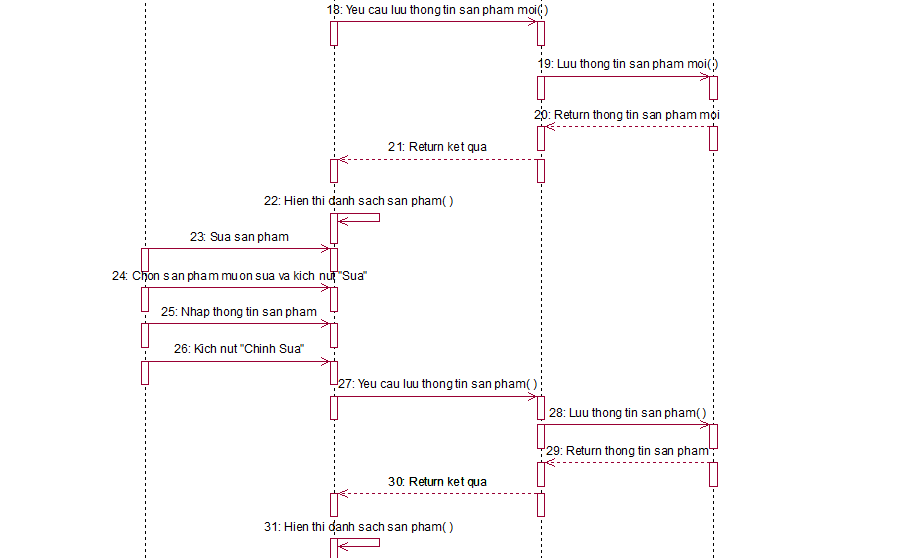


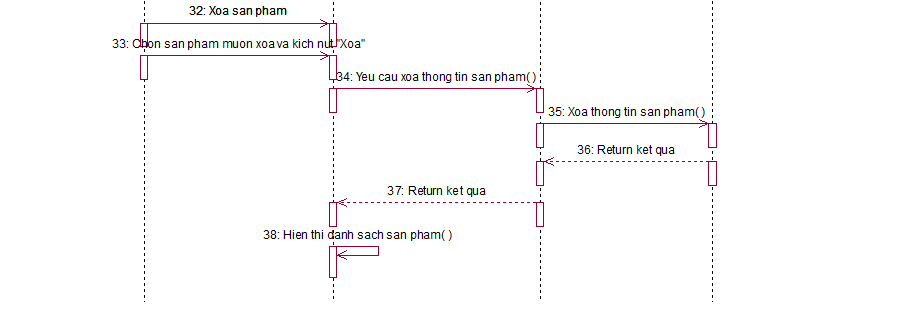


Hình 3. 15 Biểu đồ trình tự QuanLyNguoiDung

1. **Biểu đồ trình tự QuanLySanPham**

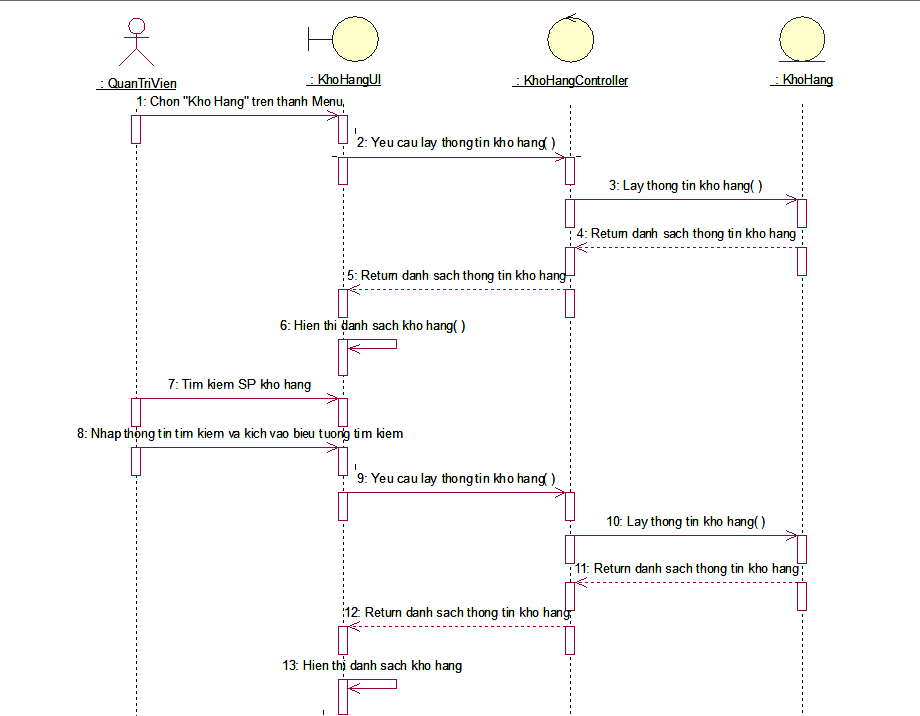


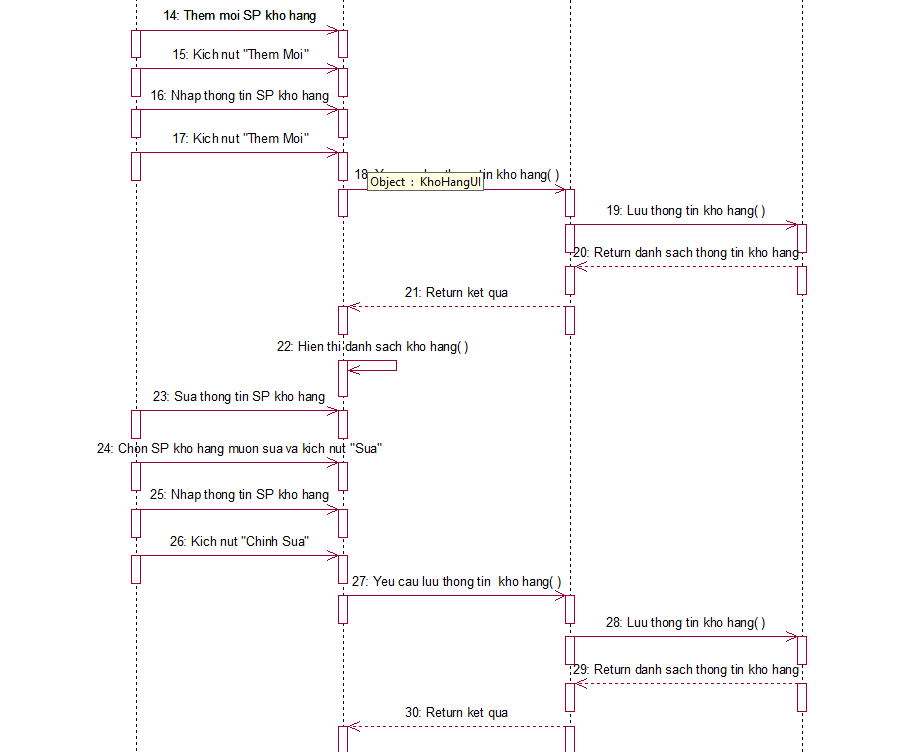


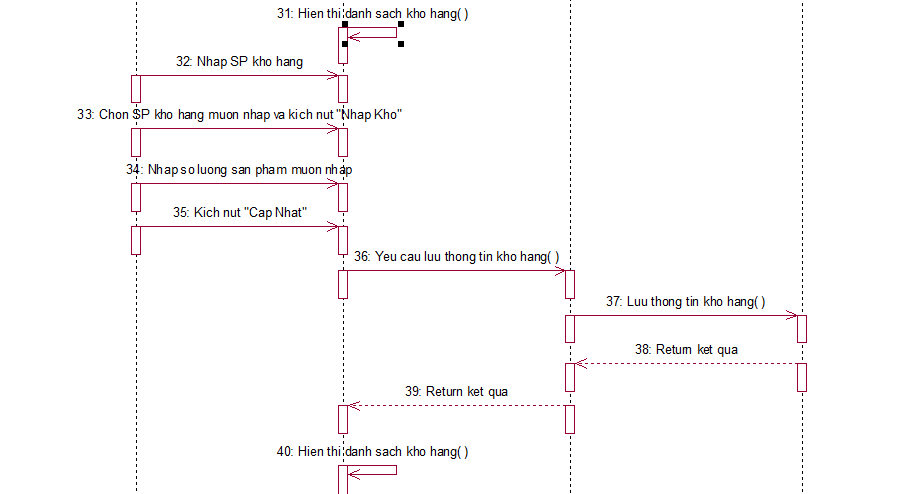


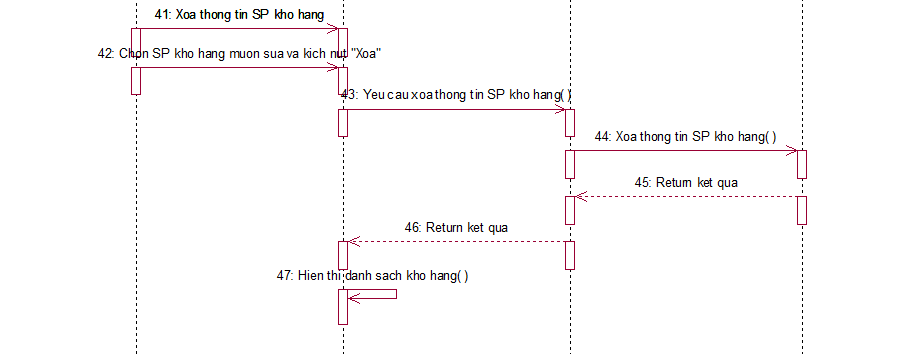
Hình 3. 16 Biểu đồ trình tự QuanLySanPham

1. **Biểu đồ trình tự QuanLyKhoHang**



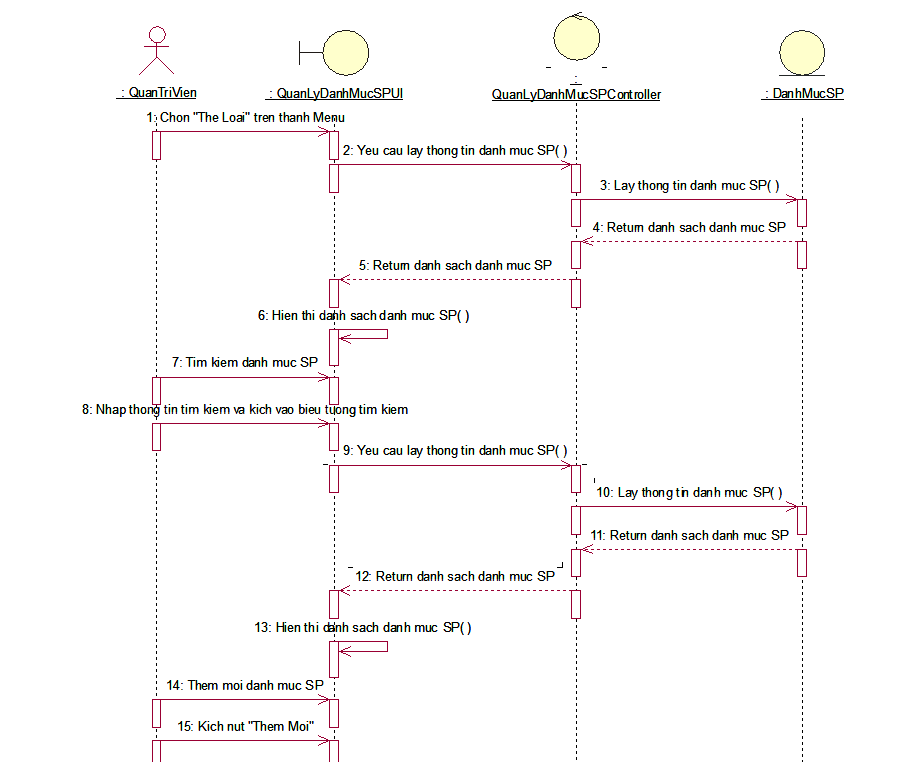


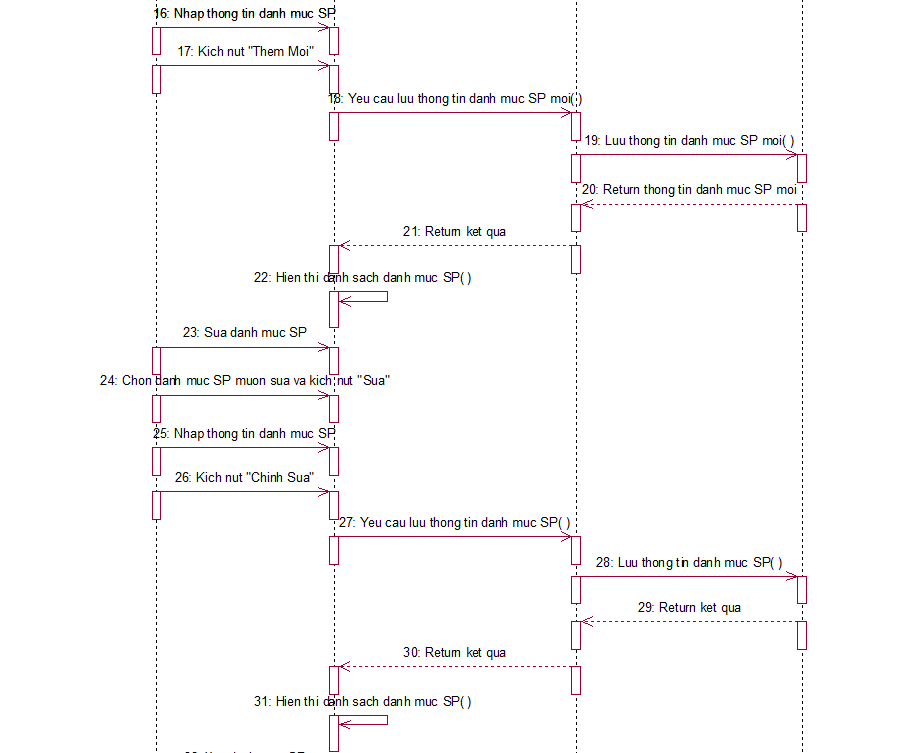


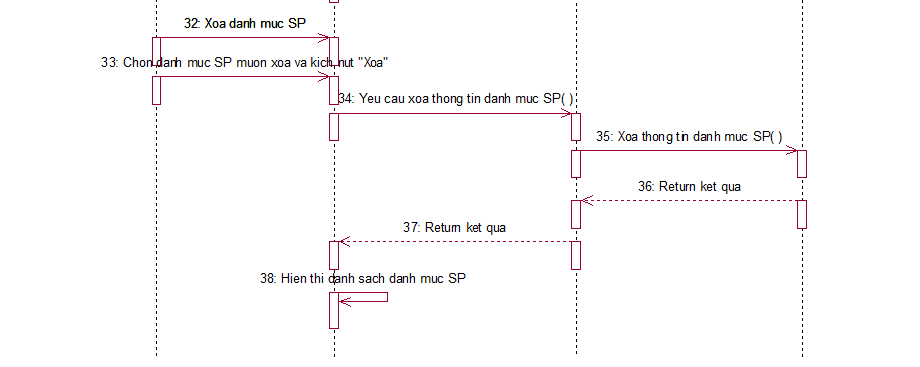


Hình 3. 17 Biểu đồ trình tự QuanLyKhoHang

1. **Biểu đồ trình tự QuanLyDanhMucSanPham**

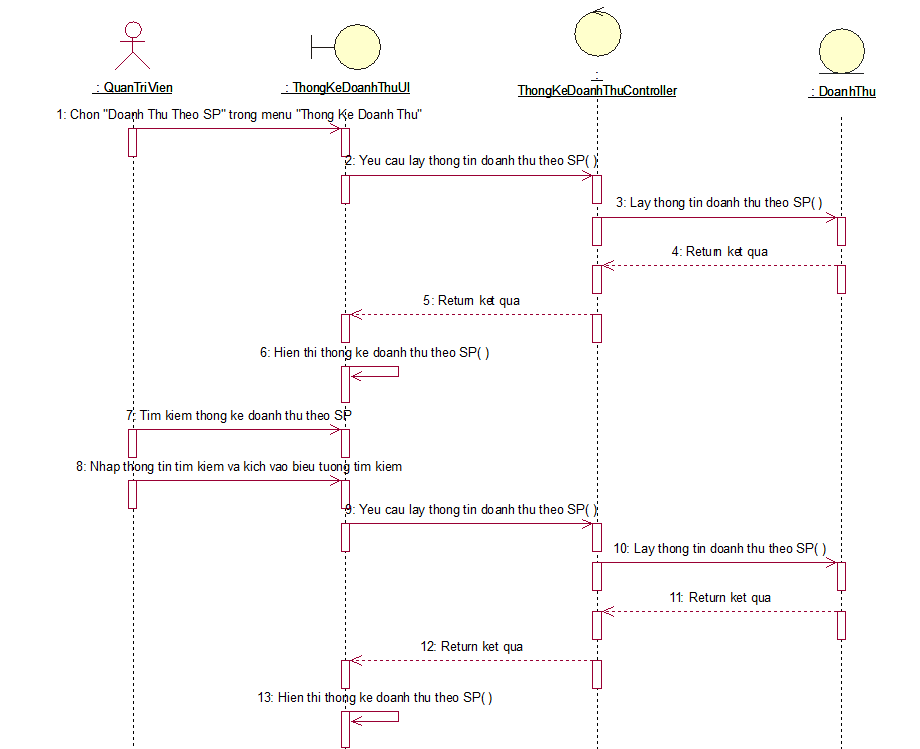


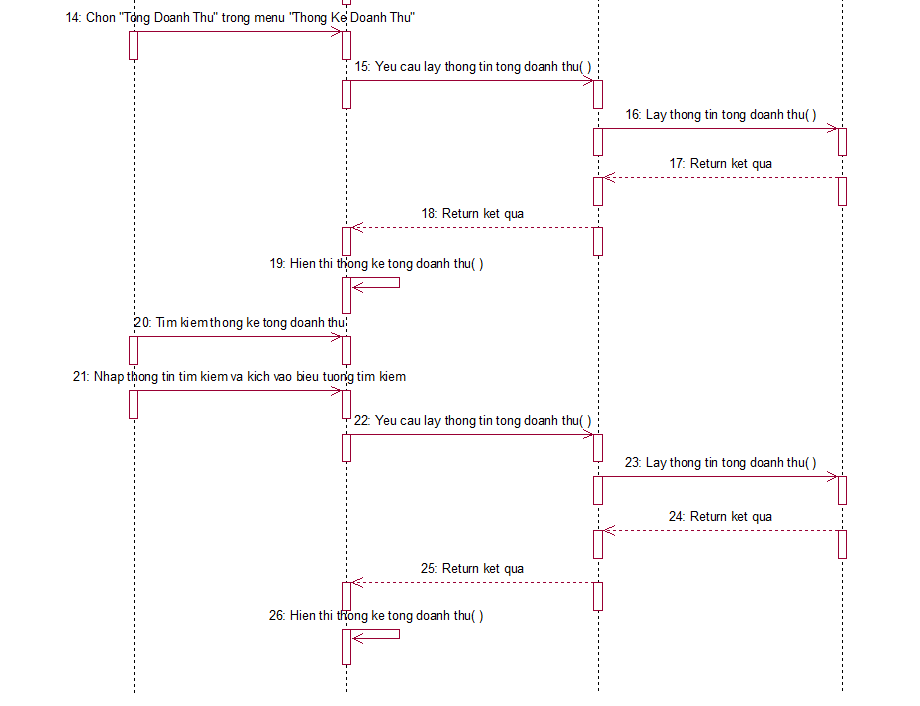




Hình 3. 18 Biểu đồ trình tự QuanLyDanhMucSanPham

1. **Biểu đồ trình tự ThongKeBaoCao**



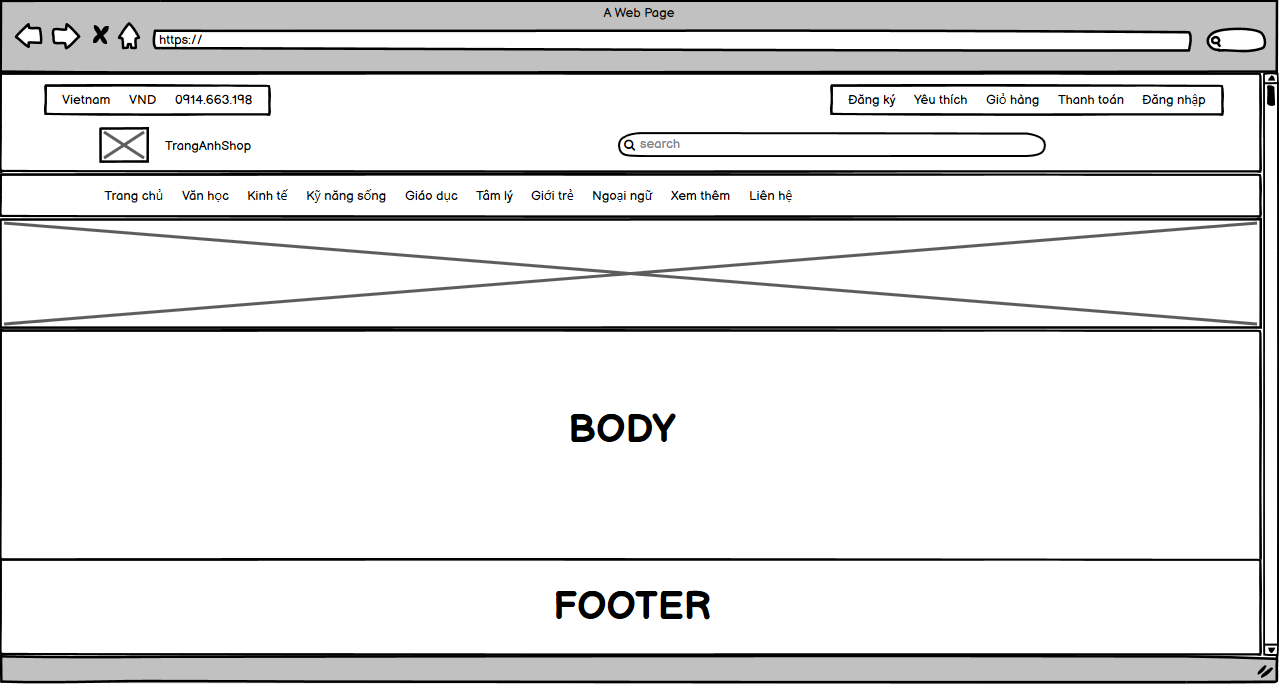


Hình 3. 19 Biểu đồ trình tự ThongKeBaoCao

## 3.2 Thiết kế giao diện

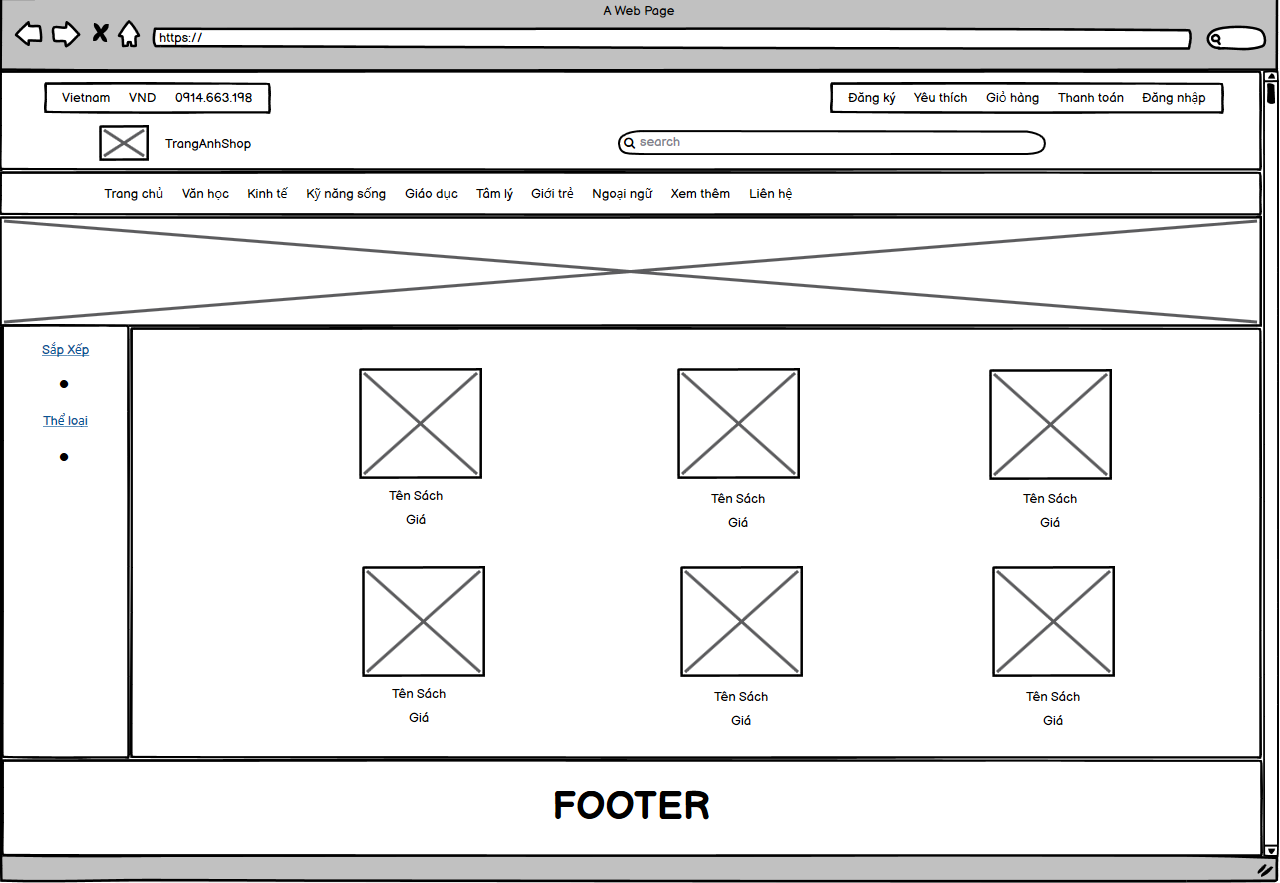
### 3.2.1 Giao diện người dùng

**3.2.1.1. Giao diện trang chủ**



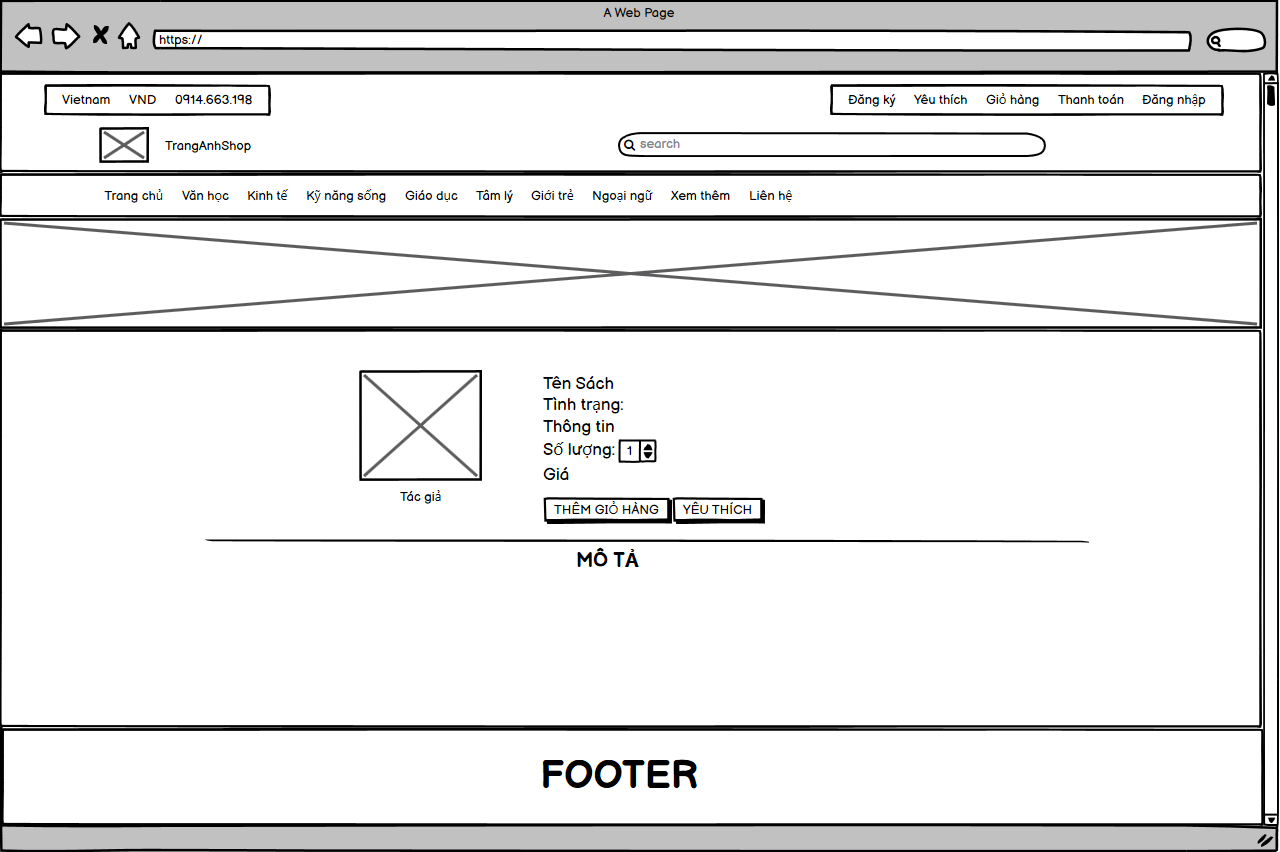
Hình 3. 20 Giao diện trang chủ

**3.2.1.2. Giao diện trang sản phẩm**



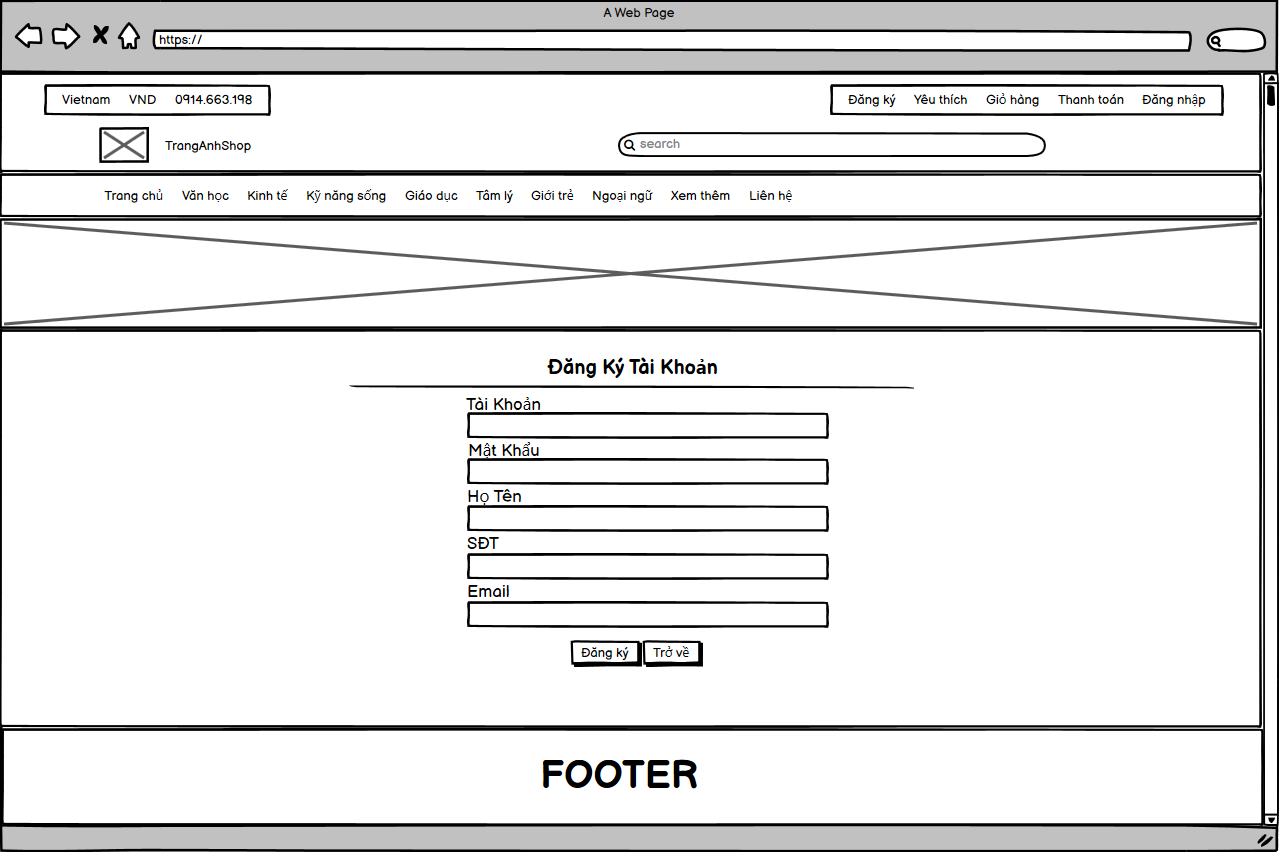
Hình 3. 21 Giao diện trang sản phẩm

**3.2.1.2. Giao diện trang chi tiết sản phẩm**



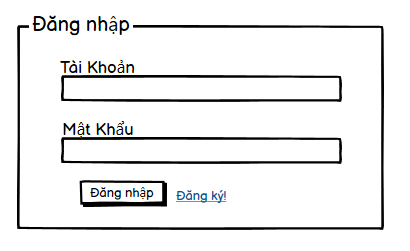
Hình 3. 22 Giao diện trang chi tiết sản phẩm

**3.2.1.3. Giao diện trang đăng ký**



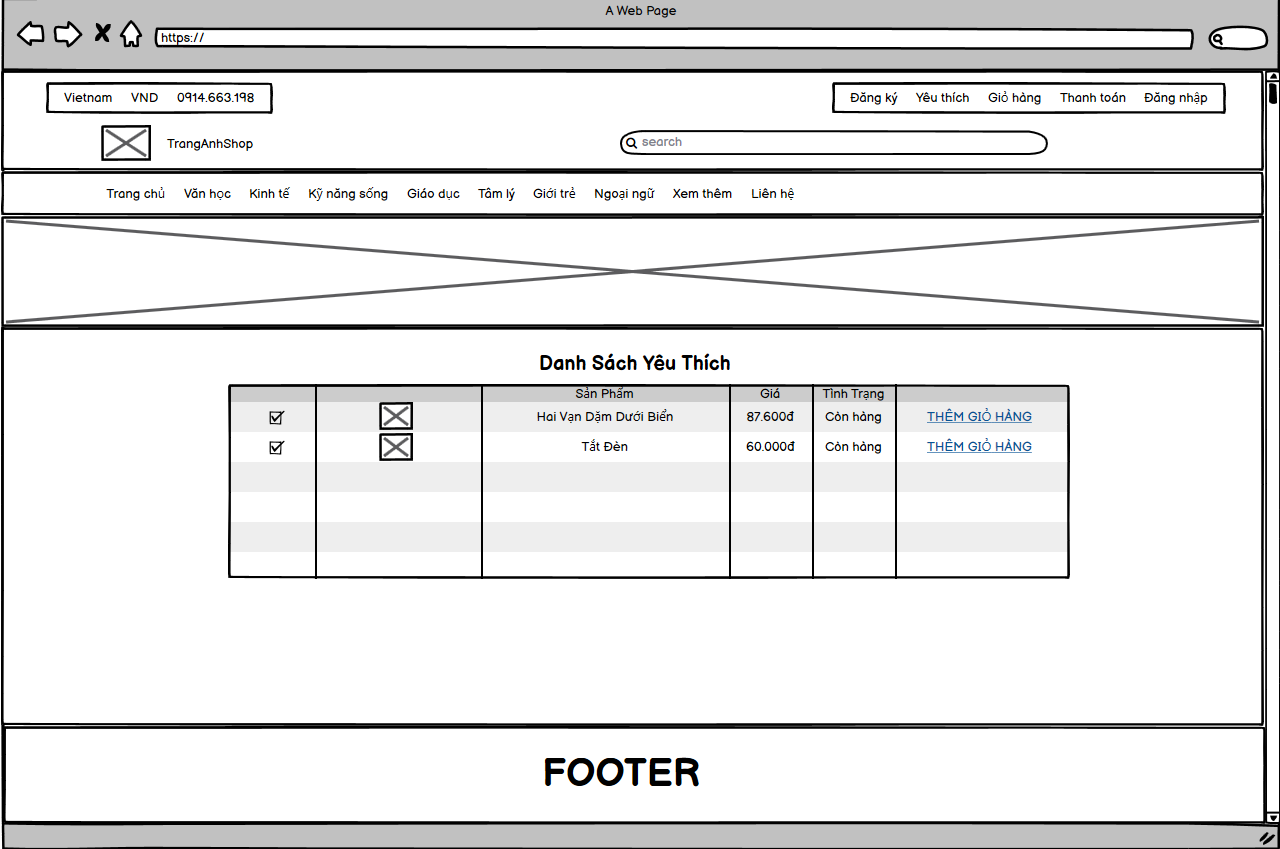
Hình 3. 23 Giao diện trang đăng ký

**3.2.1.4. Giao diện trang đăng nhập**



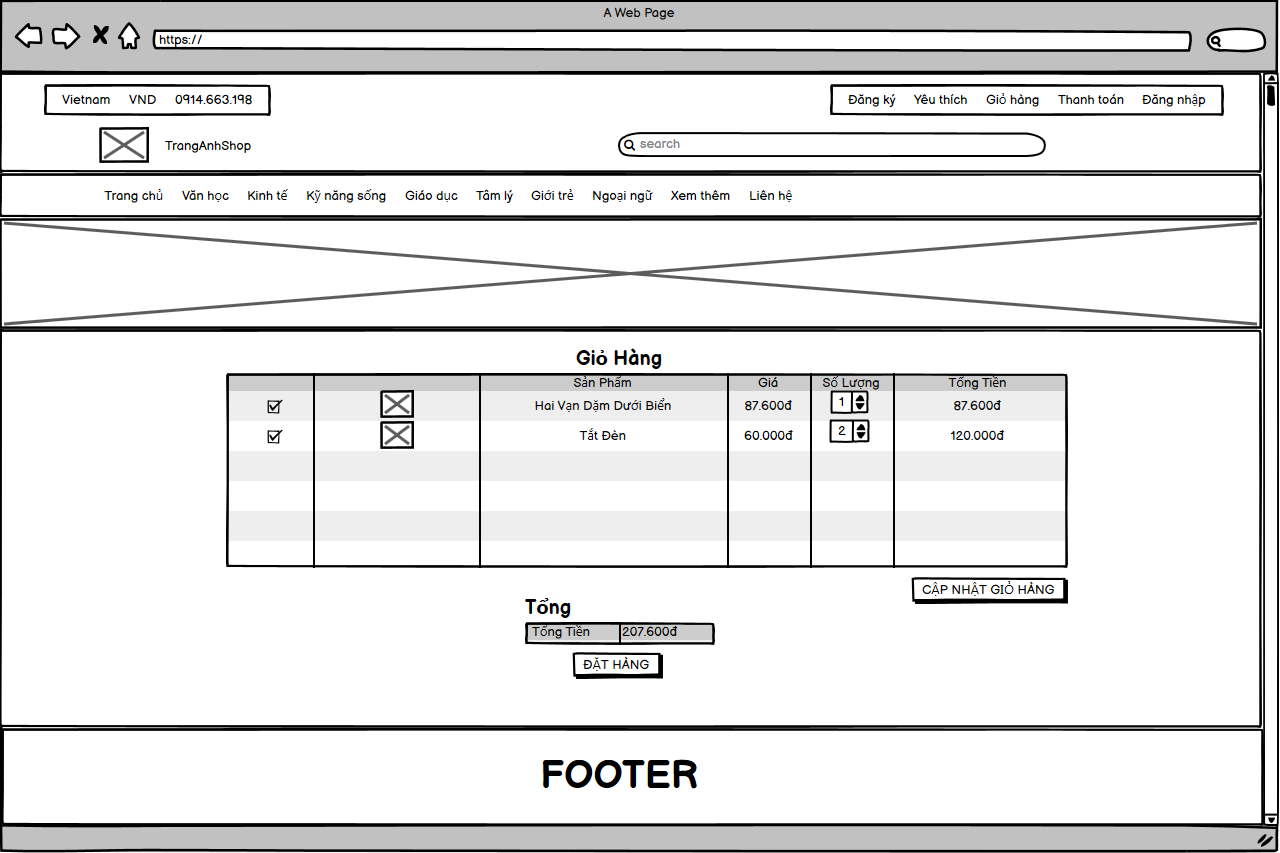
Hình 3. 24 Giao diện trang đăng nhập

**3.2.1.5. Giao diện trang danh sách yêu thích**



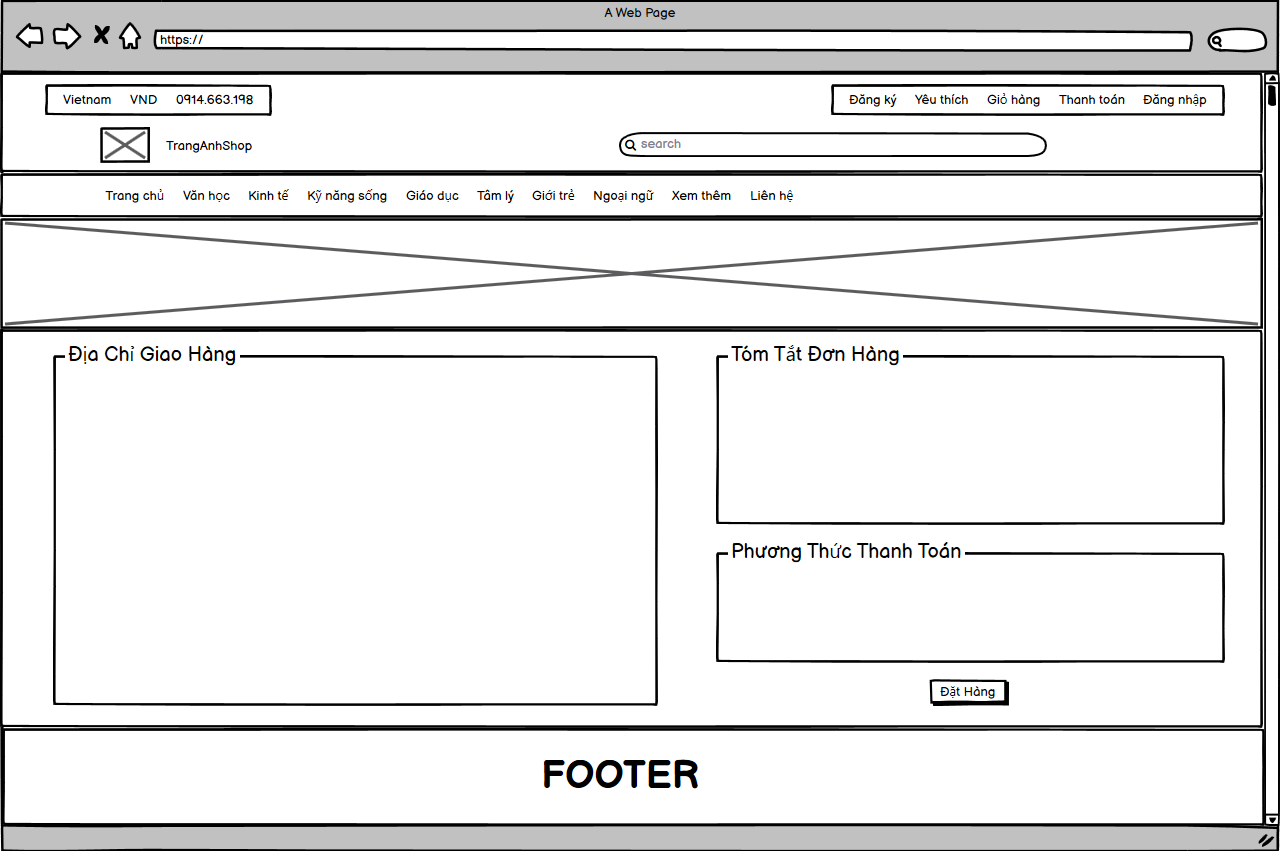
Hình 3. 25 Giao diện trang danh sách yêu thích

**3.2.1.6. Giao diện trang giỏ hàng**



Hình 3. 26 Giao diện trang giỏ hàng

**3.2.1.7. Giao diện trang đặt hàng**



Hình 3. 27 Giao diện trang đặt hàng

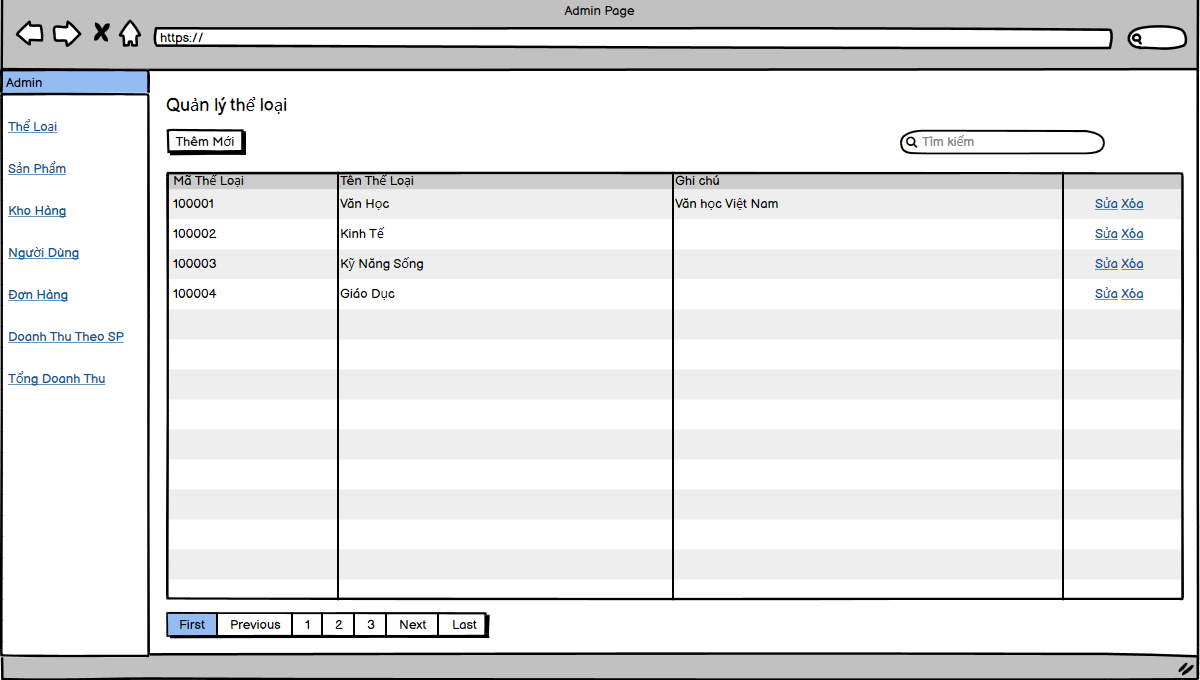
### 3.2.2 Giao diện quản trị

**3.2.2.1. Giao diện trang chủ**



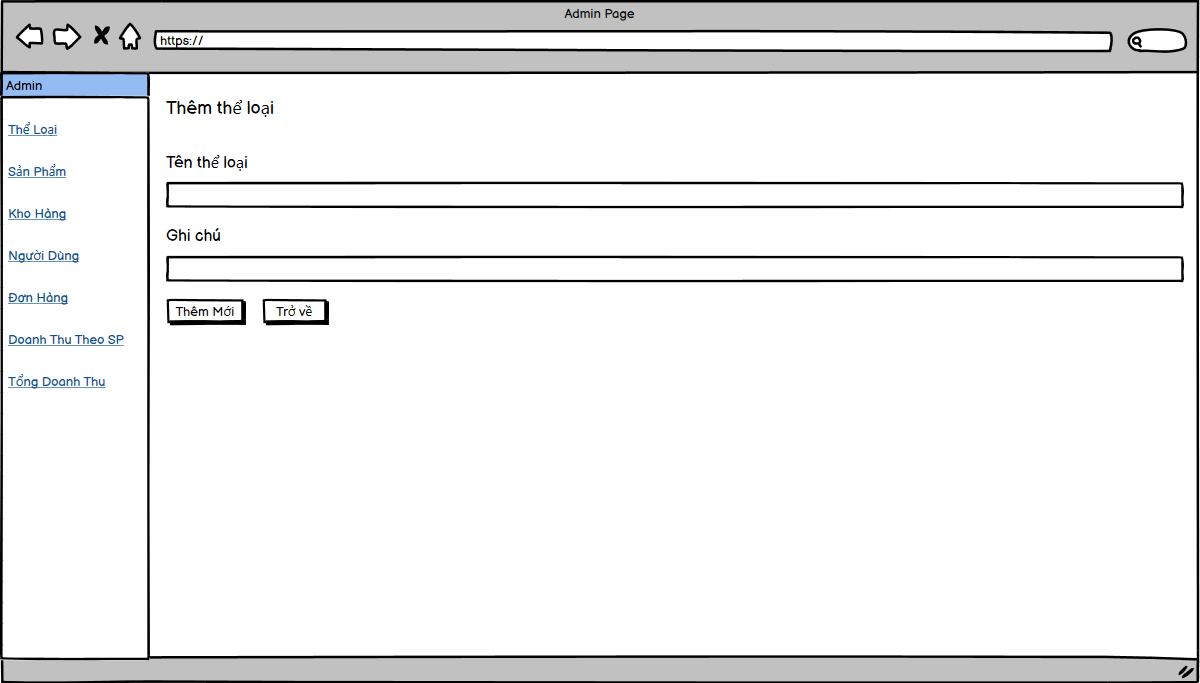
Hình 3. 28 Giao diện trang chủ admin

**3.2.2.2. Giao diện trang quản lý danh sách**



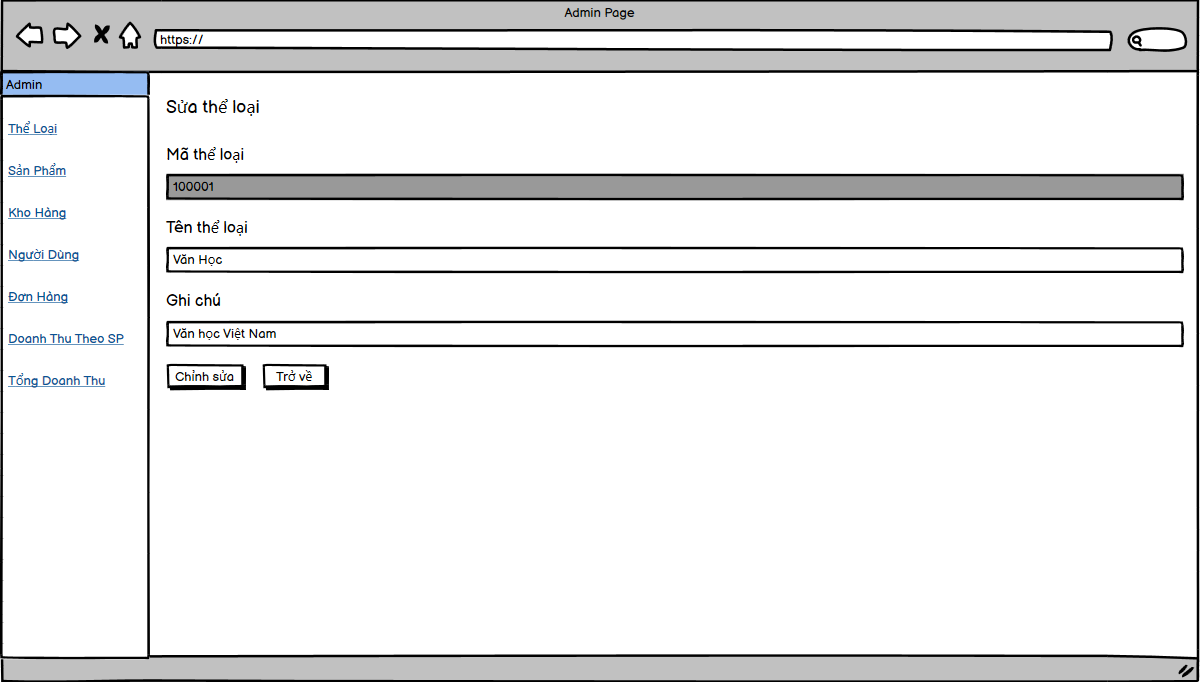
Hình 3. 29 Giao diện trang quản lý danh sách

**3.2.2.3. Giao diện trang thêm mới**



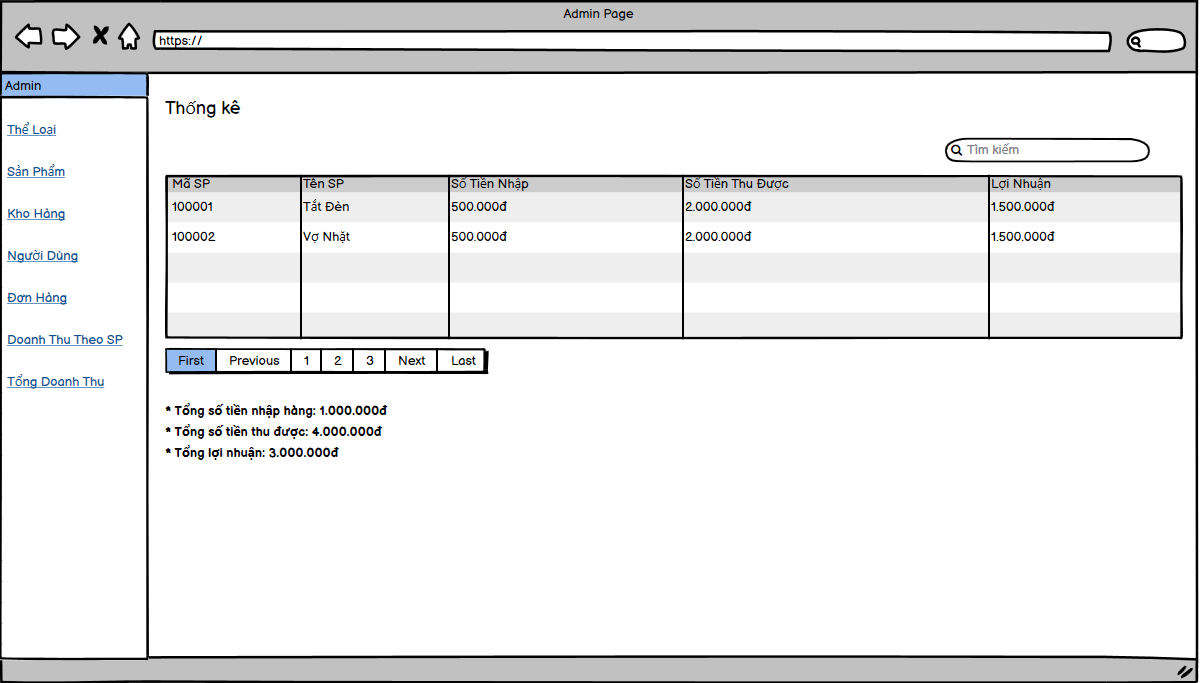
Hình 3. 30 Giao diện trang thêm mới

**3.2.2.4. Giao diện trang chỉnh sửa**



Hình 3. 31 Giao diện trang chỉnh sửa

**3.2.2.5. Giao diện trang thống kê**



Hình 3. 32 Giao diện trang thống kê

## 3.3 Cài đặt hệ thống

### 3.3.1 Chức năng người dùng

Khi bắt truy cập vào website màn hình sẽ hiển thị giao diện trang chủ với bố cục 3 phần là đầu, nội dung và chân website.

Ở phần đầu bao gồm các nội dung như số điện thoại, email, tên, logo, các banner chính giới thiệu.v.v.. của cửa hàng cùng với các chứng năng như tìm kiếm sản phẩm giỏ hàng, danh sách yêu thích, menu, đăng ký, đăng nhập.v.v...

Tiếp theo là phần nội dung của trang web bao gồm sản phẩm được sắp xếp theo danh sach sản phẩm hot, sản phẩm thịnh hành, sản phẩm mới.

### 3.3.2 Chức năng quản trị

# Kết luận

* 1. **Kết quả đạt được**

Sau thời gian thực hiện đề tài, chương trình đã hoàn thành và đạt được một số kết quả sau:

* Hiểu rõ quy trình bán sách trực tuyến
* Tìm hiểu và nắm khá rõ về các công cụ để xây dựng website như: Intellij, Mysql
* Giao diện của chương trình thân thiện, dễ sử dụng,
* Website đã giúp người dùng tiết kiệm thời gian, công sức để có được một sản phẩm ưng ý. Dễ dàng quản lý thông tin chi tiết người dùng, các thông tin về sản phẩm và những đơn hàng của người mua sản phẩm từ đó sẽ thuận tiện cho việc thanh toán và giao hàng.
* Thực hiện kiểm thử phần mềm ở mức độ lập trình.
  1. **Hạn chế của đề tài**
* Chưa thể cập nhật hết được tất cả những sản phẩm hiện đang có mặt trên thị trường.
* Website chưa được thử nghiệm trên mạng Internet.
  1. **Hướng phát triển của đề tài**

Với nỗ lực của bản thân, em đã cố gắng hoàn thành yêu cầu đề tài. Do thời gian và năng lực có hạn nên website của em mới chỉ đi sâu vào chức năng bán sản phẩm.

Hướng phát triển website trở thành một website bán hàng chuyên nghiệp. Cung cấp đầy đủ những mặt hàng hiện đang có trên thị trường với giá cả hợp lý, phải chăng. Đi kèm với bán hàng là những dịch vụ uy tín và chất lượng nhất để phục vụ đến khách hàng.

# Tài liệu tham khảo

1. Trang web: [https://stackoverflow.com//](https://laptrinhjavaweb.com/)

2. Trang web: <https://viettuts.vn/spring>/

3. Trang web: https://spring.io/