

MINI PROJECT REQUIREMENT

MỤC LỤC

[1. GIỚI THIỆU DỰ ÁN 4](#_Toc81942160)

[2. PHÂN TÍCH YÊU CẦU KHÁCH HÀNG 5](#_Toc81942161)

[2.1 USE CASE 5](#_Toc81942162)

[2.1.1 Use case đối tượng ADMIN 5](#_Toc81942163)

[2.1.2 Use case đối tượng PM 6](#_Toc81942164)

[2.1.3 Use case đối tượng USER 6](#_Toc81942165)

[2.2 Sơ đồ triển khai và yêu cầu hệ thống 7](#_Toc81942166)

[3. THIẾT KẾ ỨNG DỤNG 7](#_Toc81942169)

[3.1 MÔ HÌNH CÔNG NGHỆ ỨNG DỤNG 7](#_Toc81942170)

[3.2 THỰC THỂ 8](#_Toc81942171)

[3.2.1 Sơ đồ quan hện thực thể 8](#_Toc81942172)

[3.2.2 Chi tiết thực thể 8](#_Toc81942173)

[4.Thực hiện dự án 11](#_Toc81942174)

[4.1 Tạo CSDL với MySQL 11](#_Toc81942175)

[4.1.1 Chi tiết các bảng 11](#_Toc81942176)

[4.2 Lập trình CSDL 15](#_Toc81942177)

[4.1.1 Mô hình tổ chức lập trình CSDL 15](#_Toc81942178)

[4.1.1.1 Entity class và DAO 16](#_Toc81942179)

[4.2 Lập trình nghiệp vụ 18](#_Toc81942180)

[4.2.1 Chức năng thêm sản phẩm vào giỏ hàng và checkout giỏ hàng 18](#_Toc81942181)

[4.2.2 Chức năng quản lý người dùng 19](#_Toc81942182)

[4.2.3 Chức năng quán lý phân quyền 19](#_Toc81942183)

[5. Kiểm thử 19](#_Toc81942184)

[6. Đóng gói và triển khai 19](#_Toc81942185)

[6.1 Sản phẩm phần mềm 19](#_Toc81942186)

[6.2 Hướng dẫn cài đặt 19](#_Toc81942187)

# GIỚI THIỆU DỰ ÁN

Thiết kế và phát triển website với các yêu cầu của R2S như sau:

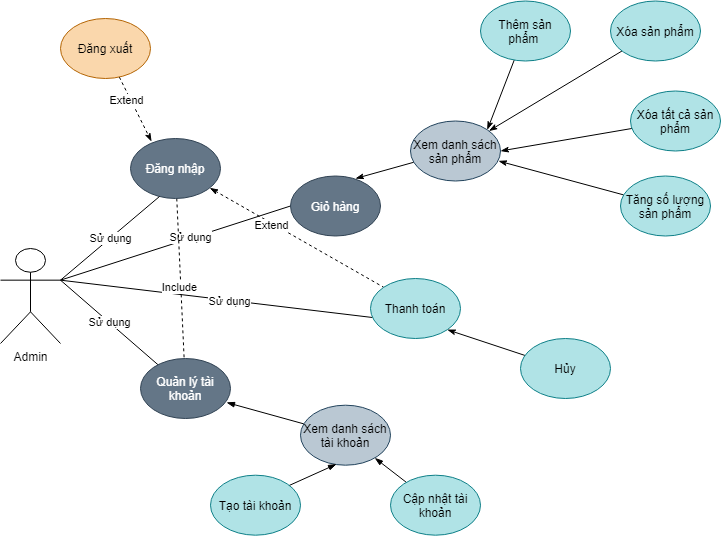
* **Yêu cầu về chức năng nghiệp vụ**
* **Quản lý tài khoản người dùng:** Quản lý các thông tin người sử dụng tài khoản bao gồm các công việc như: thêm, cập nhật thông tin tài khoản, cấp quyền đăng nhập cho người dùng.
* **Quản lý giỏ hàng:** Tài khoản thực hiện thêm các sản phẩm yêu thích vào giỏ hàng và tiến hành xóa hoặc thanh toán với các sản phẩm có trong giỏ hàng.
* **Quản lý vai trò:** Quản lý được vai trò của mỗi tài khoản để thực hiện các công việc, truy cập khác nhau.
* Đăng nhập và đăng xuất vào website.
* **Yêu cầu về mô hình**
* Sử dụng 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 lombook.
* 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 USE CASE**

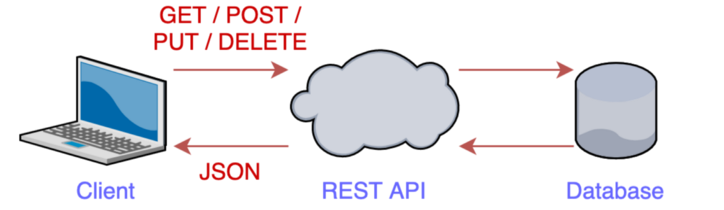
### **2.1.1 Use case đối tượng ADMIN**

### **2.1.2 Use case đối tượng PM**



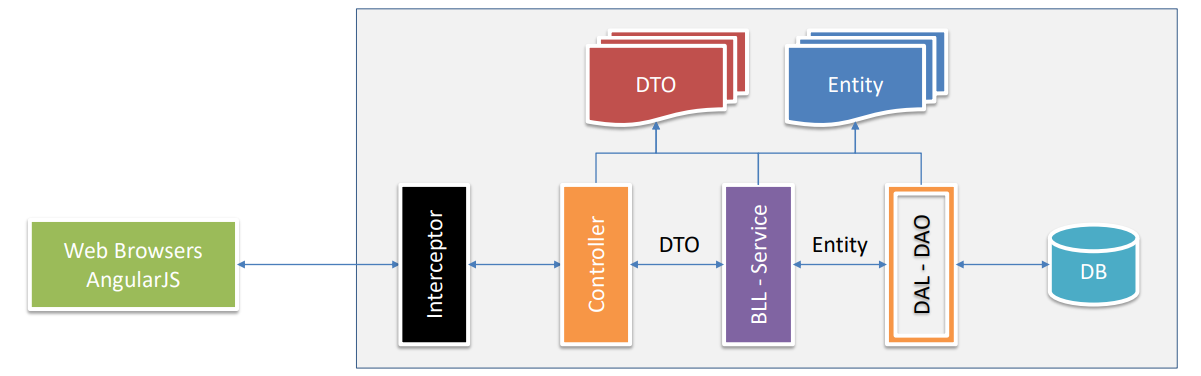
### **2.1.3 Use case đối tượng USE**R

## **2.2 Sơ đồ triển khai**



1. **THIẾT KẾ ỨNG DỤNG**

## **3.1 MÔ HÌNH CÔNG NGHỆ ỨNG DỤNG**

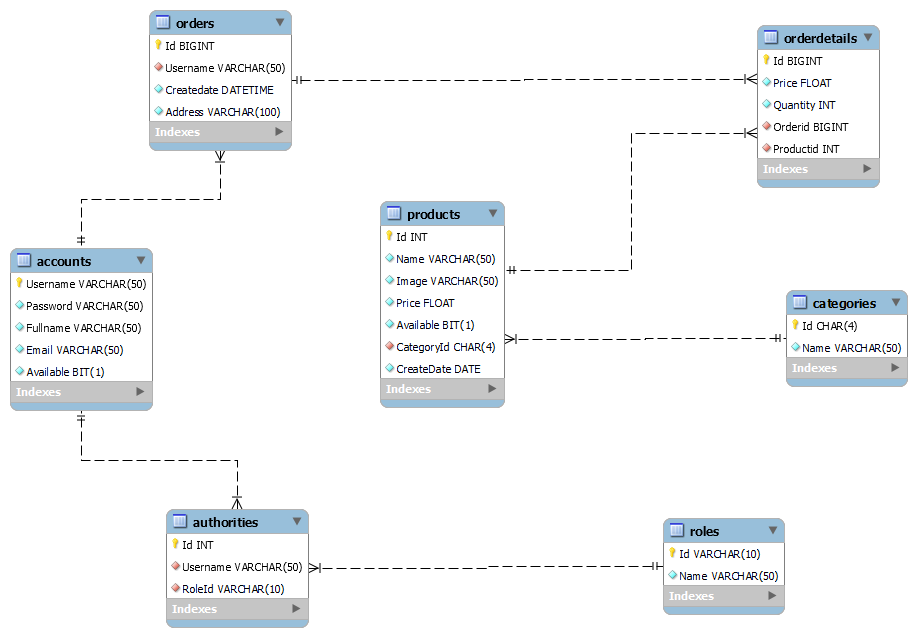


Trong đó:

* @Entity
* DAO extends JpaRepository (Data Access Layer (DAL))
* @Service (Business Logic Layer(BLL))
* @Controller, @RestController
* DTO (Data Transfer Object)

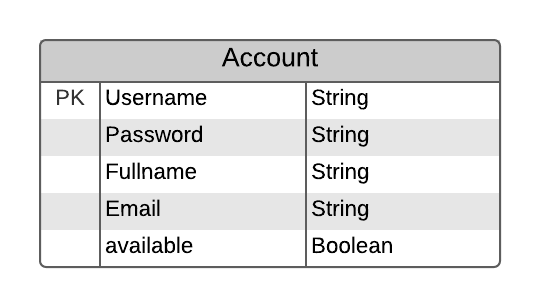
## **3.2 THỰC THỂ**

### **3.2.1 Sơ đồ quan hện thực thể**



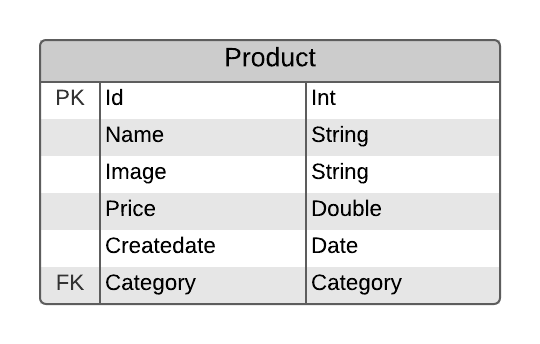
### **3.2.2 Chi tiết thực thể**

Thực thể Account



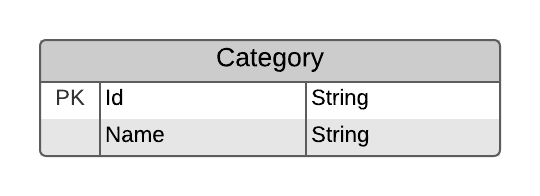
|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu Dữ Liệu | Mô Tả |
| Username | String | Tên đăng nhập |
| Password | String | Mật khẩu đăng nhập |
| Fullname | String | Họ và tên |
| Available | Boolean | Trạng thái |

Thực thể Product



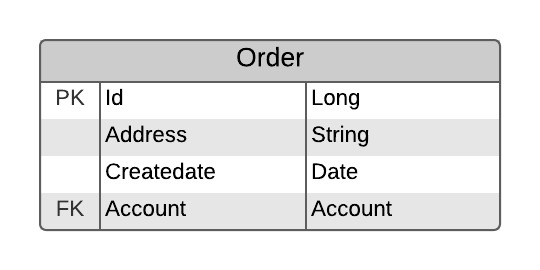
|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu Dữ Liệu | Mô Tả |
| Id | Int | Mã sản phẩm |
| Name | String | Tên sản phẩm |
| Image | String | Ảnh |
| Price | Float | Giá |
| CreateDate | Date | Ngày tạo |
| CategoryId | Boolean | Mã loại sản phẩm |

Thực thể Categories



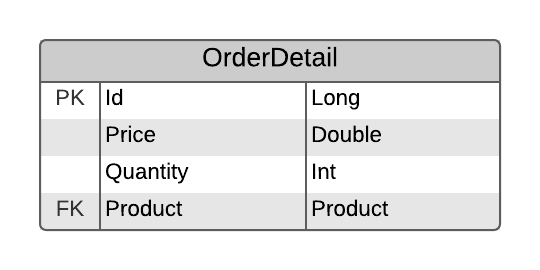
|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu Dữ Liệu | Mô Tả |
| Id | String | Mã loại sản phẩm |
| Name | String | Loại sản phẩm |

Thực thể Order



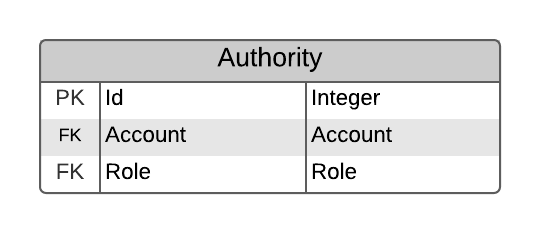
|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu Dữ Liệu | Mô Tả |
| Id | Long | Mã đơn hàng |
| Address | String | Địa chỉ |
| CreateDate | Date | Ngày tạo |
| Account | Account | Đối tượng Account |

Thực thể OderDetails



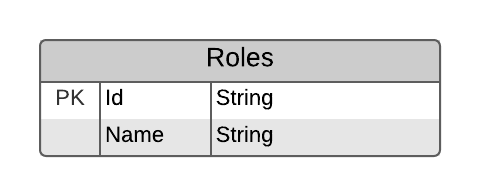
|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu Dữ Liệu | Mô Tả |
| Id | Long | Mã đơn hàng chi tiết |
| Price | String | Tổng tiền dơn hàng |
| Quantity | Int | Số lượng |
| Product | Product | Đối tượng Product |

Thực thể Authorities



|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu Dữ Liệu | Mô Tả |
| Id | Int | Mã |
| Account | Account | Đối tượng Account |
| Role | Role | Đối tượng Roles |

Thực thể Roles



|  |  |  |
| --- | --- | --- |
| Thuộc tính | Kiểu Dữ Liệu | Mô Tả |
| Id | String | Mã vai trò |
| Name | String | Tên vai trò |

4.Thực hiện dự án

4.1 Tạo CSDL với MySQL

4.1.1 Chi tiết các bảng

Bảng Account

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Username | VARCHAR(50) | PK, NOT NULL | Tên đăng nhập |
| Password | VARCHAR(45) | NOT NULL | Mật khẩu đăng nhập |
| Fullname | VARCHAR(45 | NOT NULL | Họ và tên |
| Email | VARCHAR(45) | NOT NULL |  |
| Availible | BIT | NOT NULL |  |

#### Mã lệnh tạo bảng

|  |
| --- |
| CREATE TABLE `accounts` (  `Username` varchar(50) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `Password` varchar(50) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `Fullname` varchar(50) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `Email` varchar(50) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `Available` bit(1) NOT NULL,  PRIMARY KEY (`Username`)  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8\_unicode\_ci |

Bảng Authorities

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | INT | PK, NOT NULL, AI | Mã |
| Username | VARCHAR(50) | FK, NOT NULL | Tên đăng nhập |
| RoleId | VARCHAR(10) | FK, NOT NULL | Mã vai trò |

#### Mã lệnh tạo bảng

|  |
| --- |
| CREATE TABLE `authorities` (  `Id` int NOT NULL AUTO\_INCREMENT,  `Username` varchar(45) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `RoleId` varchar(10) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  PRIMARY KEY (`Id`),  UNIQUE KEY `UKhtk0tjw6uf67w0vkhy5n3j1oq` (`Username`,`RoleId`),  KEY `RoleId` (`RoleId`),  CONSTRAINT `authorities\_ibfk\_1` FOREIGN KEY (`Username`) REFERENCES `accounts` (`Username`),  CONSTRAINT `authorities\_ibfk\_2` FOREIGN KEY (`RoleId`) REFERENCES `roles` (`Id`)  ) ENGINE=InnoDB AUTO\_INCREMENT=16 DEFAULT CHARSET=utf8mb3 COLLATE=utf8\_unicode\_ci |

Bảng Categories

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | CHAR(4) | PK, NOT NULL | Mã loại sản phẩm |
| Name | VARCHAR(50) | NOT NULL | Loại sản phẩm |

#### Mã lệnh tạo bảng

|  |
| --- |
| CREATE TABLE `categories` (  `Id` char(4) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci NOT NULL,  `Name` varchar(45) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  PRIMARY KEY (`Id`)  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8\_unicode\_ci |

Bảng Order

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | CHAR(4) | PK, NOT NULL, AI | Mã đơn hàng |
| Username | VARCHAR(50) | FK, NOT NULL | Tên đăng nhập |
| Createdate | DATETIME | NOT NULL | Ngày tạo |
| Address | VARCHAR(100) | NOT NULL |  |

#### Mã lệnh tạo bảng

|  |
| --- |
| CREATE TABLE `orders` (  `Id` bigint NOT NULL AUTO\_INCREMENT,  `Username` varchar(45) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `Createdate` datetime NOT NULL,  `Address` varchar(150) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  PRIMARY KEY (`Id`),  KEY `Username` (`Username`),  CONSTRAINT `orders\_ibfk\_1` FOREIGN KEY (`Username`) REFERENCES `accounts` (`Username`)  ) ENGINE=InnoDB AUTO\_INCREMENT=11 DEFAULT CHARSET=utf8mb3 COLLATE=utf8\_unicode\_ci |

Bảng Product

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | INT | PK, NOT NULL, AI | Mã sản phẩm |
| Name | VARCHAR(50) | NOT NULL | Tên sản phẩm |
| Image | DATETIME | NOT NULL |  |
| Price | FLOAT | NOT NULL |  |
| Createdate | DATETIME | NOT NULL | Ngày tạo |
| Available | BIT | NOT NULL |  |
| CategoryId | CHAR(4) | FK, NOT NULL | Mã loại sản phẩm |

#### Mã lệnh tạo bảng

|  |
| --- |
| CREATE TABLE `products` (  `Id` int NOT NULL AUTO\_INCREMENT,  `Name` varchar(250) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `Image` varchar(45) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `Price` float NOT NULL,  `Available` bit(1) NOT NULL,  `CategoryId` char(4) CHARACTER SET utf8 COLLATE utf8\_unicode\_ci NOT NULL,  `createdate` date NOT NULL,  PRIMARY KEY (`Id`),  KEY `CategoryId` (`CategoryId`),  CONSTRAINT `products\_ibfk\_1` FOREIGN KEY (`CategoryId`) REFERENCES `categories` (`Id`)  ) ENGINE=InnoDB AUTO\_INCREMENT=35 DEFAULT CHARSET=utf8mb3 COLLATE=utf8\_unicode\_ci |

Bảng OrderDetails

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | BIGINT | PK, NOT NULL, AI | Mã hóa đơn chi tiết |
| Price | DOUBLE | NOT NULL | Tổng tiền hóa đơn |
| Quantity | INT | NOT NULL | Số lượng |
| OrderId | BIGINT | FK, NOT NULL | Mã đơn hàng |
| ProductId | INT | FK, NOT NULL | Mã sản phẩm |

#### Mã lệnh tạo bảng

|  |
| --- |
| CREATE TABLE `orderdetails` (  `id` bigint NOT NULL AUTO\_INCREMENT,  `price` double DEFAULT NULL,  `quantity` int DEFAULT NULL,  `orderid` bigint DEFAULT NULL,  `productid` int DEFAULT NULL,  PRIMARY KEY (`id`),  KEY `FKj4gc2ja2otvwemf4rho2lp3s8` (`orderid`),  KEY `FKaltatpxipsjtcih4d1h6bn0xr` (`productid`),  CONSTRAINT `FKaltatpxipsjtcih4d1h6bn0xr` FOREIGN KEY (`productid`) REFERENCES `products` (`Id`),  CONSTRAINT `FKj4gc2ja2otvwemf4rho2lp3s8` FOREIGN KEY (`orderid`) REFERENCES `orders` (`Id`)  ) ENGINE=InnoDB AUTO\_INCREMENT=17 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci |

Bảng Roles

#### Cấu trúc bảng

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Cột | Kiểu | Ràng buộc | Ghi chú |
| Id | VARCHAR(50) | PK, NOT NULL | Mã cai trò |
| Name | VARCHAR(50) | NOT NULL | Tên vai trò |

#### Mã lệnh tạo bảng

|  |
| --- |
| CREATE TABLE `roles` (  `Id` varchar(45) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  `Name` varchar(45) CHARACTER SET utf8 COLLATE utf8\_general\_ci NOT NULL,  PRIMARY KEY (`Id`)  ) ENGINE=InnoDB DEFAULT CHARSET=utf8mb3 COLLATE=utf8\_unicode\_ci |

* 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**

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.

Account và AccountDAO

|  |  |
| --- | --- |
| THÀNH PHẦN | MÔ TẢ |
| Account | Là lớp thực thể mô tả dữ liệu Account |
| AccountDAO | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng Accounts.   * E được cụ thể hóa là Acount. * K được cụ thể hóa là String. * SQL là các câu lệnh thao tác dữ liệu trên bảng Accounts. |

Product và ProductDAO

|  |  |
| --- | --- |
| THÀNH PHẦN | MÔ TẢ |
| Product | Là lớp thực thể mô tả dữ liệu Product |
| ProductDAO | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng Products.   * E được cụ thể hóa là Product. * K được cụ thể hóa là Integer. * SQL là các câu lệnh thao tác dữ liệu trên bảng Products. |

Category và CategoryDAO

|  |  |
| --- | --- |
| THÀNH PHẦN | MÔ TẢ |
| Category | Là lớp thực thể mô tả dữ liệu Category |
| CategoryDAO | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng Categories.   * E được cụ thể hóa là Category. * K được cụ thể hóa là String. * SQL là các câu lệnh thao tác dữ liệu trên bảng Categories. |

Order và OrderDAO

|  |  |
| --- | --- |
| THÀNH PHẦN | MÔ TẢ |
| Order | Là lớp thực thể mô tả dữ liệu Order |
| OrderDAO | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng Orders.   * E được cụ thể hóa là Order. * K được cụ thể hóa là Long. * SQL là các câu lệnh thao tác dữ liệu trên bảng Orders. |

OrderDetail và OrderDetailDAO

|  |  |
| --- | --- |
| THÀNH PHẦN | MÔ TẢ |
| OrderDtail | Là lớp thực thể mô tả dữ liệu OrderDtail |
| OrderDtailDAO | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng OrderDtails.   * E được cụ thể hóa là OrderDtail. * K được cụ thể hóa là Long. * SQL là các câu lệnh thao tác dữ liệu trên bảng OrderDtails. |

Role và RoleDAO

|  |  |
| --- | --- |
| THÀNH PHẦN | MÔ TẢ |
| Role | Là lớp thực thể mô tả dữ liệu Role |
| RoleDAO | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng Roles.   * E được cụ thể hóa là Role. * K được cụ thể hóa là String. * SQL là các câu lệnh thao tác dữ liệu trên bảng Roles. |

Authorities vad AuthoritieDAO

|  |  |
| --- | --- |
| THÀNH PHẦN | MÔ TẢ |
| Authoritie | Là lớp thực thể mô tả dữ liệu Authoritie |
| AuthoritieDAO | Là lớp DAO cụ thể hóa các hoạt động truy xuất CSDL với bảng Authorities.   * E được cụ thể hóa là Authoritie. * K được cụ thể hóa là Integer. * SQL là các câu lệnh thao tác dữ liệu trên bảng Authorities. |

* 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 hiện thêm sản phẩ, nếu tồn tại thì tăng số lượng lên 1, cập nhật thông tin lên localstorage. Ngược lại lấy thông tin sản phẩm qua RestAPI, 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 tất cả 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. |

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

|  |  |  |
| --- | --- | --- |
| STT | Phương thức xử lý | Mô tả |
| 1 | Initialize() |  |
| 2 | Edit() | Hiển thị thông tin người dùng lên form |
| 3 | Create() | Thêm mới tài khoản thông qua RestAPI |
| 4 | Update() | Cập nhật thông tin tài khoản thông qua RestAPI |
| 5 | Reset() | Xóa trắng form |
| 6 | Pager() | Phân trang |

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

|  |  |  |
| --- | --- | --- |
| STT | Phương thức xử lý | Mô tả |
| 1 | Initialize() |  |
| 2 | Edit() | Hiển thị thông tin người dùng lên form |
| 3 | Create() | Thêm mới tài khoản thông qua RestAPI |
| 4 | Update() | Cập nhật thông tin tài khoản thông qua RestAPI |
| 5 | Reset() | Xóa trắng form |
| 6 | Pager() | Phân trang |

# 5. Kiểm thử

Đính kèm trong file excel.

# Đóng gói và triển khai

* 1. **Sản phẩm phần mềm**

|  |  |  |
| --- | --- | --- |
| **TT** | **Thành phần** | **Mô tả** |
| 1 | File tientppc00588\_Miniproject | File resources |
| 2 | miniproject\_r2s.sql | Cơ sở dữ liệu |
| 3 | Readme.txt | Hướng dẫn cài đặt |
| 4 | Guide.docx | Hướng dẫn sử dụng |

* 1. **Hướng dẫn cài đặt**
* Bước 1. Cài đặt hệ cơ sở dữ liệu mysql trong file tài nguyên, mật khẩu kết nối “1234”.
* Chạy SQL.sql để tạo CSDL miniproject\_r2s.

HẾT