Mục Lục

[**Chương I:Khảo sát và phân tích hệ thống** 2](#_Toc483848924)

[1. Giới thiệu và phát biểu bài toán 2](#_Toc483848925)

[2. Chức năng 2](#_Toc483848926)

[**Chương II Thiết kế và cài đặt** 3](#_Toc483848927)

[1. Thiết kế 3](#_Toc483848928)

[a. Phân tích chức năng 3](#_Toc483848929)

[b. Thiết kế CSDL 3](#_Toc483848930)

[c. Sơ đồ lớp 8](#_Toc483848931)

[d. Các sơ đồ khác 8](#_Toc483848932)

[2. Cài đặt 14](#_Toc483848933)

[a. Các framework sử dụng 14](#_Toc483848934)

[b. Các công cụ hỗ trợ 23](#_Toc483848935)

[c. Tiến hành cài đặt 24](#_Toc483848936)

[**Chương III Kết quả và hướng phát triển** 24](#_Toc483848937)

[1. Kết quả đạt được 24](#_Toc483848938)

[2. Hạn chế 25](#_Toc483848939)

[3. Hướng phát triển 25](#_Toc483848940)

**Chương I:Khảo sát và phân tích hệ thống**

## Giới thiệu và phát biểu bài toán

Ngày nay, công nghệ thông tin đã có những bước phát triển mạnh mẽ theo cả chiều rộng và sâu. Máy tính điện tử không còn là một thứ phương tiện quý hiếm mà đang ngày càng trở thành một công cụ làm việc và giải trí thông dụng của con người, không chỉ ở nơi làm việc mà còn ngay cả trong gia đình. Đặc biệt là công nghệ thông tin được áp dụng trên mọi lĩnh vực kinh tế, chính trị, xã hội Ứng dụng công nghệ thông tin và tin học hóa được xem là một trong yếu tố mang tính quyết định trong hoạt động của quốc gia, tổ chức và trong cả các cửa hàng. Nó đóng vai trò hết sức quan trọng và có thể tạo nên bước đột phá mạnh mẽ. Mạng INTERNET là một trong những sản phẩm có giá trị hết sức lớn lao và ngày càng trở nên một công cụ không thể thiếu, là nền tảng để truyền tải, trao đổi thông tin trên toàn cầu. Bằng INTERNET, chúng ta đã thực hiện được những công việc với tốc độ nhanh hơn, 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 trên khắp thế giới, làm biến đổi đáng kể bộ mặt văn hóa, nâng cao đời sống con người. Trong hoạt động sản xuất, kinh doanh, thương mại điển tử đã khẳng định được xúc tiến và thúc đẩy sự phát triển của doanh nghiệp. Đối với một cửa hàng, việc quảng bá và giới thiệu sản phẩm đến khách hàng đáp ứng nhu cầu mua sắm ngày càng cao của khách hàng sẽ là cần thiết. Vì vậy, nhóm chúng em đã thực hiện đề tài “Xây dựng Website bán hàng siêu thị online”. Cửa hàng có thể đưa các sản phẩm lên Website của mình và quản lý Website đó, khách hàng có thể đặt mua, mua hàng của cửa hàng mà không cần đến cửa hàng, cửa hàng sẽ gửi sản phẩm đến tân tay khách hàng. Website là nơi cửa hàng quảng bá tốt nhất tất cả các sản phẩm mình bán ra.

## Chức năng

Trang web bán hàng siêu thị online sẽ có một số chức năng chính được chia thành hai mảng: Người dùng và Admin.

Về phía người dùng sẽ có các tính năng chính như:

* Đăng nhập và mua hàng.
* Mua hàng không đăng nhập
* Đăng kí tài khoản và hưởng ưu đãi.
* Tìm kiếm, lọc sản phẩm,…

Về phía Admin sẽ có các tính năng chính như:

* Đăng nhập.
* Quản lí bao gồm thêm sửa xóa các đối tượng như: sản phẩm, sale, ưu thích,…
* Thống kê các thông tin cần thiết,..

# **Chương II Thiết kế và cài đặt**

## Thiết kế

### Phân tích chức năng

Xây dựng ứng dụng dạng Web, cơ sở dữ liệu My SQL, ngôn ngữ lập trình Java ứng dụng Spring boot

Ứng dụnggồm có các chức năng sau:

*- Quản lý thông tin khách hàng:*

Thêm mới, sửa chữa, xóa bỏ thông tin về khách hàng. Các thông tin về khách hàng bao gồm: mã khách hàng (id), email, tên khách hàng, địa chỉ, điện thoại liên hệ, các đặc điểm khác. Tất cả các thông tin về khách hàng đều được lưu trữ lại.

*- Quản lý thông tin hàng:*

Thêm mới, sửa chữa, xóa bỏ thông tin về hàng. Các thông tin về hàng gồm: mã hàng, tên hàng, giá, đơn vị, loại,ảnh hàng, mô tả, thông tin mô tả thêm về sản phẩm, ngày khởi tạo, lượt view, ngày chỉnh sửa thông tin.

*- Quản lý danh sách giao hàng:*

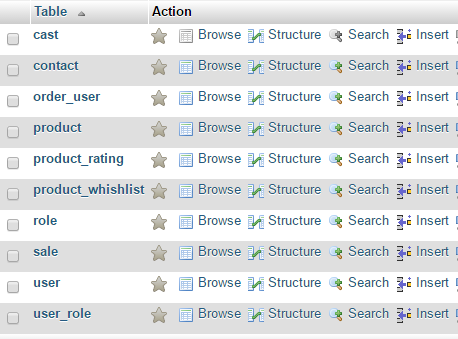
Mã giao hàng, mã sản phẩm, mã khách hàng, địa chỉ, giao hàng, ngày tạo ngày sửa.

*- Thống kê:*

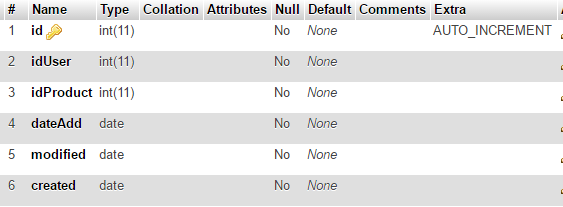
Thống kê số lượt xem, mua, ưu thích,… của sản phẩm trong siêu thị.

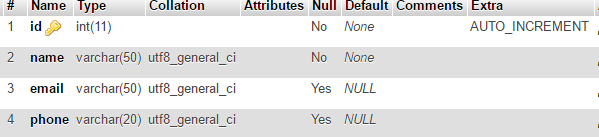
### Thiết kế CSDL

Cơ sở dữ liệu của website được thiết kế như sau.

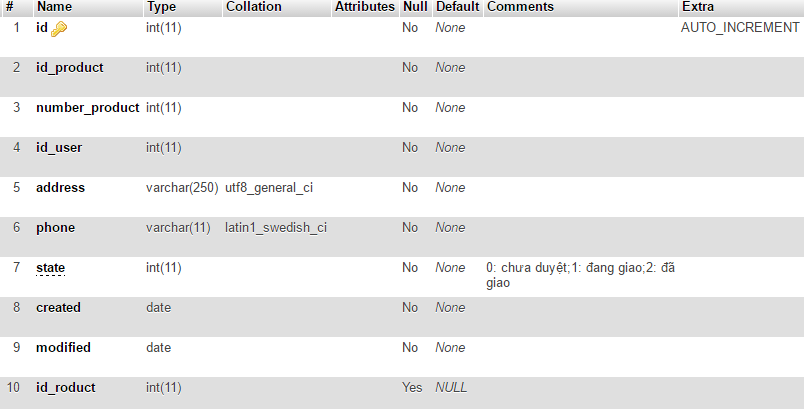


Bảng Cast

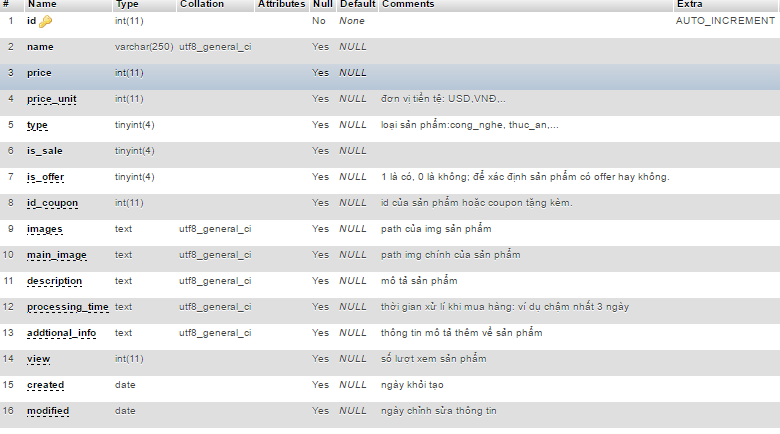




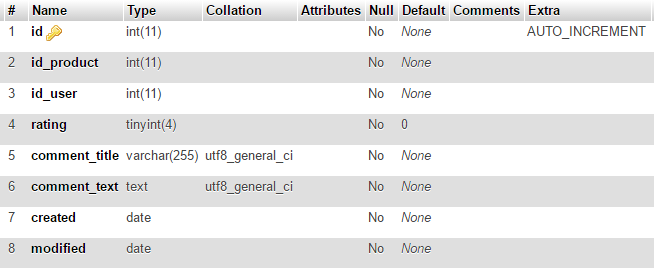
Bảng order\_user



Bảng product



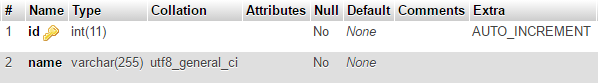
Bảng product rating



Bảng product whishlist

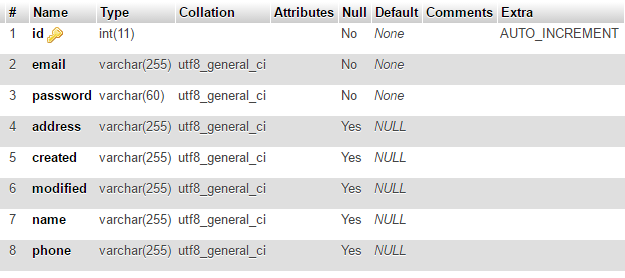


Bảng role

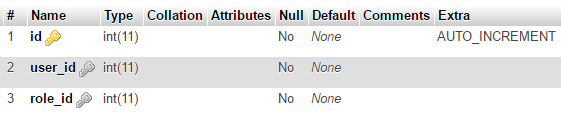


Bảng sale

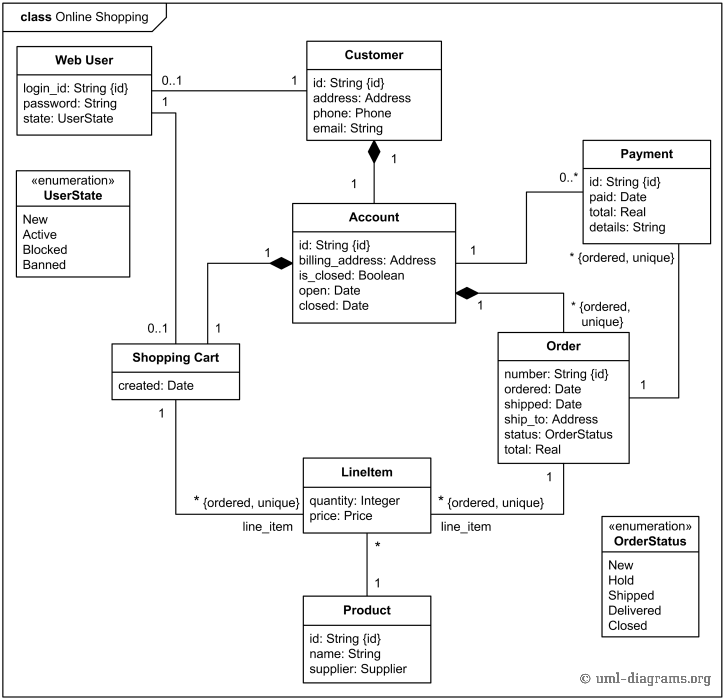
Bảng user



Bảng user\_role



### Sơ đồ lớp



### Các sơ đồ khác

SƠ ĐỒ USECASE

Xem xét Website chúng ta nhận thấy:

* Những người muốn mua hàng vào website để xem thông tin. Những người này là Khách hàng tiềm năng(Guest).
* Những người đã đặt hàng vào kiểm tra đơn hàng, thanh toán v.v.. gọi là Khách hàng (Customer).
* Về phía đơn vị bán hàng, có những người sau đây tham gia vào hệ thống:
* Quản trị hệ thống: Tạo người dùng, Phân quyền, Tạo cửa hàng, Tạo hàng

Tiếp theo ta phân tích từng actor

Trước tiên, xem xét với Actor “**Khách hàng tiềm năng**” trên trang bkc.vn để xem họ sử dụng chức năng nào?

* + Xem trang chủ
  + Xem các sản phẩm theo:

+ Theo chủng loại

+ Nhà  sản xuất

+ Tìm kiếm theo văn bản gõ vào

* + Xem chi tiết sản phẩm được chọn
  + Xem khuyến mãi
  + Xem so sánh
  + Mua hàng
  + Quản lý giỏ hàng
  + Chat với người bán hàng
  + Đăng ký tài khoản để trở thành khách hàng

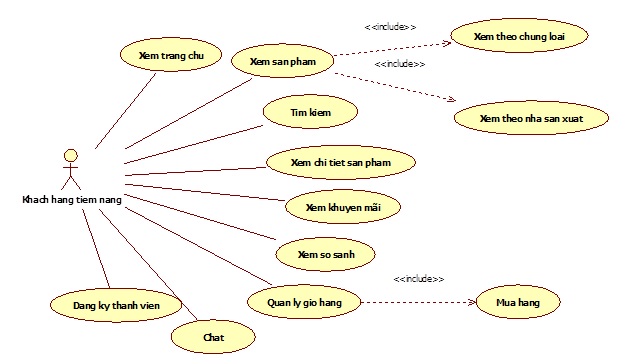
Tiếp theo, xem xét Actor “**Khách hàng**” và nhận thấy họ sử dụng chức năng:

* + Đăng nhập
  + Xem đơn hàng
  + Thanh toán

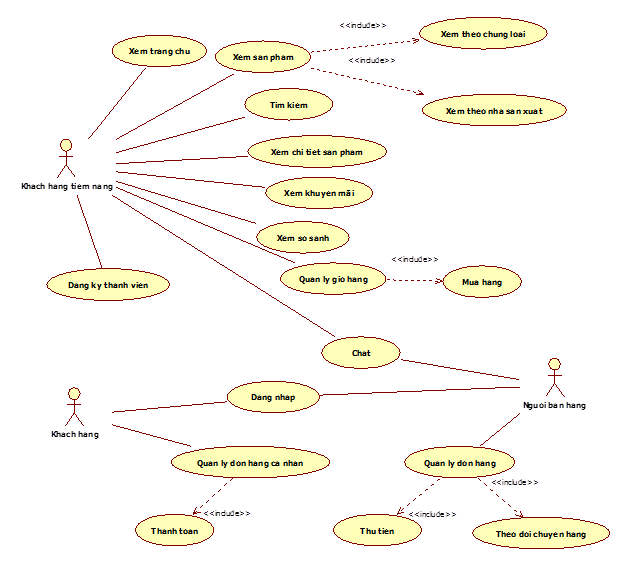
Actor Admin hay người quản lí ở đây họ sử dụng chức năng:

* Đăng nhập
* Quản lí hàng
* Quản lí user,…

Dưới đây là các sơ đồ :

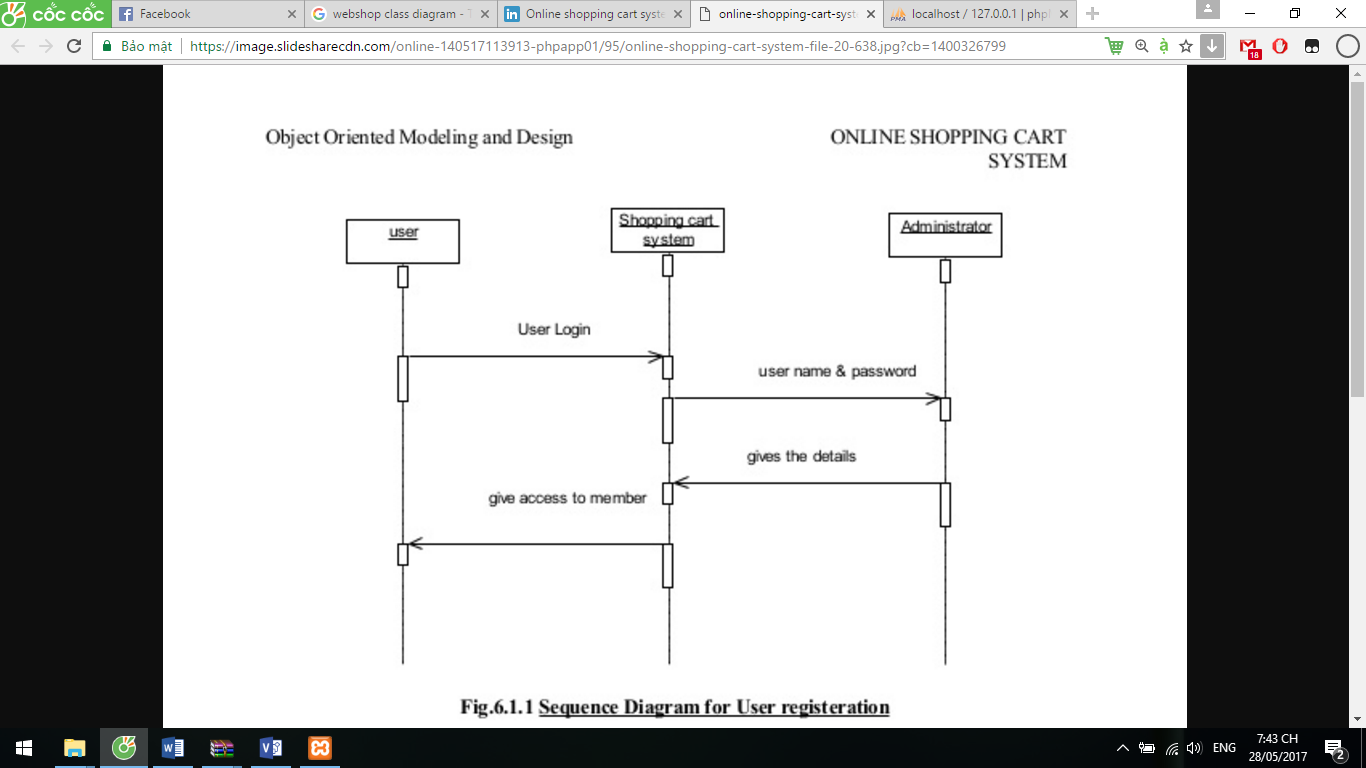




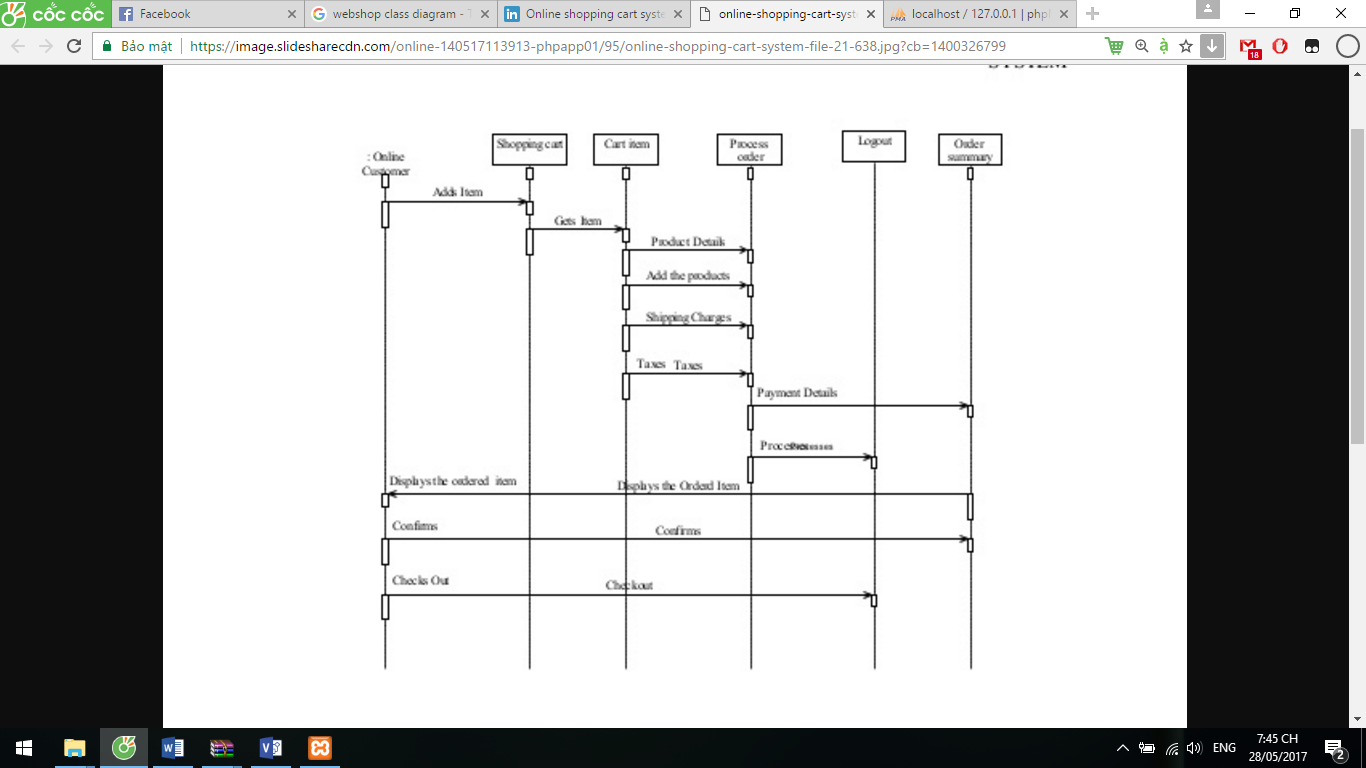


**Sequence Diagram**

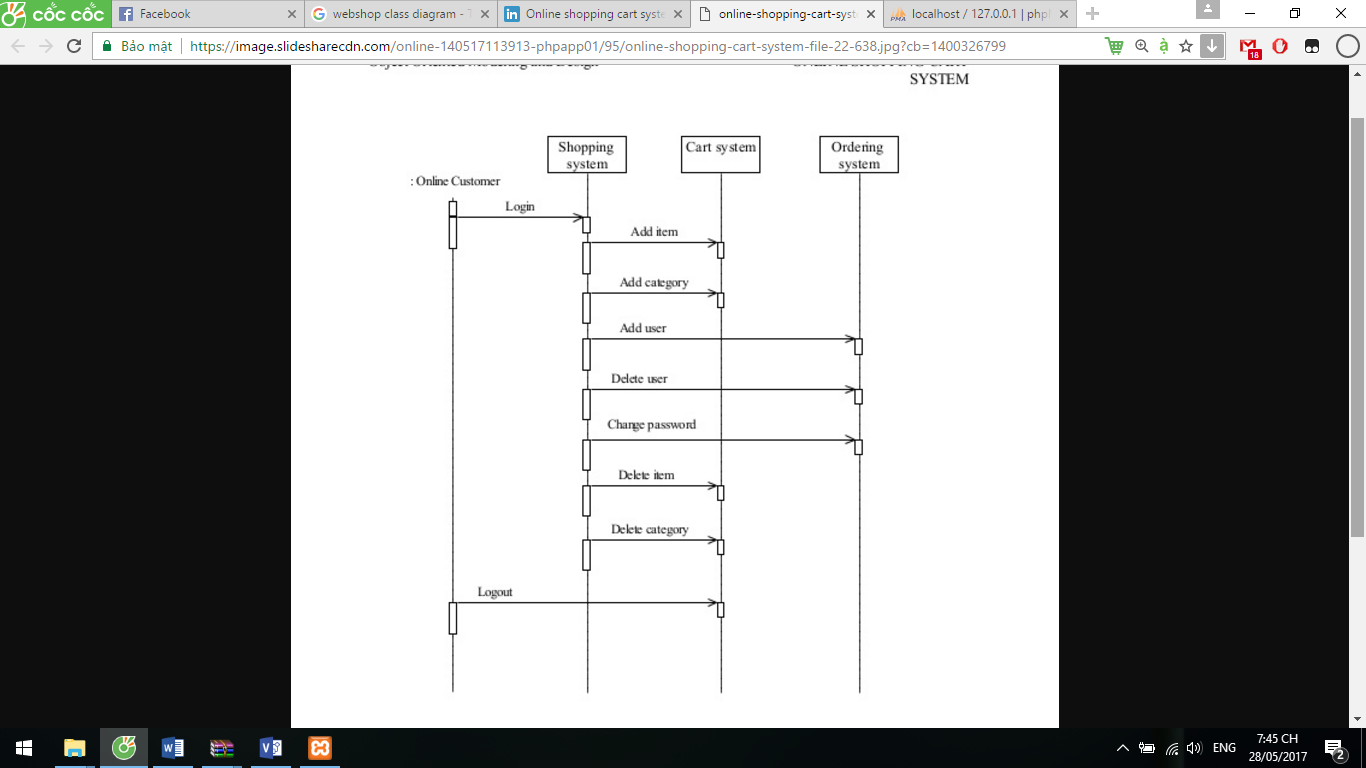
Chức năng đăng kí:



Biểu đồ cho khách hàng



Biểu đồ cho Admin



## Cài đặt

### Các framework sử dụng

#### Spring

**EJB và Spring**

Có thể nói rằng Spring Framework ra đời là nhờ sự thành công của **EJB**. EJB (Enterprise JavaBeans) là một kiến trúc phát triển dành cho các ứng dụng doanh nghiệp đòi hỏi tính mạnh mẽ và khả năng mở rộng cao. EJB là một thành phần quan trọng của nền tảng Java EE. EJB được tạo ra bởi hai gã khổng lồ công nghệ là IBM và Sun vào những năm cuối thập niên 90. Dưới sự phát triển của hai tập đoàn này, EJB đã trở nên nặng nề hơn. Tiêu chuẩn EJB cũng nhanh chóng được thông qua. Sau tất cả, EJB được ca ngợi là giải pháp tốt nhất để xây dựng các ứng dụng phân tán dành cho doanh nghiệp.

Nhưng rồi những chỉ trích về chi tiết kỹ thuật của EJB đã tăng nhanh còn hơn sự nổi tiếng của nó. EJB là một framework lớn với độ phức tạp cao. Các nhà phát triền khi làm việc với EJB đã nhận ra rằng nó không hoạt động như những gì nó đã đề ra. EJB chỉ tuyệt vời trên bàn giấy, nhưng ra thực tế đó lại là một chuyện khác. Trong những phiên bản đầu tiên, các lập trình viên phải cài đặt các lớp implementation cho 2 interface khác nhau, cần qúa nhiều mã nghi thức (ceremonial code) để đáp ứng yêu cầu của EJB ...

EJB vẫn được tiếp tục phát triển, dễ sử dụng hơn nhiều so với buổi ban đầu. Nhưng những chỉ trích vẫn còn đó, nỗi thất vọng trong cộng đồng Java ngày càng tăng lên, đòi hỏi cần có một giải pháp thay thế.

**Sự ra đời của Spring**

Vào năm 2002, Spring Framework phát hành phiên bản đầu tiên bởi Rod Johnson. Việc xây dựng các ứng dụng doanh nghiệp trở nên đơn giản và dễ dàng hơn. Rod Johnson đã cho xuất bản một cuốn sách được phổ biến rộng rãi có nhan đề "J2EE Development without EJB". Điều này đã cho Spring Framework rất nhiều động lực để thay thế EJB.

Ngày nay, Spring đã trở thành framework mã nguồn mở phổ biến nhất để xây dựng các ứng dụng doanh nghiệp. Cách tiếp cận thực tế ban đầu của Rod Johnson tiếp tục được phát triển và hướng tới một bộ công cụ hoàn chỉnh dành cho xây dựng các ứng dụng doanh nghiệp. Theo một số nguồn, trên 50% các ứng dụng web Java hiện nay đang sử dụng Spring.

Để ngăn chặn sự phức tạp trong phát triển các ứng dụng, Spring Framework thường dựa trên các quan điểm như sau:

Đơn giản hóa công việc phát triển thông qua việc sử dụng các đối tượng Java đơn giản hay còn được gọi là POJO (Plain Old Java Object)

Nới lỏng ràng buộc giữa các thành phần thông qua việc sử dụng Dependency Injection và viết các interface

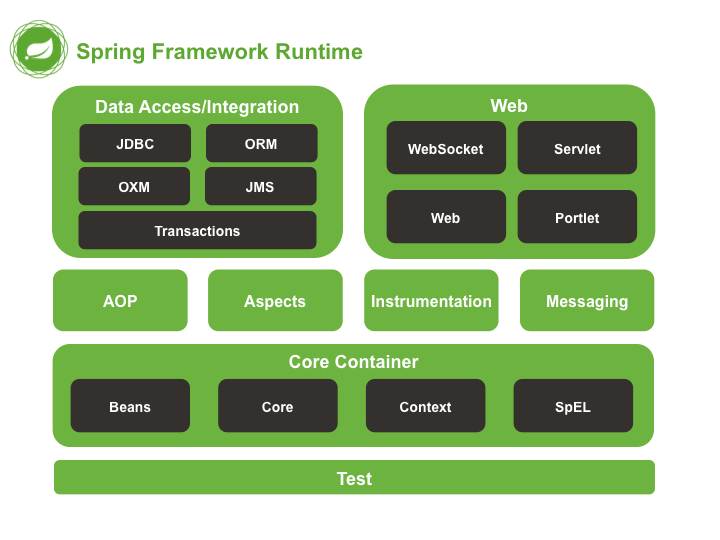
Tiếp cận lập trình khai báo bằng cách sử dụng các quy tắc (convention) và các khía cạnh (aspect) chung

Giảm thiểu các mã nghi thức và soạn sẵn (boilerplate) thông qua việc sử dụng các khuôn mẫu (template) và các khía cạnh

**Spring Core**

Spring Core chính là thành phần trung tâm, cốt lõi của Spring Framework. Đây chính là nền tảng để xây dựng nên các thành phần khác.

Lược đồ dưới đây mô tả mối quan hệ giữa Spring Core với các thành phần khác trong Spring Framework:

[[](https://s3-ap-southeast-1.amazonaws.com/kipalog.com/7ldk3o2zed_c.png)](https://s3-ap-southeast-1.amazonaws.com/kipalog.com/7ldk3o2zed_c.png)

**Spring Bean**

Spring Bean là trung tâm của Spring Core và là trái tim của một ứng dụng Spring. Trái ngược với EJB, Spring Framework được thiết kế từ lõi bằng cách sử dụng các POJO hay các Spring Bean. Spring Bean có thể được hiểu là các đối tượng Java đơn giản. Điều này khá tương đồng với nguyên lý thiết kế "Đơn trách nhiệm" (single responsibility) của Robert C Martin trong lập trình hướng đối tượng.

**Dependency Injection (DI)**

Dependency Injection (có thể dịch tiêm các thành phần phụ thuộc) là một sức mạnh nổi bật của Spring Framework. Trong các phiên bản gốc của EJB người ta không thấy sự xuất hiện của Dependency Injection, nhưng có lẽ do nhận thấy sự thành công từ Spring khi sử dụng Dependency Injection, các phiên bản EJB 3.x đã cung cấp các annotation để thực hiện inject các dependency. Dependency Injection là một mẫu thiết kế phần mềm mà các đối tượng phụ thuộc sẽ được inject vào một lớp nào đó. Dependency Injection là một implementation cụ thể của khái niệm Inversion of Control (đảo ngược điều khiển).

Ta sẽ có một ví dụ đơn giản về Dependency Injection như sau: Bạn có một web controller có nhiệm vụ lưu thông tin gửi từ form người dùng. Theo nguyên lý Đơn trách nhiệm, bạn không muốn lớp controller tương tác với cơ sở dữ liệu. Thay vào đó, bạn sẽ sử dụng một lớp service để làm công việc này. Như vậy, controller của bạn sẽ chỉ phải xử lý dữ liệu của form (get form data, validate data, ...) rồi gọi một phưong thức của lớp service được inject để lưu dữ liệu. Controller không cần phải quan tâm hay lo lắng về kết nối cơ sở dữ liệu, pooling hay bảng nào sẽ được update. Cũng như service không cần phải biết request có những thông tin gì.

**Spring Context**

Spring Context mang mọi thứ lại với nhau. Spring Context kế thừa các tính năng của Spring Bean và bổ sung các hỗ trợ cho internationalization (ví dụ như các resource bundle), event propagation, resource loading ... Ngoài ra, Spring Context cũng hỗ trợ các tính năng của Java EE như EJB, JMX và truy cập từ xa cơ bản. Interface **ApplicationContext** là tiêu điểm của Spring Context.

**Spring Expression Language (SpEL)**

Spring Expression Language là một ngôn ngữ ngắn gọn giúp cho việc cấu hình Spring Framework trở nên linh hoạt hơn.

**Các dự án trong Spring Framework**

Trên thực tế Spring Framework là một tập hợp của nhiều dự án con. Spring Core như chúng ta đã tìm hiểu là nền tảng của các dự án trong Spring Framework. Một một dự án sẽ đảm nhận một chức năng riêng trong việc xây dựng các ứng dụng doanh nghiệp. Chúng ta sẽ cùng điểm qua các dự án lớn trong Spring.

**Spring MVC**

Spring MVC được thiết kế dành cho việc xây dựng các ứng dụng nền tảng web. Đây là một dự án chúng ta không thể bỏ qua khi xây dựng các ứng dụng Java web.

**Spring Data**

Cung cấp một cách tiếp cận đúng đắn để truy cập dữ liệu từ cơ sở dữ liệu quan hệ, phi quan hệ, map-reduce và thậm chí còn hơn thế nữa.

**Spring Security**

Dự án này cung cấp các cơ chế xác thực (authentication) và phân quyền (authorization) cho ứng dụng của bạn.

**Spring Boot**

Spring Boot là một framework giúp chúng ta phát triển cũng như chạy ứng dụng một cách nhanh chóng. Sẽ giới thiệu rõ hơn trong phần sau.

**Spring Batch**

Dự án này giúp chúng ta dễ dàng tạo các lịch trình (scheduling) và tiến trình (processing) cho các công việc xử lý theo mẻ (batch job).

**Spring Integration**

Spring Integration là một implementation của Enterprise Integration Patterns (EIP). Dự án này thiết kế một kiến trúc hướng thông điệp hỗ trợ việc tích hợp các hệ thống bên ngoài.

**Spring XD**

Bạn đang băn khoăn về Big Data, tại sao không thử tìm hiểu Spring XD - mục tiêu của dự án này là đơn giản hóa công việc phát triển các ứng dụng Big Data.

**Spring Social**

Dự án này sẽ kết nối ứng dụng của bạn với các API bên thứ ba của Facebook, Twitter, Linkedin ...

#### Spring boot

Spring Boot là một dự án khá nổi bật trong hệ sinh thái Spring Framework. Nếu như trước đây, công đoạn khởi tạo một dự án Spring khá vất vả từ việc khai báo các dependency trong file pom.xml cho đến cấu hình bằng XML hoặc annotation phức tạp, thì giờ đây với Spring Boot, chúng ta có thể tạo dự án Spring một cách nhanh chóng và cấu hình cũng đơn giản hơn.

Spring boot là một project trong [Spring io platform](http://hainguyenngoc.blogspot.com/2015/03/tong-quang-ve-spring-io-platform.html) nằm ở layer IO Execution.  Nó giảm thiểu effort cần thiết để tạo production-ready, DevOps-friendly, XML-free Spring applications. Đơn giản hóa bootstrapping của các dự án Spring với source code tối thiểu, thực hiện khả năng mở rộng các tính năng hoạt động như tự động kiểm tra metrics, enpoints,... và hỗ trợ embedded containers cho phép tạo ra các tập lệnh có khả năng tự thực thi. Làm cho project dùng spring trở nên độc lập và gọn nhẹ.

Dưới đây là một số tính năng nổi bật của Spring Boot:

* Tạo các ứng dụng Spring độc lập
* Nhúng trực tiếp Tomcat, Jetty hoặc Undertow (không cần phải deploy ra file WAR)
* Các starter dependency giúp việc cấu hình Maven đơn giản hơn
* Tự động cấu hình Spring khi cần thiết
* Không sinh code cấu hình và không yêu cầu phải cấu hình bằng XML ...

Bạn có thể dùng spring boot để tạo ứng dụng Java chạy bằng command line 'java -jar' hoặc export gói war để deploy lên server như thông thường. Spring boot cung cấp cho bạn một cli tool để run 'spring scripts'.

**Yêu cầu hệ thống**

Spring Boot 1.2.2.RELEASE yêu cầu JDK 7 và Spring Framework 4.1.3 hoặc cao hơn. Bạn có thể dùng spring boot với Java 6 nhưng phải chỉnh lại cấu hình details. Yêu cầu maven (3.2+) hoặc Gradle (1.12+).

#### Bootstrap

Bootstrap là một framework HTML, CSS, và JavaScript cho phép thiết kế phát triên responsive web mobile. Nếu bạn chưa hiểu về responsive web mobile bạn có thể xem bài viết của tôi ở đây để hiểu hơn về nó “Khác nhau cơ bản giữa mobile web và Responsive Web Design“.

**Bootstrap là gì?**

Bootstrap là một framework cho phép thiết kế website reponsive nhanh hơn và dễ dàng hơn Bootstrap là bao gồm các HTML templates, CSS templates và Javascript tao ra những cái cơ bản có sẵn như: typography, forms, buttons, tables, navigation, modals, image carousels và nhiều thứ khác. Trong bootstrap có thêm các plugin Javascript trong nó. Giúp cho việc thiết kế reponsive của bạn dễ dàng hơn và nhanh chóng hơn.

Lịch sử Bootstrap Bootstrap là dược phát triển bởi Mark Otto và Jacob Thornton tại Twitter. Nó được xuất bản như là một mã nguồn mở vào tháng 8 năm 2011 trên GitHub. Tính ra đến thời điểm mình viết bài viết này nó cũng đã phát triển được 3 năm rồi. Bản bootstrap mới nhất bây giờ là bootstrap 3.

**Tại sao bạn nên sử dụng Bootstrap?**

Những điểm thuận lợi khi bản sử dụng bootstrap:

Rất dễ để sử dụng: Nó đơn giản vì nó được base trên HTML, CSS và Javascript chỉ cẩn có kiến thức cơ bản về 3 cái đó là có thể sử dụng bootstrap tốt.  
Tính năng Responsive: Bootstrap’s xây dựng sẵn reponsive css trên các thiết bị phones, tablets, và desktops Mobile: Trong Bootstrap 3 mobile-first styles là một phần của core framework Tương thích với trình duyệt: Nó tương thích với tất cả các trình duyệt (Chrome, Firefox, Internet Explorer, Safari, and Opera).

#### Jquery

Với sự phát triển rất mau lẹ của Internet, người dùng ngày càng quan tâm hơn đến hình thức của một trang web. Trước đây một trang web chỉ cần có banner, nội dung và ít footer hời hợt là đã được cho là một trang web hoàn chỉnh. Nhưng bây giờ trang web đó phải có banner bắt mắt, nội dung hay và còn nhiều hiệu ứng lạ mắt khác nữa thì mới có thể thu hút được người đọc.

Chính vì thế những web designer bắt đầu chú ý đến các thư viện JavaScript mở như jQuery để tạo ra các hiệu ứng có thể tương tác trực tiếp với người đọc một cách nhanh chóng và dễ dàng hơn rất nhiều là sử dụng thuần JavaScript.

Nhưng nếu bạn là người mới làm quen với jQuery bạn sẽ thấy không biết phải bắt đầu từ đâu vì jQuery cũng giống như bất cứ thư viện nào khác cũng có rất nhiều functions. Cho dù bạn có đọc phần tài liệu hướng dẫn sử dụng của jQuery thì bạn vẫn thấy rất phức tạp và khó hiểu. Nhưng bạn yên tâm một điều là jQuery có cấu trúc rất mạch lạc và theo hệ thống. Cách viết code của jQuery được vay mượn từ các nguồn mà các web designer đa phần đã biết như HTML và CSS. Nếu từ trước đến nay bạn chỉ là Designer chứ không phải coder, bạn cũng có thể dễ dàng học jQuery vì kiến thức về CSS giúp bạn rất nhiều khi bắt đầu với jQuery.

Nhận thấy jQuery còn khá mới mẻ với nhiều bạn và nó cũng là thư viện được đông đảo người sử dụng. Izwebz giới thiệu đến các bạn loạt bài về jQuery. Trong loạt bài này chúng ta sẽ tìm hiểu về jQuery và các tính năng của nó. Trước hết bạn nên biết jQuery có thể làm được những gì.

**Những gì Jquery có thể làm**

**Hướng tới các thành phần trong tài liệu HTML**. Nếu không sử dụng thư viện JavaScript này, bạn phải viết rất nhiều dòng code mới có thể đạt được mục tiêu là di chuyển trong cấu trúc cây (hay còn gọi là DOM = Document Object Model) của một tài liệu HTML và chọn ra các thành phần liên quan. Jquery cho phép bạn chọn bất cứ thành phần nào của tài liệu để “vọc” một cách dễ dàng như sử dụng CSS.

**Thay đổi giao diện của một trang web.** CSS là công cụ rất mạnh để định dạng một trang web nhưng nó có một nhược điểm là không phải tất cả các trình duyệt đều hiển thị giống nhau. Cho nên jQuery ra đời để lấp chỗ trống này, vì vậy các bạn có thể sử dụng nó để giúp trang web có thể hiển thị tốt trên hầu hết các trình duyệt. Hơn nữa jQuery cũng có thể thay đổi class hoặc những định dạng CSS đã được áp dụng lên bất cứ thành phần nào của tài liệu HTML ngay cả khi trang web đó đã được trình duyệt load thành công. Thay đổi nội dung của tài liệu. Jquery không phải chỉ có thể thay đổi bề ngoài của trang web, nó cũng có thể thay đổi nội dung của chính tài liệu đó chỉ với vài dòng code. Nó có thể thêm hoặc bớt nội dung trên trang, hình ảnh có thể được thêm vào hoặc đổi sang hình khác, danh sách có thể được sắp xếp lại hoặc thậm chí cả cấu trúc HTML của một trang web cũng có thể được viết lại và mở rộng. Tất cả những điều này bạn hoàn toàn có thể làm được nhờ sự giúp đỡ của API (Application Programming Interface = Giao diện lập trình ứng dụng).

**Tương tác với người dùng.** Cho dù công cụ bạn dùng có mạnh mẽ đến mấy, nhưng nếu bạn không có quyền quyết định khi nào nó được sử dụng thì công cụ đó cũng coi như bỏ. Với thư viện javaScript như jQuery, nó cho bạn nhiều cách để tương tác với người dùng ví dụ như khi người dùng nhấp chuột vào đường link thì sẽ có gì xảy ra. Nhưng cái hay của nó là không làm cho code HTML của bạn rối tung lên chính là nhờ các Event Handlers. Hơn nữa Event Handler API sẽ bảo đảm rằng trang web của bạn tương thích hầu hết với các trình duyệt, điều này đã và đang làm đau đầu rất nhiều các web designer.

**Tạo hiệu ứng động cho những thay đổi của tài liệu.** Để tương tác tốt với người dùng, các web designer phải cho người dùng thấy được hiệu ứng gì sẽ xảy ra khi họ làm một tác vụ nào đó. Jquery cho phép bạn sử dụng rất nhiều hiệu ứng động như mờ dần, chạy dọc chạy ngang v.v.. và nếu vẫn chưa đủ, nó còn cho phép bạn tự tạo ra các hiệu ứng của riêng mình.

**Lấy thông tin từ server mà không cần tải lại trang web.** Đây chính là công nghệ ngày càng trở nên phổ biến Asynchronous JavaScript And XML (AJAX), nó giúp người thiết kế web tạo ra những trang web tương tác cực tốt và nhiều tính năng. Thư viện jQuery loại bỏ sự phức tạp của trình duyệt trong quá trình này và cho phép người phát triển web có thể tập trung vào các tính năng đầu cuối. Đơn giản hoá các tác vụ javaScript. Ngoài những tính năng như đã nêu ở trên, jQuery còn cho phép bạn viết code javaScript đơn giản hơn nhiều so với cách truyền thống như là các vòng lặp và điều khiển mảng.

**Tại sao jQuery làm việc tốt**

Người dùng ngày càng quan tâm hơn đến Dynamic HTML, đó cũng là nền móng cho sự ra đời của những javaScript Frameworks. Có frameworks thì chỉ tập trung vào một vài tính năng vừa nêu ở trên, có cái thì ráng bao gồm tất cả những hiệu ứng, tập tính và nhồi nhét vào một package. Để đảm bảo là một thư viện “nhanh gọn nhẹ” nhưng vẫn “ngon bổ rẻ” với các tính năng đã nêu ở trên, jQuery sử dụng những chiến lược sau:

**Tận dụng kiến thức về CSS.** Các jQuery Selector hoạt động y chang như CSS Selector với cùng cấu trúc và cú pháp. Chính vì thế thư viện jQuery là cửa ngõ cho các web designer muốn thêm nhiều tính năng hơn nữa cho trang web của mình. Bởi vì điều kiện tiên quyết để trở thành một web designer chuyên nghiệp là khả năng sử dụng CSS thuần thục. Với kiến thức có sẵn về CSS, bạn sẽ có sự khởi đầu thuận lợi với jQuery.

**Hỗ trợ Plugin.** Để tránh bị rơi vào trạng thái quá tải tính năng, jQuery cho phép người dùng tạo và sử dụng Plugin nếu cần. Cách tạo một plugin mới cũng khá đơn giản và được hướng dẫn cụ thể, chính vì thế cộng đồng sử dụng jQuery đã tạo ra một loạt những plugin đầy tính sáng tạo và hữu dụng.

**Xoá nhoà sự khác biệt giữa trình duyệt.** Một thức tế tồn tại là mỗi một hệ thống trình duyệt lại có một kiểu riêng để đọc trang web. Dẫn đến một điều làm đau đầu các web designer là làm thế nào để cho trang web có thể hiển thị tốt trên mọi trình duyệt. Cho nên đôi khi người ta phải làm hẳn một phần code phức tạp để đảm bảo rằng trang web của họ được hiển thị gần như tương đồng ở các trình duyệt phổ biến. Jquery giúp bạn thêm một lớp bảo vệ cho sự khác biệt của trình duyệt và giúp quá trình này diễn ra dễ dàng hơn rất nhiều.

**Luôn làm việc với Set.** Ví dụ khi chúng ta yêu cầu jQuery tìm tất cả các thành phần có class là delete và ẩn chúng đi. Chúng ta không cần phải loop qua từng thành phần được trả về. Thay vào đó, những phương pháp như là hide() được thiết kế ra để làm việc với set thay vì từng thành phần đơn lẻ. Kỹ thuật này được gọi là vòng lặp ẩn, điều đó có nghĩa là chúng ta không phải tự viết code để loop nữa mà nó vẫn được thực thi, chính vì thế code của chúng ta sẽ ngăn hơn rất nhiều.

**Cho phép nhiều tác vụ diễn ra trên cùng một dòng.** Để tránh phải sử dụng những biến tạm hoặc các tác vụ lặp tốn thời gian, jQuery cho phép bạn sử dụng kiểu lập trình được gọi là Chaining cho hầu hết các method của nó. Điều đó có nghĩa là kết quả của các tác vụ được tiến hành trên một thành phần chính là thành phần đó, nó sẵn sàng cho tác vụ tiếp theo được áp dụng lên nó. Những chiến lược được nêu ở trên giúp kích thước của jQuery rất nhỏ bé chỉ khoảng trên dưới 20Kb dạng nén. Nhưng vẫn đảm bảo cung cấp cho chúng ta những kỹ thuật để giúp code trên trang nhỏ gọn và mạch lạc.

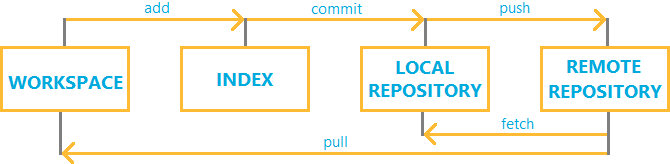
Jquery sở dĩ trở nên phổ biến là do cách sử dụng đơn giản và bên cạnh đó còn có một cộng đồng sử dụng mạnh mẽ vẫn ngày ngày phát triển thêm Plugin và hoàn thiện những tính năng trọng tâm của jQuery. Cho dù thực tế là vậy, nhưng jQuery lại là thư viện javaScript hoàn toàn miễn phí cho mọi người sử dụng. Tất nhiên nó được bảo vệ bởi luật GNU Public License và MIT License, nhưng bạn cứ yên tâm là bạn có thể sử dụng nó trong hầu hết các trường hợp kể cả thương mại lẫn cá nhân.

### Các công cụ hỗ trợ

1. Xampp
2. Eclipse
3. Git và ToitorseGit

Đối với các dự án phát triển phần mềm lớn nhỏ, việc quản lý mã nguồn theo cách thông thường sẽ mất nhiều thời gian, chi phí và kém hiệu quả. Các thành viên trong nhóm phát triển sẽ phải tìm một cách nào đó để phiên bản làm việc trên máy tính cá nhân là mới nhất. Khi dự án có sự thay đổi, việc phục hồi lại thời điểm cần thiết là khá khó khăn. Do đó, chúng ta cần một phương pháp nào đó có thể khắc phục được các nhược điểm kể trên. Một trong những phương pháp hiệu quả nhất là sử dụng một hệ thống **Version Control System** (VCS) mà tiêu biểu là Git.

Quy trình làm việc là rất quan trọng và cần được rèn luyện nhiều nếu người sử dụng muốn thành thạo trong việc sử dụng Git. Quy trình làm việc của Git được biểu diễn bằng sơ đồ sau:



**Đặc điểm chung :**

* Thao tác cả Git và Mercurial
* Dễ dàng sử dụng, trực quan
* Theo sát git-flow
* So sánh tập tin và merge

Có nhiều lập trình viên sử dụng, quản lý Git dưới dạng command line, và để đạt được mức độ thuần thục và nhớ các câu lệnh cũng như quy trình (git-flow) để làm việc với Git thì quả thật mất thời gian rất dài.

Theo cá nhân em thì hãy để phần mềm làm việc đó, hãy tập trung cho việc code. Do vậy em đã giới thiệu một phần mềm GUI được sử dụng để quản lý Git.

Nhóm em sử dụng chủ yếu 2 phần mềm GUI sau:

[Tortoise Git](https://tortoisegit.org/)Được phát triển bởi cộng đồng mở nguồn mở

* Tích hợp với trình đơn trong menu của Explorer
* Tích hợp với hệ thống quản lý tracking issue
* Là phần mềm mã nguồn mở

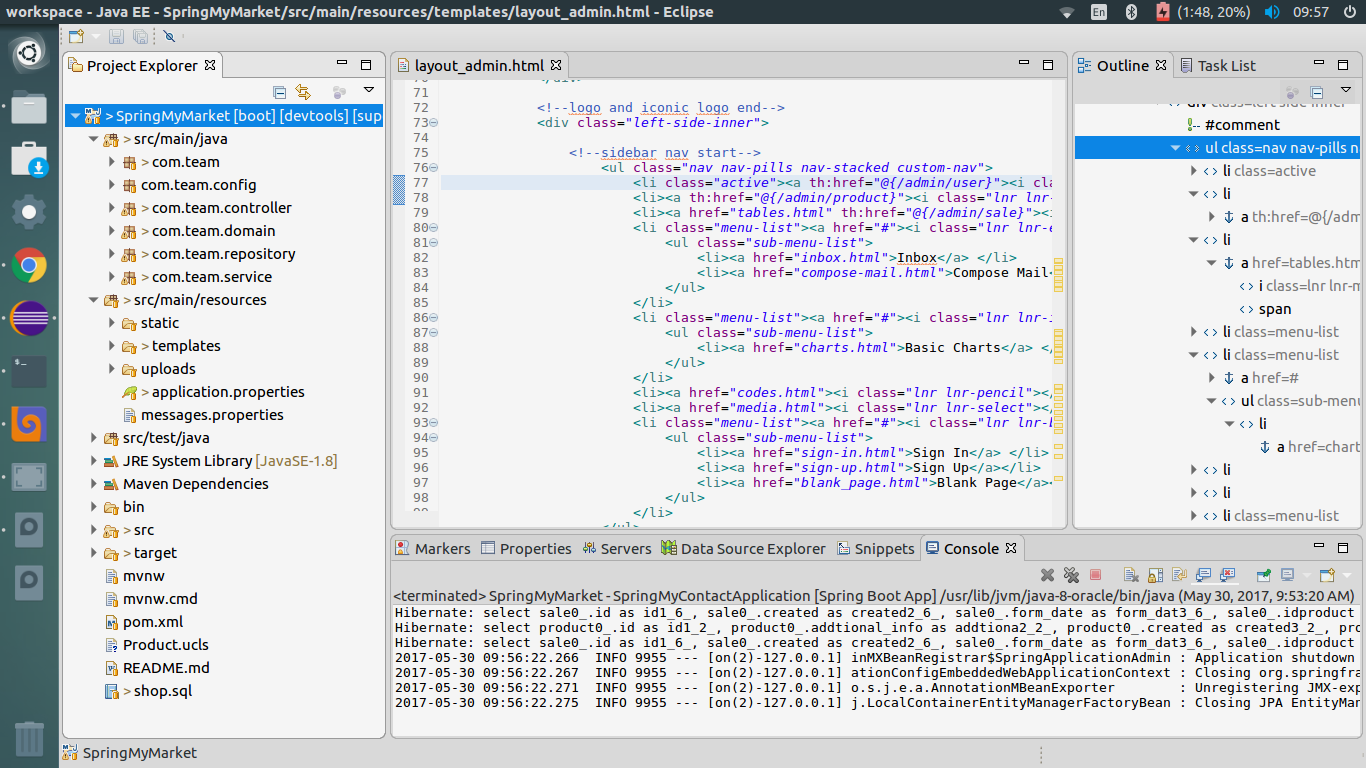
[Smart Git](http://www.syntevo.com/smartgit/)Được phát triển bởi nhà sản xuất phần mềm Syntevo

* Tương tác cả Git, Mercurial và cả SVN
* Sử dụng với nhiều nhà cung cấp: Github, Bitbucket, Atlassian Stash, Beanstalk, Codebase…

### Tiến hành cài đặt

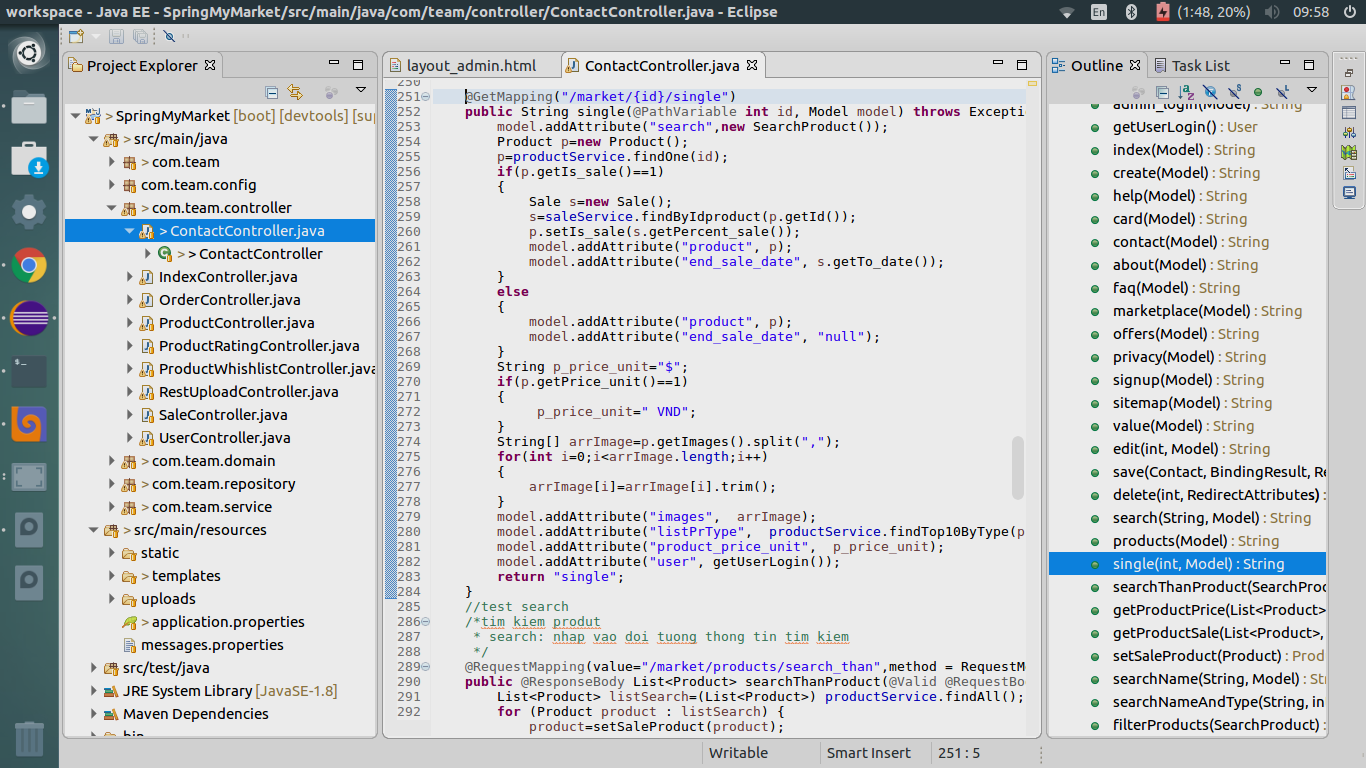
#### Các code cơ bản

* Cấu trúc chương trình:

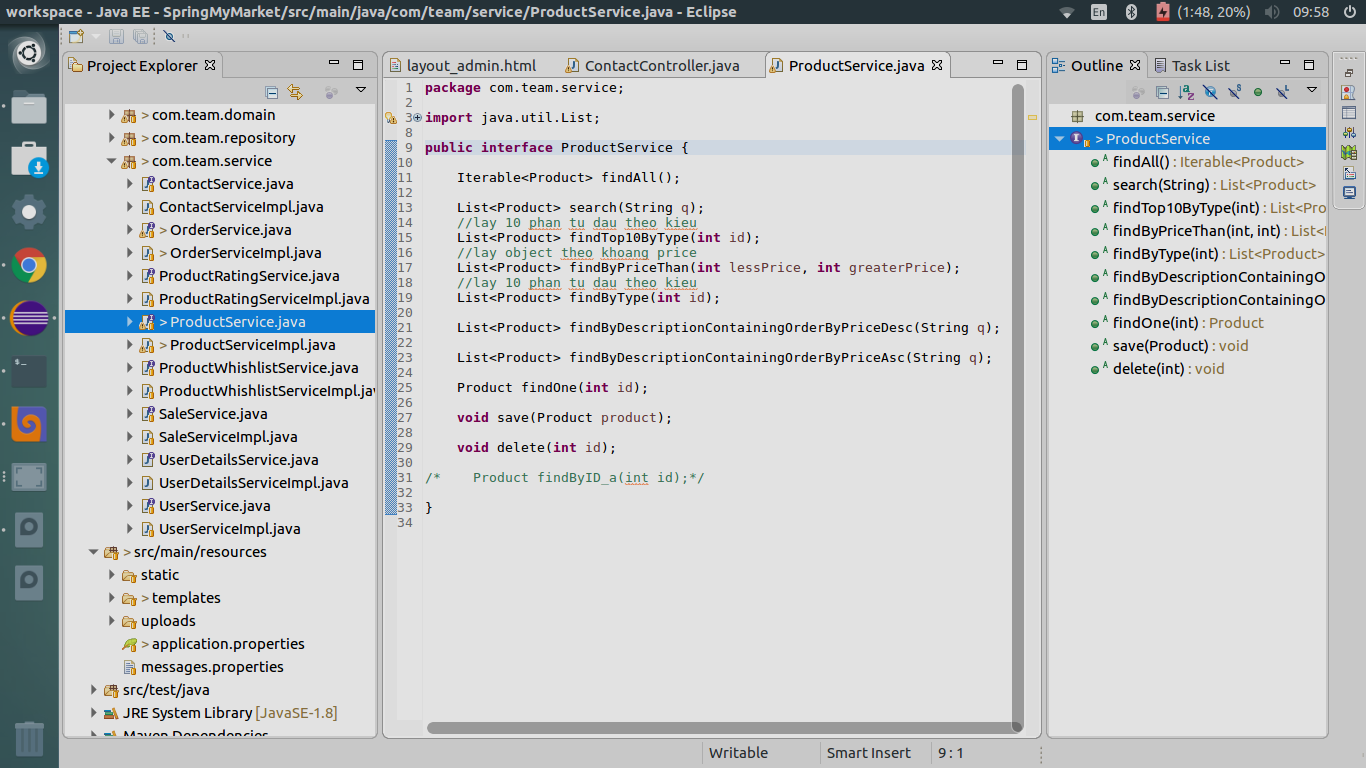


* Một số code cài đặt

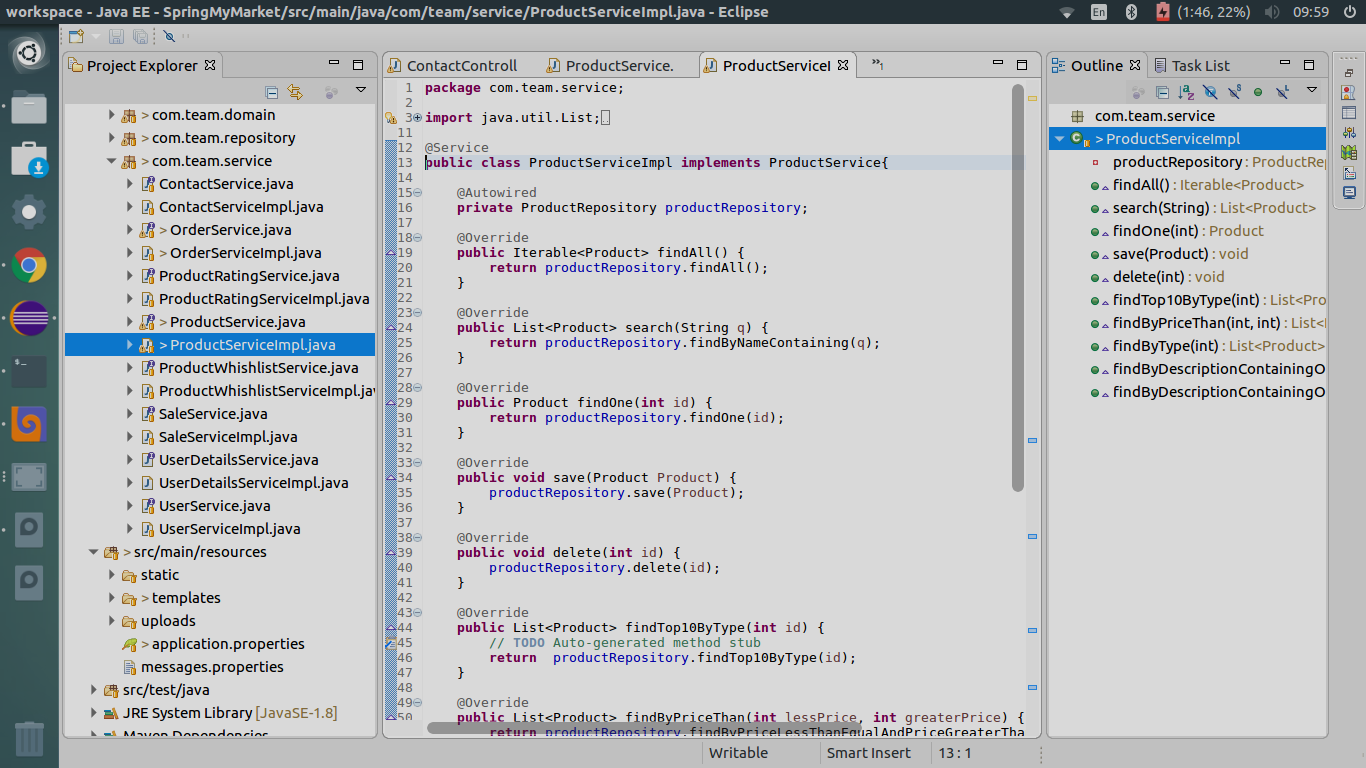
Controller



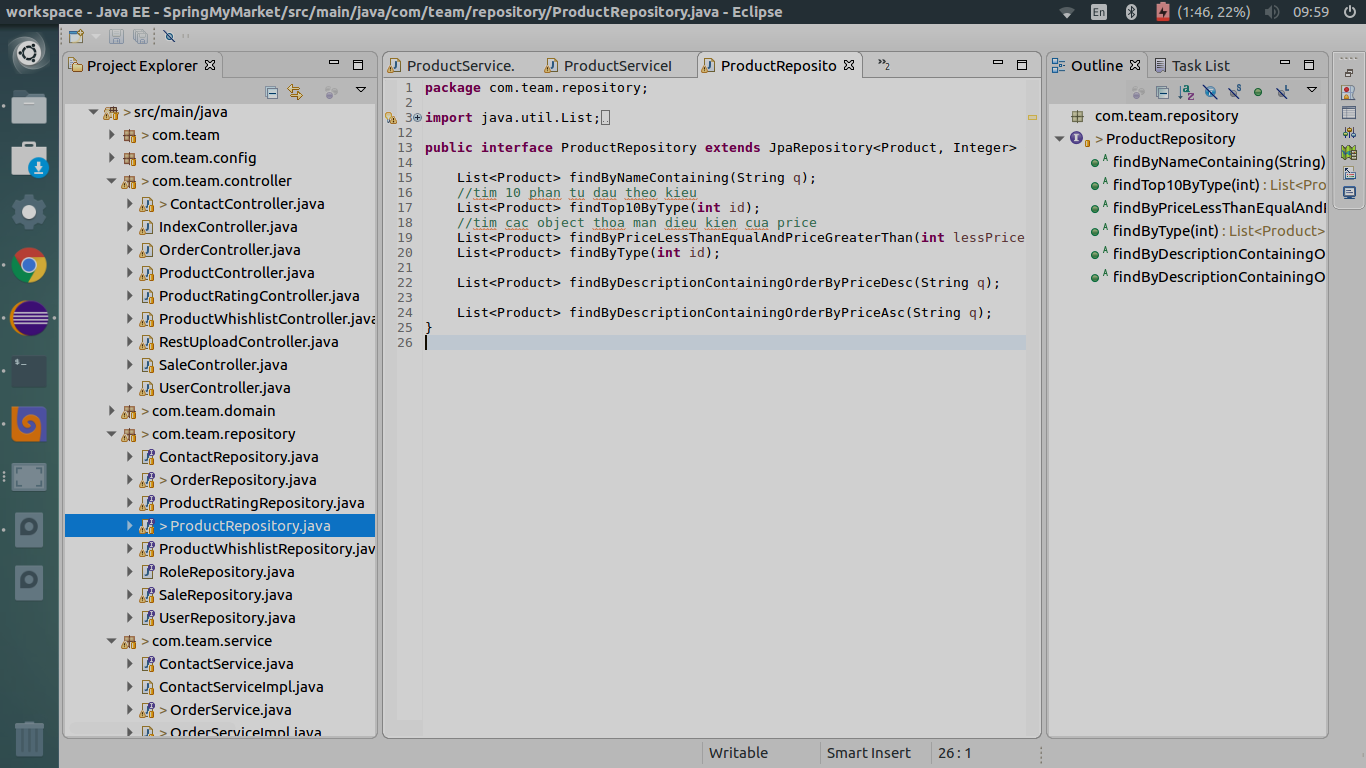
Service



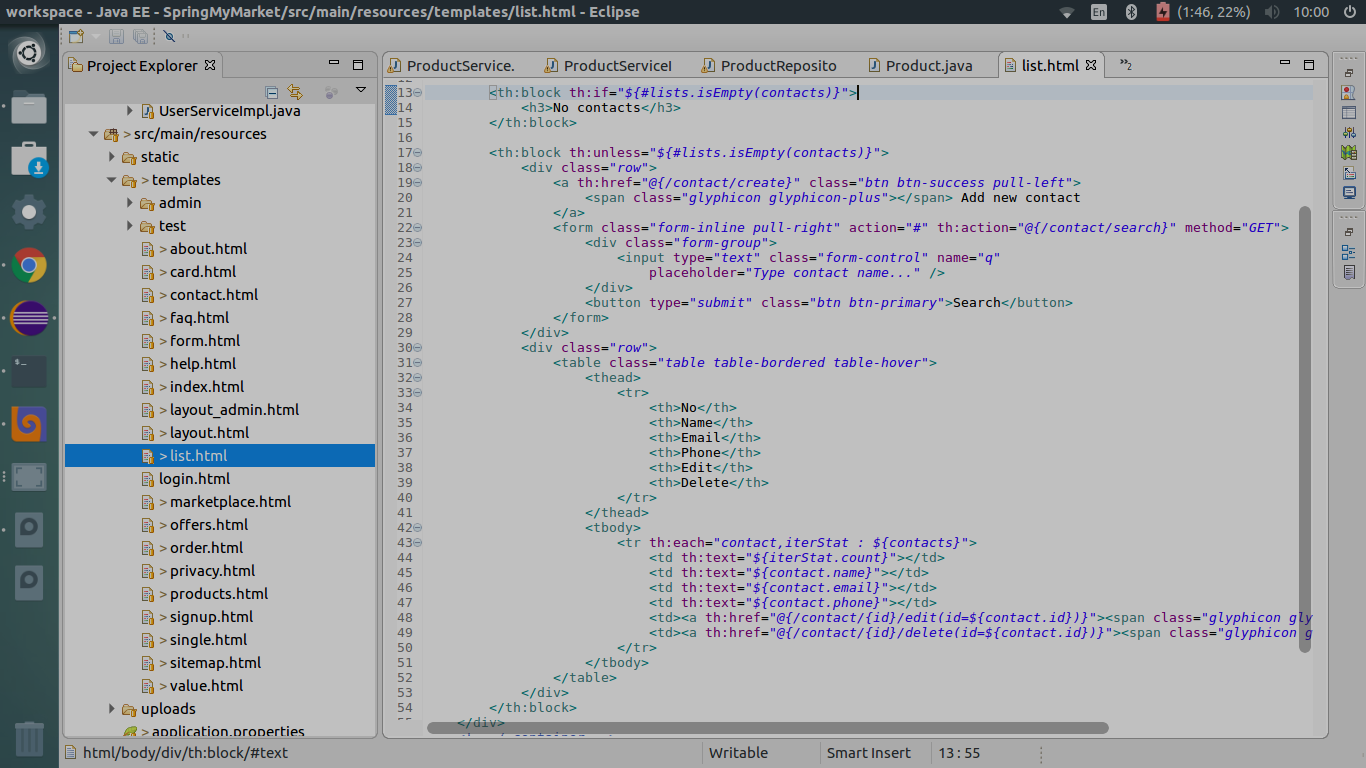
Service implement



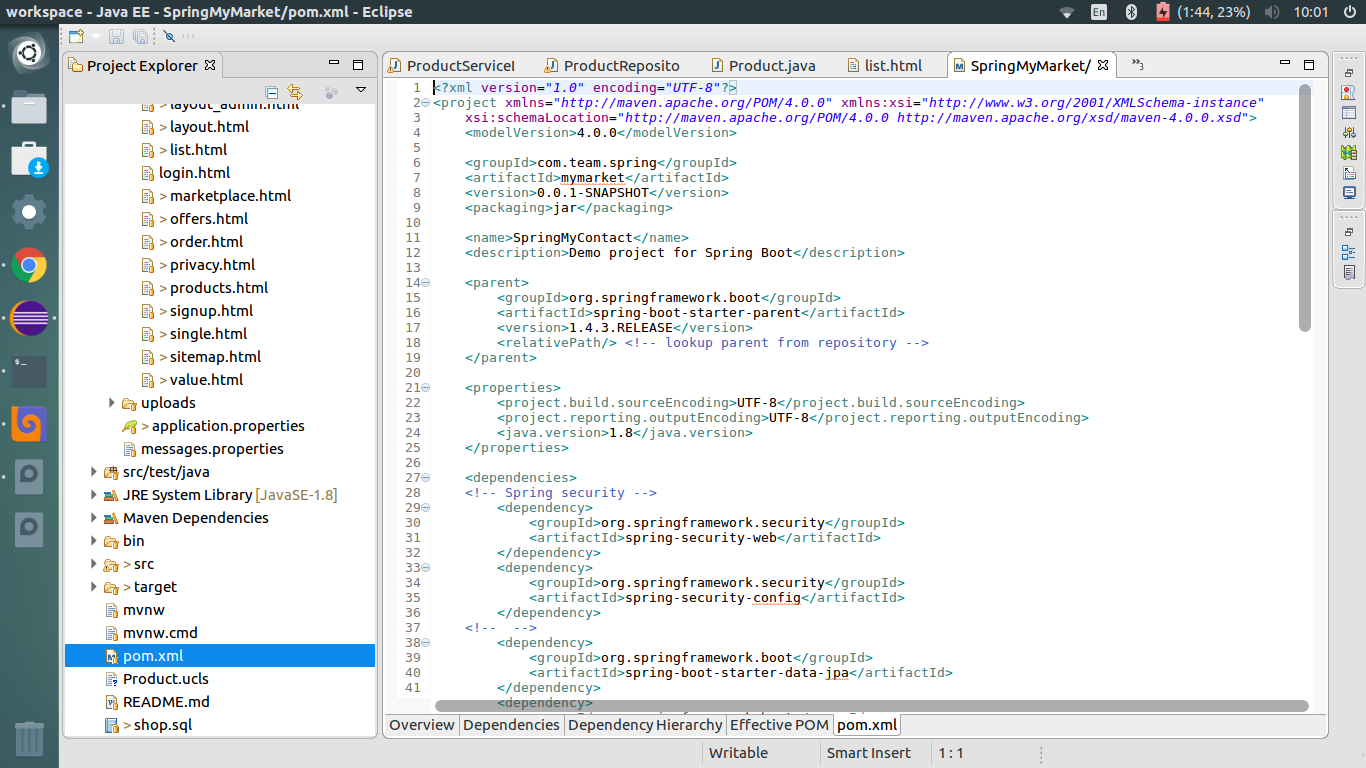
Repository



Thymeleaf

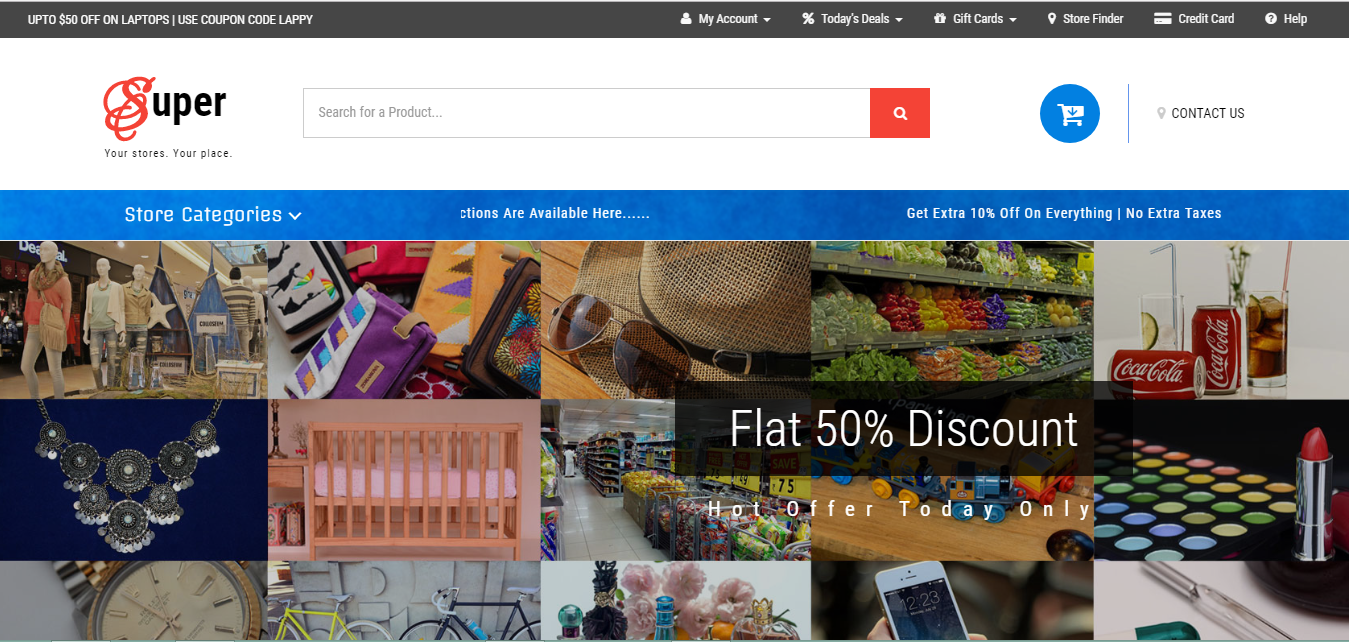


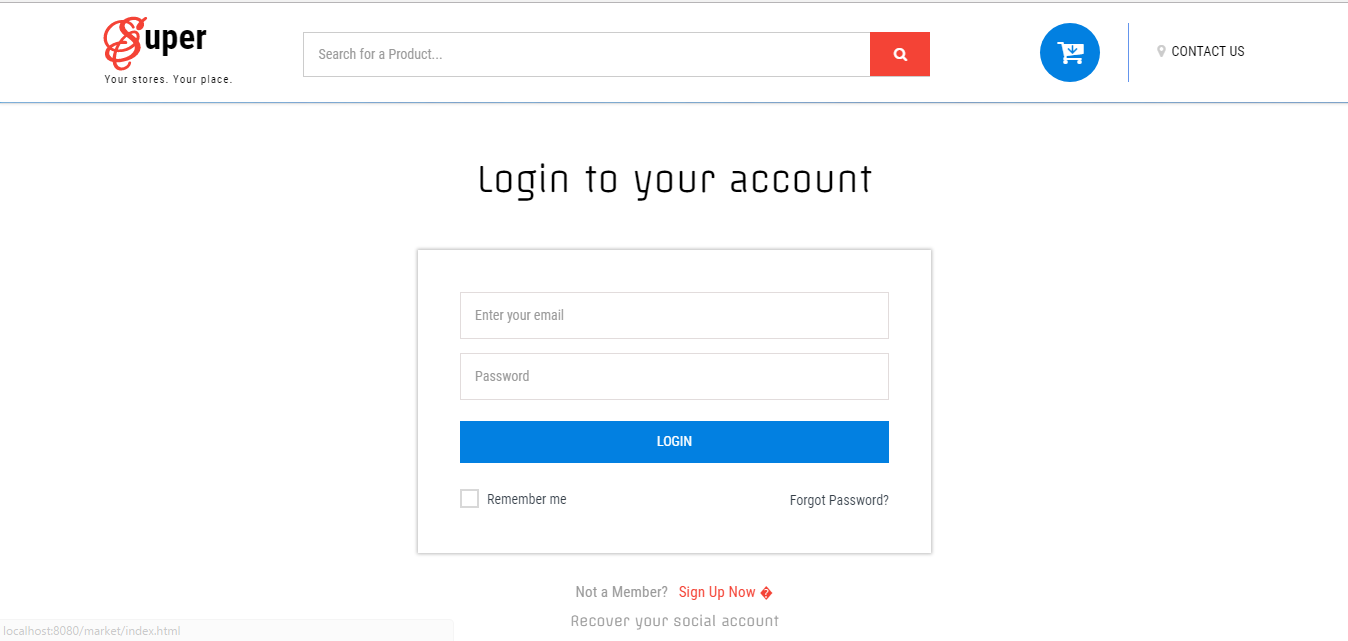
Pom.xml



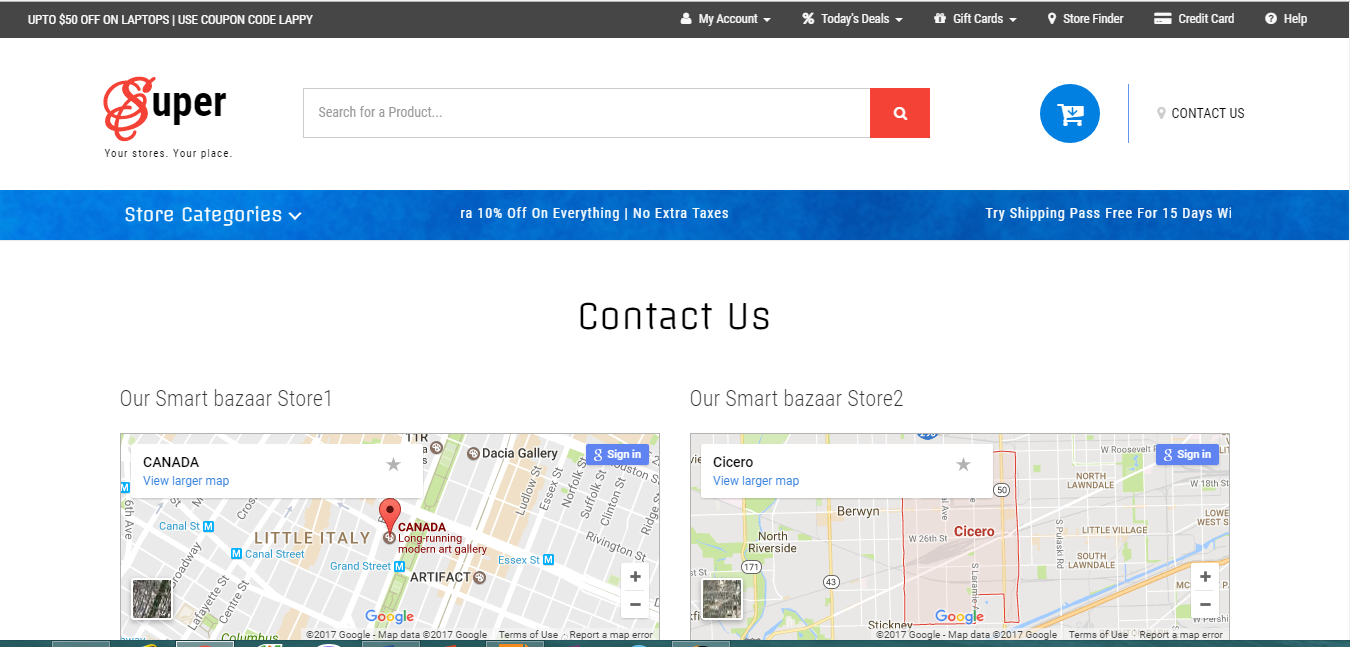
#### Giao diện và tính năng

MỘT SỐ GIAO DIỆN VÀ CHỨC NĂNG

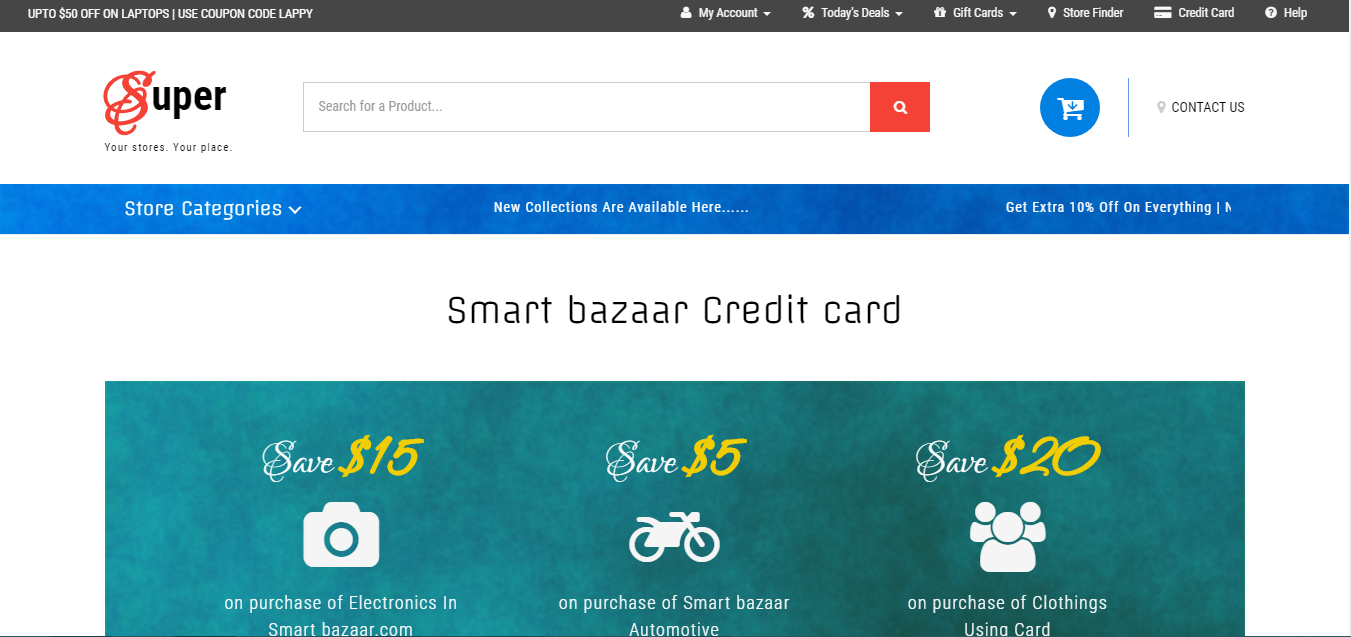


Trang chủ

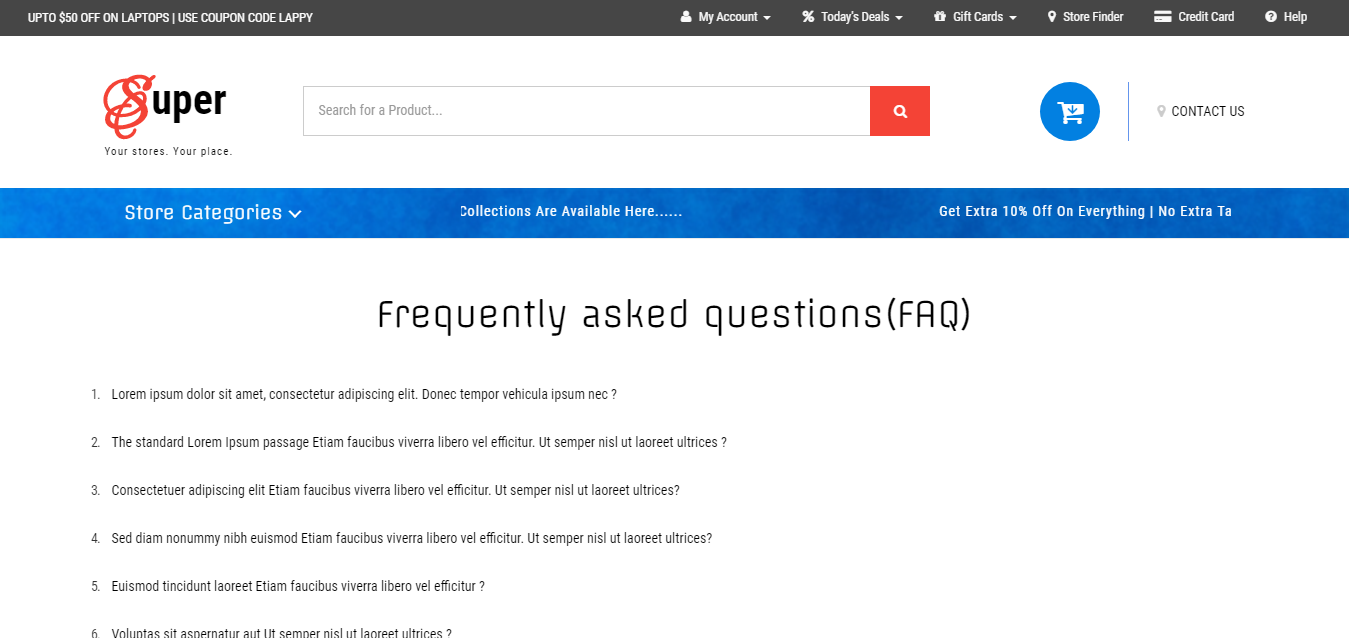
Login



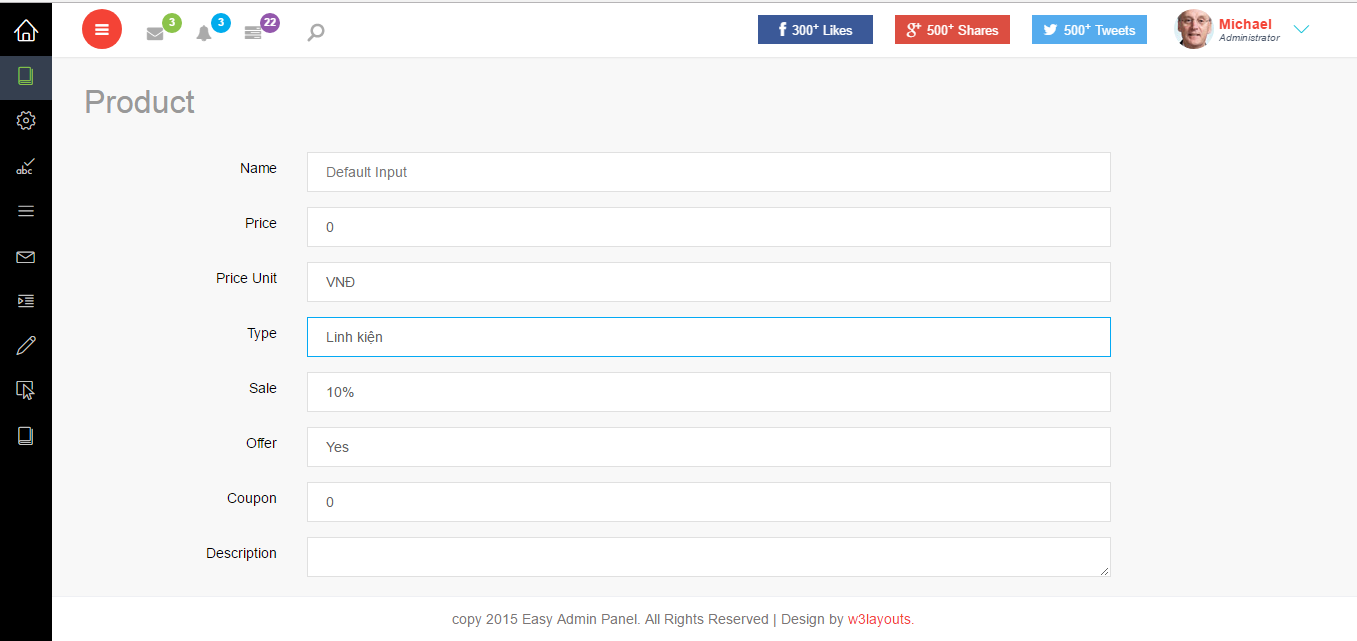
Contact



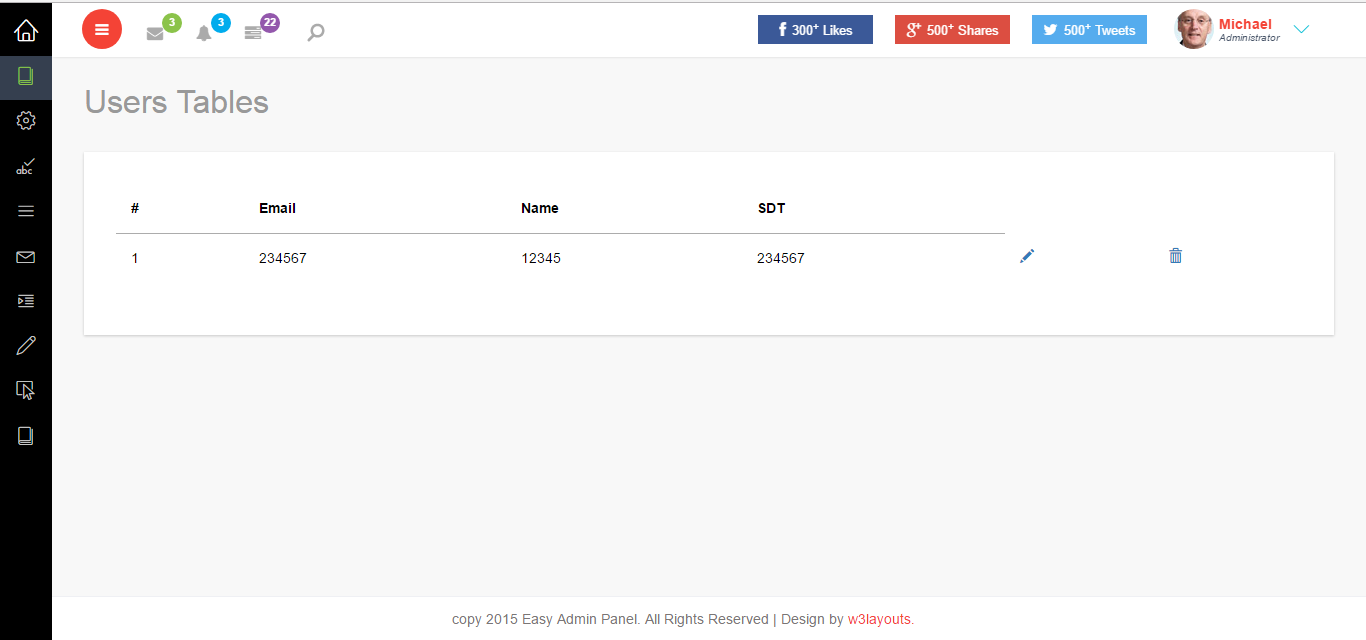
Card



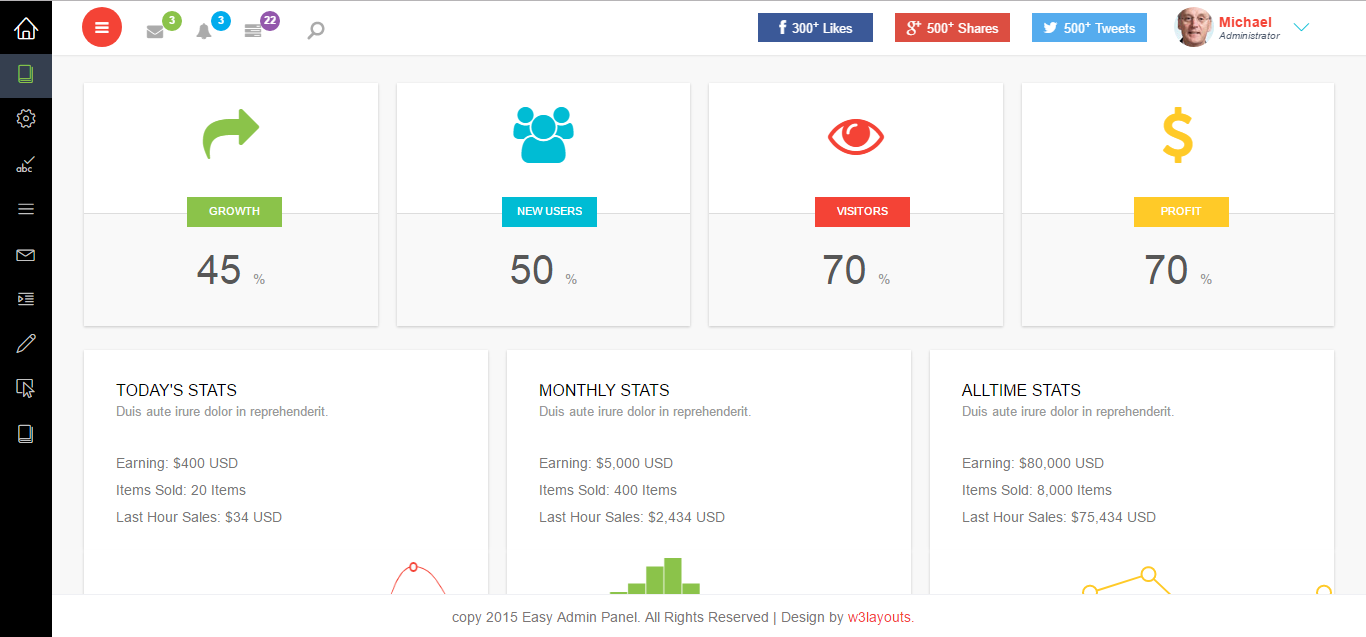
FAQ



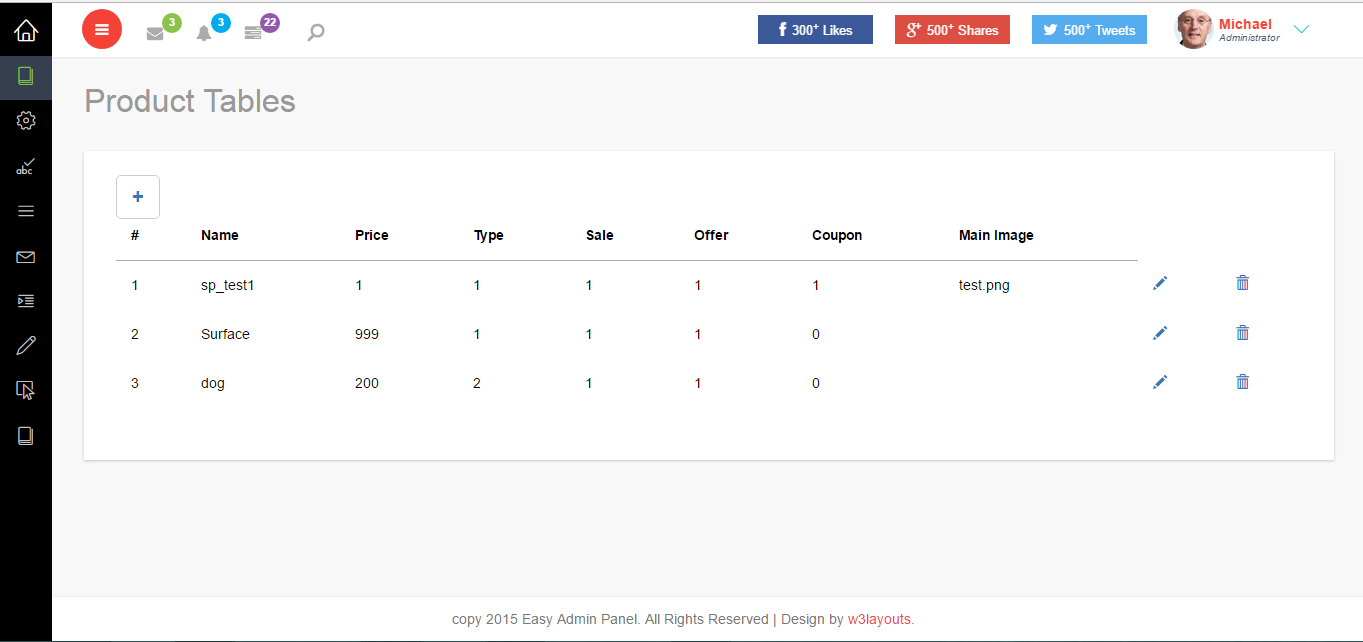
Create



User



Admin



# **Chương III Kết quả và hướng phát triển**

## Kết quả đạt được

Nghiên cứu và trình bày một cách có hệ thống các nội dung cơ bản về Spring framework và một số công nghệ, kỹ thuật khác trong việc xây dựng một ứng dụng Java trên nền web.

Về mặt lý thuyết:

* + Nắm được kiến trúc tổng quan của Spring Framework cũng như các nguyên lý cơ bản và cơ chế hoạt động của framework này.
  + Nắm được mô hình web MVC trong Spring framework và các cơ chế để bảo mật một ứng dụng web được hỗ trợ trong module Spring Security.

Về mặt ứng dụng:

* Thiết kế và cài đặt được ứng dụng web với mô hình web server và data server  
  tách biệt nhau, giao tiếp với nhau thông qua ActiveMQ.
* Xây dựng được ứng dụng web dạng SPA mà việc giao tiếp với server thông  
  qua RESTful service.
* Thiết kế giao diện người dùng với HTM5 và Bootstrap có hỗ trợ responsive.
* Xây dựng thành công ứng dụng quản lý bán hàng với đầy đủ các chức năng cơ bản.
* Xây dựng thành công một ứng dụng quản lý bán hàng trên nền web có áp dụng các công nghệ mới trong quá trình phát triển một ứng dụng Web hiện đại.

## Hạn chế

Các chức năng cơ bản của ứng dụng đã hoàn thành tốt, tuy nhiên do yêu cầu và phạm vi đề tài chỉ dừng lại ở mức demo công nghệ nên ứng dụng chưa hấp dẫn.

Hạn chế về vật chất, không co host trên internet co cài đặt đầy đủ các phần mềm cần thiết tạo kho khăn trong việc thực hiện demo chương trình trên internet.

Vấn đề triển khai những công nghệ nêu trên không khó về mặt kỹ thuật nhưng phải xây dựng một cơ sở hạ tầng đầy đủ và tốt để có thể thực thi ứng dụng.

## Hướng phát triển

Website nhóm phát triển chỉ dừng ở demo và chưa hoàn thiện một số tính năng còn thiếu. Nhưng các tính năng chính đã khá đầy đủ và hoàn thiện.

Do vậy qua thời gian này nhóm sẽ tiếp tục hoàn thiện các tính năng còn thiếu.