**Kế hoạch PT&TKPM**

**Hệ thống Quản lý Cửa hàng Thiết bị Điện tử trực tuyến**

**Tuần 1: Phân tích đề bài**

# **Actor**

* **Khách hàng(Customer)**: Người dùng cuối, mua sản phẩm
* **Quản trị viên(Admin)**: Quản lý toàn bộ hệ thống
* **Cổng thanh toán (Payment Gateway)**: Xử lý giao dịch thanh toán online

1. **Use Case**

## **2.1. Khách hàng (Customer)**

* **Chức năng:**
* **Đăng ký/Đăng nhập tài khoản**: Tạo và quản lý hồ sơ cá nhân.
* **Tìm kiếm/lọc sản phẩm**: Lọc theo loại, thương hiệu, giá, khuyến mãi.
* **Xem chi tiết sản phẩm**: Hình ảnh, mô tả, đánh giá, tình trạng tồn kho.
* **Quản lý giỏ hàng**: Thêm, sửa, xóa sản phẩm trong giỏ hàng.
* **Đặt hàng trực tuyến**: Chọn địa chỉ giao hàng, phương thức thanh toán.
* **Thanh toán**: Kết nối với Payment Gateway.
* **Theo dõi đơn hàng**: Kiểm tra trạng thái đơn hàng.
* **Lịch sử đơn hàng**: Xem các đơn hàng đã mua.
* **Yêu cầu hỗ trợ**: Gửi yêu cầu hỗ trợ hoặc chat với CSKH.
* **Đánh giá sản phẩm và dịch vụ**: Để lại nhận xét, đánh giá sao.
* **Nhận thông báo**: Nhận thông báo về các sự kiện.

## **2.2. Quản trị viên(Admin)**

* **Chức năng:**
* **Đăng nhập hệ thống quản trị:**
* Truy cập hệ thống quản trị với quyền cao nhất để quản lý toàn bộ hoạt động cửa hàng**.**
* **Quản lý người dùng và tài khoản**:
* Xem danh sách tất cả khách hàng và nhân viên đã đăng ký tài khoản.
* Khóa hoặc mở khóa tài khoản khách hàng có hành vi gian lận.
* Chỉnh sửa thông tin tài khoản hoặc hỗ trợ khi khách bị mất mật khẩu.
* Theo dõi lịch sử mua hàng và lịch sử thanh toán.
* **Quản lý sản phẩm và kho**:
* Thêm, sửa, xóa sản phẩm với đầy đủ thông tin: tên, mô tả, hình ảnh, giá, thương hiệu, danh mục.
* Theo dõi và cập nhật số lượng tồn kho theo thời gian thực.
* Nhập kho sản phẩm mới từ nhà cung cấp và ghi nhận xuất kho khi có hàng lỗi hoặc trả hàng.
* Nhận cảnh báo khi sản phẩm sắp hết hàng hoặc tồn kho quá cao.
* **Quản lý đơn hàng**:
* Xem danh sách đơn hàng theo trạng thái: Chờ xác nhận, Đã xác nhận, Đang đóng gói, Đang giao hàng, Hoàn thành, Đã hủy.
* Xác nhận đơn hàng trước khi đóng gói và giao cho khách.
* Cập nhật trạng thái đơn hàng theo tiến trình xử lý.
* Hủy đơn hàng khi khách yêu cầu hoặc có sự cố.
* Xử lý trả hàng, hoàn tiền và liên kết với **Payment Gateway** để thực hiện giao dịch hoàn tiền.
* In hóa đơn, phiếu giao hàng và báo cáo liên quan đến đơn hàng.
* **Chăm sóc khách hàng:**
* Xem thông tin hồ sơ khách hàng và lịch sử mua hàng.
* Trả lời và xử lý các yêu cầu hỗ trợ của khách hàng qua hệ thống chat hoặc email.
* Ghi chú các thông tin quan trọng về khách hàng (khách VIP, khách thường xuyên khiếu nại, blacklist).
* Quản lý và duyệt các đánh giá sản phẩm, phản hồi đánh giá có nội dung vi phạm hoặc tiêu cực.
* **Quản lý thanh toán**:
* Theo dõi và xác nhận các giao dịch thanh toán của khách hàng.
* Xử lý hoàn tiền khi đơn hàng bị hủy hoặc khách trả hàng.
* Kết nối với **Payment Gateway** để đảm bảo thanh toán chính xác và minh bạch.
* **Quản lý khuyến mãi và marketing:**
* Tạo và quản lý mã giảm giá, voucher khuyến mãi.
* Thiết lập các chương trình giảm giá theo sản phẩm, thương hiệu hoặc toàn bộ giỏ hàng.
* **Báo cáo và thống kê**:
* Thống kê doanh thu theo ngày, tuần, tháng, sản phẩm hoặc danh mục.
* Báo cáo tồn kho: hàng sắp hết, hàng tồn nhiều, hàng bán chạy.
* Theo dõi tỷ lệ đơn hàng thành công, hủy, trả hàng.

## **2.3. Cổng thanh toán (Payment Gateway)**

* **Chức năng:**
* **Xử lý thanh toán trực tuyến:**
* Nhận yêu cầu thanh toán từ hệ thống khi khách hàng đặt hàng.
* Xác thực thông tin thanh toán (số thẻ, tài khoản ví điện tử, hạn mức giao dịch).
* Kiểm tra tính hợp lệ của giao dịch trước khi chuyển tiền.
* **Xác thực bảo mật giao dịch:**
* Hỗ trợ xác thực **OTP**, **3D Secure** hoặc **xác thực sinh trắc học** để chống gian lận.
* Mã hóa dữ liệu thanh toán bằng các chuẩn bảo mật **SSL/TLS** và **PCI DSS**.
* Phát hiện và chặn các giao dịch bất thường hoặc đáng ngờ.
* **Kết nối với ngân hàng và ví điện tử:**
* Liên kết với nhiều ngân hàng và dịch vụ ví điện tử để hỗ trợ thanh toán đa phương thức: Thẻ tín dụng/ghi nợ, Ví điện tử, Chuyển khoản ngân hàng trực tuyến.
* **Thông báo kết quả giao dịch:**
* Trả về kết quả thanh toán thành công hoặc thất bại cho hệ thống cửa hàng và khách hàng.
* Cung cấp mã giao dịch để theo dõi, đối soát về sau.
* **Hỗ trợ hoàn tiền (Refund):**
* Xử lý yêu cầu hoàn tiền từ hệ thống khi khách hàng hủy đơn hoặc trả hàng.
* Đảm bảo tiền được hoàn lại về đúng tài khoản ban đầu của khách hàng.
* **Ghi log và báo cáo giao dịch:**
* Lưu trữ toàn bộ lịch sử giao dịch để phục vụ đối soát, kiểm tra và chống gian lận.
* Cung cấp báo cáo tổng hợp về số lượng, giá trị, và trạng thái giao dịch.

## **Yêu cầu phi chức năng**

## **3.1. Hiệu suất**

* Trang web tải trong dưới 5 giây khi có < 1000 người dùng truy cập cùng lúc.
* Hệ thống có thể xử lý tối thiểu 500 đơn hàng/phút.
* Chịu tải flash sale/pre-order

## **3.2. Bảo mật**

* Mọi dữ liệu nhạy cảm (mật khẩu, thanh toán) phải được mã hóa.
* Tuân thủ quyền riêng tư; hóa đơn điện tử; chính sách BH/đổi trả
* TLS 1.2+; dữ liệu nhạy cảm mã hoá at-rest AES-256.

## **3.3. Khả năng mở rộng**

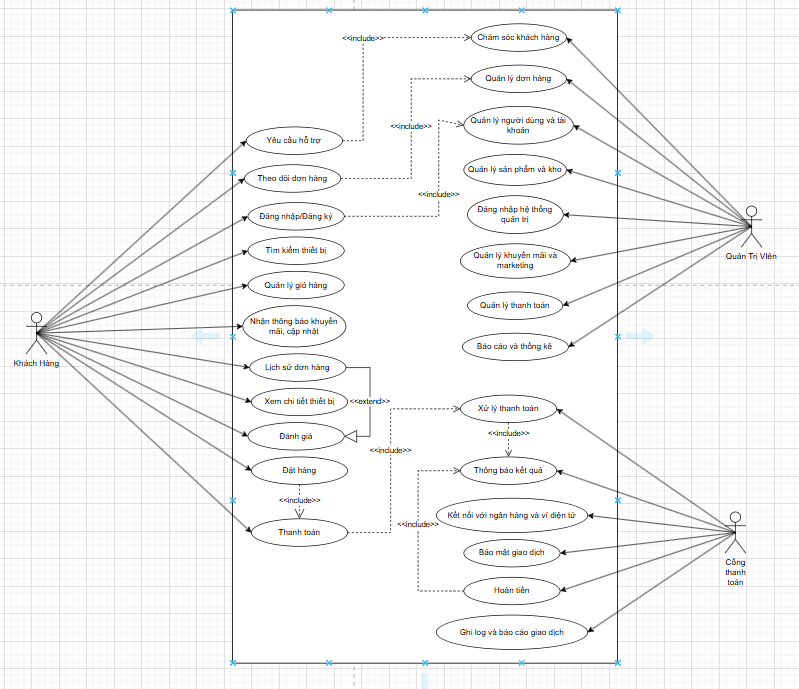
* Có thể nâng cấp để hỗ trợ gấp đôi số lượng sản phẩm và khách hàng trong vòng 6 tháng mà không ảnh hưởng hiệu suất.

## **3.4. Khả năng sử dụng**

* Giao diện thân thiện, người mới có thể đặt hàng trong < 2 phút.

**Tuần 2: Mô hình hóa Use Case và Kịch bản**

**Use Case Diagram**

****

## Kịch Bản Chi Tiết :

Các Trường Hợp Sử Dụng (Use Case) Quan trọng:

1. Đặt Hàng
2. Thanh Toán
3. Hủy Đơn Hàng & Hoàn Tiền

### 1. Trường Hợp Sử Dụng: Đặt Hàng

#### Kịch Bản Chính 1.0: Mua Sắm Kết Hợp (Thành công)

Mục tiêu: Khách hàng tạo một đơn hàng kết hợp Laptop và Điện thoại thành công.

* Truy cập và Lựa chọn: Khách hàng đã đăng nhập tìm kiếm và chọn mua Điện thoại đời mới cùng với Máy tính xách tay cấu hình cao.
* Kiểm tra Tồn kho và Cấu hình:
  + Khách hàng xem trang chi tiết. Hệ thống xác nhận cả hai sản phẩm đều hiển thị "Còn hàng" và Khách hàng chọn các tùy chọn cấu hình phù hợp (màu sắc, dung lượng).
* Thao tác Giỏ hàng và Giữ hàng: Khách hàng thêm cả hai sản phẩm vào giỏ.
  + Điểm Kiểm soát Tồn kho: Hệ thống ngay lập tức tạo yêu cầu giữ hàng tạm thời (khoảng 45 phút) cho 1 đơn vị tồn kho của mỗi sản phẩm. Tồn kho khả dụng trên hệ thống bị giảm đi.
* Thủ tục Đặt hàng: Khách hàng kiểm tra Tổng giá trị đơn hàng, bao gồm cả thuế VAT, và nhấn "Tiến hành Thanh toán".
* Điền Thông tin Giao nhận: Khách hàng xác nhận thông tin nhận hàng, chọn phương thức vận chuyển "Nhanh" và chấp nhận phí vận chuyển.
* Tạo Đơn hàng: Khách hàng nhấn "Xác nhận Đặt hàng".
  + Phản hồi Hệ thống: Hệ thống ghi nhận đơn hàng #DH9001.
  + Trạng thái đơn hàng được thiết lập là "Chờ thanh toán". Dữ liệu đơn hàng được khóa và chuyển sang Trường Hợp Sử Dụng Thanh Toán.

#### Kịch Bản Phụ 1.1: Lỗi Kiểm Tra Tồn Kho (Hết hàng đột ngột)

Mục tiêu: Xử lý khi sản phẩm đã hết hàng ngay trước thời điểm Khách hàng xác nhận mua.

* Tình huống: Khách hàng chọn mua Máy tính X (chỉ còn 1 chiếc). Cùng lúc đó, giao dịch thanh toán từ Khách hàng khác cho chiếc máy cuối cùng này được hoàn tất.
* Hành động của Khách hàng: Khách hàng nhấn "Xác nhận Đặt hàng" cho Máy tính X.
* Phản hồi Hệ thống (Kiểm định): Hệ thống kiểm tra tồn kho tại thời điểm tạo đơn, phát hiện số lượng tồn kho bằng 0.
* Thông báo Lỗi Nghiêm trọng: Hệ thống hiển thị cảnh báo: "Rất tiếc! Sản phẩm [Máy tính X] vừa được bán hết. Vui lòng xóa sản phẩm này khỏi giỏ hàng hoặc chọn mẫu thay thế."
* Hệ quả: Việc tạo đơn hàng bị chặn. Khách hàng không thể tiếp tục với sản phẩm đã hết hàng.

### 2. Trường Hợp Sử Dụng: Thanh Toán

#### Kịch Bản Chính 2.0: Thanh toán Thẻ Tín dụng (Thành công)

Mục tiêu: Khách hàng thanh toán thành công đơn hàng #DH9001 bằng thẻ tín dụng.

* Chọn Phương thức: Khách hàng chọn "Thanh toán bằng Thẻ Tín dụng/Ghi nợ".
* Xác thực và Giao dịch: Khách hàng nhập thông tin thẻ.
  + Cổng thanh toán (CTTT) tiến hành xác thực bảo mật (ví dụ: OTP) và gửi yêu cầu giao dịch tới Ngân hàng phát hành.
* Kiểm soát Giao dịch: CTTT nhận phản hồi "Giao dịch Được chấp thuận" từ Ngân hàng.
* Cập nhật Trạng thái & Kho:
  + Hệ thống cập nhật trạng thái đơn hàng #DH9001 thành "Đã thanh toán, Chờ xử lý".
  + Lượng hàng đã được trừ kho vĩnh viễn và lệnh giữ hàng tạm thời bị hủy.
  + Hệ thống tự động gửi Email/Tin nhắn Xác nhận Thanh toán kèm theo hóa đơn điện tử.

#### Kịch Bản Phụ 2.1: Thanh Toán Thất Bại (Thẻ bị Từ chối do Số dư)

Mục tiêu: Xử lý trường hợp thẻ không đủ tiền thực hiện giao dịch.

* Tình huống: Khách hàng cố gắng thanh toán đơn hàng mua Điện thoại trị giá 30 triệu đồng.
* Sự cố: Tài khoản ngân hàng của Khách hàng không đủ số dư.
* Phản hồi Lỗi: CTTT nhận mã lỗi "Không đủ tiền trong tài khoản" từ Ngân hàng.
* Thông báo Lỗi: Hệ thống hiển thị thông báo rõ ràng: "Thanh toán không thành công. Thẻ của quý khách không đủ số dư. Vui lòng kiểm tra lại thẻ hoặc chọn phương thức thanh toán khác."
* Hành động Khắc phục: Khách hàng chọn phương thức thanh toán "Thanh toán khi nhận hàng (COD)".
* Hệ quả: Đơn hàng được cập nhật thành "Chờ xử lý, Thanh toán COD", và tiến trình xử lý đơn hàng được tiếp tục.

### 3. Trường Hợp Sử Dụng: Hủy Đơn Hàng & Hoàn Tiền

#### Kịch Bản Chính 3.0: Hủy Đơn Hàng (Đã Thanh toán) và Hoàn tiền thành công

Mục tiêu: Khách hàng hủy đơn hàng đã thanh toán và được hoàn tiền tự động.

* Yêu cầu Hủy: Khách hàng truy cập đơn hàng #DH9004 (trạng thái: "Đã thanh toán, Chờ đóng gói") và nhấn "Hủy đơn hàng".
* Kiểm soát Trạng thái: Hệ thống kiểm tra: Đơn hàng CHƯA được bàn giao cho đơn vị vận chuyển → Cho phép hủy.
* Cập nhật và Hoàn kho: Khách hàng xác nhận lý do hủy.
  + Trạng thái đơn hàng cập nhật thành "Đã Hủy".
  + Hệ thống hoàn trả tồn kho ngay lập tức và tự động kích hoạt lệnh Hoàn tiền thông qua CTTT.
* Phản hồi Hoàn tiền: Gửi email xác nhận: "Yêu cầu hủy đơn hàng đã hoàn tất. Số tiền [X đồng] sẽ được hoàn trả về tài khoản/thẻ của quý khách trong 5-7 ngày làm việc."

#### Kịch Bản Phụ 3.1: Chặn Hủy Đơn Hàng (Đã Xuất kho)

Mục tiêu: Ngăn chặn việc hủy khi hàng đã rời khỏi kho và đang trên đường giao.

* Tình huống: Khách hàng muốn hủy đơn hàng #DH9005 (mua Máy tính).
* Trạng thái Hiện tại: Đơn hàng đã được xuất kho (trạng thái: "Đang giao hàng" - mã vận đơn đã được gán).
* Phản hồi Hệ thống: Khi Khách hàng truy cập, nút "Hủy đơn hàng" bị ẩn hoặc bị vô hiệu hóa.
* Thông báo & Hướng dẫn: Hệ thống hiển thị thông báo: "Đơn hàng của quý khách đang được vận chuyển. Vui lòng không hủy. Nếu không còn nhu cầu, quý khách có thể từ chối nhận hàng khi đơn vị vận chuyển giao tới, hoặc tiến hành trả hàng sau khi nhận theo quy định."

**Tuần 3: Thiết kế Lớp và Tạo cơ sở code**

# **Các lớp đối tượng, thuộc tính và phương thức**

## 1.1. Người dùng (User)

Đây là lớp **cha** cho tất cả người dùng (khách hàng, quản trị viên).

**Thuộc tính:**

* name (String): Họ tên người dùng.
* email (String): Địa chỉ email.
* password (String): Mật khẩu đăng nhập.
* phone (String): Số điện thoại.
* address (String): Địa chỉ mặc định.
* status (boolean): Trạng thái tài khoản (Active/Locked).
* createdAt (Date): Ngày tạo tài khoản.
* role (String): Vai trò.

**Phương thức:**

* createUser(userInfo: Object): Tạo tài khoản mới (nhân viên, admin, customer).
* login(email, password): Đăng nhập.
* logout(): Đăng xuất.
* changePassword(password, newPassword): Đổi mật khẩu.
* updateProfile(name: String, phone: String, address: String): Cập nhật thông tin cá nhân.

## 1.2. **Khách hàng (Customer)**

**Đối tượng: Customer lớp con của User**

**Thuộc tính (Attributes):**

* customerID (Int): Mã định danh khách hàng.
* addresses (List<String>): Địa chỉ giao hàng mặc định hoặc danh sách nhiều địa chỉ.
* paymentMethods (List<String>): Các phương thức thanh toán đã lưu (thẻ, ví điện tử...)

**Phương thức (Methods):**

* searchProduct(filters: Object): Tìm kiếm hoặc lọc sản phẩm.
* viewProductDetail(productID: String): Xem chi tiết sản phẩm.
* manageCart(action: String, productID: String, quantity: Int): Thêm/Sửa/Xóa sản phẩm trong giỏ hàng.
* placeOrder(orderInfo: Object): Đặt hàng trực tuyến.
* trackOrder(orderID: String): Theo dõi trạng thái đơn hàng.
* viewOrderHistory(): Xem lịch sử mua hàng.
* requestSupport(issueInfo: Object): Gửi yêu cầu hỗ trợ hoặc chat với CSKH.
* rateProduct (productID: String, rating: Int, comment: String): Đánh giá sản phẩm.
* receiveNotification(notification: Object): Nhận thông báo hệ thống.
* applyDiscountCode(code: String): Áp dụng mã khuyến mãi khi đặt hàng.

1.3. **Quản trị viên (Admin)**

**1.3.1. Lớp Admin (Kế thừa từ User)**

Lớp này chỉ giữ thông tin quản trị, quyền hạn và liên kết đến các nhóm chức năng.

**Thuộc tính:**

* adminID (int): Mã quản trị viên.

**1.3.2. Các lớp quản lý chuyên biệt**

Theo nguyên tắc Single Responsibility Principle (SRP), mỗi nhóm chức năng được tách thành một lớp riêng.

\* **Lớp UserManagement (Quản lý người dùng)**

Dùng để quản lý khách hàng hoặc nhân viên.

**Thuộc tính:**

* customers (List<Customer>): Danh sách khách hàng.
* admins (List<Admin>): Danh sách quản trị viên.

**Phương thức:**

* getCustomerProfile(customerID: Int): Lấy thông tin chi tiết khách hàng .
* updateCustomerProfile(customerID: Int, newProfile: Object): Cập nhật thông tin khách hàng.
* deleteCustomerProfile(customerID: Int): Xóa thông tin khách hàng.
* lockCustomerAccount(userID: Int): Khóa tài khoản khách hàng.
* unlockCustomerAccount(userID: Int): Mở khóa tài khoản khách hàng.
* getAdminProfile(adminID: Int): Lấy thông tin chi tiết quản trị viên.
* updateAdminProfile(adminID: Int, newProfile: Object): Cập nhật thông tin quản trị viên.
* deleteAdminProfile(adminID: Int): Xóa thông tin quản trị viên.

\* **Lớp ProductManagement (Quản lý sản phẩm)**

**Thuộc tính:**

- products (List<Product>): Danh sách sản phẩm trong hệ thống.

- categories (List<Category>): Danh mục sản phẩm.

***Phương thức:***

* addProduct(productInfo: Object): Thêm sản phẩm mới.
* updateProduct(productID: String, newInfo: Object): Sửa thông tin sản phẩm.
* deleteProduct(productID: String): Xóa sản phẩm.
* getProduct(productID: String): Lấy thông tin chi tiết sản phẩm.
* searchProduct(criteria: Object): Tìm kiếm/lọc sản phẩm.

**\* Lớp InventoryManagement (Quản lý kho hàng)**

**Thuộc tính:**

- products (List<Product>): Danh sách sản phẩm trong hệ thống.

- inventory (Inventory): Quản lý tồn kho chi tiết.

***Phương thức:***

* **getLowStockProducts()**: Lấy danh sách các sản phẩm có số lượng tồn kho thấp.
* checkProductAvailability(productID: String): Kiểm tra hàng còn hay hết.
* alertLowStock(): Cảnh báo sản phẩm sắp hết hàng.

**\* Lớp OrderManagement (Quản lý đơn hàng)**

***Thuộc tính:***

* orders (List<Order>): Danh sách đơn hàng.
* completedOrders (List<Order>): Danh sách đơn hàng đã hoàn tất.
* canceledOrders (List<Order>): Danh sách đơn hàng đã hủy.

***Phương thức:***

* viewOrders(status: String): Xem danh sách đơn hàng theo trạng thái.
* confirmOrder(orderID: String): Xác nhận đơn hàng.
* updateOrderStatus(orderID: String, newStatus: String ): Cập nhật trạng thái đơn hàng.
* cancelOrder(orderID: String ): Hủy đơn hàng.
* processReturnOrder(orderID: String): Xử lý trả hàng.
* getOrderDetail(orderID: String): Xem chi tiết đơn hàng (sản phẩm, khách hàng, thanh toán).
* searchOrders(criteria: Object): Tìm kiếm/lọc đơn hàng (theo ngày, khách hàng, trạng thái).
* assignShipment(orderID: String, shipmentInfo: Object): Gán thông tin vận chuyển (liên kết với lớp Shipment/Delivery).
* trackOrder(orderID: String): Theo dõi trạng thái vận chuyển.
* refundOrder(orderID: String): Hoàn tiền thông qua Payment Gateway.
* printInvoice(orderID: String): In hóa đơn hoặc phiếu giao hàng.

**\* Lớp CustomerSupportManagement (Chăm sóc khách hàng)**

**Thuộc tính:**

* supportRequests (List<SupportRequest>): Danh sách yêu cầu hỗ trợ.
* activeRequests (List<SupportRequest>): Danh sách yêu cầu đang xử lý.
* resolvedRequests (List<SupportRequest>): Danh sách yêu cầu đã xử lý.

**Phương thức:**

* getRequest() : lấy tất cả yêu cầu.
* removeSupportRequest(requestID: String) : Xóa yêu cầu khỏi hệ thống
* moderateReview(reviewID: Int): Duyệt hoặc xóa đánh giá vi phạm.

**\* Lớp PaymentManagement (Quản lý thanh toán)**

**Thuộc tính:**

* payments(List<Payment>): Danh sách giao dịch thanh toán.
* gateway(PaymentGateway): Cổng thanh toán .
* availableMethods(List<PaymentMethod>): Danh sách phương thức thanh toán.

**Phương thức:**

* createPayment(orderID: String, customerID: String, amount: Double, methodID: String): Xác nhận giao dịch thanh toán.
* processPayment(paymentID: String): Gọi PaymentGateway để xử lý thanh toán.
* refundPayment(paymentID: String, reason: String): Hoàn tiền một giao dịch.
* getPaymentHistory(customerID: String): Lấy lịch sử thanh toán của khách hàng.
* listAvailableMethods():Liệt kê tất cả phương thức thanh toán có sẵn.

***\** Lớp DiscountManagement (Quản lý khuyến mãi)**

Thuộc tính:

* discountCodes (List<Discount>): Danh sách mã giảm giá.
* activeDiscounts (List<Discount>): Danh sách mã đang có hiệu lực.
* expiredDiscounts (List<Discount>): Danh sách mã đã hết hạn.

Phương thức:

* createDiscountCode(discountCodes: Discount): Tạo mã giảm giá.
* updateDiscountCode(codeID: String, newInfo: Object ): Sửa thông tin mã giảm giá.
* deleteDiscountCode(codeID: String ): Xóa mã giảm giá.
* listActiveDiscounts(): Liệt kê các mã đang có hiệu lực.
* expireDiscount(codeID: String): Ngưng hiệu lực mã giảm giá.

**\* Lớp ReportManagement (Báo cáo và thống kê)**

**Thuộc tính:**

* reportsHistory (List<Report>): Lưu trữ lịch sử báo cáo đã tạo.

***Phương thức:***

* generateSalesReport(timeRange: String): Báo cáo doanh thu.
* generateStockReport(): Báo cáo tồn kho.
* generateOrderReport(): Báo cáo tỷ lệ đơn hàng thành công hoặc bị hủy.
* generateCustomerReport(timeRange: String): Báo cáo về khách hàng (khách hàng mới, khách hàng trung thành, tần suất mua hàng).
* generateDiscountReport(timeRange: String): Báo cáo hiệu quả các chương trình khuyến mãi.
* exportReport(reportID: String, format: String): Xuất báo cáo ra các định dạng (PDF, Excel).
* scheduleReport(type: String, frequency: String): Lên lịch tạo báo cáo định kỳ.

**1.2.4. Mối quan hệ giữa các lớp**

Admin có thể gọi đến các lớp quản lý thông qua composition:

* Admin → UserManagement
* Admin → ProductManagement
* Admin → InventoryManagement
* Admin → OrderManagement
* Admin → CustomerSupportManagement
* Admin → PaymentManagement
* Admin → DiscountManagement
* Admin → ReportManagement

**1.4. Sản phẩm (Product)**

Đối tượng: Product

***Thuộc tính (Attributes):***

* productID (String): Mã sản phẩm.
* name (String): Tên sản phẩm.
* description (String): Mô tả sản phẩm.
* price (Double): Giá bán.
* brand (String): Thương hiệu.
* category (String): Danh mục sản phẩm.
* stockQuantity (int): Số lượng tồn kho.
* discount (int): Giảm giá (nếu có).
* images (List<String>): Danh sách hình ảnh sản phẩm.
* status (String): Trạng thái (Đang bán/Ngừng bán).
* warranty (String): Thông tin bảo hành.

**Phương thức (Methods):**

* getProductInfo(): Lấy thông tin sản phẩm
* updateStock(quantity: int): Cập nhật tồn kho.
* applyDiscount(discountPercent: int): Áp dụng khuyến mãi.
* updateProductInfo(newInfo: Object): Sửa thông tin sản phẩm
* checkAvailability(): Boolean → Kiểm tra còn hàng không.
* calculateFinalPrice(): Double → Tính giá sau giảm giá.
* addImage(imageURL: String) → Thêm ảnh sản phẩm.
* removeImage(imageURL: String) → Xóa ảnh sản phẩm.

**1.5. Kho hàng (Inventory)**

**Đối tượng**: Inventory

**Thuộc tính (Attributes):**

* inventoryID(String): Mã đơn hàng.
* product (Product): Sản phẩm trong kho.
* quantity(Int): Số lượng sản phẩm trong kho.
* lastUpdated (Date): Ngày cập nhật kho.

**Phương thức (Methods):**

* increaseStock(product : Product,quantity : Int ): Tăng sản phẩm vào kho.
* decreaseStock(product : Product,quantity : Int ): Giảm sản phẩm vào kho.
* alertLowStock(productID: String): Cảnh báo khi số lượng tồn kho dưới mức an toàn.
* checkStock(productID: String): Kiểm tra số lượng tồn kho.

**1.6. Đơn hàng (Order)**

Đối tượng: Order

**Thuộc tính (Attributes):**

* orderID (String): Mã đơn hàng.
* customerID (int): Khách hàng đặt đơn.
* orderItems (List<OrderItem>): Danh sách sản phẩm trong đơn.
* totalAmount (Double): Tổng tiền đơn hàng.
* status (String): Trạng thái (Chờ xác nhận, Đang giao, Hoàn thành, Đã hủy...).
* paymentMethod (String): Phương thức thanh toán.
* shippingAddress (String): Địa chỉ giao hàng.
* createdAt (Date): Ngày đặt hàng.
* updatedAt (Date): Ngày cập nhật trạng thái.

**Phương thức (Methods):**

* updateStatus(newStatus: String): Cập nhật trạng thái đơn hàng.
* calculateTotal(): Tính tổng tiền đơn hàng.
* cancelOrder(): Hủy đơn hàng.
* generateInvoice(): Xuất hóa đơn.
* processPayment(): Xử lý thanh toán đơn hàng.
* refundOrder(): Hoàn tiền (liên kết PaymentManagement).
* trackShipment(): Theo dõi tình trạng giao hàng.

**1.7. Yêu cầu hỗ trợ (SupportRequest)**

**Thuộc tính (Attributes):**

* requestID(String): Mã yêu cầu.
* customerID(String): Mã khách hàng.
* title(String): Tiêu đề hoặc chủ đề yêu cầu (ví dụ: "Vấn đề thanh toán").
* description(String): Mô tả chi tiết vấn đề mà khách hàng gặp phải.
* status(String): Trạng thái yêu cầu: "Pending" (chờ xử lý), "Active" (đang xử lý), "Resolved" (đã giải quyết).
* createdAt(DateTime): Thời điểm khách hàng gửi yêu cầu.
* **resolvedAt**(DateTime): Thời điểm yêu cầu xử lý xong.

**Phương thức (Methods):**

* viewCustomerProfile(customerID: Int): Xem thông tin khách hàng.
* updateStatus(newStatus: String) : Cập nhật trạng thái yêu cầu (Pending → Active → Resolved).
* addResponse(response: String) : Gửi phản hồi cho khách hàng.
* getRequest() : Lấy thông tin yêu cầu.

**1.8. Thanh toán (Payment)**

**Thuộc tính (Attributes):**

* paymentID(String): Mã thanh toán.
* orderID(String): Mã đơn hàng liên quan đến thanh toán.
* customerID(String): Mã khách hàng thực hiện thanh toán.
* amount(Double):Số tiền thanh toán.
* method(PaymentMethod): Phương thức thanh toán được sử dụng.
* amount (String): Số tiền giao dịch.
* transactionDate(DateTime): Ngày giờ giao dịch.
* status (String): Trạng thái giao dịch (Thành công/Thất bại/Đang xử lý).
* transactionID(String):Mã giao dịch từ cổng thanh toán.

**Phương thức (Methods):**

* processPayment(): Xử lý thanh toán.
* refundPayment(reason: String): Hoàn tiền cho giao dịch.
* getPaymentDetails(): Lấy thông tin chi tiết thanh toán.

**1.9. Phương thức thanh toán (PaymentMethod)**

**Thuộc tính (Attributes):**

* methodID(String): Mã phương thức thanh toán.
* methodName(String): Tên phương thức thanh toán.

**Phương thức (Methods):**

* activate(): Bật phương thức thanh toán.
* deactivate(): Tắt phương thức thanh toán.
* getDetails(): Lấy thông tin phương thức thanh toán.

**1.10. Cổng thanh toán (PaymentGateway)**

**Thuộc tính (Attributes):**

* gatewayName(String): Tên cổng thanh toán (VNPay, PayPal, Stripe).
* apiKey(String):API key để kết nối dịch vụ.
* endpointURL(String): URL endpoint của cổng thanh toán.
* transactionID(String):Mã giao dịch từ cổng thanh toán.

**Phương thức (Methods):**

* connect(): Kết nối đến cổng thanh toán.
* sendPayment(payment: Payment): Gửi yêu cầu thanh toán ra ngoài.
* verifyTransaction(transactionID: String): Xác minh giao dịch.
* refundTransaction(transactionID: String): Hoàn tiền qua cổng thanh toán.

**1.11. Khuyến mãi (Discount)**

Đối tượng: Discount

**Thuộc tính (Attributes):**

* discountID (String): Mã của chương trình khuyến mãi
* code (String): Mã khuyến mãi .
* discountType (String): Loại giảm (Percent/Fixed).
* discountValue (Double): Giá trị giảm (% hoặc số tiền).
* maxDiscount (Double): Giảm tối đa.
* minOrderValue (Double): Giá trị đơn hàng tối thiểu.
* usageLimit (Int): Giới hạn số lần sử dụng.
* usedCount (Int): Số lần đã sử dụng.
* startDate (Date): Ngày bắt đầu.
* endDate (Date): Ngày kết thúc.
* applicableCategories (List<String>): Danh mục sản phẩm áp dụng.
* status (String): Hoạt động/Hết hạn/Ngưng hiệu lực.
* description(String): Mô tả chi tiết về khuyến mãi.

**Phương thức (Methods):**

* validatePromotion(code: String): Kiểm tra tính hợp lệ của mã.
* isExpired(): Boolean → Kiểm tra hết hạn.
* isApplicable(orderID: String): Boolean → Kiểm tra điều kiện áp dụng.
* calculateDiscount(orderAmount: Double): Double → Tính số tiền được giảm.
* getDiscountDetail(codeID: String): Lấy thông tin chi tiết mã giảm giá.
* validateDiscountCode(codeID: String, orderInfo: Object): Kiểm tra tính hợp lệ (còn hạn, đúng điều kiện áp dụng, còn lượt dùng).

# **1.12. Report(Báo cáo)**

**Attributes**

* reportID(int): Mã báo cáo.
* name (String): Tên báo cáo.
* reportType(String): Loại báo cáo: "Sales", "Stock", "Order", "Customer", "Discount".
* generatedAt(DateTime): Thời điểm báo cáo được tạo.
* timeRange(DateTime): Khoảng thời gian báo cáo (VD: "01/09/2025 - 30/09/2025").
* scheduledTime(DateTime): Thời gian báo cáo sẽ được tạo nếu lên lịch định kỳ.

**Methods**

* export(format: String): Xuất báo cáo ra file, trả về đường dẫn file.
* schedule(frequency: String, startTime: DateTime): Lên lịch tạo báo cáo định kỳ.
* getDetails():Trả về thông tin chi tiết báo cáo để hiển thị.

# **1.13. Category (Danh mục sản phẩm)**

**Attributes**

* categoryID (int): Mã danh mục.
* name (String): Tên danh mục.
* description (String): Mô tả danh mục.
* parentCategoryID (int): Mã danh mục cha.
* status (String): Trạng thái (ACTIVE/INACTIVE).
* imageURL (String): Đường dẫn ảnh đại diện.
* createdDate (Date): Ngày tạo.
* updatedDate (Date): Ngày cập nhật.

**Methods**

* createCategory (name: String, description: String, parentCategoryID: int, imageURL: String): Tạo danh mục mới.
* updateCategory (categoryID: int, name: String, description: String, parentCategoryID: int, status: String, imageURL: String): Cập nhật danh mục.
* deleteCategory (categoryID: int): Xóa danh mục.
* getCategoryInfo (categoryID: int): Lấy thông tin chi tiết danh mục.
* listSubCategories (parentCategoryID: int): Liệt kê các danh mục con.
* searchCategory (keyword: String): Tìm kiếm danh mục theo từ khóa.
* listProducts (categoryID: int): Liệt kê sản phẩm thuộc danh mục.

**1.14. Cart (Giỏ hàng của khách)**

**Attributes**

* cartID (int): Mã giỏ hàng.
* customerID (int): Mã khách hàng.

**Methods**

* addItem(productID: Int, quantity: Int): Thêm sản phẩm vào giỏ.
* removeItem(productID: Int): Xóa sản phẩm khỏi giỏ.
* updateItem(productID: Int, quantity: Int): Cập nhật số lượng sản phẩm.
* getCartItems(): Lấy danh sách sản phẩm trong giỏ.
* getItemCount(): Trả về số lượng sản phẩm trong giỏ.
* calculateTotal(): Tính tổng tiền giỏ hàng.
* applyPromotion(promoID: String): Áp dụng khuyến mãi.
* clearCart(): Xóa toàn bộ giỏ hàng.
* saveCart(): Lưu giỏ hàng.
* restoreCart(cartID: Int): Khôi phục giỏ hàng đã lưu.

**1.15.**  **CartItem (Sản phẩm trong giỏ hàng)**

**Đối tượng**: CartItem

**Thuộc tính (Attributes):**

* cartItemID (Int): Mã chi tiết giỏ hàng.
* cartID (Int): Mã giỏ hàng.
* product (Product): Sản phẩm.
* addedAt (Date): Ngày thêm vào giỏ.

**Phương thức (Methods):**

* updateQuantity(newQuantity: Int): Cập nhật số lượng.
* calculateSubtotal(): Tính thành tiền.
* getCartItemInfo(): Lấy thông tin chi tiết sản phẩm trong giỏ

## **1.16. OrderItem (Sản phẩm trong đơn hàng)**

**Đối tượng**: OrderItem

**Thuộc tính (Attributes):**

* orderItemID (Int): Mã chi tiết đơn hàng.
* orderID (String): Mã đơn hàng liên kết.
* product (Product): Sản phẩm.

**Phương thức (Methods):**

* calculateSubtotal(): Double → Tính thành tiền sau giảm giá.
* getOrderItemInfo(): Lấy thông tin chi tiết sản phẩm trong đơn.

## **1.17. Review (Đánh giá sản phẩm)**

**Thuộc tính (Attributes):**

* reviewID (Int): Mã định danh đánh giá.
* productID (String): Sản phẩm được đánh giá.
* customerID (Int): Khách hàng đánh giá.
* rating (Int): Điểm số (1–5).
* title (String): Tiêu đề ngắn cho đánh giá.
* comment (String): Nội dung đánh giá.
* images (List<String>): Hình ảnh kèm theo (nếu có).
* status (String): Trạng thái (Hiển thị/Ẩn/Đang duyệt).
* date (Date): Ngày đăng.

**Phương thức (Methods):**

* addReview(productID: String, customerID: Int, rating: Int, title: String, comment: String, images: List<String>): Tạo đánh giá mới.
* editReview(reviewID: Int, newRating: Int, newTitle: String, newComment: String, newImages: List<String>): Sửa đánh giá.
* deleteReview(reviewID: Int): Xóa đánh giá.
* approveReview(reviewID: Int): Duyệt đánh giá (cho admin).
* reportReview(reviewID: Int): Báo cáo đánh giá vi phạm.

**1.18.** **Notification (Thông báo)**

**Đối tượng**: Notification

**Thuộc tính (Attributes):**

* notificationID (Int): Mã thông báo.
* userID (Int): Người nhận thông báo.
* message (String): Nội dung thông báo.
* type (String): Loại thông báo (Đơn hàng, Thanh toán, Khuyến mãi, Hệ thống).
* status (Boolean): Trạng thái (Đã đọc/Chưa đọc).
* priority (String): Mức ưu tiên (Normal/High).
* createdAt (Date): Thời gian tạo.
* expiredAt (Date): Ngày hết hạn thông báo (nếu có).

**Phương thức (Methods):**

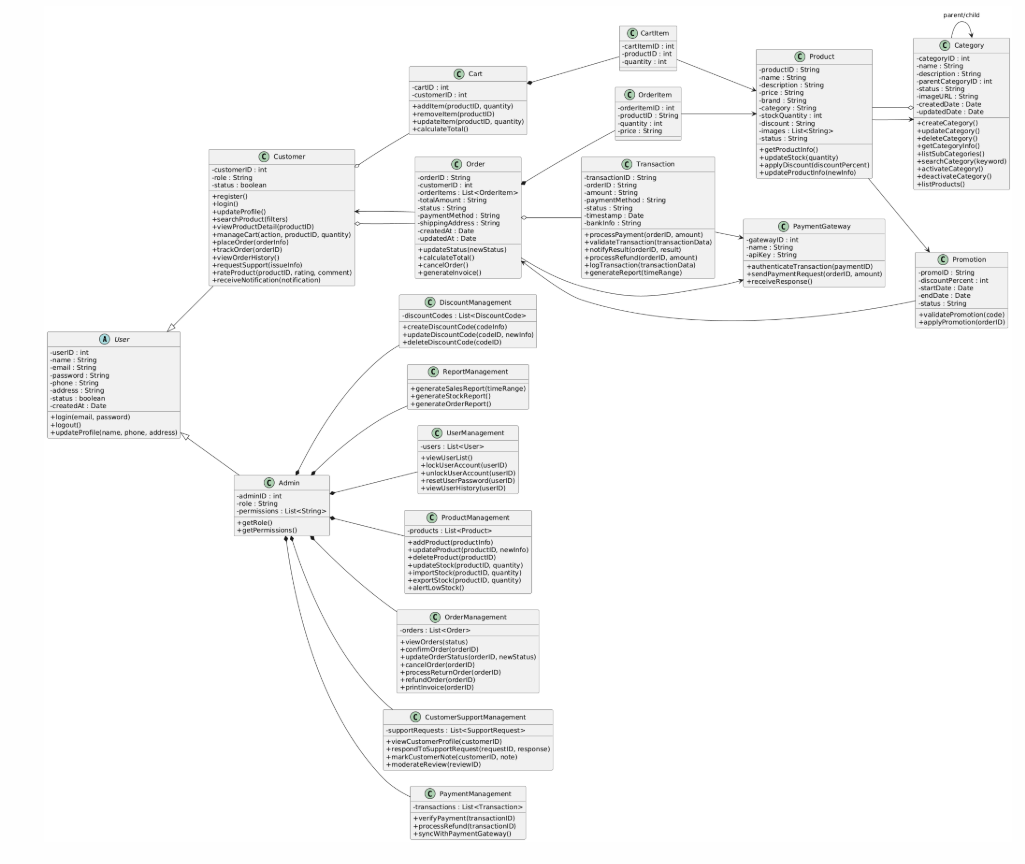
* sendNotification(userID: Int, message: String, type: String, priority: String): Gửi thông báo.
* markAsRead(notificationID: Int): Đánh dấu thông báo đã đọc.
* deleteNotification(notificationID: Int): Xóa thông báo.
* listUserNotifications(userID: Int): Liệt kê tất cả thông báo của một người dùng.

# **Mối liên hệ**

* Association (Kết hợp): Hai lớp có sự tương tác với nhau.
* Aggregation (Bao gồm – Has-A): Lớp A chứa lớp B nhưng B có thể tồn tại độc lập với A.
* Composition (Thành phần – Part-Of): Lớp A chứa lớp B, B không thể tồn tại nếu không có A.
* Inheritance (Kế thừa – Is-A): Lớp con kế thừa từ lớp cha.
* Các lớp người dùng có thể kế thừa từ lớp **User (abstract class)**

| **Lớp 1** | **Quan hệ** | **Lớp 2** | **Bội số** | **Ghi chú** |
| --- | --- | --- | --- | --- |
| **Order** | **Composition** | **OrderItem** | **1-N** | **Xóa Order ⇒ OrderItem bị xóa** |
| **Customer** | **Aggregation** | **Cart** | **1-1** | **Mỗi khách có đúng 1 giỏ** |
| **Cart** | **Composition** | **CartItem** | **0-N** | **Giỏ có thể trống; xóa Cart ⇒ CartItem bị xóa.** |
| **CartItem** | **Association** | **Product** | **N-1** | **N CartItem tham chiếu 1 Product** |
| **OrderItem** | **Association** | **Product** | **N-1** | **N OrderItem tham chiếu 1 Product.** |
| **Customer** | **Aggregation** | **Order** | **1/0-N** | **1 khách có thể chưa đặt đơn nào.** |
| **Order** | **Aggregation** | **Transacion** | **1/0-N** | **1 đơn có thể có nhiều giao dịch (thanh toán/hoàn tiền** |
| **PaymentGateway** | **Association (xử lý)** | **Transacion** | **1-N** | **1 cổng thanh toán xử lý nhiều giao dịch** |
| **Order** | **Association (sử dụng)** | **PaymentGateway** | **N-1** | **Mỗi Order dùng 1 cổng tại thời điểm trả tiền.** |
| **Promotion** | **Association (áp dụng)** | **Order** | **1/0-N** | **1 khuyến mãi áp dụng cho nhiều đơn; mỗi Order 0–1 Promotion** |
| **Promotion** | **Association (theo SP)** | **Product** | **N-M** | **Cần bảng nối PromotionProduct.** |
| **Category (cha)** | **Association (self)** | **Category (con)** | **1/0-N** | **Cây danh mục; con 0–1 cha** |
| **Category** | **Aggregation** | **Product** | **1-N** | **1 danh mục có nhiều sản phẩm** |
| **Product** | **Association** | **Category** | **N-1** | **1 sản phẩm thuộc 1 danh mục chính.** |
| **Admin** | **Composition** | **UserManagement** | **1-1** | **Admin sở hữu module** |
| **UserManagement** | **Aggregation** | **User** | **1-N** | **Quản lý nhiều người dùng** |
| **Admin** | **Composition** | **ProductManagement** | **1-1** |  |
| **ProductManagement** | **Aggregation** | **Product** | **1-N** |  |
| **Admin** | **Composition** | **OrderManagement** | **1-1** |  |
| **OrderManagement** | **Aggregation** | **Order** | **1-N** |  |
| **Admin** | **Composition** | **CustomerSupportManagement** | **1-1** |  |
| **CustomerSupportManagement** | **Aggregation** | **SupportRequest** | **1-N** |  |
| **Admin** | **Composition** | **PaymentManagement** | **1-1** |  |
| **PaymentManagement** | **Aggregation** | **Transaction** | **1-N** |  |
| **PaymentManagement** | **Association (đồng bộ)** | **PaymentGateway** | **1-N** | **Hệ thống có thể tích hợp nhiều cổng** |
| **Admin** | **Composition** | **DiscountManagement** | **1-1** |  |
| **DiscountManagement** | **Aggregation** | **DiscountCode** | **1-N** |  |
| **DiscountManagement** | **Aggregation/Association** | **Promotion** | **1–N** | **Quản lý nhiều chương trình KM.** |
| **Admin** | **Composition** | **ReportManagement** | **1-1** |  |
| **ReportManagement** | **Dependency (đọc dữ liệu)** | **Order / Product / Transaction** | **1-N** | **Dùng dữ liệu để lập báo cáo.** |

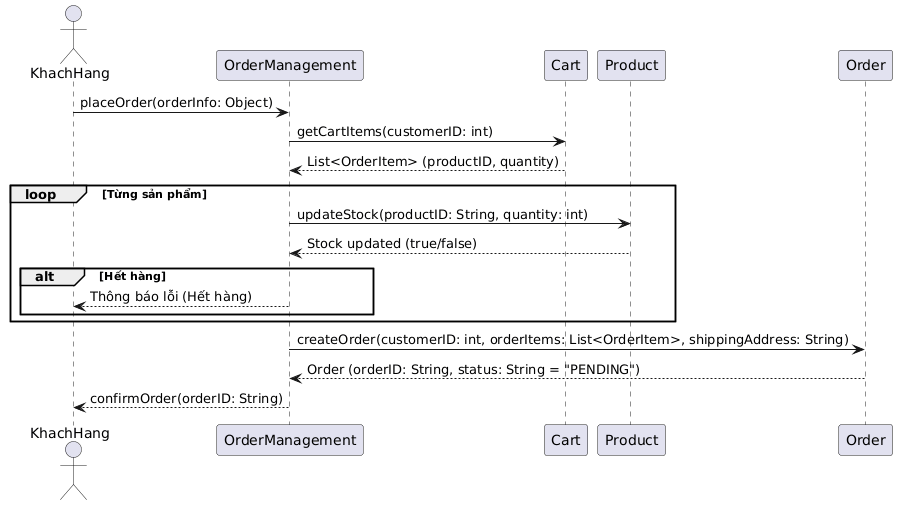
**Biểu đồ lớp**

****

**Tuần 4: Thiết kế Tương tác và Giao diện người dùng (UI)**

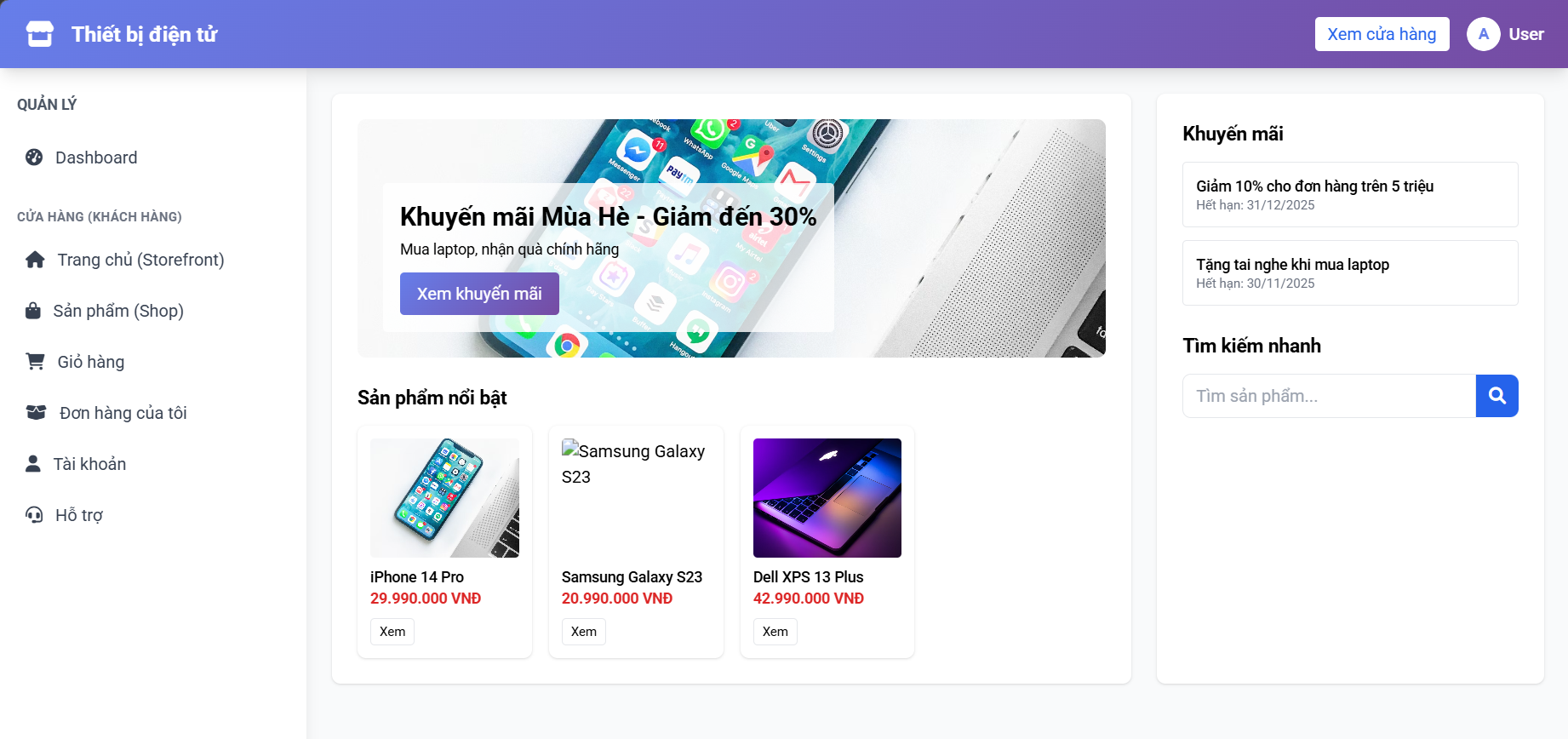
# **Biểu đồ trình tự (Sequence Diagram)**

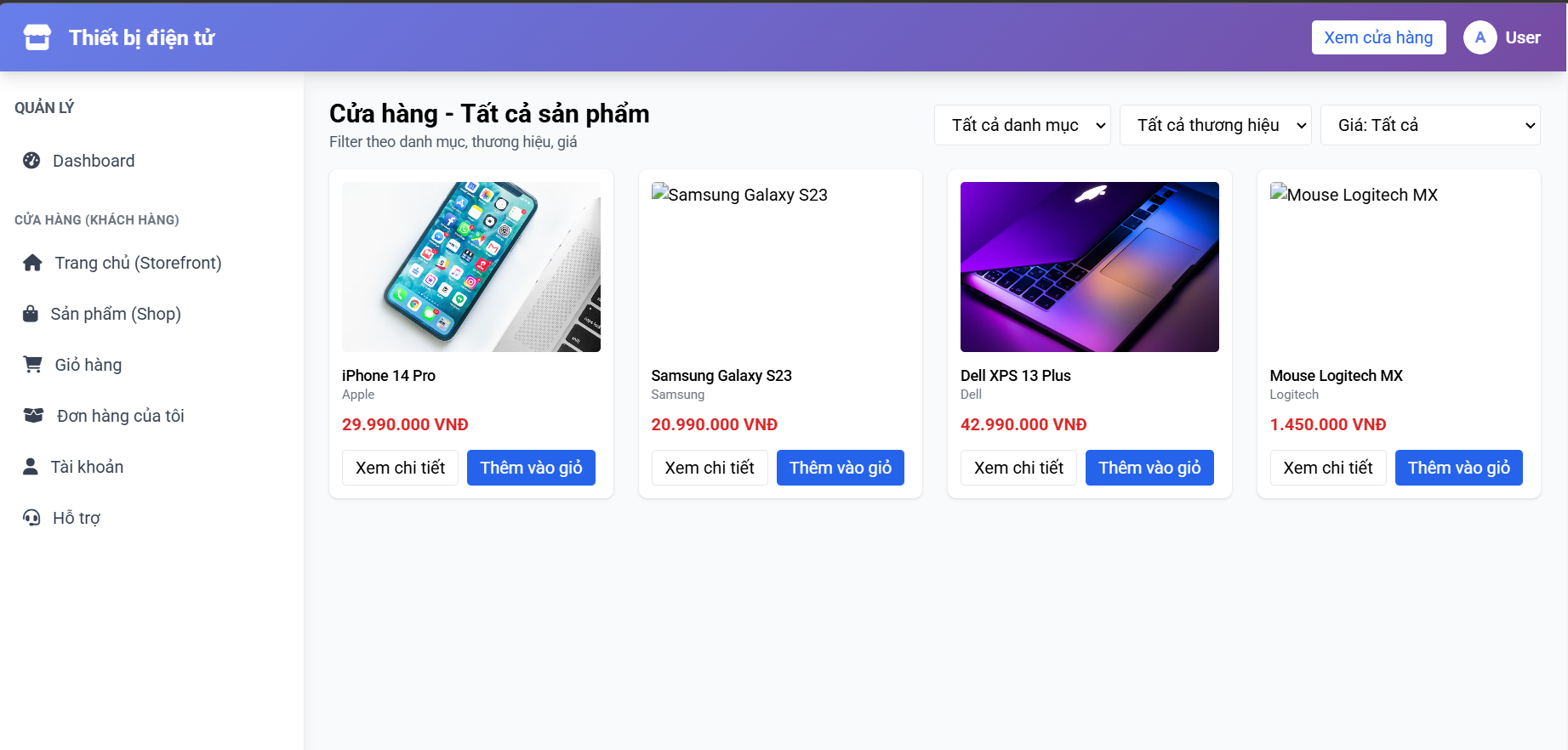
Đặt hàng:



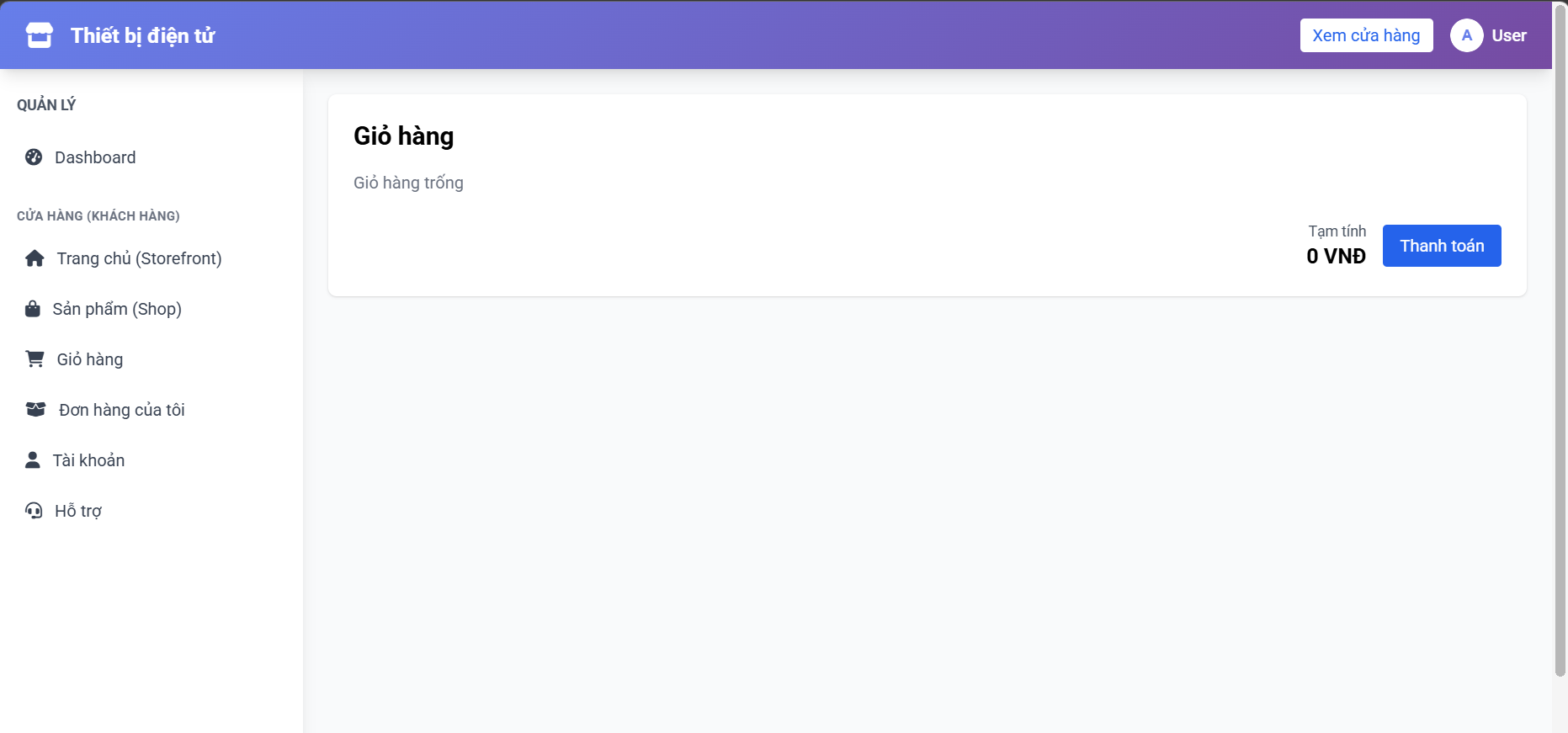
# **Thiết kế giao diện**

2.1. Giao diện người dùng

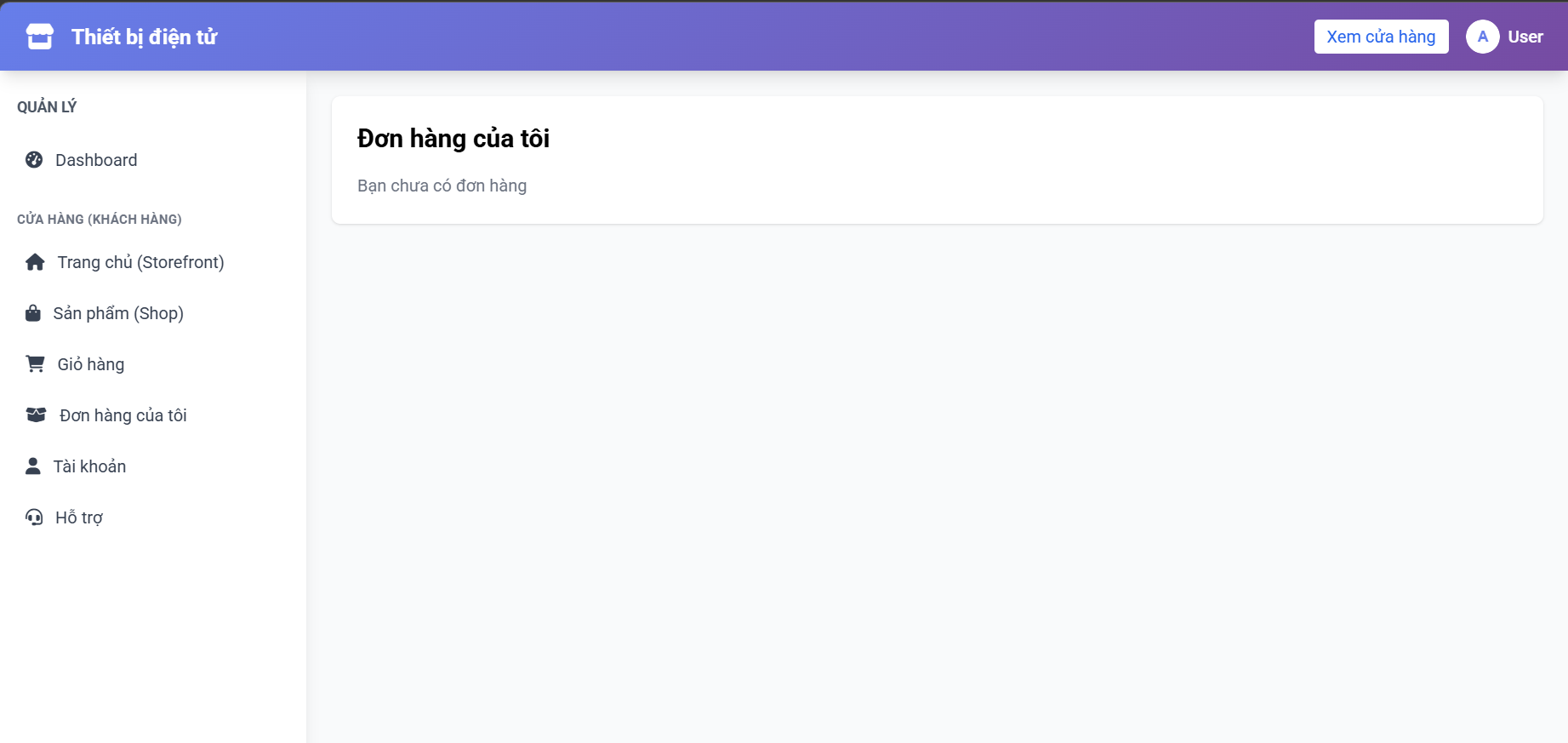


2.1.1. Giao diện xem các sản phẩm của người dùng

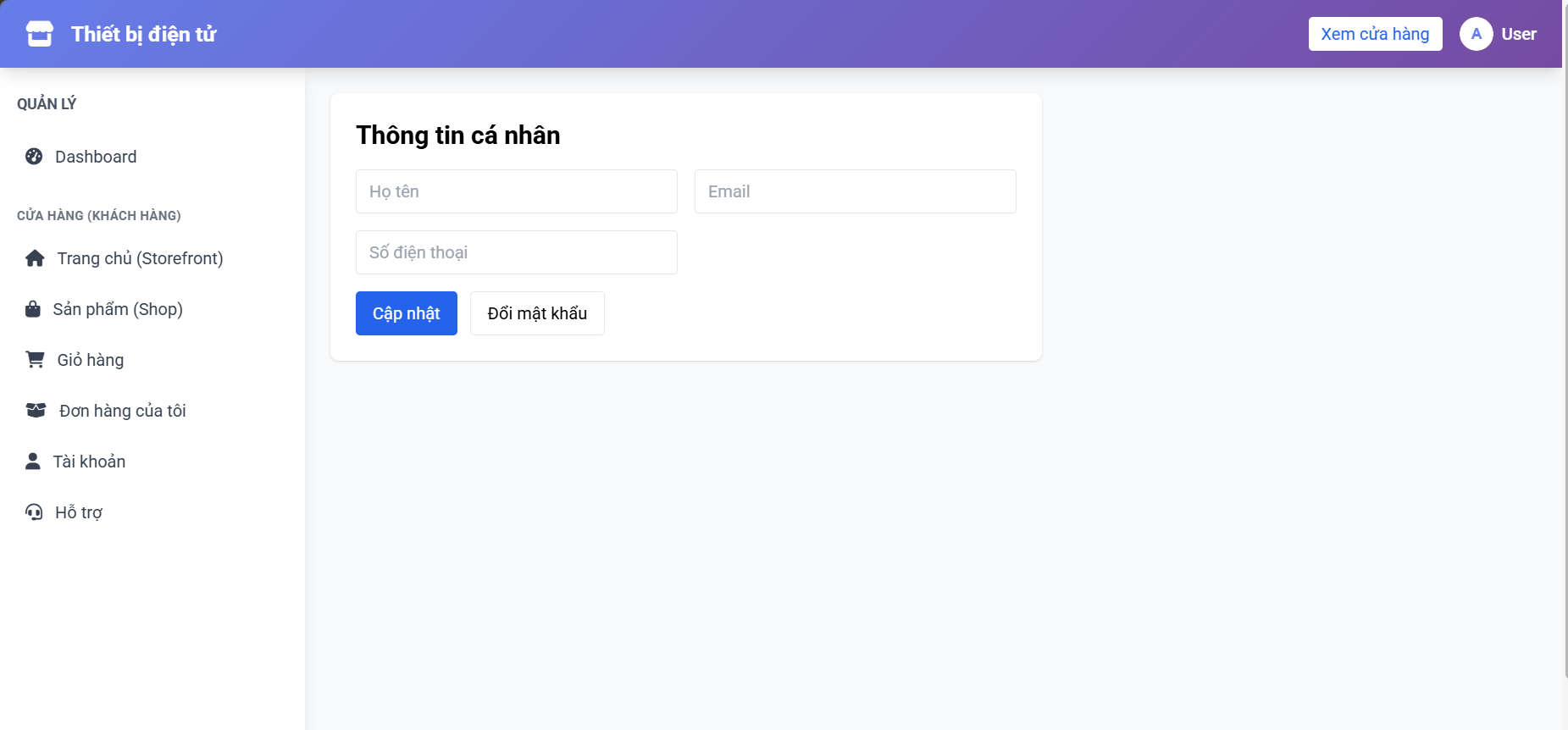
2.1.2. Giao diện kiểm tra giỏ hàng của người dùng



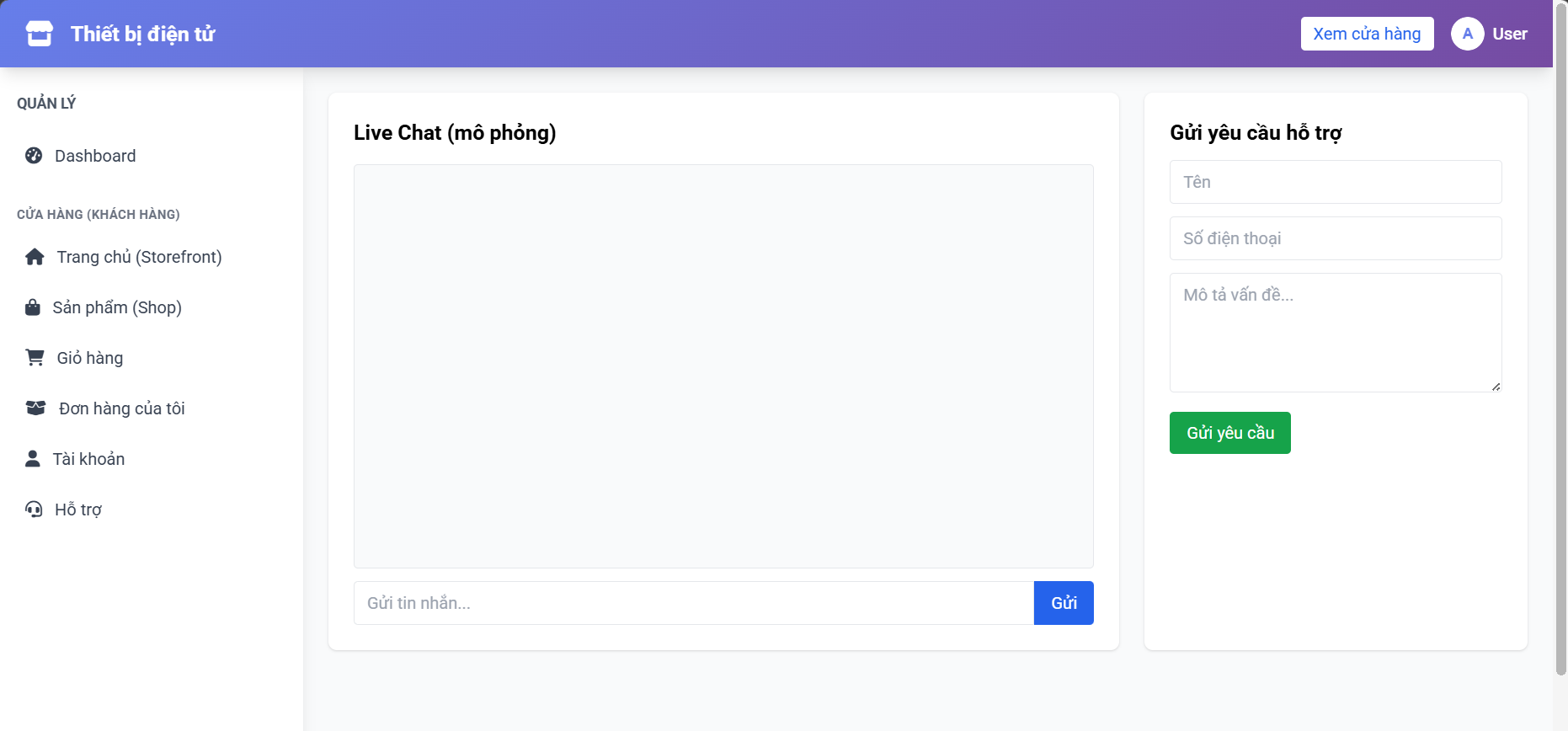
2.1.3. Giao diện kiểm tra đơn hàng của người dùng



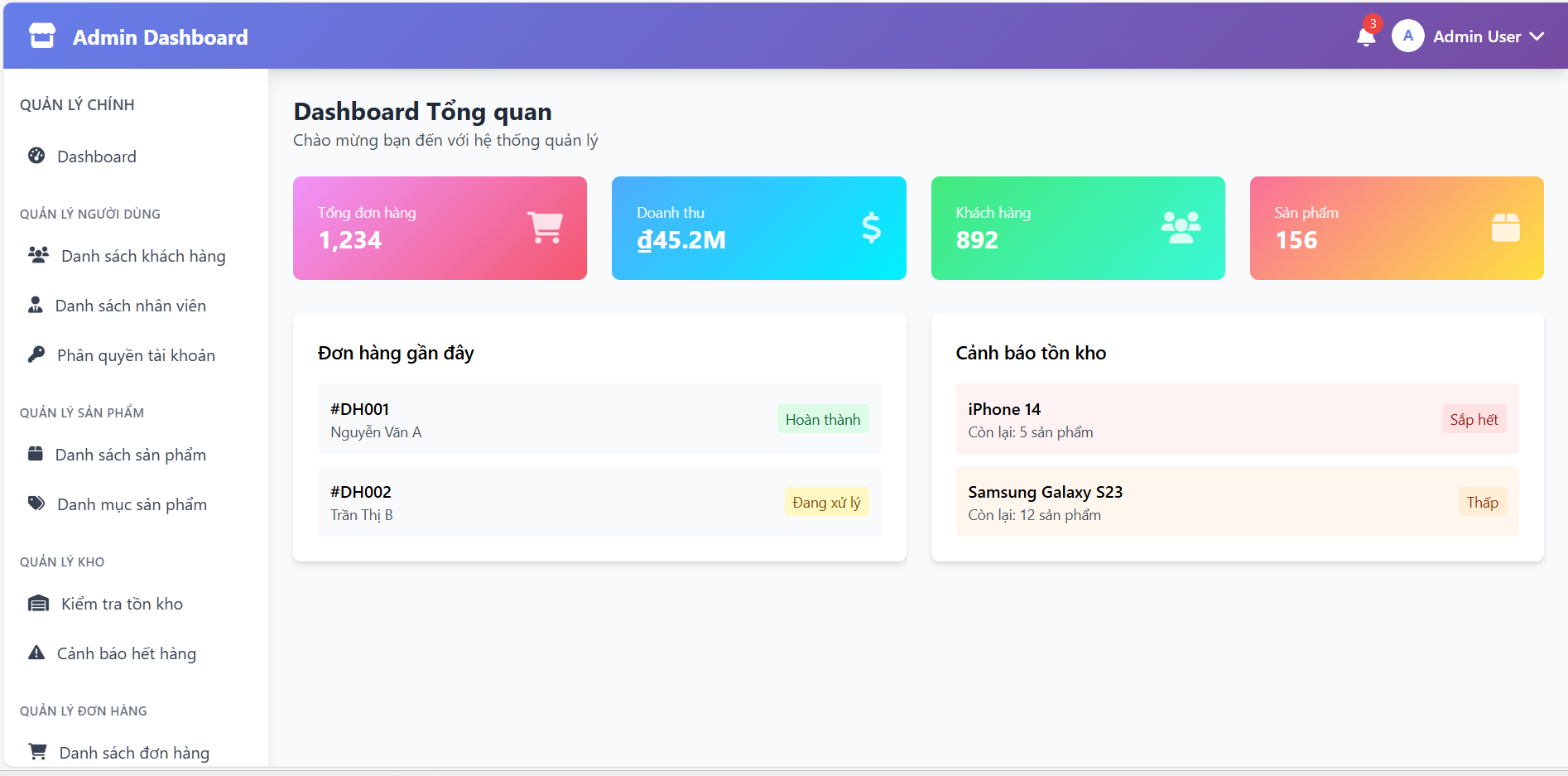
2.1.4. Giao diện cập nhập các thông tin bảo mật của khách hàng

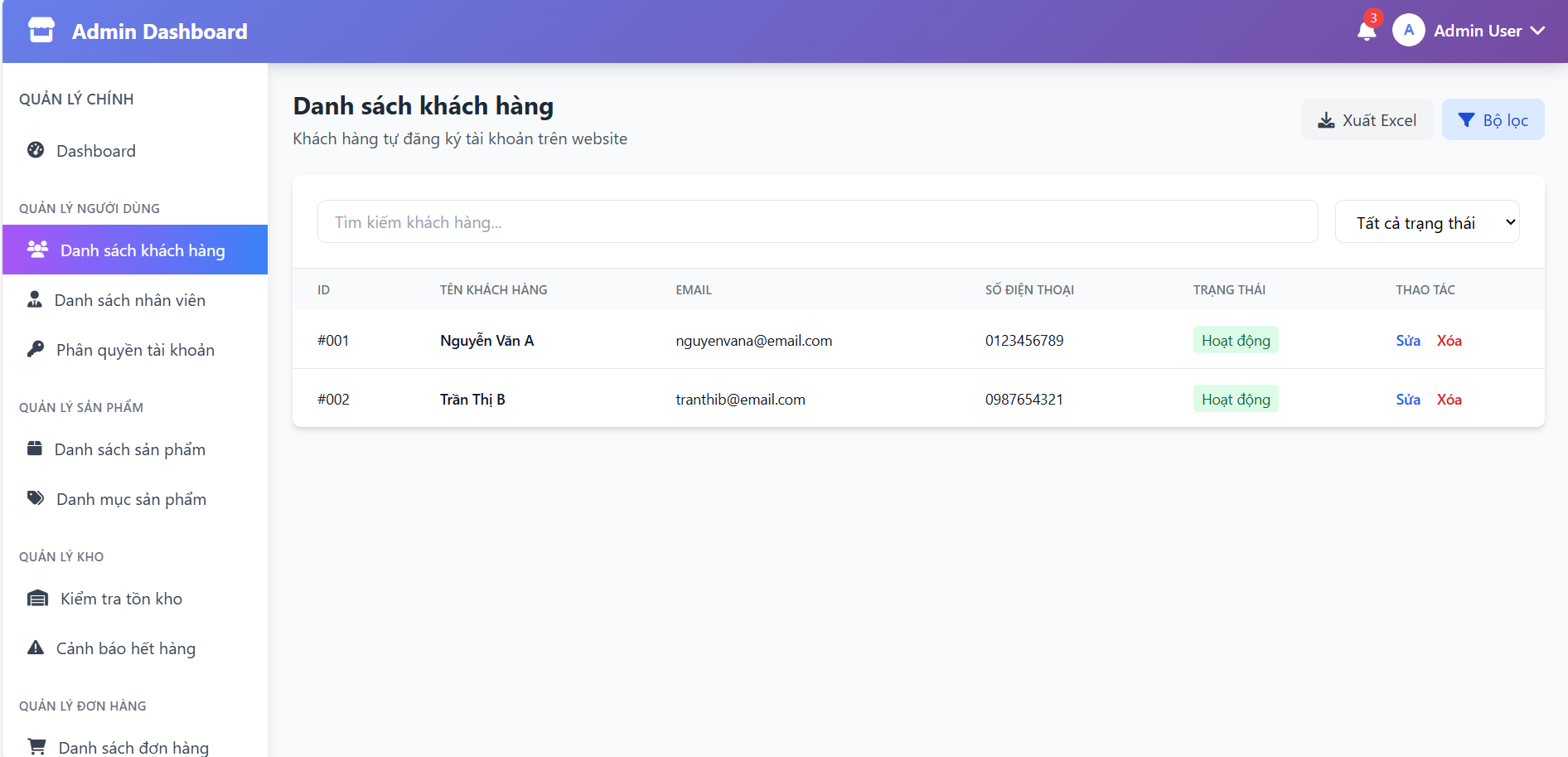


2.1.5. Giao diện chat với nhân viên CSKH của người dùng

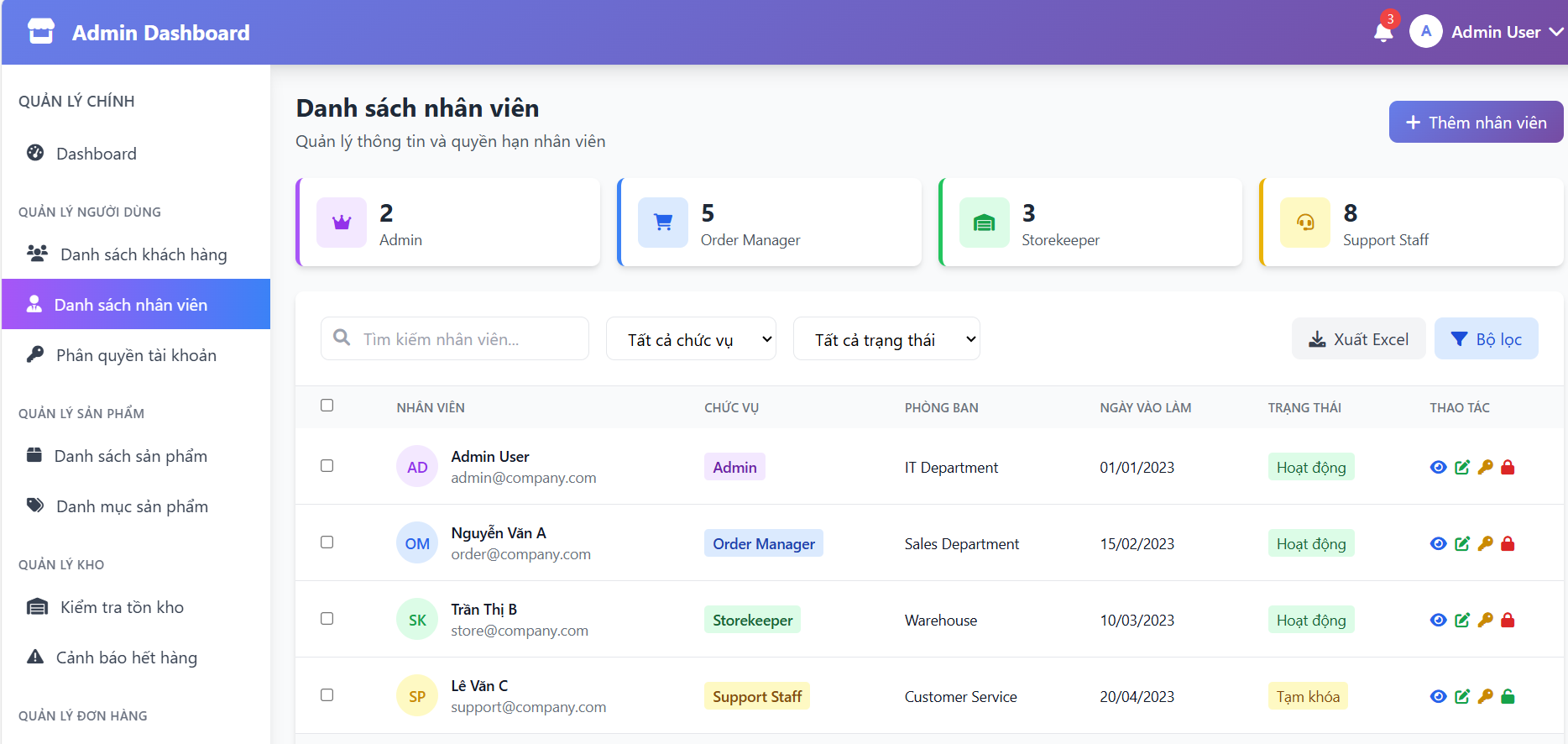


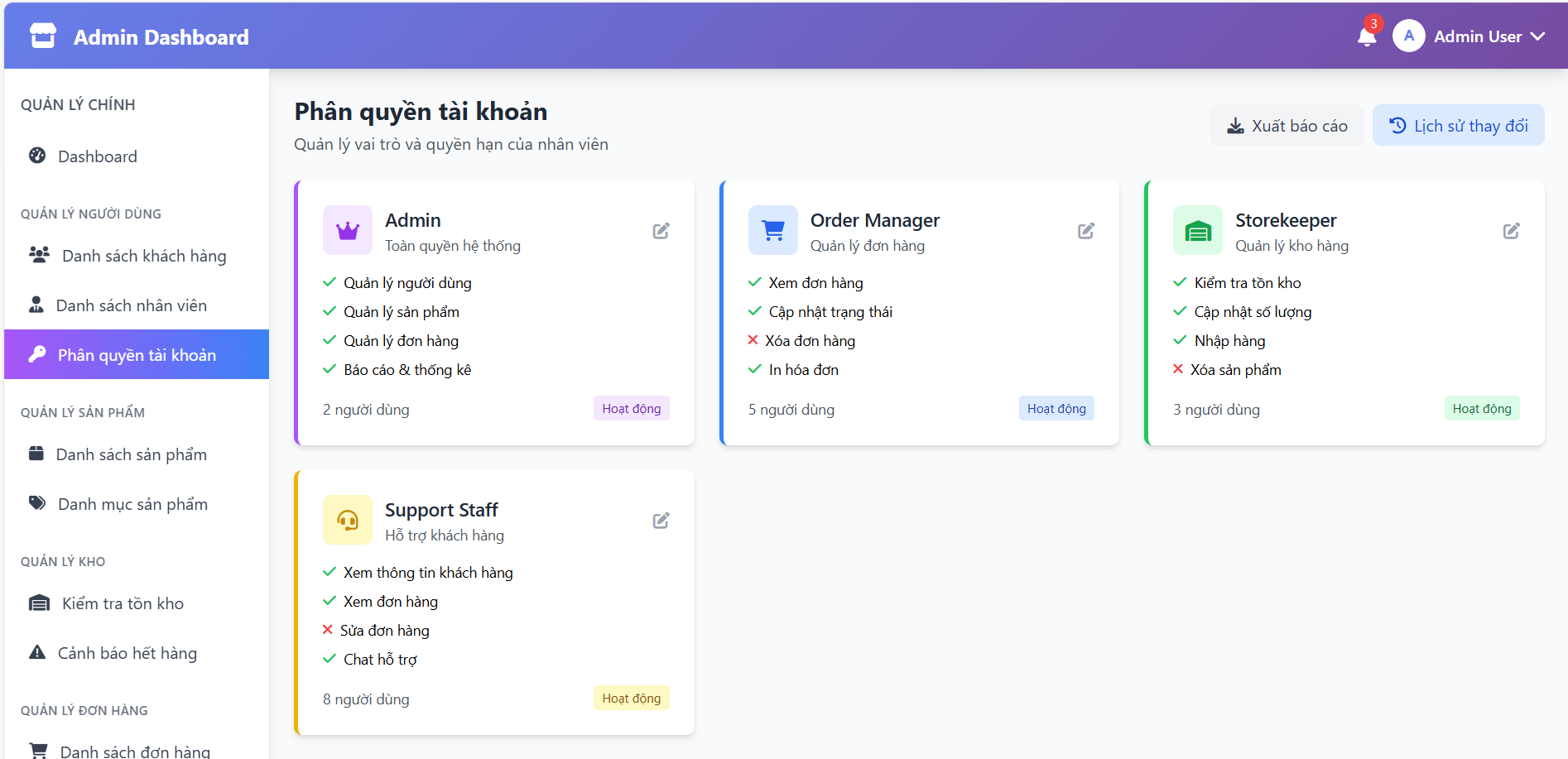
2.2. Giao diện quản lý của quản trị viên



2.2.1 Quản trị viên quản lý con người

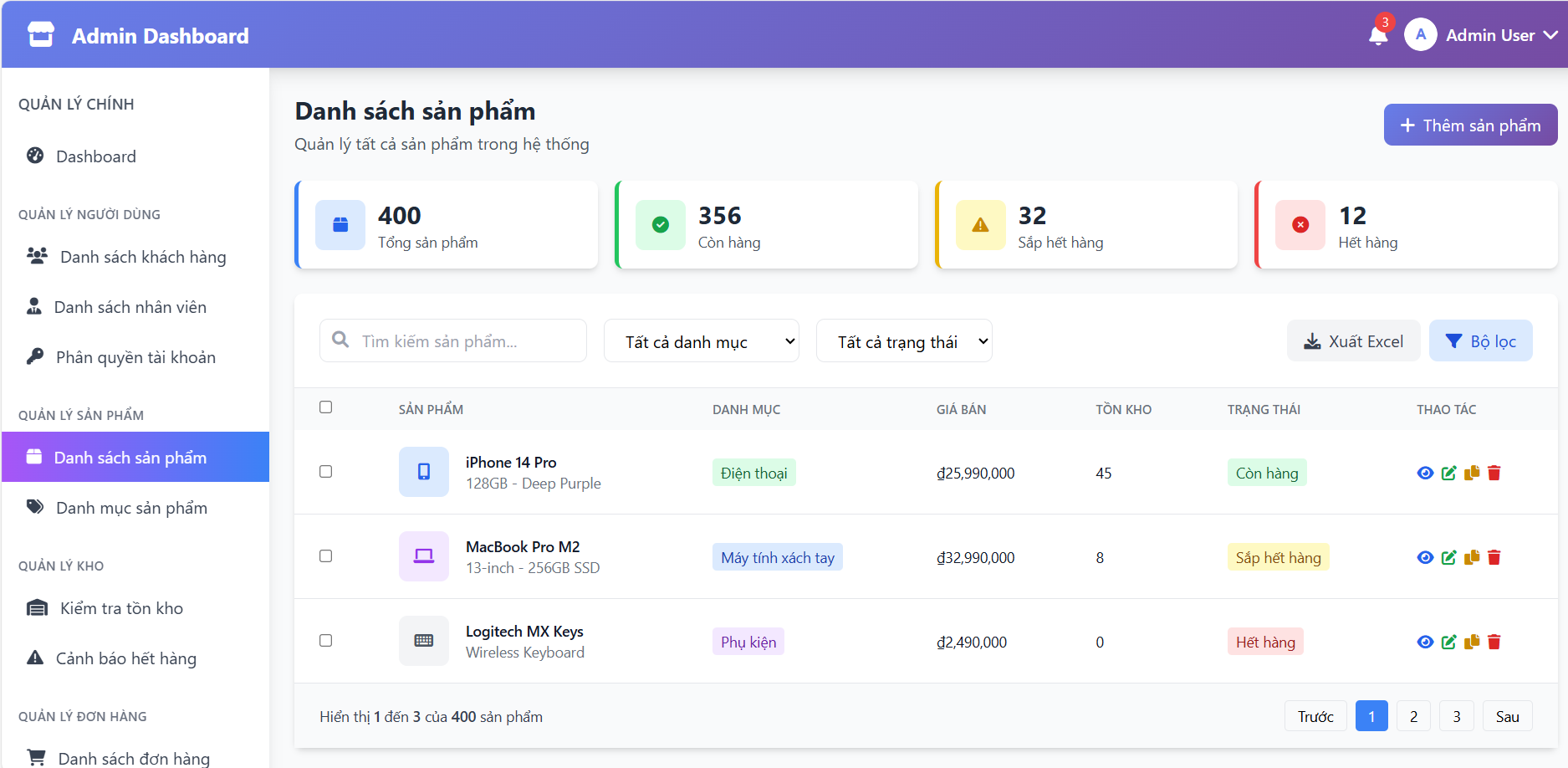
2.2.1.1. Quản trị viên quản lý các nhân viên

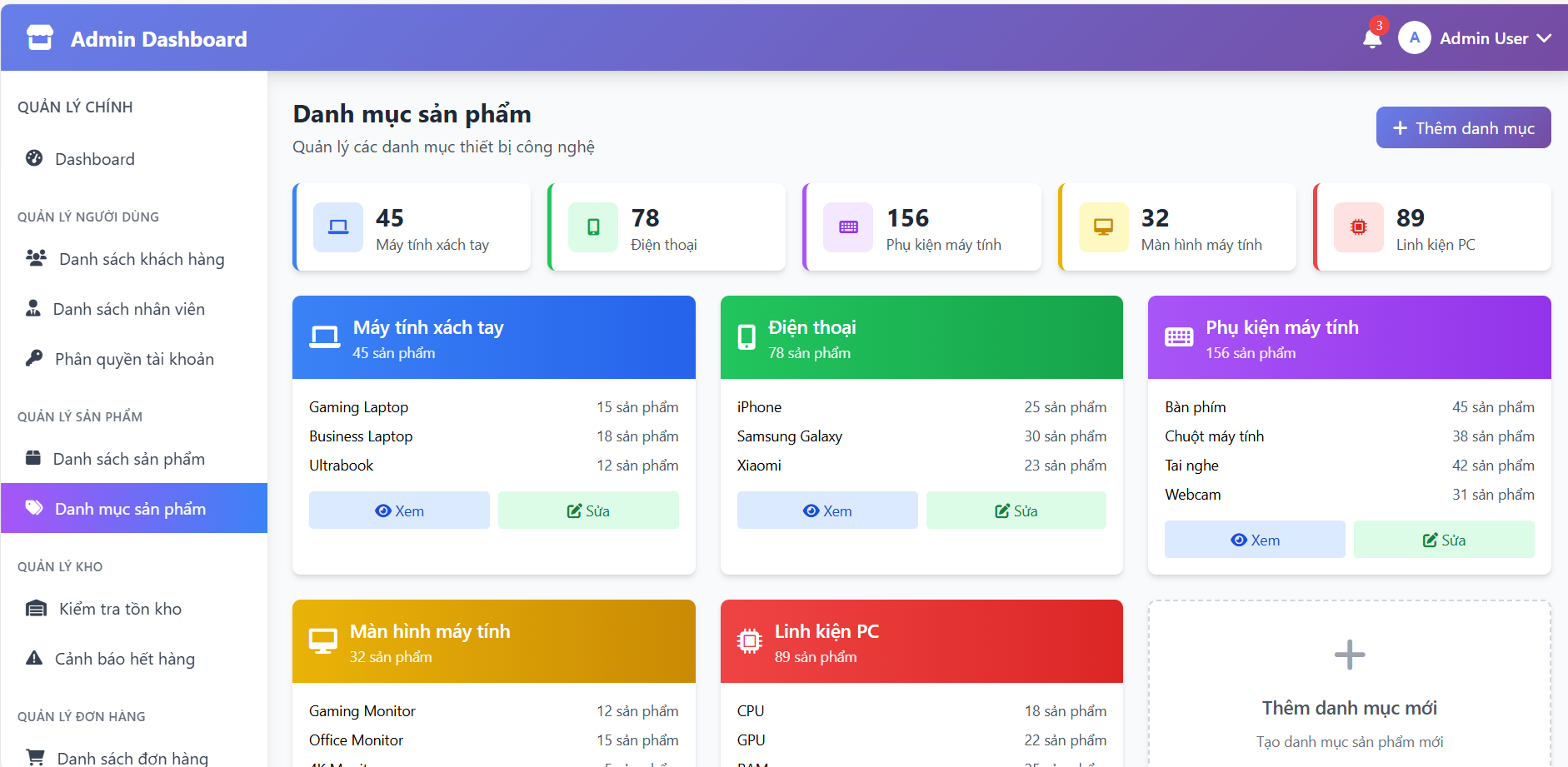


2.2.1.3. Quản trị viên phân quyền cho các tài khoản của nhân viên 

2.2.2. Quản trị viên quản lý sản phẩm

2.2.2.1. Danh sách các sản phẩm

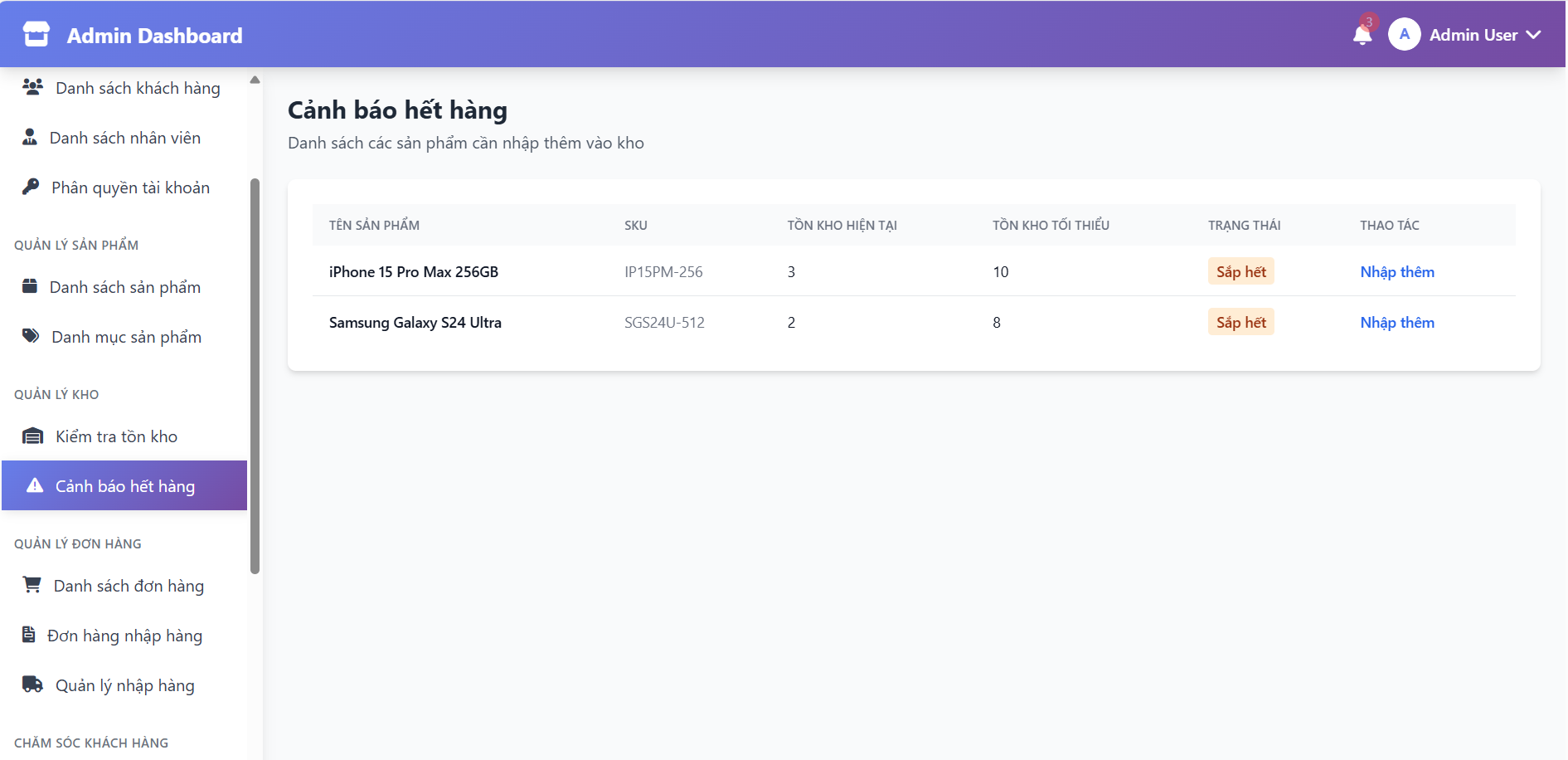


2.2.2.2. Danh mục của các sản phẩm mà cửa hàng quản lý

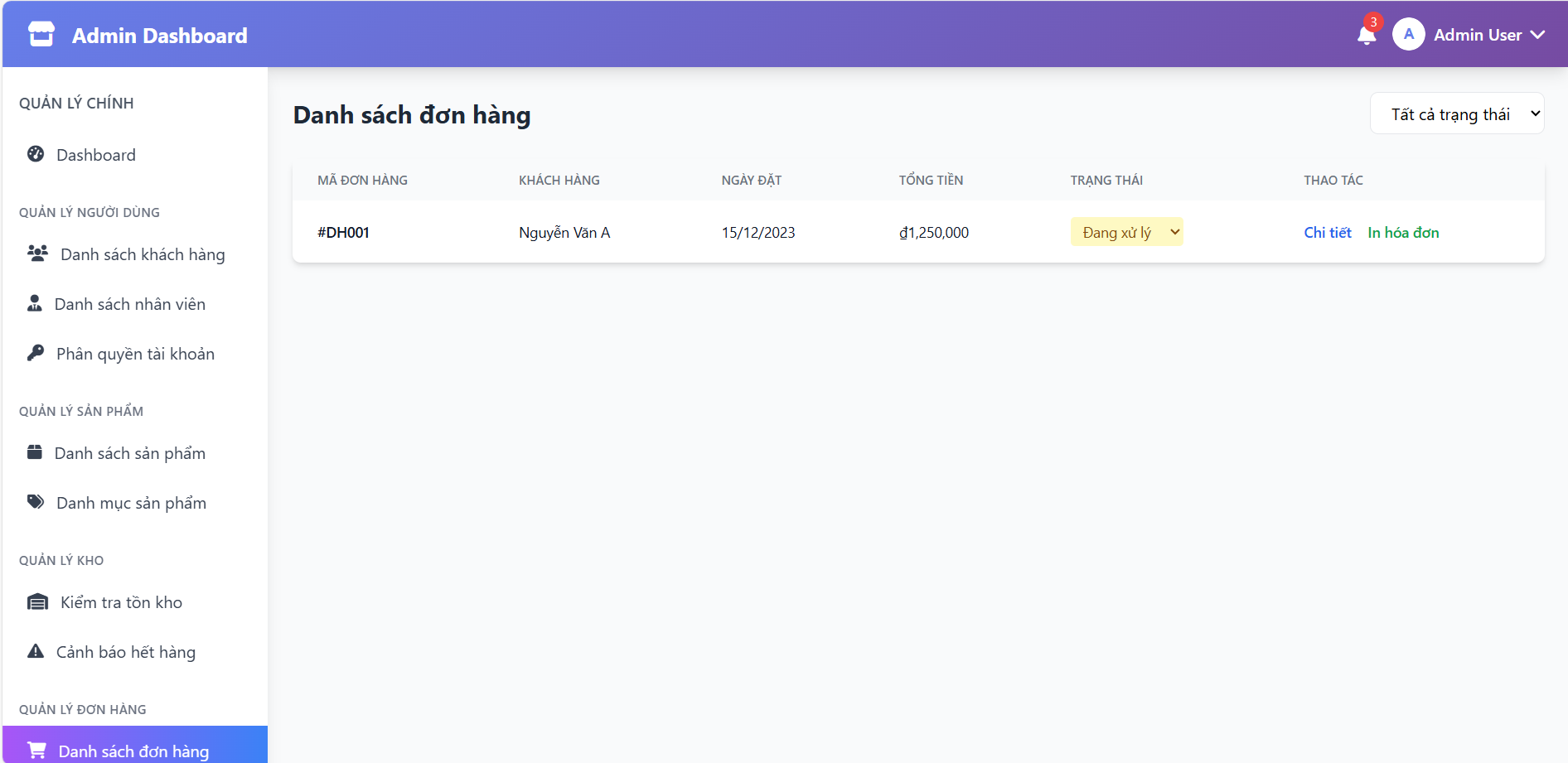
2.2.3. Quản lý kho

2.2.3.1. Kiểm tra hàng tồn kho



2.2.3.2. Cảnh báo hết hàng

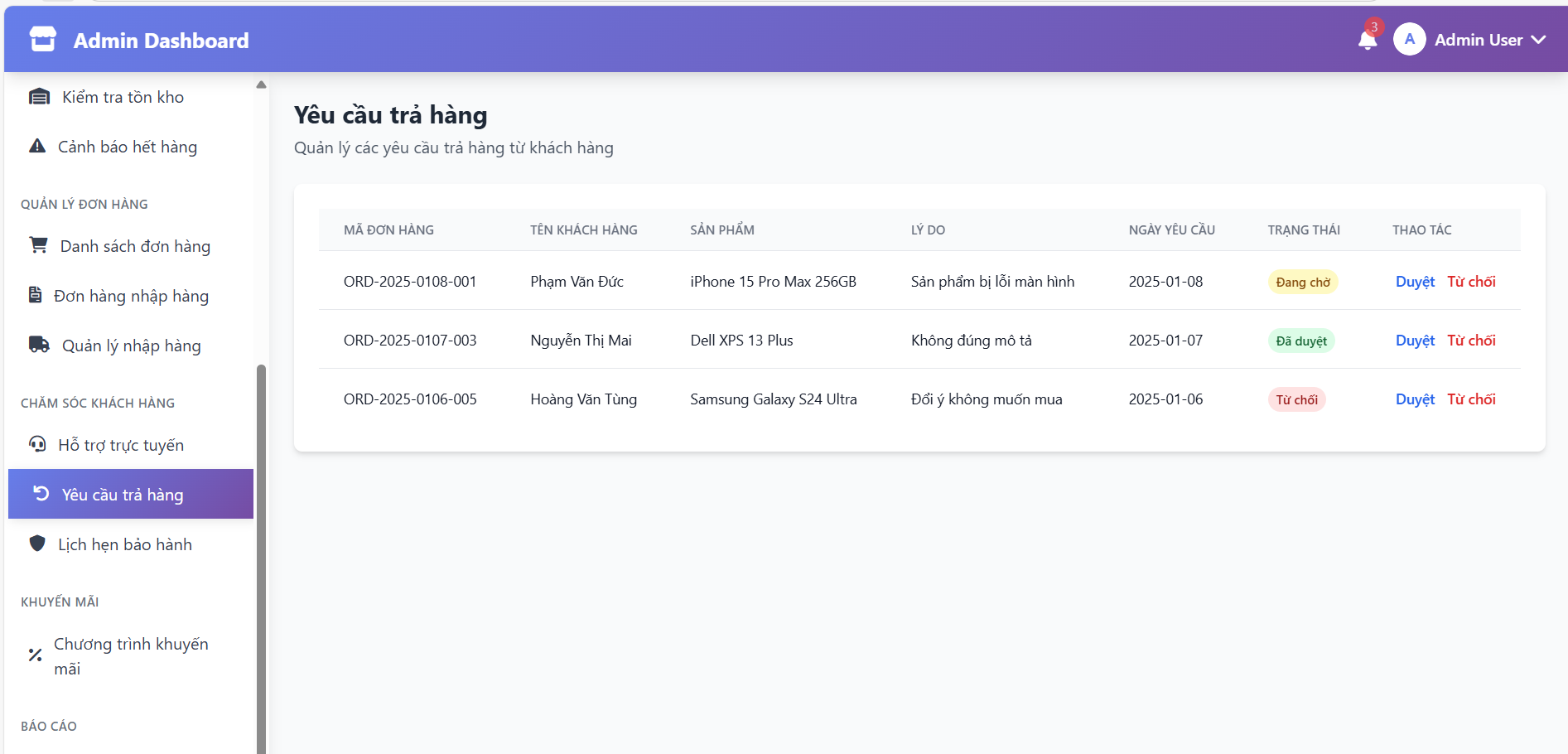
2.2.4. Danh sách các đơn hàng



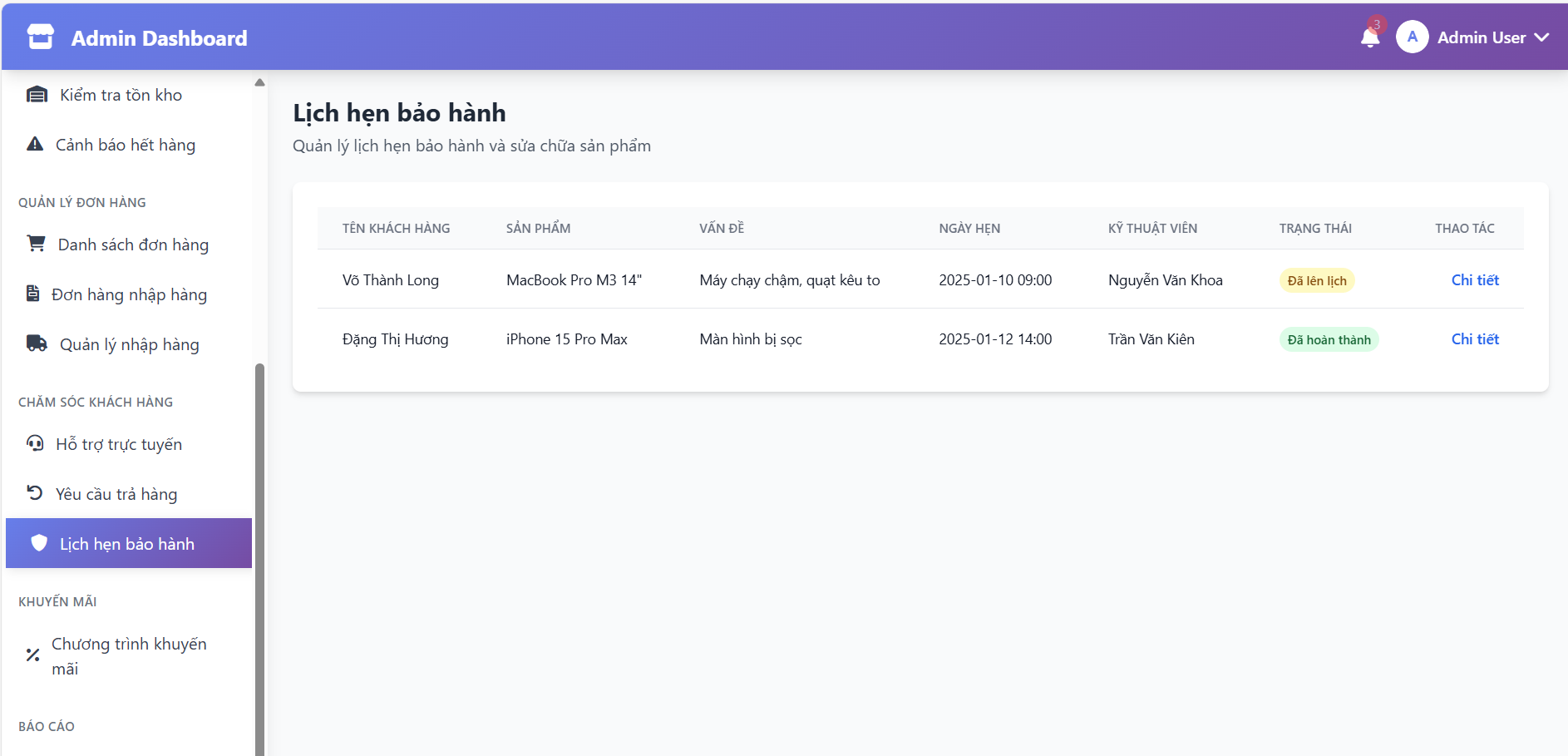
2.2.5. Chăm sóc khách hàng

2.2.5.1. Hỗ trợ trực tuyến

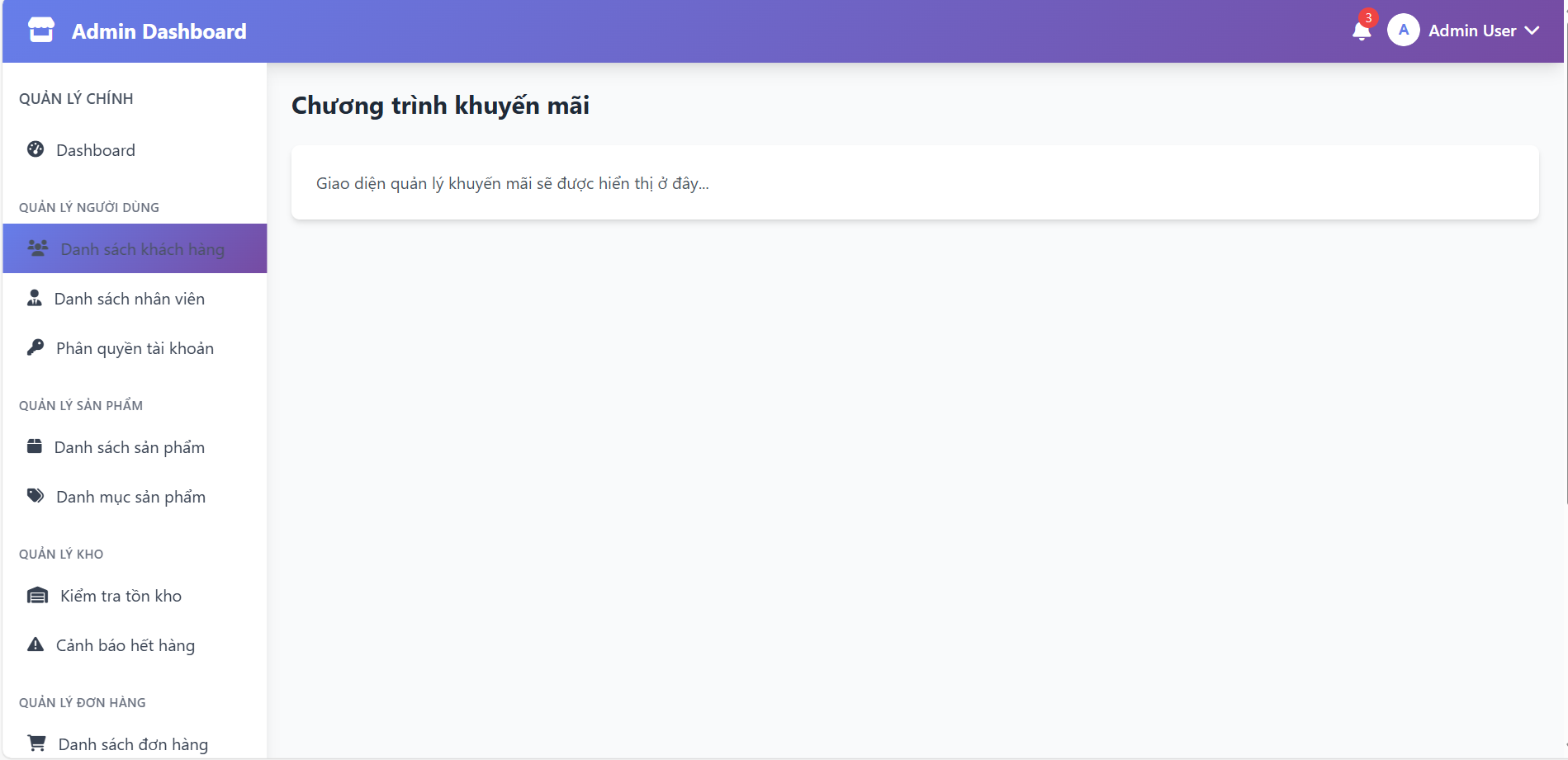


2.2.5.2. xử lý các yêu cầu muốn trả hàng

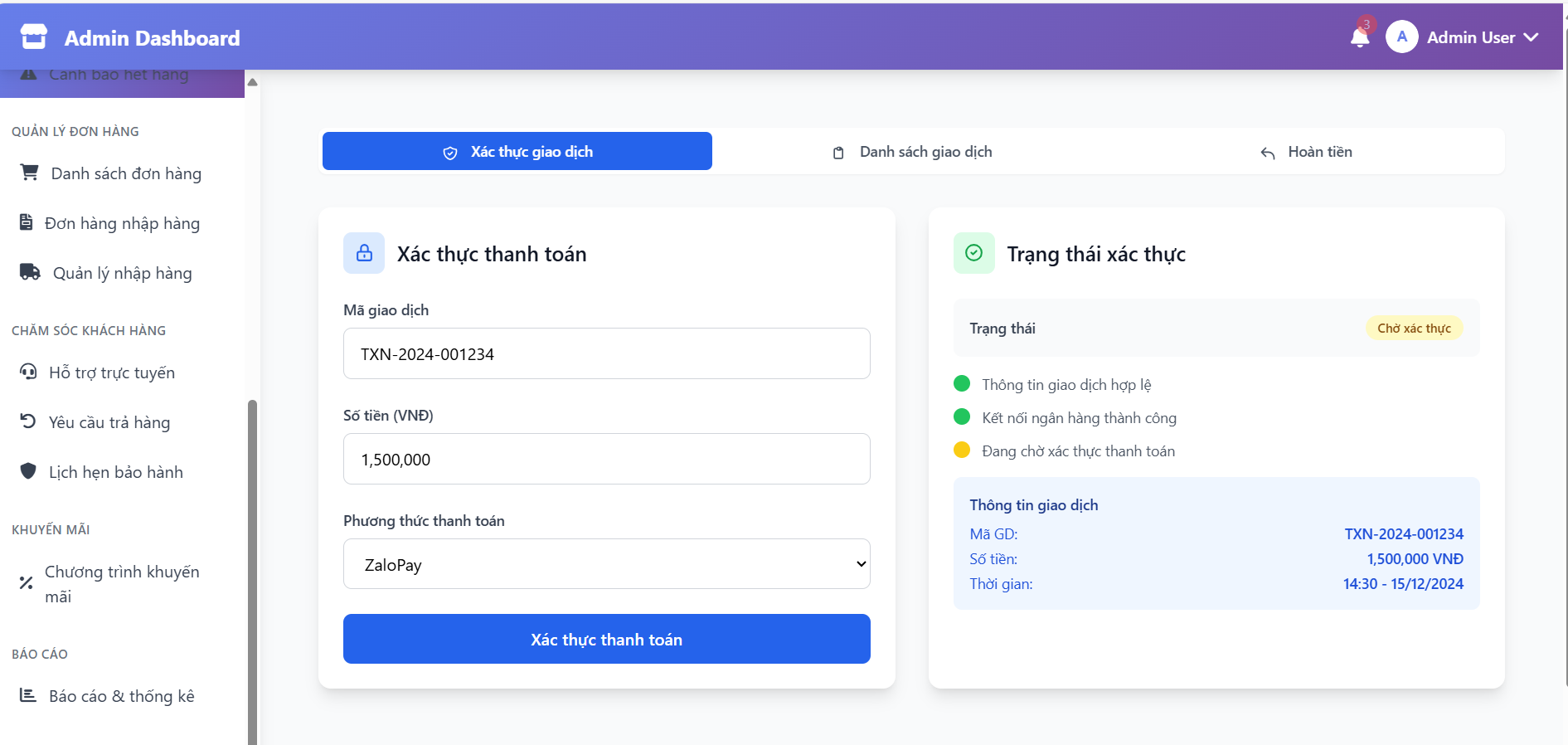
2.2.5.3. Hỗ trợ đặt lịch bảo hành

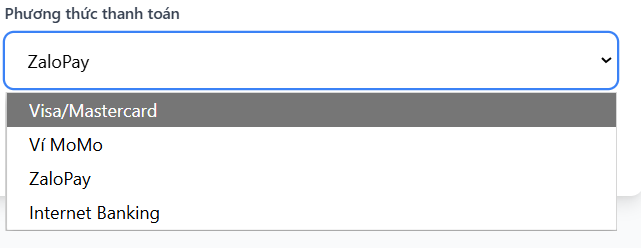


2.2.5. Khuyến mãi

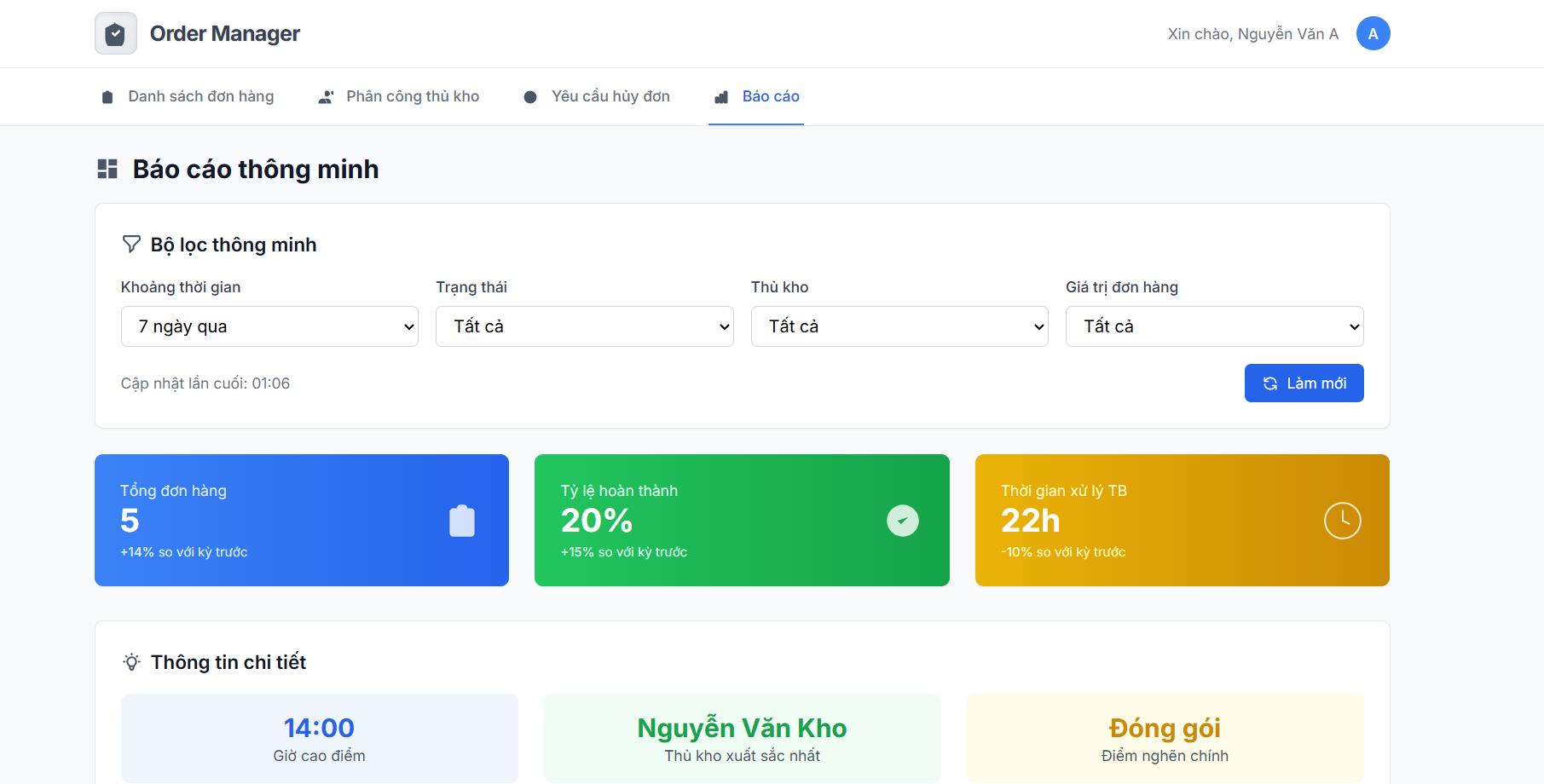


2.2.6. Thanh toán



2.2.6.1 Các phương thức thanh toán

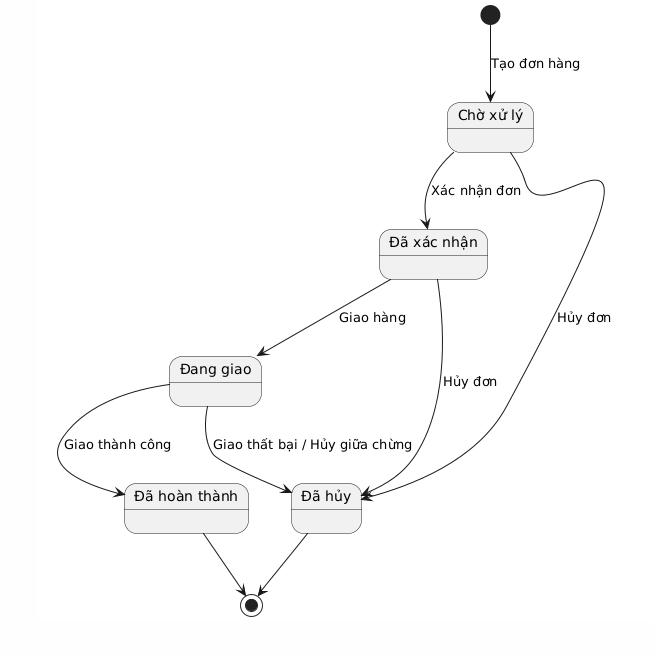
2.2.7. Thống kê và báo cáo

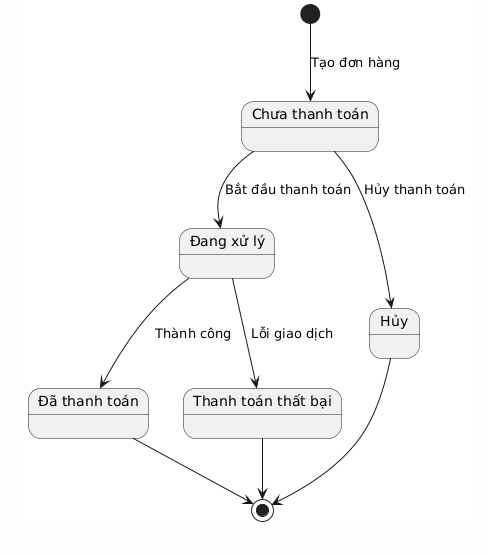


|  |
| --- |

**Tuần 5: Thiết kế Hành vi và Trạng thái**

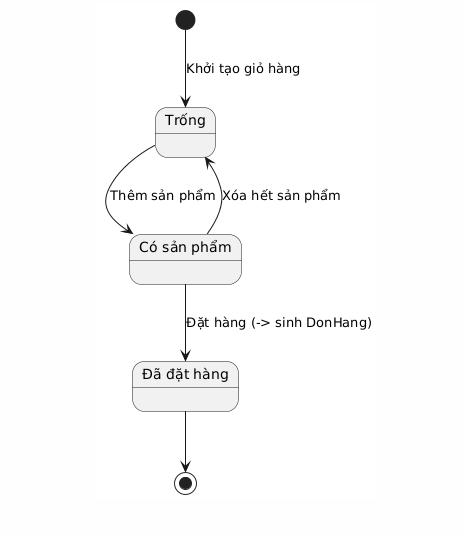
1. **Biểu đồ trạng thái**

****

****

****

****

****