NỘI DUNG

[**I.** **Thành viên nhóm và nhiệm vụ** 3](#_Toc523508347)

[**II.** **Spring Boot** 3](#_Toc523508348)

[1. Spring Boot là gì và tại sao phải sử dụng Spring Boot 3](#_Toc523508349)

[2. Những cái hay của Spring Boot 4](#_Toc523508350)

[**III.** **Đề tài và lý do chọn** 4](#_Toc523508351)

[1. Đề tài 4](#_Toc523508352)

[2. Lý do 5](#_Toc523508353)

[**IV.** **Phân tích thiết kế cơ sở dữ liệu** 5](#_Toc523508354)

[1. Các thực thể 5](#_Toc523508355)

[2. Các mối quan hệ 5](#_Toc523508356)

[3. Thuộc tính thực thể 6](#_Toc523508357)

[4. Sơ đồ ERD 7](#_Toc523508358)

[**V.** **Quá trình xây dựng project** 7](#_Toc523508359)

[1. Cài đặt Spring tool suite cho Eclipse 7](#_Toc523508360)

[2. Tạo project với **Spring Boot** 12](#_Toc523508361)

[3. Cấu trúc của project 14](#_Toc523508362)

[4. Cấu trúc file **pom.xml** 14](#_Toc523508363)

[5. Cấu trúc file **application.properties** 16](#_Toc523508364)

[**VI.** **Các chức năng của trang web** 17](#_Toc523508365)

[1. Chức năng đăng nhập, đăng xuất và phân quyền 17](#_Toc523508366)

[2. Chức năng hiển thị dữ liệu 20](#_Toc523508367)

[3. Chức năng thêm 23](#_Toc523508368)

[a. Thêm user 23](#_Toc523508369)

[b. Thêm sản phẩm 24](#_Toc523508370)

[4. Chức năng sửa 25](#_Toc523508371)

[5. Chức năng xóa 26](#_Toc523508372)

[6. Chức năng tìm kiếm dữ liệu. 26](#_Toc523508373)

[**VII.** **Kết quả đạt được** 27](#_Toc523508374)

[**VIII.** **Thuận lợi và khó khăn** 27](#_Toc523508375)

[1. Thuận lợi 27](#_Toc523508376)

[2. Khó khăn 27](#_Toc523508377)

1. **Thành viên nhóm và nhiệm vụ**

* Nguyễn Văn Tuấn:
* Nhiện vụ: thiết kế back-end, thực hiện chức năng thêm, xóa, sửa và hiển thị, up hình ảnh và hiển thị hình ảnh.
* Dương Thành Vũ:
* Nhiệm vụ: thực hiện chức năng đăng nhập bằng gmail, phân quyền, đăng nhập bằng user có trong CSDL, load dữ liệu của user sau khi đăng nhập, mã hóa mật khẩu của user.

1. **Spring Boot**
2. Spring Boot là gì và tại sao phải sử dụng Spring Boot

* Springframework đã tồn tại hơn một thập kỷ nay, cũng giống như bao framework khác, nó giúp rất nhiều trong việc xây dựng các ứng dụng doanh nghiệp ( Enterprise application) bằng việc áp dụng các khái niệm như MVC, Inversion Of Control (IoC), AOP (aspect-oriented programming) với nhiều module mạnh mẽ cho security, dependency management,... Nhưng đã biết, trong ứng dụng Spring để sử dụng các tính năng của nó chúng ta phải config rất nhiều bằng cách viết những file cấu hình khá là phức tạp, nào là web.xml để config dispatcher servlet, nào là xxx-servlet.xml,... Mặc dù trong Spring 2.5 đã giới thiệu annotation để giảm thiểu số file ấy mà thay bằng config bằng java, chúng ta vẫn chẳng thể tránh những config phiền toái ấy.
* Ví dụ như việc config transaction, các filter hay quản lý các dependency chúng ta phải config trong những file xml hoặc trong code java. Việc config như thế đòi hỏi khá nhiều thời gian cũng công sức, thay vì phải ngồi config từng chút một như vậy thì để thời gian cho business logic có phải là hơn không? Hơn nữa, việc quản lý các dependency quả là một việc làm vô nghĩa, tại sao chúng ta phải đầu từ thời gian vào công việc này? Phải giải quyết xung đột giữa chúng hay những vấn đề về version?
* Hiểu được như thế, các nhà phát triển đã cho ra Spring Boot với mục tiêu chính là giảm thiếu (hoặc hoàn toàn) các config lùm xùm trong Spring, giúp những lập trình viên có thể tập trung vào business logic hơn từ đó nâng cao chất lượng của sản phẩm. Tất cả những gì bạn cần là viết mã logic cho application hoặc rất rất ít các config nguyên thủy của Spring.
* Nói chung, Spring boot là một project được xây dựng dựa trên những cái chung mà trong các project spring nào cũng có, từ đó giúp giảm bớt những config rắc rối hay phải quản lý các version để tập trung vào code logic hơn, nâng cao chất lượng sản phẩm.

1. Những cái hay của Spring Boot

* Spring Boot đã mang lại rất nhiều những tiện ích đáng kinh ngạc trong việc phát triển các ứng dụng Spring, nhưng có 4 tiện ích chính mà nó đã thực hiện là:
* Tự động config (automatic configuration): Spring boot có thể tự động config những thứ cơ bản mà hầu hết các ứng dụng Spring đều phải có.
* Khởi xướng phụ thuộc (starter dependencies): Bạn chỉ việc khai báo những loại tính năng bạn cần, nó sẽ đảm bảo các phụ thuộc cần thiết sẽ được thêm vào project.
* Giao diện command-line (The command-line interface): Đây là sự lựa chọn, Spring boot cho phép bạn chỉ cần viết code logic không cần xây dựng project như truyền thống.
* The Actuator (tạm dịch là truyền động): Nó cho phép bạn những gì đã xảy ra bên trong một ứng dụng đang chạy.

1. **Đề tài và lý do chọn**
2. Đề tài

* Đề tài của nhóm để tham gia bảo vệ của môn là một trang web bán điện thoại cơ bản, sử dụng Framework là spring boot.
* Trang web có các chức năng cần thiết và đơn giản như: thêm, xóa, sửa, load dữ liệu, đăng nhập, phân quyền, sử dụng cookie để giảm tải cho server và có thể đăng nhập bằng gmail.

1. Lý do

* Với xu thế phát triễn của thời đại 4.0 như hiện nay thì tất cả mọi người đều làm việc một cách tấp nập và thứ mà không thể thiếu trong cuộc sống đó một chiếc điện thoại.
* Nhưng đâu phải ai cũng có thời gian rảnh rỗi mà đi quẹo lựa những chiếc điện thoại thay vào đó họ sử dụng thời gian rảnh để nghĩ ngơi lấy lại sức cho công việc tiếp theo.
* Vì thế các trang web mua sắm điện thoại đã ra đời nhằm phục vụ cho việc mua sắm mà không cần phải tốn quá nhiều thời gian, chỉ có kết nối internet và có thể truy cập vào trang web, tha hồ mà quẹo lựa mua sắm. Sau khi mua xong không cần đi đến nơi bán để nhận hàng mà đã có dịch vụ ship đến tận nơi.

1. **Phân tích thiết kế cơ sở dữ liệu**
2. Các thực thể

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **USER** |  | **PRODUCT** |  | **INVOICE** |  | **INVOICEDETAIL** |

1. Các mối quan hệ

* Một bản invoice thì có thể có nhiều invoicedetail, ngược lại nhiều bản invoiceadetail thì có một bản invoice.

|  |  |  |
| --- | --- | --- |
| **INVOICE** | **1 N** | **INVOICEDETAIL** |
|  |

* Một bản product thì có thể có nhiều invoicedetail, ngược lại nhiều bản invoicedetail thì có một bản product.

|  |  |  |
| --- | --- | --- |
| **PRODUCT** | **1 N** | **INVOICEDETAIL** |
|  |

1. Thuộc tính thực thể

**INVOICE**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **TÊN THUỘC TÍNH** | **KIỂU DỮ LIỆU** | **NOT NULL**  **NULL** | **MẶC ĐỊNH** | **CHI CHÚ** | **KHOÁ** |
| 1 | idinvoice | INT(11) | NOT NULL | NOT NULL |  | PK |
| 2 | customername | VARCHAR(50) | NULL | NULL |  |  |
| 3 | numberphone | VARCHAR(20) | NULL | NULL |  |  |
| 4 | address | VARCHAR(250) | NULL | NULL |  |  |
| 5 | status | VARCHAR(50) | NOT NULL | NOT NULL |  |  |

**PRODUCT**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **TÊN THUỘC TÍNH** | **KIỂU DỮ LIỆU** | **NOT NULL**  **NULL** | **MẶC ĐỊNH** | **CHI CHÚ** | **KHOÁ** |
| 1 | idproduct | INT(11) | NOT NULL | NOT NULL |  | PK |
| 2 | productname | VARCHAR(50) | NULL | NULL |  |  |
| 3 | quantity | VARCHAR(10) | NULL | NULL |  |  |
| 4 | price | VARCHAR(10) | NULL | NULL |  |  |
| 5 | image | VARCHAR(250) | NULL | NULL |  |  |

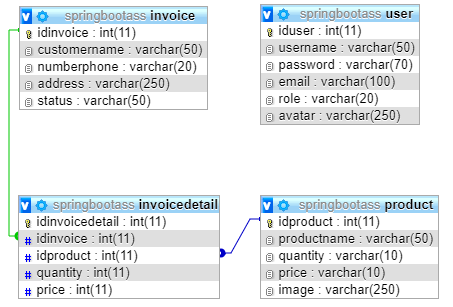
**INVOICEDETAIL**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **TÊN THUỘC TÍNH** | **KIỂU DỮ LIỆU** | **NOT NULL**  **NULL** | **MẶC ĐỊNH** | **CHI CHÚ** | **KHOÁ** |
| 1 | idinvoicedetail | INT(11) | NOT NULL | NOT NULL |  | PK |
| 2 | idinvoice | INT(11) | NOT NULL | NOT NULL |  | FK |
| 3 | idproduct | INT(11) | NOT NULL | NOT NULL |  | FK |
| 4 | quantity | INT(11) | NOT NULL | NOT NULL |  |  |
| 5 | price | INT(11) | NOT NULL | NOT NULL |  |  |

**USER**

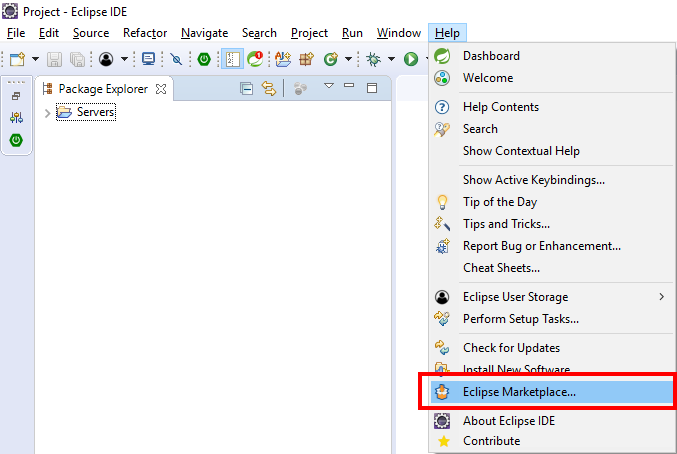
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STT** | **TÊN THUỘC TÍNH** | **KIỂU DỮ LIỆU** | **NOT NULL**  **NULL** | **MẶC ĐỊNH** | **CHI CHÚ** | **KHOÁ** |
| 1 | iduser | INT(11) | NOT NULL | NOT NULL |  | PK |
| 2 | username | VARCHAR(50) | NULL | NULL |  |  |
| 3 | password | VARCHAR(70) | NULL | NULL |  |  |
| 4 | email | VARCHAR(100) | NULL | NULL |  |  |
| 5 | role | VARCHAR(20) | NULL | NULL |  |  |
| 6 | avatar | VARCHAR(250) | NULL | NULL |  |  |

1. Sơ đồ ERD

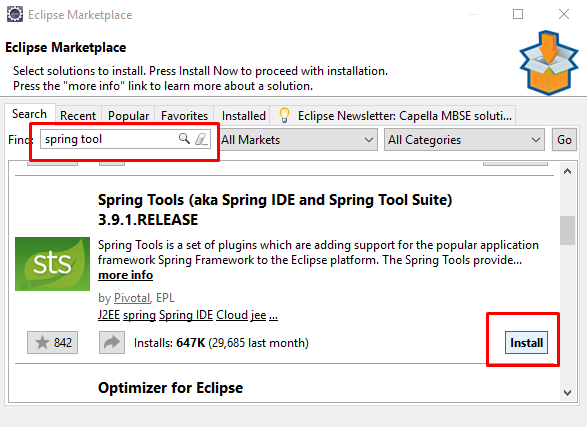


1. **Quá trình xây dựng project**
2. Cài đặt Spring tool suite cho Eclipse

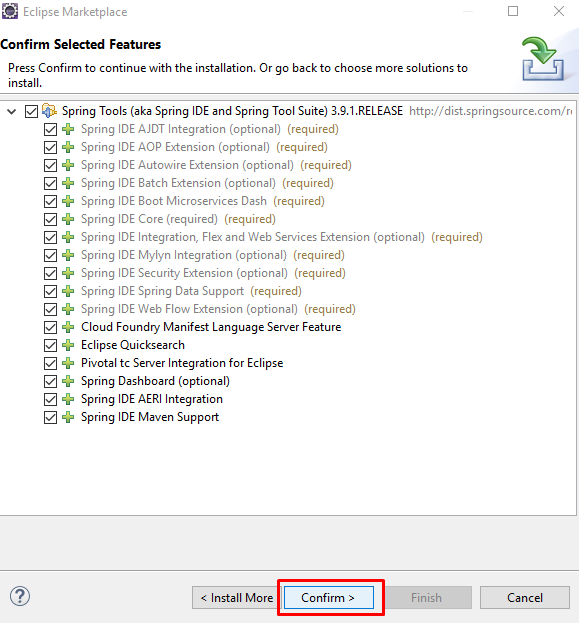
* Spring tool suite (STS) là một extension mở rộng cho eclipse, chuyên dùng để hỗ trợ phát triển các ứng dụng bằng Spring Framework.
* Truy cập Eclipse Marketplace bằng cách: **Help > Eclip Marketplace…**



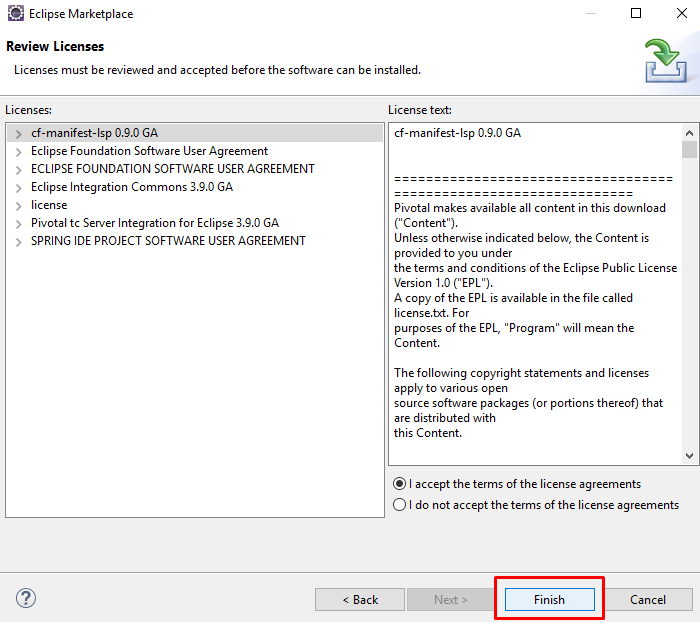
* Gõ từ khóa **sts** hoặc **spring tool** và click chọn **Install**.



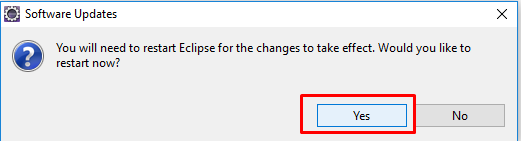
* Click chọn tất cả > **Confirm**.



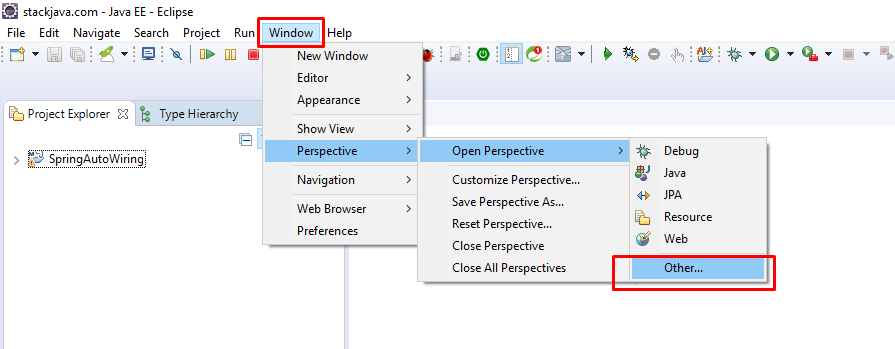
* Click accept > **Finish**.

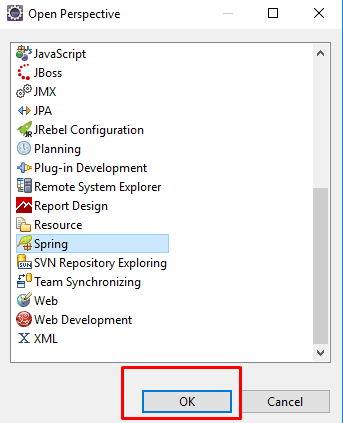


* Sau khi cài đặt xong elipse sẽ yêu cầu restart lại để áp dụng spring tool vào eclipse, click chọn **Yes** để restart lại.

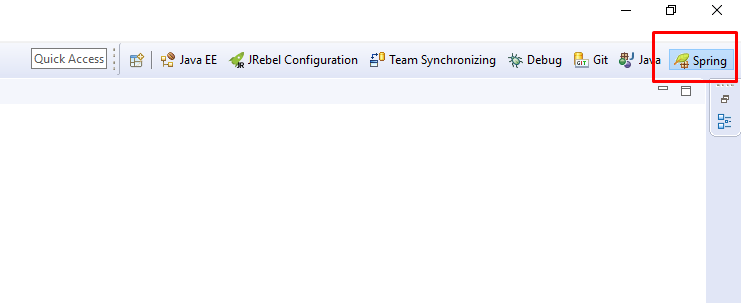


* Hiển thị tính năng spring tool bằng cách: **Window > Perspect > Open Perspect > Other > Spring**.



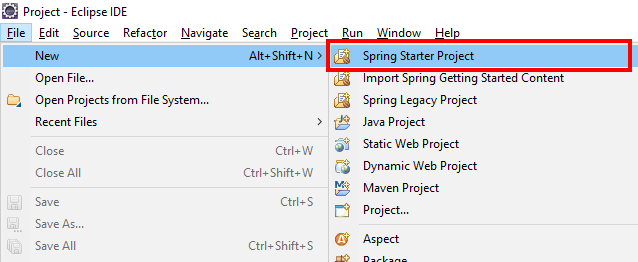


* Sau khi chọn xong thì biểu tượng của spring tool sẽ hiển thị ở góc trên bên phải của eclipse.

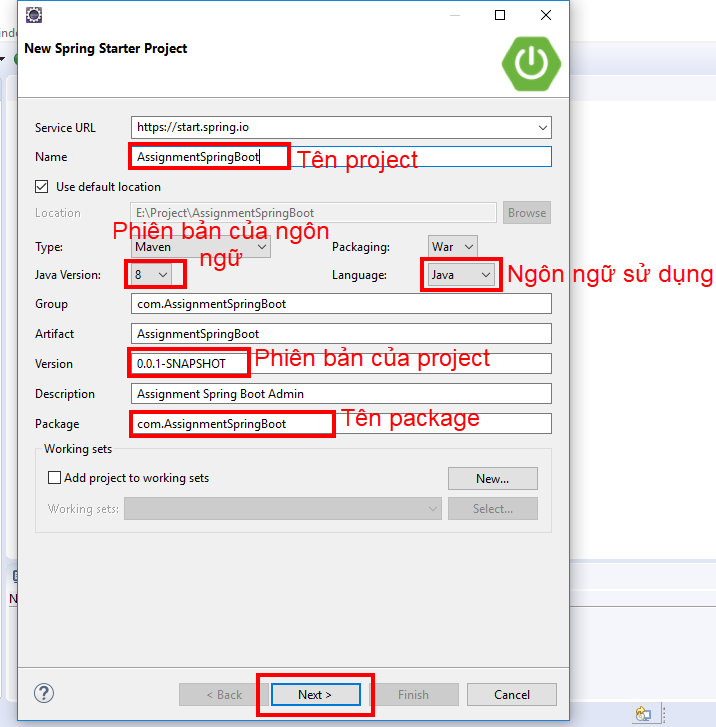


1. Tạo project với **Spring Boot**

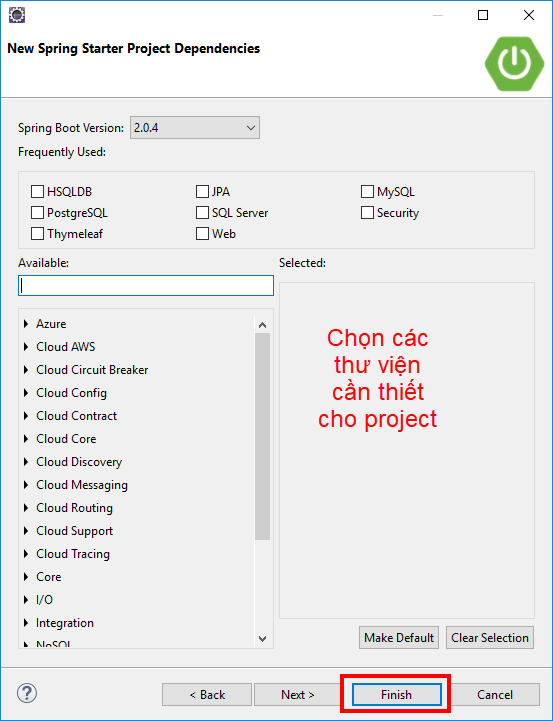
* Bằng cách: **File > New > Spring Started Project**.



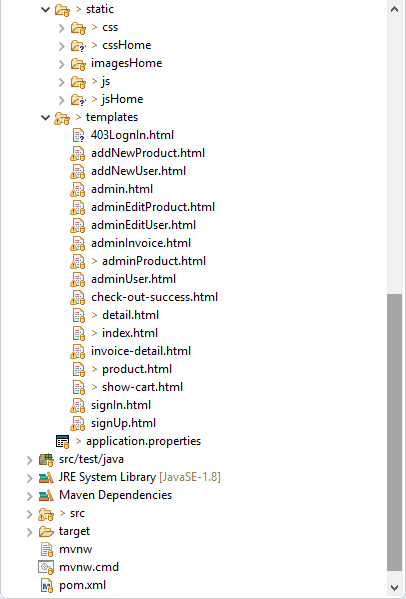
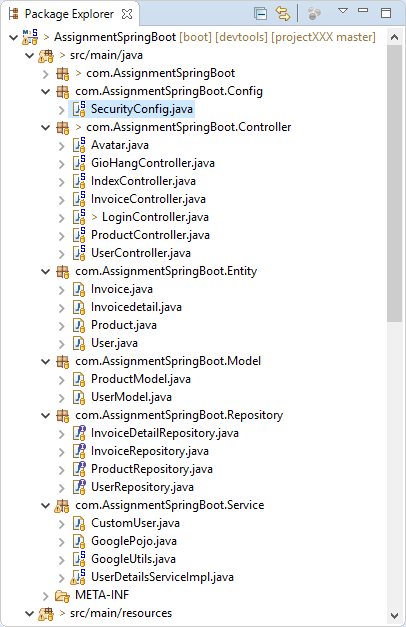
* Điền các thông tin cần thiết cho một project > **Next**.



* Chọn các thư viện cần thiết cho project > **Finish**.



1. Cấu trúc của project



1. Cấu trúc file **pom.xml**

<?xml version=*"1.0"* encoding=*"UTF-8"*?>

<project xmlns=*"http://maven.apache.org/POM/4.0.0"*

xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*

xsi:schemaLocation=*"http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd"*>

<modelVersion>4.0.0</modelVersion>

<groupId>com.AssignmentSpringBoot</groupId>

<artifactId>AssignmentSpringBoot</artifactId>

<version>0.0.1-SNAPSHOT</version>

<packaging>war</packaging>

<name>AssignmentSpringBoot</name>

<description>Assignment Spring Boot Admin</description>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>2.0.4.RELEASE</version>

<relativePath /> <!-- lookup parent from repository -->

</parent>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>

<java.version>1.8</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-data-jpa</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-thymeleaf</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-security</artifactId>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<scope>runtime</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-tomcat</artifactId>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

</dependency>

<!-- thư viện thêm sau -->

<dependency>

<groupId>commons-io</groupId>

<artifactId>commons-io</artifactId>

<version>2.4</version>

</dependency>

<dependency>

<groupId>org.apache.httpcomponents</groupId>

<artifactId>fluent-hc</artifactId>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

</plugin>

</plugins>

</build>

</project>

1. Cấu trúc file **application.properties**

##### DATABASE #####

spring.datasource.url=jdbc:mysql://localhost:3306/springbootass?useSSL=false&characterEncoding=UTF-8

spring.datasource.username=root

spring.datasource.password=123123

spring.jpa.show-sql=true

spring.jpa.hibernate.ddl-auto=update

spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5Dialect

##### SIZE IMAGE #####

spring.servlet.multipart.max-file-size=30MB

spring.servlet.multipart.max-request-size=30MB

##### LOCATION IMAGE #####

vitriluufile=E:/Project/projectXXX/AssignmentSpringBoot/src/main/resources/static/imagesHome

##### LOGN IN EMAIL #####

google.app.id=599638690061-eqb4pglotqp9b7uqad097jnm5dj93n33.apps.googleusercontent.com

google.app.secret=dfUq12sAOjvSDPwvIKd5Xm48

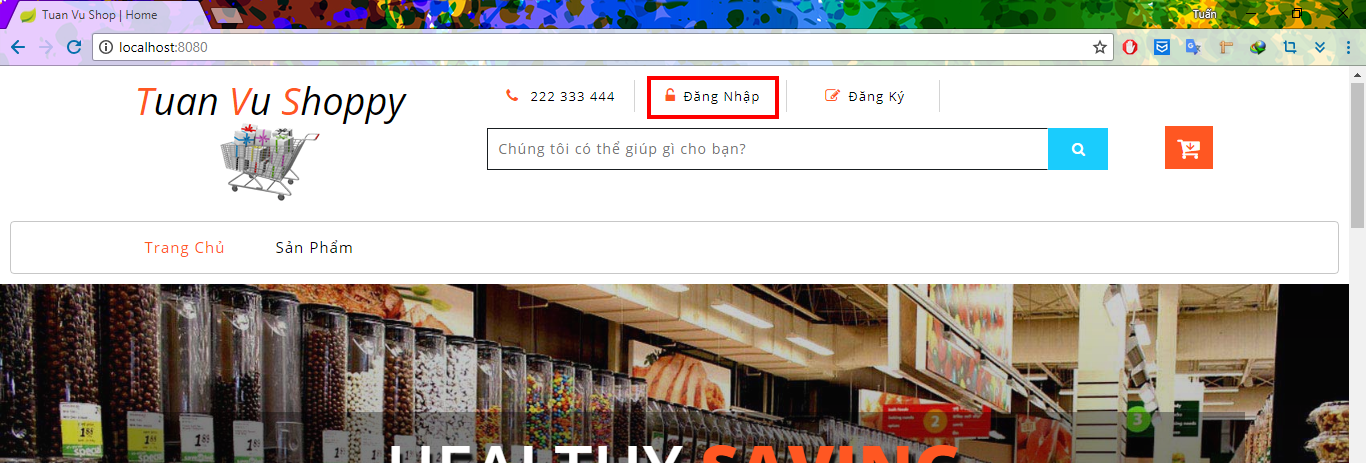
google.redirect.uri=http://localhost:8080/login-google

google.link.get.token=https://accounts.google.com/o/oauth2/token

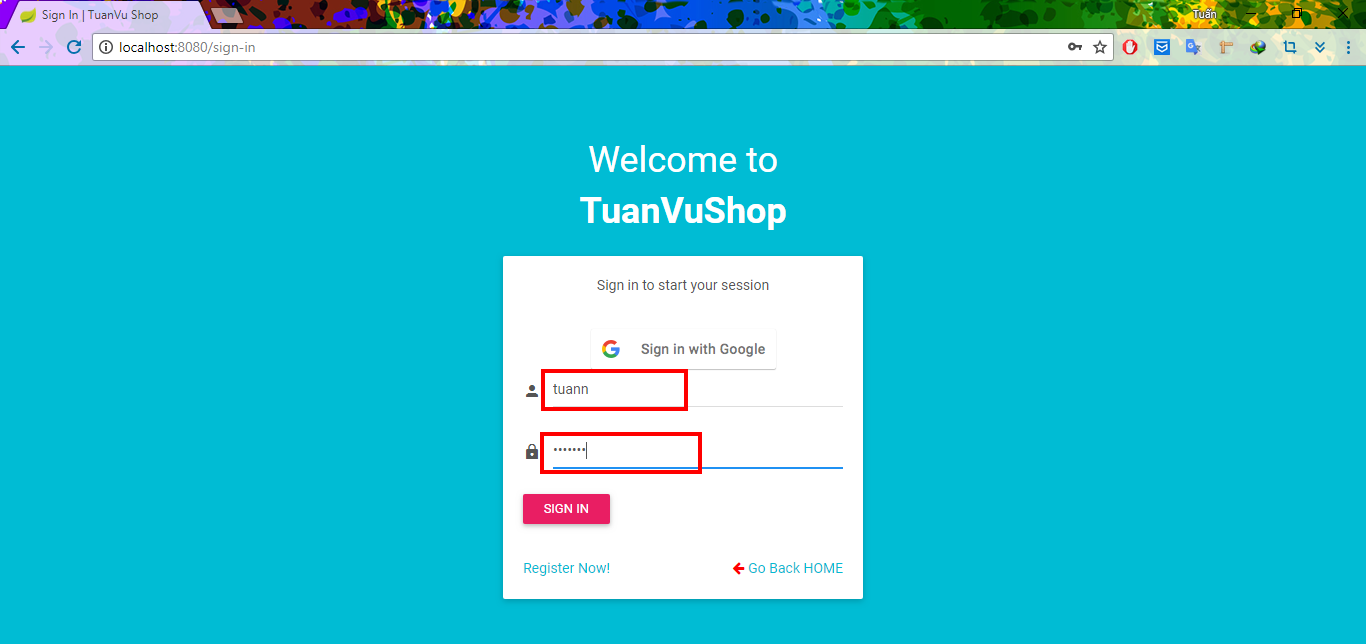
google.link.get.user\_info=https://www.googleapis.com/oauth2/v1/userinfo?access\_token=

1. **Các chức năng của trang web**
2. Chức năng đăng nhập, đăng xuất và phân quyền

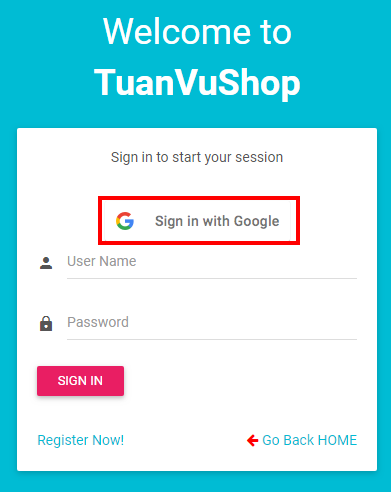
* Sau khi truy cập trang web click **Đăng Nhập**.



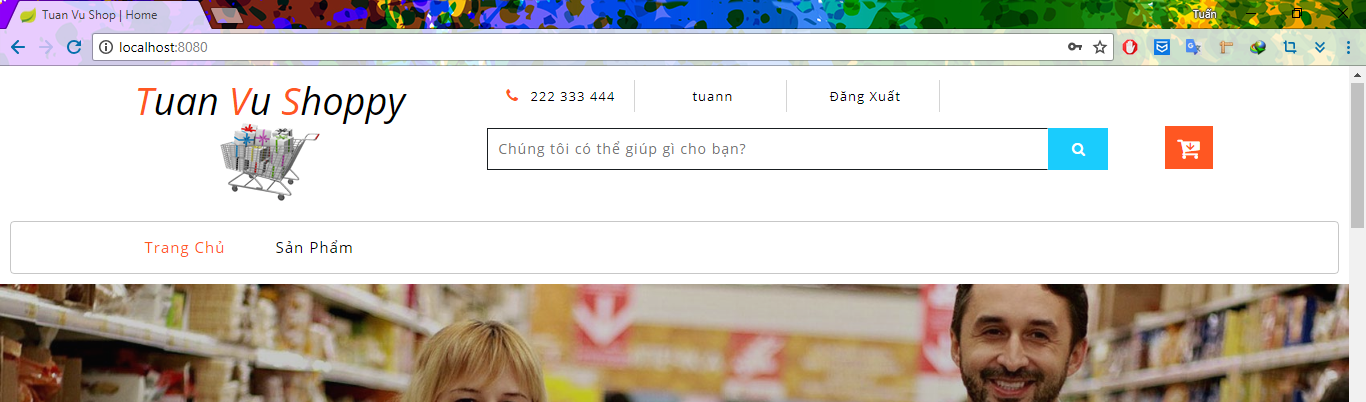
* Sử dụng tài khoản trong CSDL để đăng nhập.



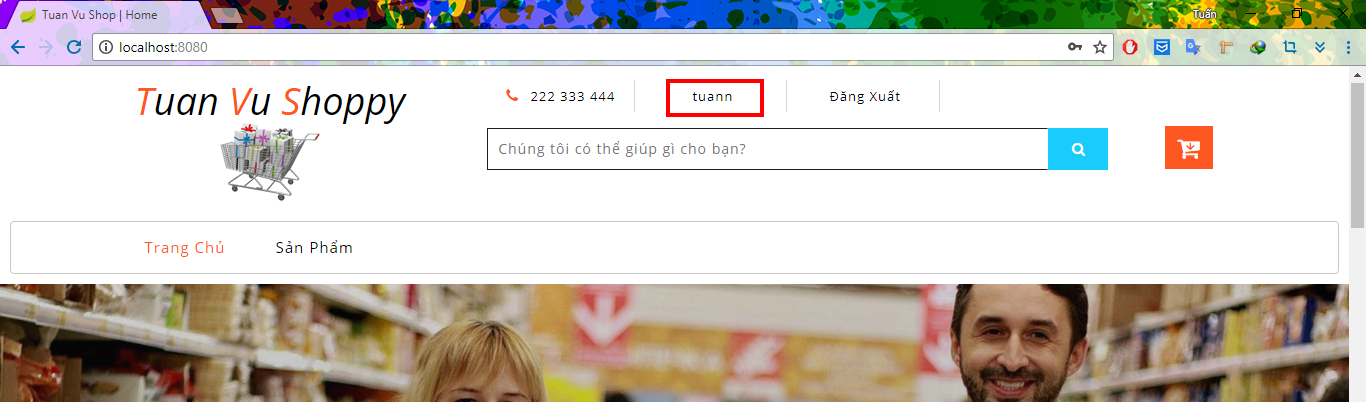
* Hoặc sử dụng **gmail** để đăng nhập (gmail sau khi đăng nhập mặc định quyền sẽ là quyền user).

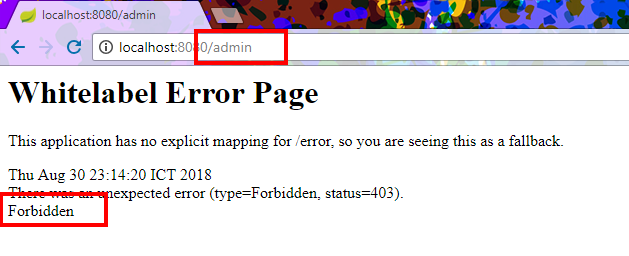


* Đây là sau khi đã đăng nhập.

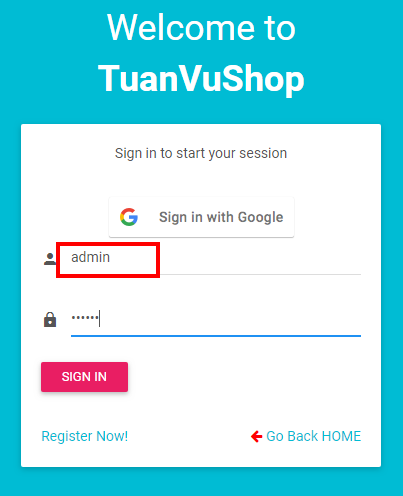


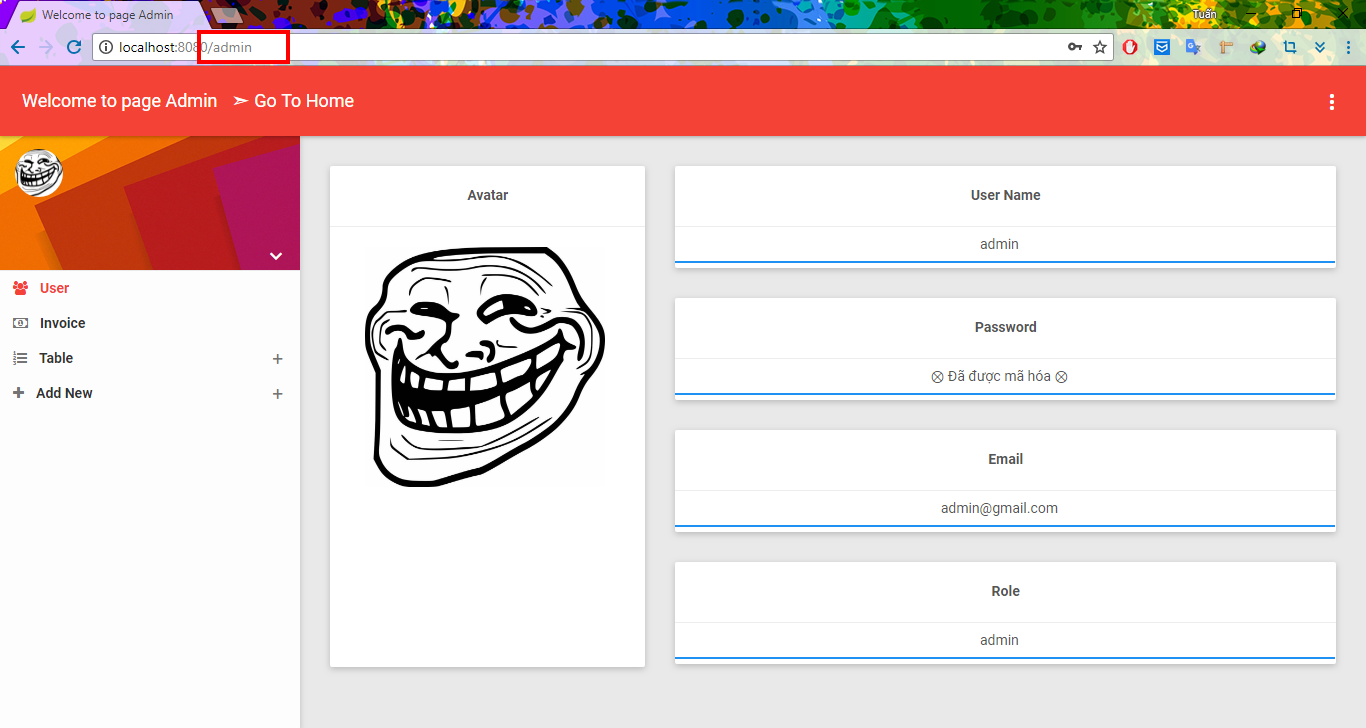
* Click vào tên người dùng để chuyển đến trang dùng của **admin**.
* Và tài khoản có quyền là **user** thì không thể truy cập vào trang dùng của **admin** được.



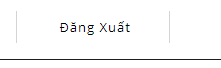
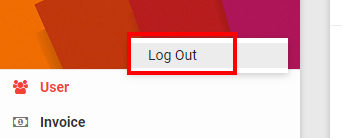


* Sử dụng tài khoản có quyền là admin thì sau khi đăng nhập thay vì sẽ hiển thị trang chủ thì trang sẽ chuyển hướng đến trang dùng của admin.



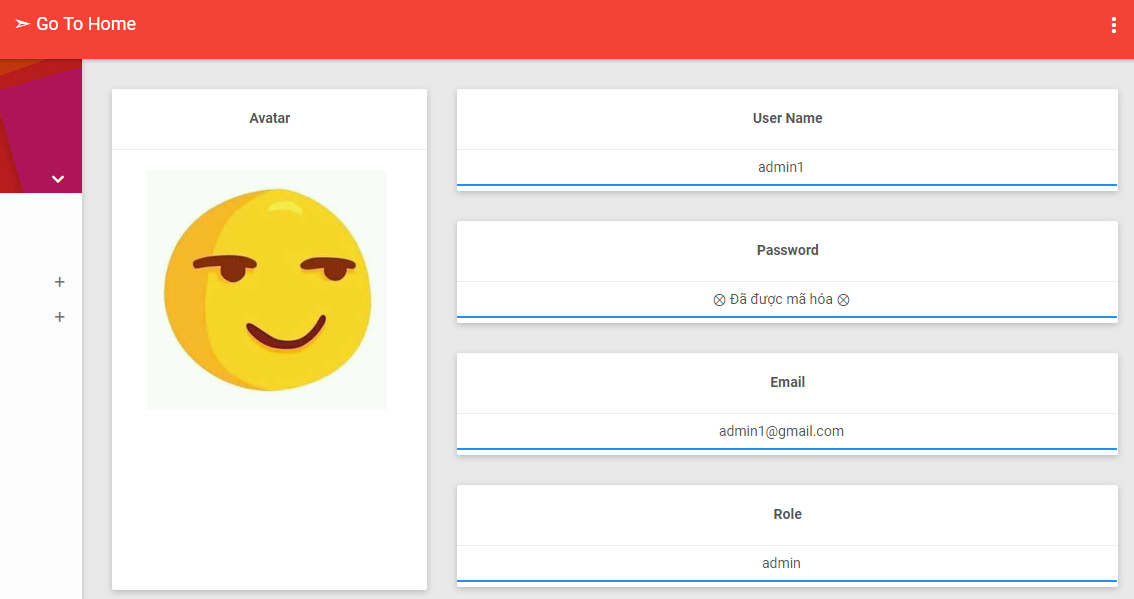


* Click **Log Out** để đăng xuất ở trang admin và click đăng xuất ở trang cho người đùng có quyền là user.

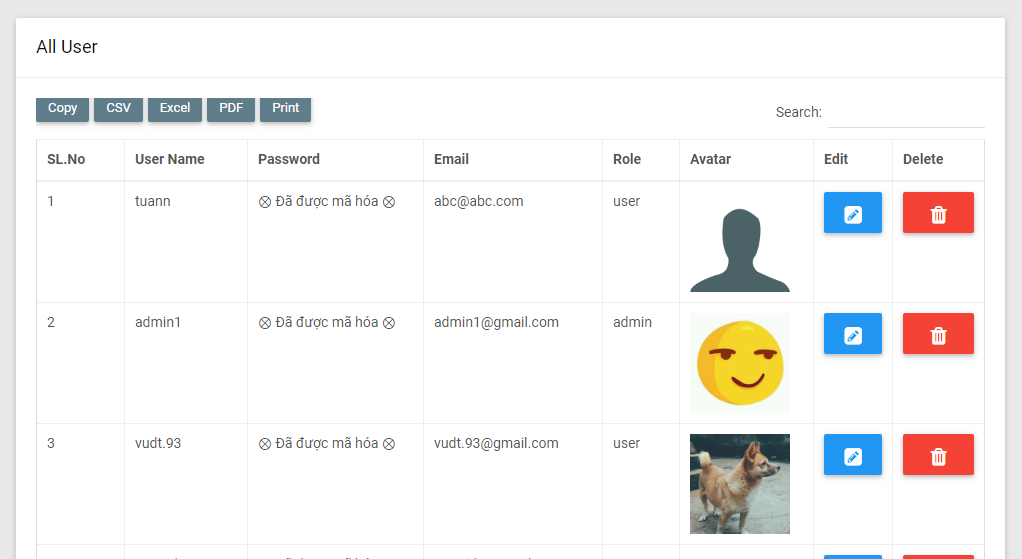


1. Chức năng hiển thị dữ liệu

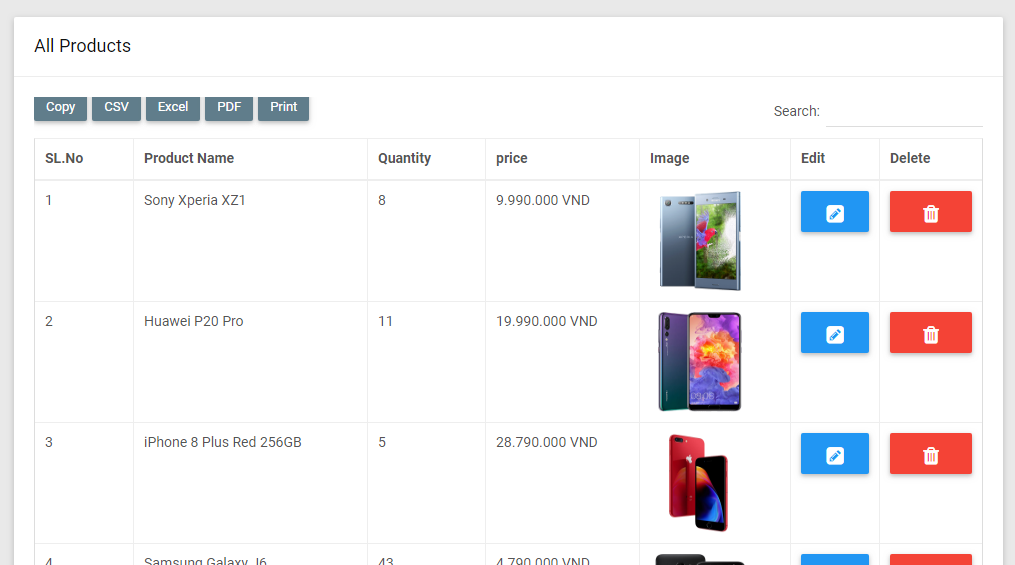
* Gồm có hiển thị dữ liệu thông tin của user đã đăng nhập, dữ liệu của tất cả user, dữ liệu của sản phẩm, dữ liệu của hóa đơn đặt hàng và chi tiết hóa đơn.
* Hiển thị thông tin của user sau khi đăng nhập gồm hình ảnh, username, password, role, email.



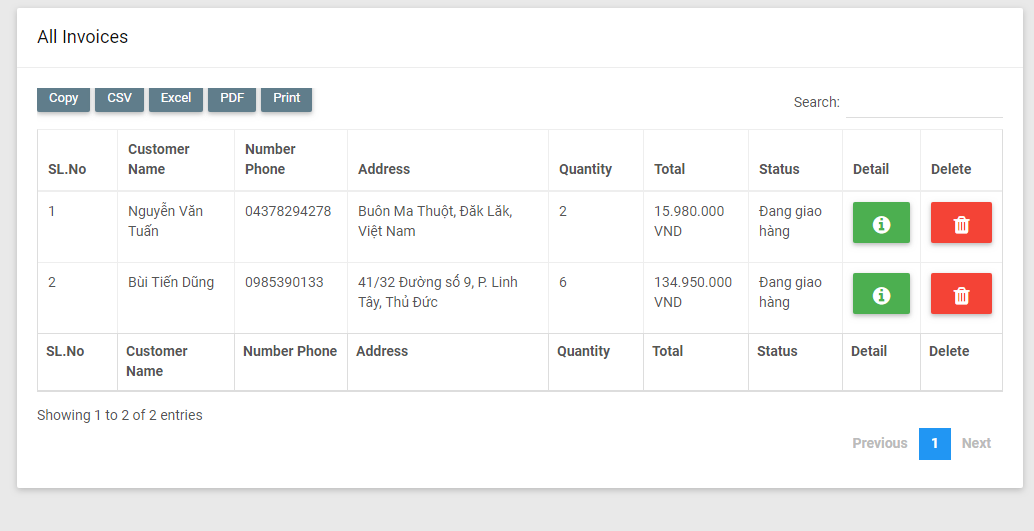
* Hiển thị dữ liệu tất cả user gồm username, password, email, role, hình ảnh.



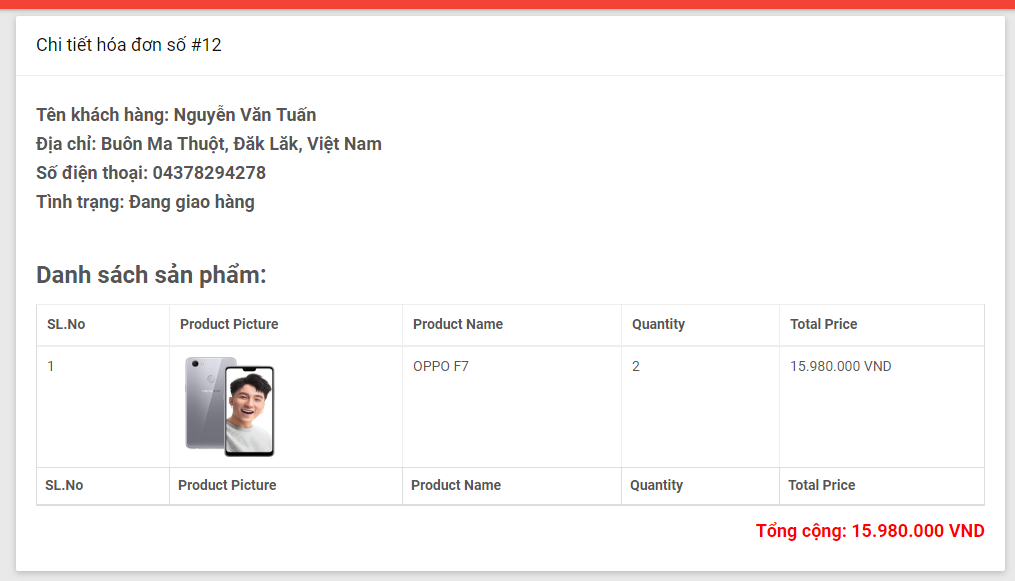
* Hiển thị dư liệu của sản phẩm gồm tên sản phẩm, số lượng, giá, ảnh.



* Hiển thị dữ liệu hóa đơn gồm tên người mua, số điện thoại, địa chỉ, số lượng sản phẩm, tổng tiền, tình trạng đơn hàng, chi tiết hóa đơn.

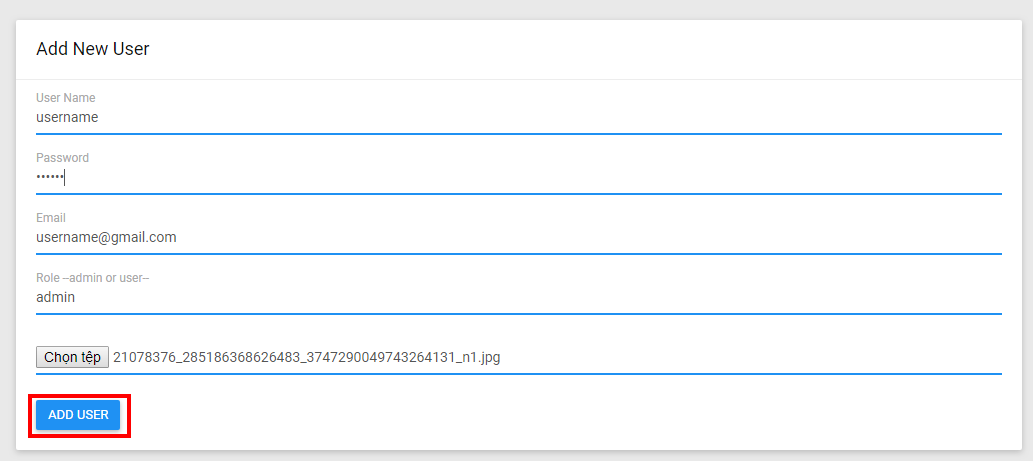


* Hiển thị dữ liệu của chi tiết hóa đơn gồm các thông tin của hóa đơn và chi tiết các sản phẩm.

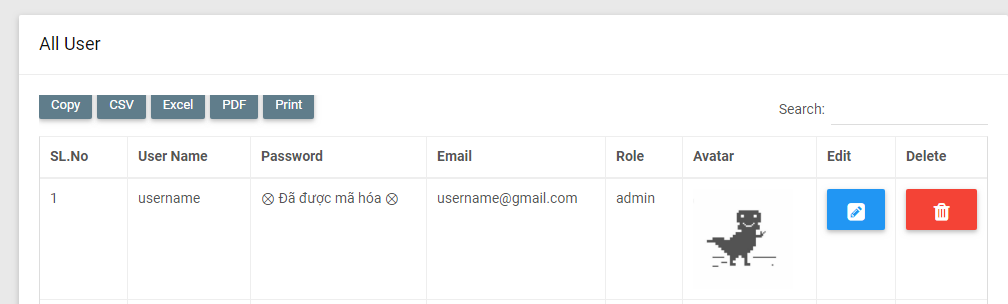


1. Chức năng thêm
2. Thêm user

* Bắt buộc phải nhập đầy đủ các thông tin của một user.

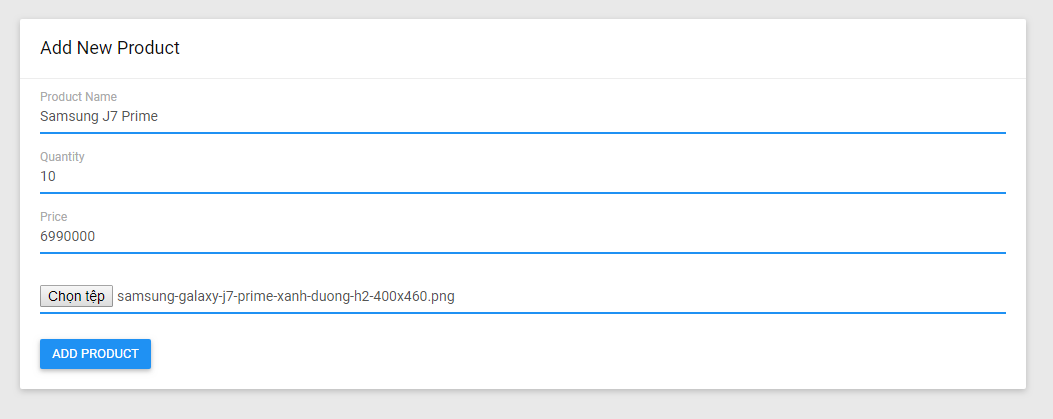


* Như vậy là đã thêm mới thành công một user.

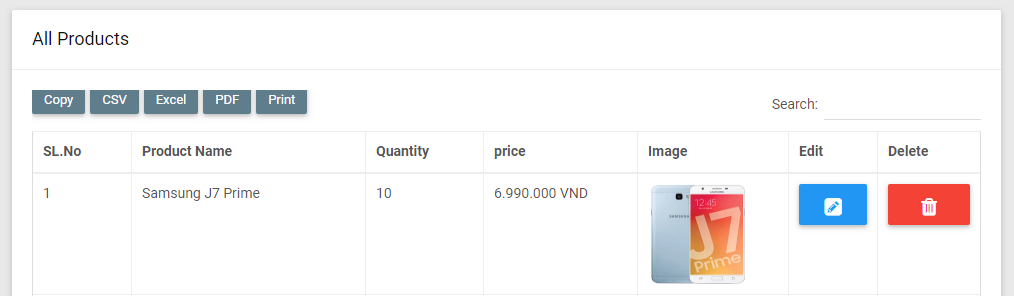


1. Thêm sản phẩm

* Nhập đầy đủ các thông tin cho một sản phẩm.

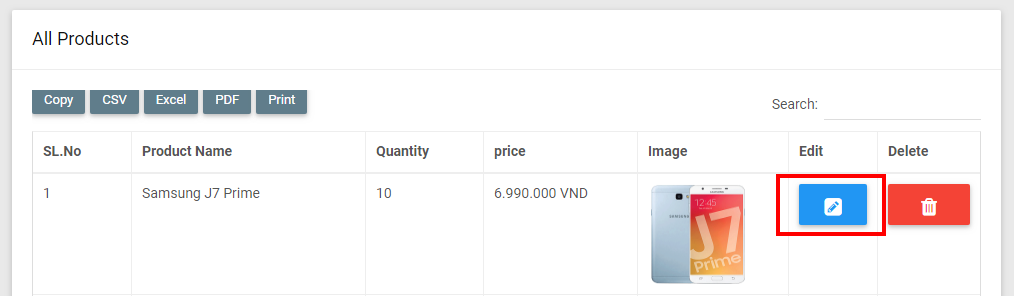


* Như vậy là đã thêm mới một sản phẩm thành công.

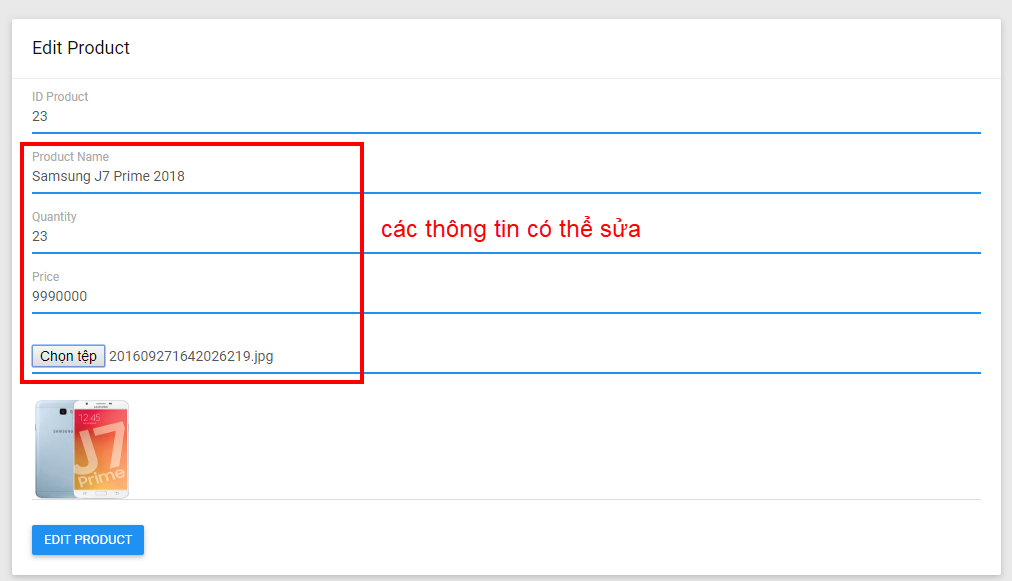


1. Chức năng sửa

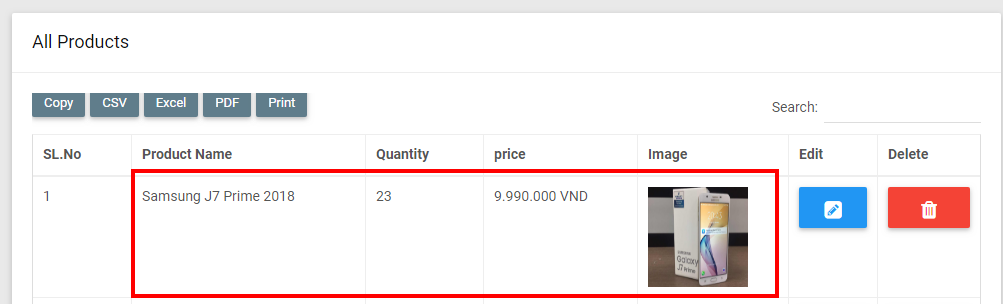
* Click **Edit** để thực hiện sửa.



* Điền các thông tin cần sửa.

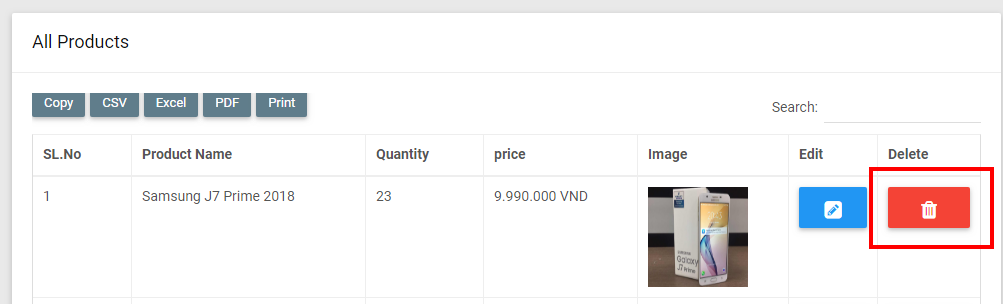


* Như vậy là đã sửa thành công.

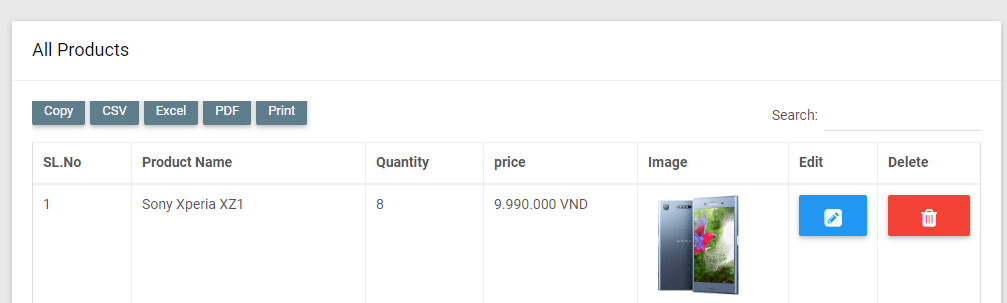


1. Chức năng xóa

* Click **Delete** để xóa sản phẩm.

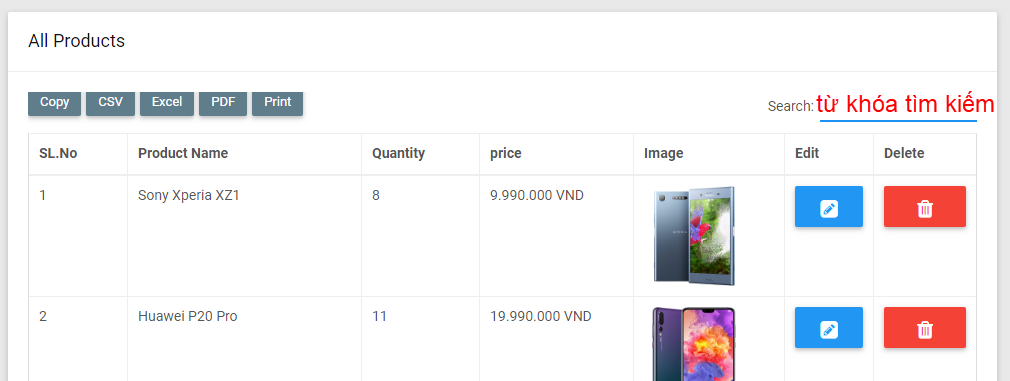


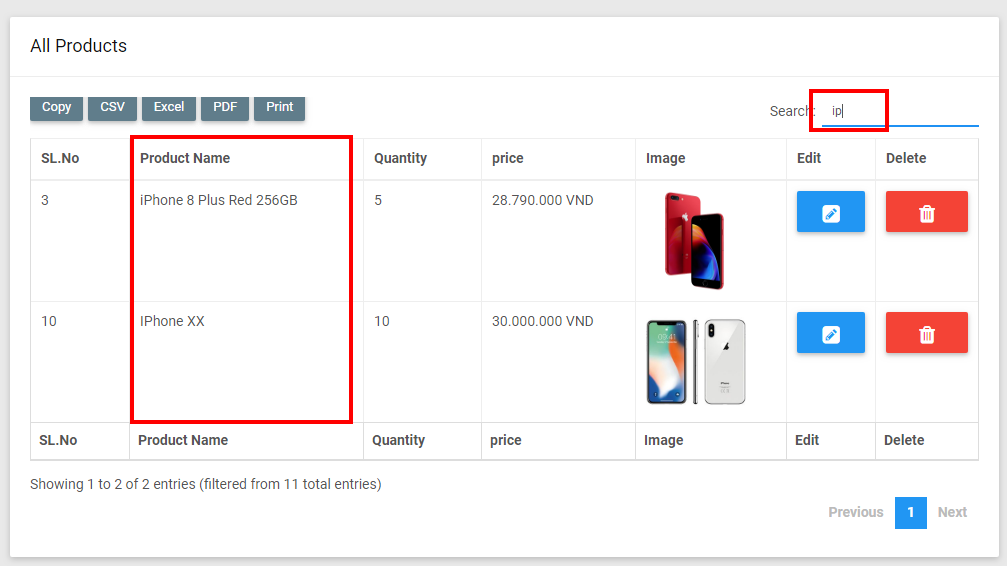
* Như vậy là đã xóa thành công.



1. Chức năng tìm kiếm dữ liệu.

* Nhằm hỗ trợ khi dữ liệu quá nhiều.
* Nhập từ khóa tìm kiếm.





1. **Kết quả đạt được**

* Hiểu thêm được một spring framework, biết cách sử dụng spring boot và nhưng thứ đi kèm với spring boot.
* Nâng cao được quá trình tự học, quang trọng nhất là quá trình làm việc theo nhóm tuy là nhóm chỉ có 2 thành viên.
* Biết cách phân chia công việc của nhau, đặt biệt là không đùn đẩy công việc cho nhau.
* Hiểu thêm về java hơn và có kỹ năng lập trình với một framework mới.

1. **Thuận lợi và khó khăn**
2. Thuận lợi

* Có sự giúp đỡ nhiệt tình từ các bạn trong lớp, đặt biệt là từ thầy.
* Có nguồn tham khảo rộng lớn và phong phú từ internet.
* Các thành viên trong nhóm hòa đồng và công việc được phân chia hợp lý dẫn đến công việc không bị động lại.

1. Khó khăn

* Vì đây là môn tự học nên vẫn còn nhiều điều mới dẫn đến việc hiểu rõ hết rất khó.
* Spring boot cũng là một framework nhưng cách thức hoạt động lại không giống các framework khác nên còn bỡ ngỡ.
* Vì mới tìm hiểu về spring boot nên còn nhiều thiếu sót trong quá trình tạo nên project với spring boot.
* Và các thành viên trong nhóm thì ở cách xa nhau nên quá trình trao đổi cũng khó khăn và tốn thời gian.

\*\*\*\*\*\*\*\*\*\*\*\* CẢM ƠN ĐÃ ĐỌC HẾT \*\*\*\*\*\*\*\*\*\*\*\*