|  |  |  |
| --- | --- | --- |
|  |  |  |

|  |
| --- |
| FPT POLYTECHNIC |
| MINI PROJECT REQUIREMENT |
| TRẦN MINH HẬU – PC01058 (ỨNG DỤNG PHẦN MỀM) |

|  |
| --- |
| 7/9/2021 |



MỤC LỤC

[1. Giới Thiệu Dự Án](#_Toc81854216)

[2. Phân Tích Yêu Cầu Khách Hàng](#_Toc81854217)

[2.1 Sơ Đồ Use case](#_Toc81854218)

[2.2 Sơ đồ triển khai và yêu cầu hệ thống](#_Toc81854219)

[2.2.1 Mô hình triển khai](#_Toc81854220)

[3. Thiết kế ứng dụng](#_Toc81854222)

[3.1 mô hình công nghệ ứng dụng](#_Toc81854223)

[3.2 Thực Thể](#_Toc81854225)

[4. Thực hiện dự án](#_Toc81854228)

[4.1 Tạo CSDL với MySQL](#_Toc81854229)

[**4.1.1 Chi tiết các bảng**](#_Toc81854230)

[4.2 Lập trình CSDL](#_Toc81854231)

[**4.2.1 Mô hình tổ chức lập trình CSDL**](#_Toc81854232)

[4.3 Lập trình nghiệp vụ](#_Toc81854233)

[5. Kiểm thử](#_Toc81854234)

[6. Đóng gói và triển khai](#_Toc81854235)

# **Giới Thiệu Dự Án**

Thiết kế và phát triển website với các yêu cầu như sau:

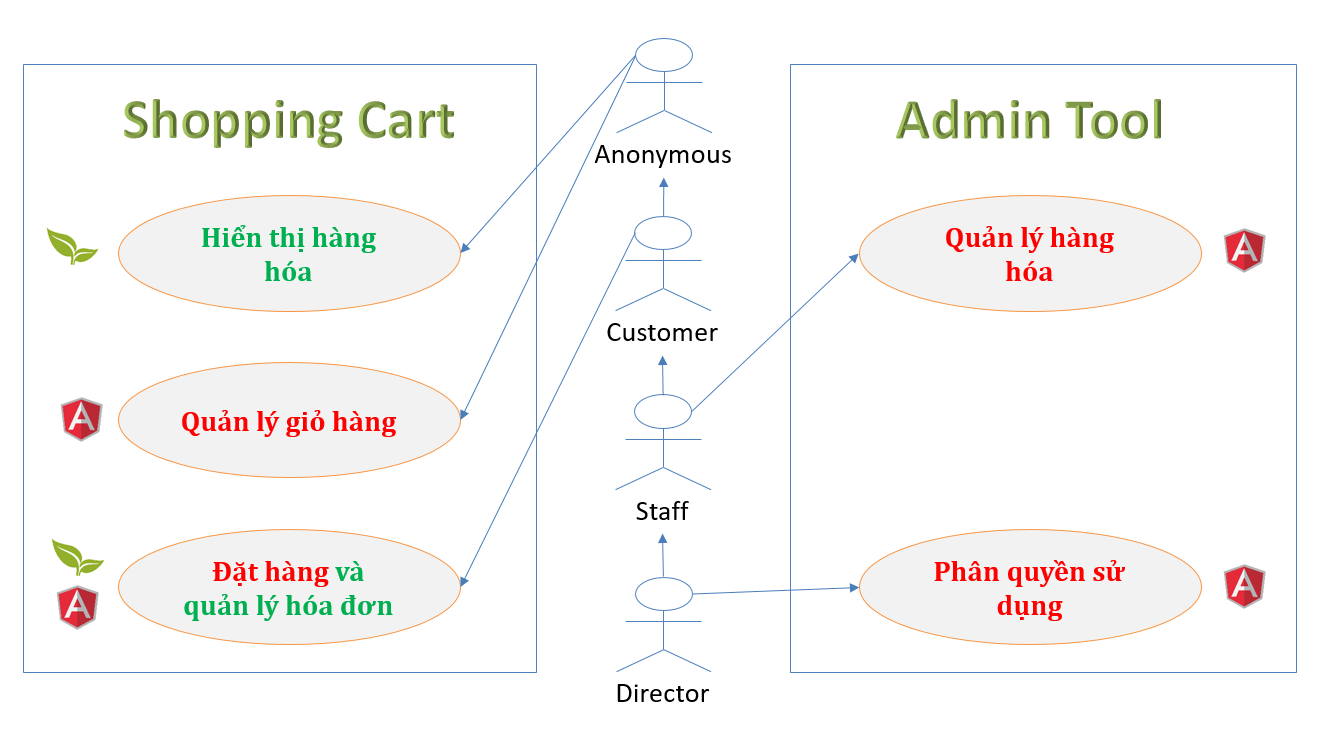
* Yêu cầu về chức năng nghiệp vụ:
* Quản lý người dung(account): quản lý các thông tin người sử dụng bao gồm các công việc như: thêm mới, cập nhật thông tin tài khoản, xoá tài khoản.
* Quản lý vai trò(role): quản lý được vai trò của mỗi tài khoản để thực hiện các công việc khác nhau.

Login vào website và logout.

* Quản lý giỏ hàng service: tài khoản thực hiện chọn các sản phẩm yêu thích vào giỏ hàng có thể thêm bớt số lượng và tiến hành checkout với các sản phẩm có trong giỏ hàng.
* Yêu cầu về mô hình:
* Sử dụng lombook, spring boot, mysql để lập trình phía back-end. Back end được thiết kế và lập trình để cung cấp các API để truy xuất ở front-end.
* Sử dụng angular để lập trình giao diện front-end.
* Sử dụng swagger để document các API.

# **Phân Tích Yêu Cầu Khách Hàng**

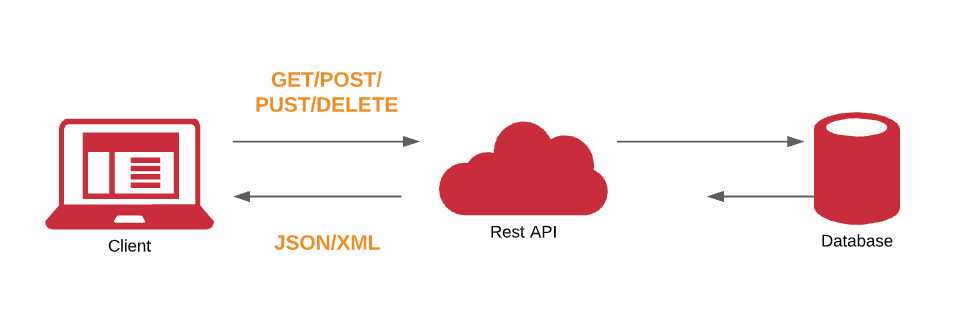
## 2.1 Sơ Đồ Use case



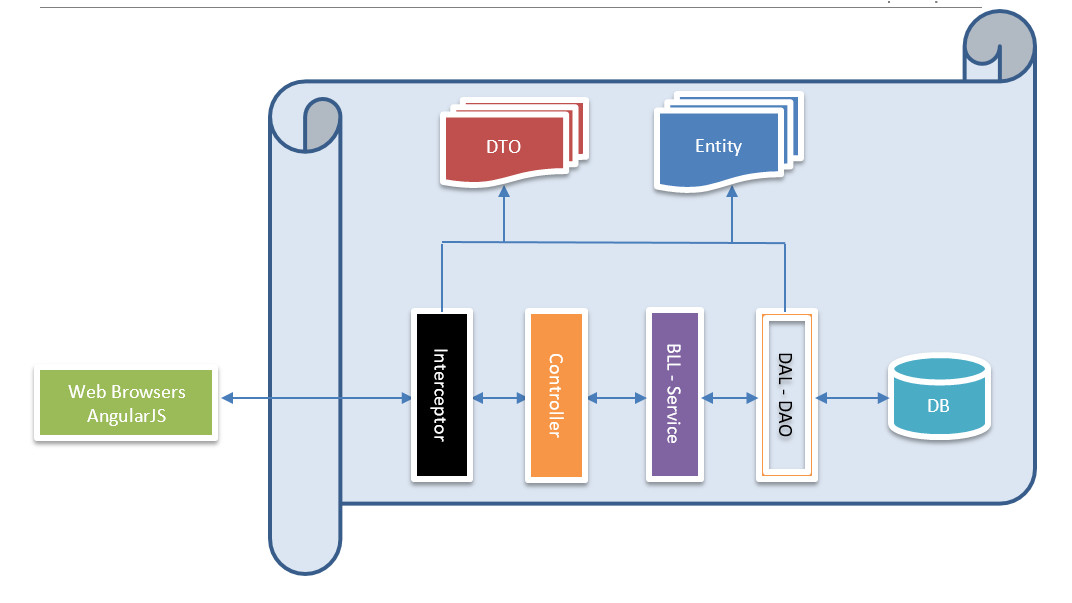
## **2.2 Sơ đồ triển khai và yêu cầu hệ thống**

## 2.2.1 Mô hình triển khai

* Hệ quản trị CSDL MySQL và CSDL sẽ được cài đặt và sử dụng trên một máy chuyên biệt.

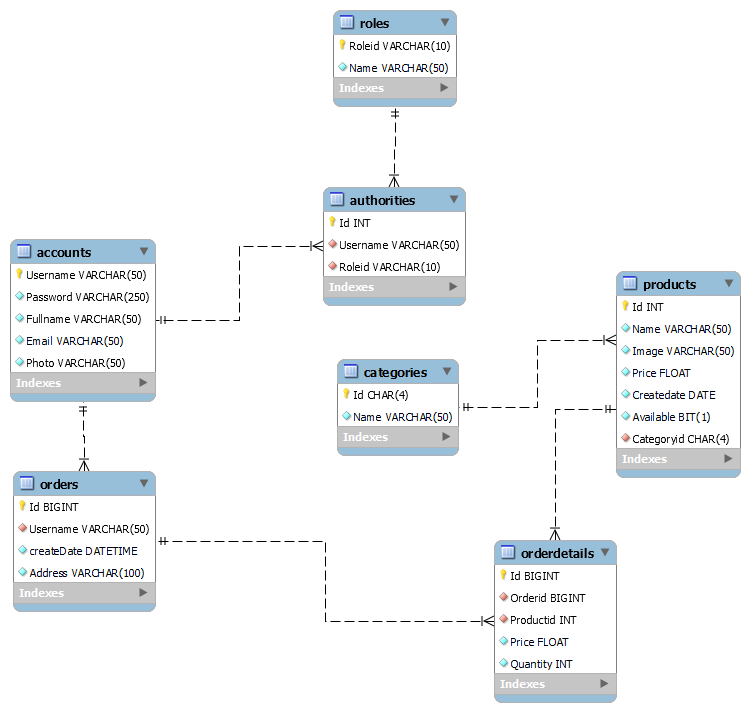


1. **Thiết kế ứng dụng**
   1. **mô hình công nghệ ứng dụng**

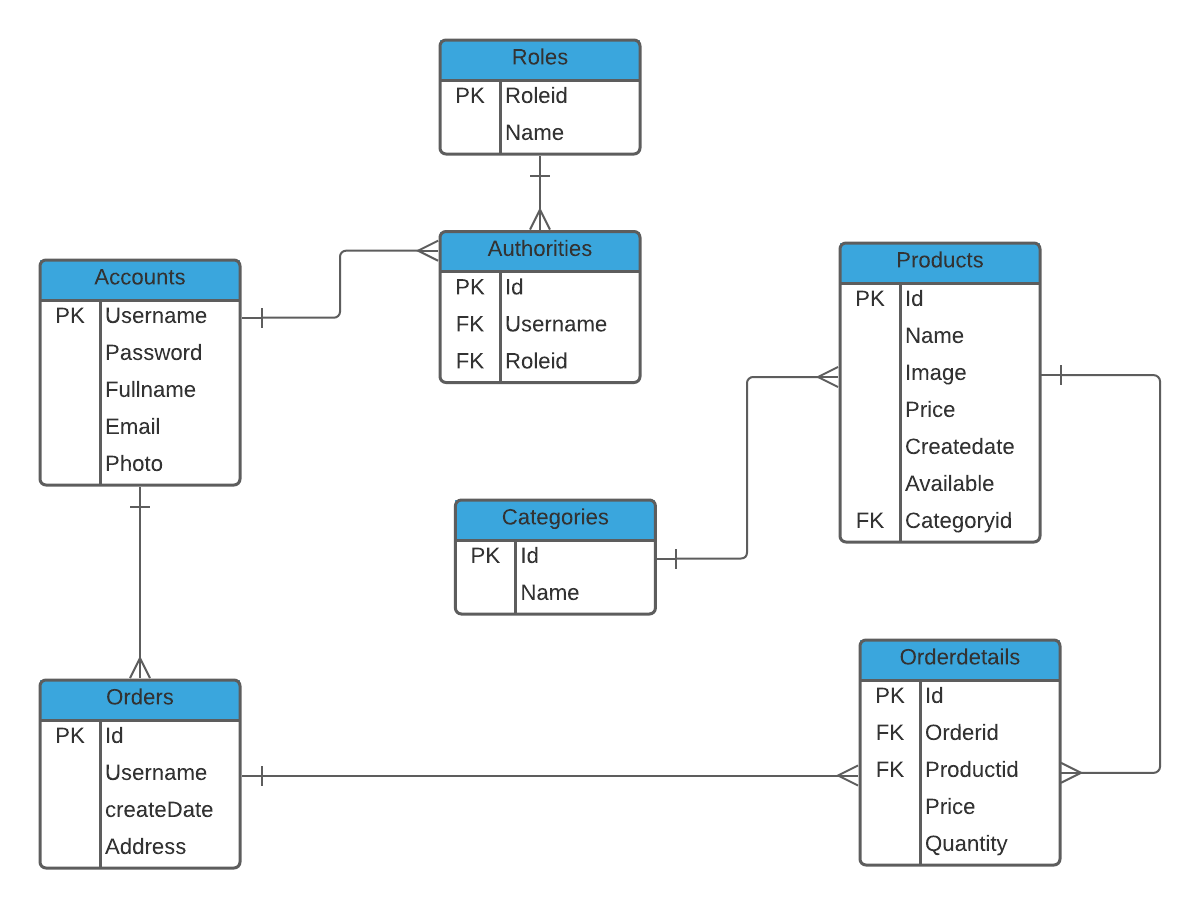


Trong đó:

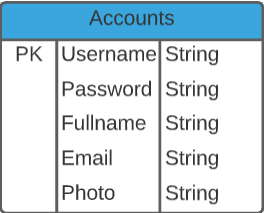
* @Entity
* DAO extends JpaRepository (Data Access Layer (DAL))
* @Service (Business Logic Layer(BLL))
* @Controller, @RestController
* DTO (Data Transfer Object)
  1. **Thực Thể**
     1. **Sơ đồ database diagram**

****

* + 1. **Sơ đồ quan hệ thực thể**

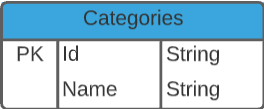
**3.2.2 Chi tiết thực thể**

* + - 1. **Thực thể Account**



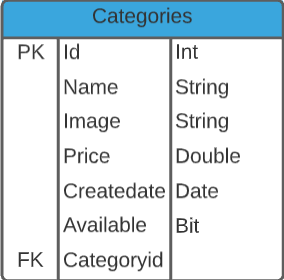
|  |  |  |
| --- | --- | --- |
| Thuộc Tính | Kiểu Dữ Liệu | Mô Tả |
| Username | String | Username |
| Password | String | Password |
| Fullname | String | Full name |
| Email | String | Email |
| Photo | String | Image |

* + - 1. **Thực thể Categories**



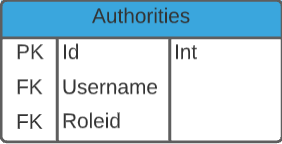
|  |  |  |
| --- | --- | --- |
| Thuộc Tính | Kiểu Dữ Liệu | Mô Tả |
| Id | String | Id category |
| Name | String | Name category |

* + - 1. **Thực thể Products**



|  |  |  |
| --- | --- | --- |
| Thuộc Tính | Kiểu Dữ Liệu | Mô Tả |
| Id | Int | Id product |
| Name | String | Name product |
| Image | String | Image product |
| Price | Double | Giá |
| Createdate | Date | Ngày khởi tạo |
| Available | Boolean | Trạng thái |
| Categoryid |  | Category id |

* + - 1. **Thực thể loại Authorities**



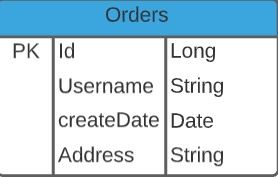
|  |  |  |
| --- | --- | --- |
| Thuộc Tính | Kiểu Dữ Liệu | Mô Tả |
| Id | Integer | Id product |
| Username |  | Username |
| Roleid |  | Roleid |

* + - 1. **Thực thể Roles**



|  |  |  |
| --- | --- | --- |
| Thuộc Tính | Kiểu Dữ Liệu | Mô Tả |
| Roleid | String | Id roles |
| Name | String | Name role |

* + - 1. **thực thể orders**



|  |  |  |
| --- | --- | --- |
| Thuộc Tính | Kiểu Dữ Liệu | Mô Tả |
| Id | Long | Id order |
| Username |  | Username |
| Createdate | Date | Ngày khởi tạo |
| Addesss | String | Địa chỉ |

* + - 1. **thực thể orderdetails**



|  |  |  |
| --- | --- | --- |
| Thuộc Tính | Kiểu Dữ Liệu | Mô Tả |
| Id | Long | Id order details |
| OrderId |  | Order id |
| Price | Double | Giá |
| Quantity | Integer | Số lượng |
| Productid |  | Product id |

1. **Thực hiện dự án**
   1. **Tạo CSDL với MySQL**
      1. **Chi tiết các bảng**

#### **Bảng Accounts**

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Username | VARCHAR(50) | PK, NOT NULL | User name |
| Password | VARCHAR(250) | NOT NULL | Pass word |
| Fullname | VARCHAR(50) | NOT NULL | Full name |
| Email | VARCHAR(50) | NOT NULL | Email |
| Photo | VARCHAR(50) | NOT NULL | Hình ảnh |

#### **Bảng Authorities**

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | INT | PK, NOT NULL, AUTO\_INCREMENT | Id authorities |
| Username | VARCHAR(50) | FK, NOT NULL | User name |
| RoleId | VARCHAR(10) | FK, NOT NULL | Id role |

#### **Bảng Categories**

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | CHAR(4) | PK, NOT NULL | Id category |
| Name | VARCHAR(50) | NOT NULL | Name category |

#### **Bảng Orders**

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | CHAR(4) | PK, NOT NULL, AUTO\_INCREMENT | Id order |
| Username | VARCHAR(50) | FK, NOT NULL | Name category |
| Createdate | DATETIME | NOT NULL | Create date |
| Address | VARCHAR(150) | NOT NULL | Địa chỉ |

#### **Bảng Products**

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | INT | PK, NOT NULL, AUTO\_INCREMENT | Id product |
| Name | VARCHAR(50) | NOT NULL | Name product |
| Image | VARCHAR(50) | NOT NULL | Hình ảnh |
| Price | FLOAT | NOT NULL | Giá |
| Createdate | DATE | NOT NULL | Create Date |
| Available | BIT | NOT NULL | Trạng thái |
| CategoryId | CHAR(4) | FK, NOT NULL | Id category |

#### **Bảng OrderDetails**

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | BIGINT | PK, NOT NULL, AUTO\_INCREMENT | Id OrderDetails |
| Price | FLOAT | NOT NULL | Giá |
| Quantity | INT | NOT NULL | Số lượng |
| OrderId | BIGINT | FK, NOT NULL | Order id |
| ProductId | INT | FK, NOT NULL | Product id |

#### **Bảng Roles**

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | VARCHAR(10) | PK, NOT NULL | Id role |
| Name | VARCHAR(50) | NOT NULL | Name role |

* 1. **Lập trình CSDL**
     1. **Mô hình tổ chức lập trình CSDL**

DB

Entity

DTO

|  |  |  |  |
| --- | --- | --- | --- |
| **Controller**  **(Behaviors)** | **Service**  **(Services)** | **Repository**  **(DAOs)** | **Database**  **(Tables)** |

* + - 1. **Entity class và DAO**
         1. **Class Diagram**
* Các lớp Entity là lớp dùng để miêu tả các dữ liệu từ các bảng tương ứng trong CSDL.
* Các lớp DAO dùng để truy xuất CSDL.
  1. **Lập trình nghiệp vụ**
     1. **Chức năng thêm sản phẩm vào giỏ hàng và checkout giỏ hàng**

|  |  |  |
| --- | --- | --- |
| STT | Phương thức xử lý | Mô tả |
| 1 | Cart.add(id) | Thực hiển kiểm tra xem sản phẩm đã tồn tại chưa, nếu tồn tại thì “qty” tăng lên 1, cập nhật thông tin vào localstorage. Ngược lại lấy thông tin sản phẩm qua RestAPI “/rest/products/{id} , lưu thông tin sản phẩm vừa tìm được lên localstorage. |
| 2 | Cart.remove(id) | Thực hiện xóa sản phẩm ra khỏi localstorage. |
| 3 | Cart.clear() | Xóa các sản phẩm được lưu trong localstorage. |
| 4 | Cart.count() | Tính số sản phấm có trong giỏ hàng. |
| 5 | Cart.amount() | Tính tổng tiển cần phải thanh toán trong giỏ hàng. |
| 6 | Cart.saveToLocalStorage() | Lưu sản phẩm được chọn vào localstorage. |
| 7 | Cart.loadFormLocalStorage() | Lấy thông tin được lưu ở localstorage lưu vào mảng có tên items. |
| 8 | Purchase() | Thực hiện đặt hàng |

* + 1. **Chức năng quản lý người dùng**

|  |  |  |
| --- | --- | --- |
| STT | Phương thức xử lý | Mô tả |
| 1 | Initialize() |  |
| 2 | Edit() | Load thông tin người dùng lên form |
| 3 | Create() | Thêm mới tài khoản thông qua RestAPI `/rest/account/, item` |
| 4 | Update() | Cập nhật thông tin tài khoản thông qua RestAPI `/rest/account/${item.username}, item` |
| 5 | Delete() | Xóa thông tin tài khoản thông qua RestAPI `/rest/account/delete/${item.username}` |
| 6 | Reset() | Reset thông tin người dùng |
| 7 | Pager() | Phân trang |

* + 1. **Chức năng quán lý phân quyền**

|  |  |  |
| --- | --- | --- |
| STT | Phương thức xử lý | Mô tả |
| 1 | Initialize() |  |
| 2 | $scope.grant\_authority() | Cấp quyền thông qua RestAPI `/rest/authorities, authority` |
| 3 | $scope.revoke\_authority () | Xoá quyền được cấp thông qua RestAPI `/rest/authorities/${authority.id}` |

# **Kiểm thử**

# **Đóng gói và triển khai**

* 1. **Sản phẩm phần mềm**

|  |  |  |
| --- | --- | --- |
| **TT** | **Thành phần** | **Mô tả** |
| 1 | 1 file source project | File resources |
| 2 | Minir2s.sql | Cơ sở dữ liệu |
| 3 | Readme.txt | Hướng dẫn cài đặt |
| 4 | Guide.docx | Hướng dẫn sử dụng |

* 1. **Hướng dẫn cài đặt**
* Cài đặt và giải nén file hautmpc01058\_Mini\_Project.rar.
* Cài đặt và import file MiniR2S.sql để tạo database minir2s.
  1. **Hướng dẫn sửa dụng**
* Khởi chạy project, mở trình duyệt chạy với dường dẫn http:/localhost:8080/ để truy cập vào ứng dụng.
* Đăng nhập với tài khoản customer,MinhHau với vai trò ROLE\_USER, director với vai trò ROLE\_ADMIN, staff với vai trò ROLE\_PM.

= HẾT =