**ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**

**\*\*\*\*\***



**BÁO CÁO BÀI TẬP LỚN OOP**

**ĐỀ TÀI: QUẢN LÝ BÁN HÀNG**

**Giảng viên hướng dẫn**: Lê Đức Hậu

**Mã học phần**: IT3103

**Mã lớp học:** 151964

**Nhóm 4:**  Phạm Công Hào 20215045

Nguyễn Phúc Mạnh 20215087

Phạm Xuân Phong 20215115

Tống Chí Quang 20210725

Nguyễn Trung Sơn 20215132

Mục Lục

[I) Giới thiệu bài toán 3](#_Toc185710162)

[II) Thiết kế hệ thống 4](#_Toc185710163)

[III) Phát triển hệ thống 7](#_Toc185710164)

[IV) Biểu đồ lớp 8](#_Toc185710165)

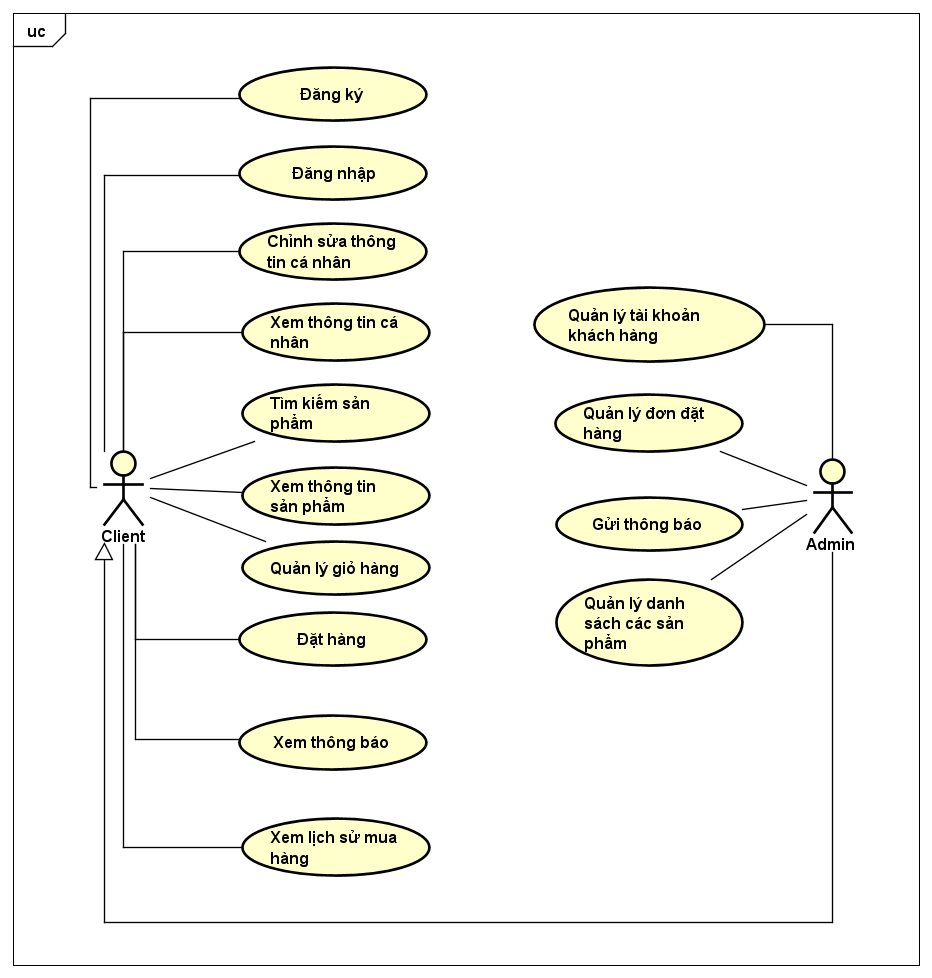
[V) Đánh giá trong nhóm 9](#_Toc185710166)

# I) Giới thiệu bài toán

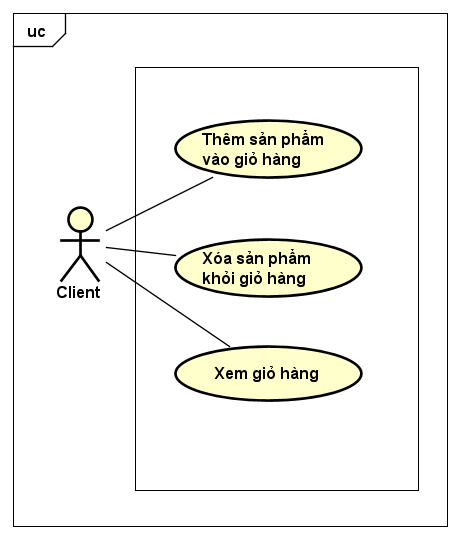
* Quản lý bán hàng là một trong những khía cạnh quan trọng trong việc vận hành một doanh nghiệp, đặc biệt trong các cửa hàng bán lẻ, siêu thị, hoặc nền tảng thương mại điện tử. Mục tiêu của bài toán quản lý bán hàng là tạo ra một hệ thống hiệu quả để theo dõi và xử lý các giao dịch mua bán, quản lý hàng tồn kho, khách hàng, và các chi tiết liên quan đến sản phẩm và dịch vụ.
* Bọn em xây dựng một hệ thống quản lý bán hàng, người dùng được chia thành hai nhóm chính: **Client (Khách hàng)** và **Admin (Quản trị viên)**. Mỗi nhóm có những quyền hạn và nhiệm vụ khác nhau khi tương tác với hệ thống. Dưới đây là góc nhìn của từng nhóm người dùng về bài toán quản lý bán hàng:
* **Client (Khách hàng):**
  + **Đăng ký và đăng nhập**: Khách hàng có thể tạo tài khoản cá nhân hoặc đăng nhập vào tài khoản đã có. Hệ thống cần hỗ trợ các chức năng bảo mật như mã hóa mật khẩu và xác thực người dùng.
  + **Duyệt sản phẩm**: Khách hàng có thể duyệt qua danh mục sản phẩm, xem chi tiết các mặt hàng như mô tả, hình ảnh, giá, và thông tin khác. Hệ thống hỗ trợ tìm kiếm sản phẩm theo tên, danh mục, hoặc các bộ lọc khác (giá, độ phổ biến, đánh giá)
  + **Thêm sản phẩm vào giỏ hàng**: Khách hàng có thể thêm sản phẩm vào giỏ hàng, điều chỉnh số lượng và xem tổng giá trị. Giỏ hàng có thể được lưu trữ và tiếp tục vào các phiên mua sau.
  + **Thực hiện thanh toán**: Khách hàng có thể theo dõi trạng thái của các đơn hàng đã đặt (đang xử lý, đã giao, đã thanh toán, hủy bỏ).
  + **Quản lý tài khoản cá nhân**: Khách hàng có thể chỉnh sửa thông tin cá nhân (địa chỉ, số điện thoại, email). Khách hàng cũng có thể thay đổi mật khẩu và quản lý các cài đặt tài khoản của mình
* **Admin (Quản trị viên)**
  + **Quản lý sản phẩm:** Quản trị viên có thể thêm, sửa, xóa sản phẩm trong hệ thống, bao gồm các thuộc tính như tên sản phẩm, mô tả, giá, số lượng tồn kho, và hình ảnh. Cập nhật thông tin về sản phẩm, bao gồm cả việc điều chỉnh giá và kiểm soát tồn kho.
  + **Quản lý khách hàng:** Admin có thể xem danh sách khách hàng, chỉnh sửa thông tin cá nhân của họ, và theo dõi lịch sử mua hàng. Quản trị viên có thể xác thực hoặc chặn tài khoản khách hàng nếu cần thiết.
  + **Quản lý đơn hàng:** Admin có thể xem và xử lý các đơn hàng của khách hàng, bao gồm việc thay đổi trạng thái đơn hàng, từ "Đang xử lý" đến "Đã giao" hoặc "Đã hủy". Quản trị viên có thể kiểm tra thông tin đơn hàng chi tiết và thay đổi nếu cần (ví dụ: hủy đơn hàng, thêm sản phẩm).
  + **Quản lý thanh toán và hóa đơn:** Admin có thể theo dõi trạng thái thanh toán của đơn hàng (đã thanh toán, chưa thanh toán, đang xử lý). Hệ thống cũng sẽ hỗ trợ việc phát hành hóa đơn và gửi chúng cho khách hàng sau khi giao dịch hoàn tất.
  + **Quản lý tồn kho:** Admin có thể theo dõi và kiểm soát số lượng sản phẩm tồn kho. Cập nhật tự động khi có các giao dịch bán hàng hoặc nhập kho mới.

# II) Thiết kế hệ thống

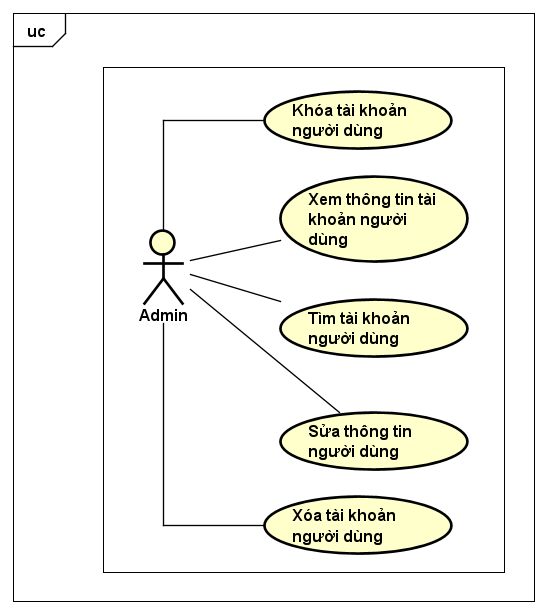
* **Biểu đồ User Case**

****

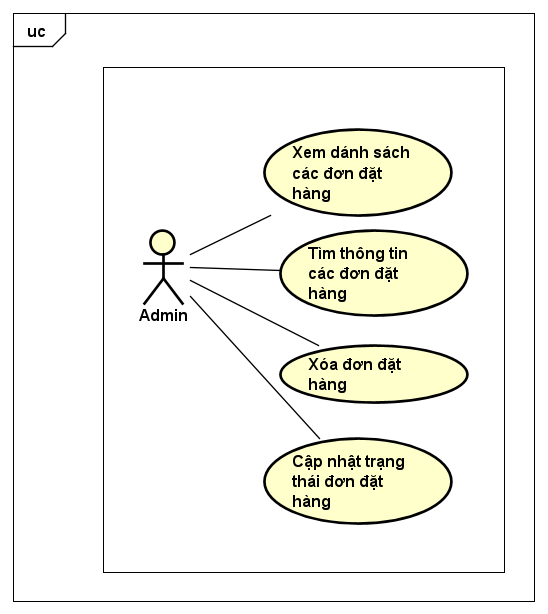
* **Biểu đồ phân rã các chức năng:** 
  + Phân rã chức năng Quản lý giỏ hàng của Client



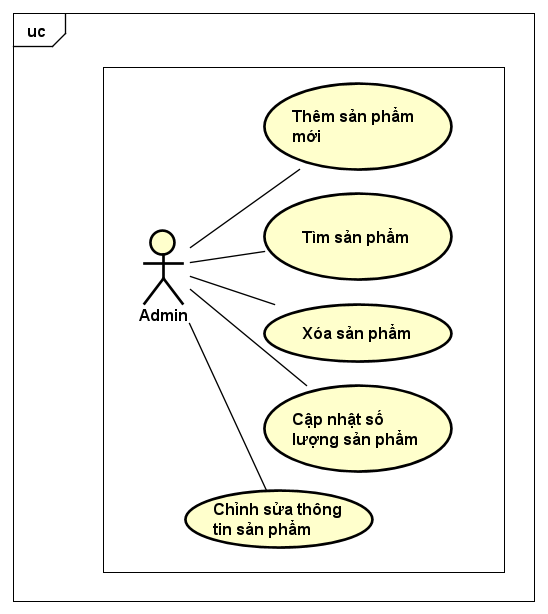
* Phân rã chức năng Quản lý tài khoản khách hàng của Admin



* Phân rã chức năng Quản lý đơn đặt hàng của Admin



* Phân rã chức năng Quản lý danh sách các sản phẩm của Admin



# III) Phát triển hệ thống

Hệ thống được phát triển theo Spring MVC, một module trong **Spring Framework** được sử dụng phổ biến để phát triển các ứng dụng web theo mô hình **MVC (Model-View-Controller)**.

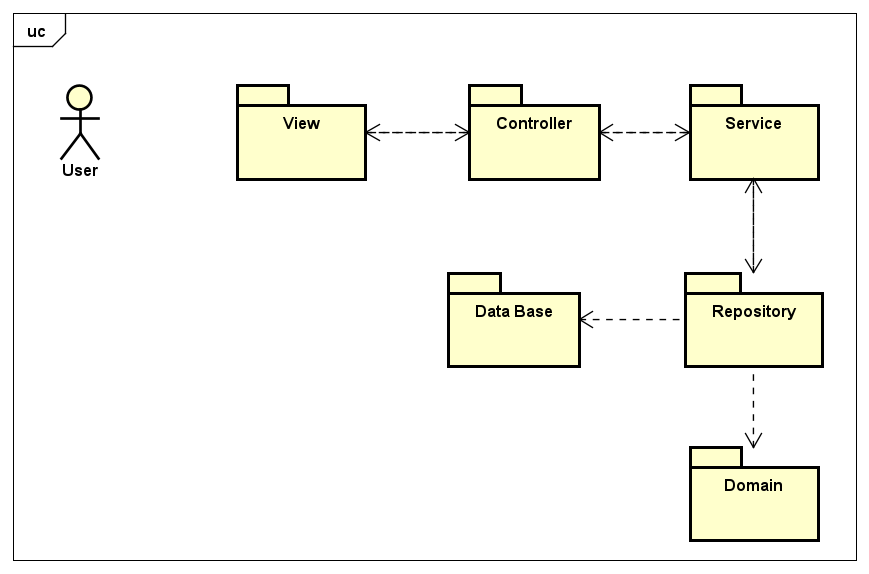
Spring MVC là một phần của Spring Framework, cung cấp các công cụ và cấu trúc để phát triển ứng dụng web một cách dễ dàng và hiệu quả. Nó tuân theo mô hình **Model-View-Controller**, giúp tách biệt rõ ràng giữa các thành phần **Logic xử lý (Controller)**, **Giao diện (View)** và **Dữ liệu (Model)**.

Các thành phần chính trong Spring MVC:

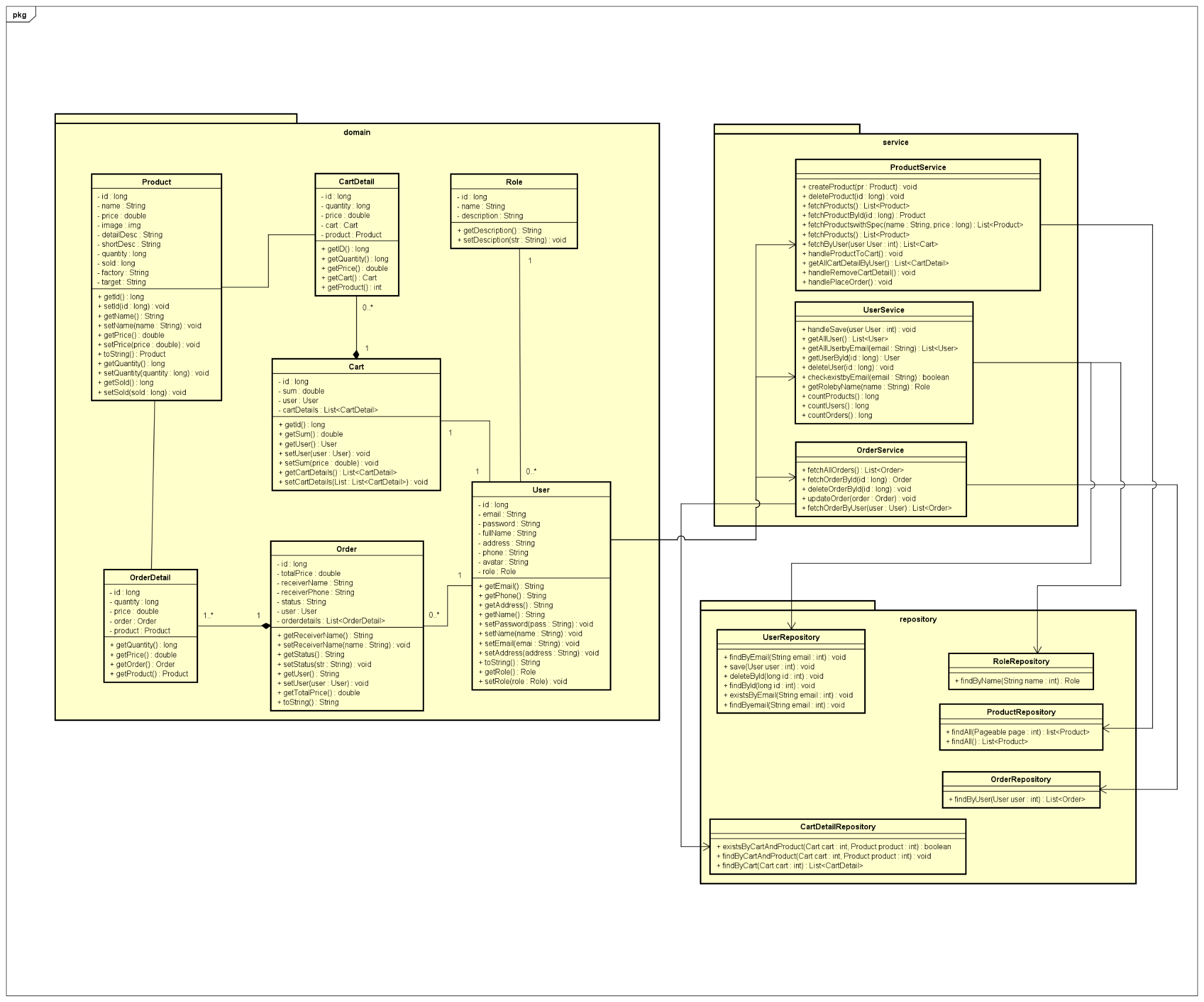
* **DispatcherServlet:** Là Front Controller chịu trách nhiệm điều phối tất cả các yêu cầu HTTP đến các Controller. Được định nghĩa trong file cấu hình Spring (XML hoặc Java).
* **Controller:** Xử lý logic ứng dụng. Nhận yêu cầu từ người dùng, tương tác với Model và trả về dữ liệu hoặc View.
* **Model**: Đại diện cho dữ liệu “các lớp”.
* **View:** Hiển thị dữ liệu cho người dùng, có thể sử dụng các công nghệ như Thymeleaf, JSP, FreeMarker, hoặc Mustache.

Trong dự án Quản lý bán hàng bọn em chia các thành phần chính theo các package để dễ quản lý cụ thể:

* **View(webApp):** để hiển thị giao diện người dùng.
* **Controller:** kiểm tra các yêu cầu từ User nếu hợp lệ chuyển đến Service.
* **Thư mục service:** chứa các lớp xử lý logic. Nếu là CRUD thì gọi đến các phương thức bên trong các lớp Repository, nếu là các phương thức phức tạp sẽ được triển khai bên trong Service.
* **Thư mục repository:** Chứa các interface để tương tác với cơ sở dữ liệu (thực hiện các câu lệnh truy vấn, CRUD).
* **Thư mục domain:** chứa các lớp đại diện cho dữ liệu vd: User, Cart, Order, Product, OrderDetail, CartDetail, Role.

****

# IV) Biểu đồ lớp

****

* **Chi tiết về biểu đồ lớp:**
  + **Lớp Cart và lớp CartDetail:** quan hệ 1-N.
  + **Lớp Order và lớp OrderDetail:** quan hệ 1-N.
  + **Lớp User và lớp Order:** quan hệ 1-N.
  + **Lớp User và Cart:** quan hệ 1-1.
  + **Lớp User và lớp Role:** quan hệ N-1. “mỗi tài khoản có một loại vai trò “Admin, client”, mỗi vai trò có thể có nhiều tài khoản.”
  + **Lớp OrderService:**
* fetchAllOrders(): lấy danh sách tất cả các đơn hàng
* fetchOrderById(): tìm đơn hàng theo ID
* deleteOrderById(): xóa một đơn hàng và toàn bộ chi tiết đơn hàng trong cơ sở dữ liệu liên quan đến nó.
* updateOrder(): cập nhật trạng thái của đơn hàng.
* fetchOrderByUser(): lấy danh sách các đơn hàng của một User.
* **Lớp UserService:**
* getAllUser(): lấy thông tin toàn bộ User trong cơ sở dữ liệu.
* getAllUserByEmail(): lấy danh sách tất cả User dựa vào email.
* getUserById(): tìm User thông qua id.
* deleteUser(): xóa một người dùng dựa vào id.
* getPolebyName(): xác định và trả về vai trò của 1 User.
* countUsers(): trả về số lượng User trong cơ sở dữ liệu.
* countProducts(): trả về số lượng sản phẩm của cửa hàng.
* countOrders(): trả về số lượng đơn hàng.
* **Lớp ProductService:**
  + creteProduct(): thêm sản phẩm mới.
  + fetchProducts(): lấy danh sách sản phẩm trong của hàng.
  + fetchProductById(): trả về sản phẩm theo Id.
  + deleteProduct(): xóa một sản phẩm khỏi của hàng dựa trên Id.
  + handleProductToCart(): thêm một sản phẩm vào giỏ hàng.
  + getAllCartDetailByUser(): lấy danh sách chi tiết giỏ hàng của User.
  + handleUpdateCartBeforeCheckou(): Cập nhật số lượng các sản phẩm trong giỏ hàng.
  + handlePlaceOrder(): Xử lý đơn đặt hàng.

# V) Đánh giá trong nhóm

|  |  |  |
| --- | --- | --- |
| STT | Tên thành viên | Phần trăm đóng góp |
| 1 | Phạm Công Hào 20215045 | 20% |
| 2 | Tống Chí Quang 20210725 | 20% |
| 3 | Nguyễn Trung Sơn 20215132 | 20% |
| 4 | Nguyễn Phúc Mạnh 20215087 | 20% |
| 5 | Phạm Xuân Phong 20215115 | 20% |

# 