

1. Package: product

- **Nhiệm vụ:** UC001 (View Product Detail - Customer), UC002 (View Product Detail - Product Manager), UC003 (Manage Product).

Class Diagram: Vẽ biểu đồ cho package product.

1.1 ProductController

- **States:** Không có (là controller, không lưu trữ trạng thái).

Attributes:

- productService: ProductService - Sử dụng để gọi các thao tác liên quan đến sản phẩm.

Operations:

- getFeaturedProducts(): ResponseEntity<List<ProductDTO>> - Lấy danh sách sản phẩm nổi bật.
- getProductList(role: String): ResponseEntity<List<ProductDTO>> - Lấy danh sách sản phẩm theo vai trò.
- getProductDetails(userId: String, productId: String, role: String): ResponseEntity<T extends ProductDTO> - Lấy chi tiết sản phẩm theo vai trò.
- addProduct(productDTO: ProductDTO): ResponseEntity<String> - Thêm sản phẩm mới.
- updateProduct(productId: String, productDTO: ProductDTO): ResponseEntity<String> - Cập nhật sản phẩm.
- deleteProduct(productId: String): ResponseEntity<String> - Xóa sản phẩm.
- searchProduct(query: String, role: String): ResponseEntity<List<ProductDTO>> - Tìm kiếm sản phẩm theo từ khóa và vai trò.
- getRelatedProducts(productId: String): ResponseEntity<List<ProductDTO>> - Lấy danh sách sản phẩm liên quan.

Methods:

- getFeaturedProducts: Gọi productService.getFeaturedProducts, trả về danh sách ProductDTO.
- getProductList: Gọi productService.getProductList, trả về danh sách theo vai trò.
- getProductDetails: Gọi productService.getProductDetails, trả về CustomerProductDTO hoặc ManagerProductDTO dựa trên vai trò.
- addProduct: Gọi productService.addProduct, trả về thông báo thành công/thất bại.
- updateProduct: Gọi productService.updateProduct, trả về thông báo cập nhật.
- deleteProduct: Gọi productService.deleteProduct, trả về thông báo xóa.
- searchProduct: Gọi productService.searchProducts, trả về danh sách sản phẩm theo vai trò.

- **getRelatedProducts:** Gọi productService.suggestRelatedProducts, trả về danh sách sản phẩm liên quan.
-

1.2 ProductService (Interface)

- **States:** Không có (interface không có trạng thái).
- **Attributes:** Không có (interface).

Operations:

- **getFeaturedProducts():** List<ProductDTO> - Lấy sản phẩm nổi bật.
- **getProductList(role: String):** List<ProductDTO> - Lấy danh sách sản phẩm theo vai trò.
- **getProductDetails<T extends ProductDTO>(userId: String, productId: String, role: String):** T - Lấy chi tiết sản phẩm theo vai trò.
- **searchProducts<T extends ProductDTO>(userId: String, keyword: String, role: String):** List<T> - Tìm kiếm sản phẩm theo từ khóa và vai trò.
- **suggestRelatedProducts(productId: String):** List<ProductDTO> - Gợi ý sản phẩm liên quan.
- **addProduct(productDTO: ProductDTO):** ManagerProductDTO - Thêm sản phẩm mới.
- **updateProduct(productId: String, productDTO: ProductDTO):** ManagerProductDTO - Cập nhật sản phẩm.
- **deleteProduct(productId: String):** void - Xóa sản phẩm.
- **validateProductDetails(productDTO: ProductDTO):** Boolean - Kiểm tra tính hợp lệ sản phẩm.
- **checkProductQuantity(productId: String):** int - Kiểm tra số lượng tồn kho.
- **validateDeletion(productId: String):** void - Kiểm tra trước khi xóa.
- **checkDeleteLimit(userId: String):** Boolean - Kiểm tra giới hạn xóa của user.
- **checkDeletionConstraints(productId: String):** Boolean - Kiểm tra ràng buộc xóa.
- **checkOrdersAffected(productId: String):** Boolean - Kiểm tra đơn hàng bị ảnh hưởng.
- **mapToBaseDTO(product: Product):** ProductDTO - Chuyển Product thành ProductDTO. (*Private*)

Methods: Không có (interface chỉ khai báo operations).

1.3 ProductServiceImpl

- **States:** Không có (là service, không lưu trữ trạng thái lâu dài).

Attributes:

- **productRepository:** ProductRepository - Thao tác với database.

- `productDetailFetcherFactory`: `ProductDetailFetcherFactory` - Lấy chi tiết sản phẩm theo vai trò.
 - `productSearchFetcherFactory`: `ProductSearchFetcherFactory` - Tìm kiếm sản phẩm theo vai trò.
 - `auditLogService`: `AuditLogServiceImpl` - Ghi log hành động.
Operations: (Như interface `ProductService`).
Methods:
 - `getFeaturedProducts`: Gọi `productRepository.findFeaturedProducts`, ánh xạ sang `ProductDTO`.
 - `getProductList`: Gọi `productRepository.findAll`, lọc theo vai trò, trả về `ProductDTO`.
 - `getProductDetails`: Gọi `productDetailFetcherFactory.getFetcher(role).fetchProductDetails`, ghi log qua `auditLogService`.
 - `searchProducts`: Gọi `productSearchFetcherFactory.getFetcher(role).searchProducts`, ghi log.
 - `suggestRelatedProducts`: Gọi `productRepository.findRelatedProducts`, trả về `ProductDTO`.
 - `addProduct`: Validate qua `validateProductDetails`, lưu qua `productRepository`, ghi log, trả về `ManagerProductDTO`.
 - `updateProduct`: Validate, cập nhật qua `productRepository`, ghi log, trả về `ManagerProductDTO`.
 - `deleteProduct`: Kiểm tra `checkDeletionConstraints` và `checkOrdersAffected`, xóa qua `productRepository`, ghi log.
 - `validateProductDetails`: Kiểm tra các trường (`name`, `price`, v.v.) hợp lệ.
 - `checkProductQuantity`: Gọi `productRepository.findById`, trả về `stockQuantity`.
 - `validateDeletion`: Kiểm tra `checkDeleteLimit` và các ràng buộc khác.
-

1.4 ProductDTO

- **States**: Không có (là DTO, không có trạng thái riêng).

Attributes:

- `productId`: String - Mã sản phẩm.
 - `name`: String - Tên sản phẩm.
 - `description`: String - Mô tả sản phẩm.
 - `price`: float - Giá sản phẩm.
 - `specifications`: String - Thông số kỹ thuật.
- Operations**: Không có.
Methods: Getter/setter.
-

1.5 ManagerProductDTO

- **States:** Không có (là DTO).

Attributes:

- Kế thừa từ ProductDTO.
- value: float - Giá trị sản phẩm.
- barcode: String - Mã vạch.
- stockQuantity: int - Số lượng tồn kho.
- warehouseEntryDate: Date - Ngày nhập kho.
- editHistory: List<String> - Lịch sử chỉnh sửa.

Operations: Không có.

Methods: Getter/setter.

1.6 CustomerProductDTO

States:

- "AVAILABLE": Còn hàng.
- "OUT_OF_STOCK": Hết hàng. (*Dựa trên availability*)

Attributes:

- Kế thừa từ ProductDTO.
- availability: String - Trạng thái còn hàng.
- images: String - Đường dẫn hình ảnh.
- relatedProducts: List<String> - Danh sách ID sản phẩm liên quan.

Operations: Không có.

Methods: Getter/setter.

1.7 ProductRepository (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- findFeaturedProducts(): List<Product> - Lấy sản phẩm nổi bật.
- findAll(): List<Product> - Lấy tất cả sản phẩm.
- findById(productId: String): Optional<Product> - Tìm theo ID.
- findByKeyword(keyword: String): List<Product> - Tìm theo từ khóa.
- findRelatedProducts(productId: String): List<Product> - Tìm sản phẩm liên quan.
- save(product: Product): Product - Lưu sản phẩm.
- deleteById(productId: String): void - Xóa sản phẩm.

Methods: Định nghĩa giao diện JPA.

1.8 Product

States:

- "AVAILABLE": Còn hàng (khi stockQuantity > 0).
- "OUT_OF_STOCK": Hết hàng (khi stockQuantity = 0).

Attributes:

- productId: String - Mã sản phẩm.
- name: String - Tên sản phẩm.
- description: String - Mô tả sản phẩm.
- price: float - Giá sản phẩm.
- value: float - Giá trị sản phẩm.
- barcode: String - Mã vạch.
- stockQuantity: int - Số lượng tồn kho.
- warehouseEntryDate: Date - Ngày nhập kho.
- editHistory: List<String> - Lịch sử chỉnh sửa.
- specifications: String - Thông số kỹ thuật.
- images: List<ProductImage> - Danh sách hình ảnh.

Operations: Không có.

Methods: Getter/setter.

1.9 ProductImage

- **States:** Không có (là entity phụ).

Attributes:

- id: Long - Mã hình ảnh.
- url: String - Đường dẫn hình ảnh.
- product: Product - Sản phẩm liên quan.

Operations: Không có.

Methods: Getter/setter.

1.10 ProductSearchFetcher (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- `searchProducts<T extends ProductDTO>(keyword: String): List<T>` - Tìm kiếm sản phẩm theo vai trò.
Methods: Không có.
-

1.11 CustomerProductSearchFetcher

- **States:** Không có (là triển khai cụ thể).
- **Attributes:** Không có.

Operations:

- `searchProducts(keyword: String): List<CustomerProductDTO>` - Tìm kiếm cho khách hàng.
Methods:
 - `searchProducts`: Gọi `productRepository.findByKeyword`, ánh xạ sang `CustomerProductDTO`.
-

1.12 ManagerProductSearchFetcher

- **States:** Không có (là triển khai cụ thể).
- **Attributes:** Không có.

Operations:

- `searchProducts(keyword: String): List<ManagerProductDTO>` - Tìm kiếm cho quản lý.
Methods:
 - `searchProducts`: Gọi `productRepository.findByKeyword`, ánh xạ sang `ManagerProductDTO`.
-

1.13 ProductSearchFetcherFactory

- **States:** Không có (là factory).
- **Attributes:** Không có.

Operations:

- `getFetcher(role: String): ProductSearchFetcher` - Trả về fetcher theo vai trò.
Methods:
 - `getFetcher`: Trả về `CustomerProductSearchFetcher` hoặc `ManagerProductSearchFetcher` dựa trên role.
-

1.14 ProductDetailFetcher (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- `fetchProductDetails<T extends ProductDTO>(productId: String): T` - Lấy chi tiết sản phẩm theo vai trò.
Methods: Không có.
-

1.15 CustomerProductDetailFetcher

- **States:** Không có (là triển khai cụ thể).
- **Attributes:** Không có.

Operations:

- `fetchProductDetails(productId: String): CustomerProductDTO` - Lấy chi tiết cho khách hàng.
Methods:
 - `fetchProductDetails`: Gọi `productRepository.findById`, ánh xạ sang `CustomerProductDTO`.
-

1.16 ManagerProductDetailFetcher

- **States:** Không có (là triển khai cụ thể).
- **Attributes:** Không có.

Operations:

- `fetchProductDetails(productId: String)`: `ManagerProductDTO` - Lấy chi tiết cho quản lý.
Methods:
 - `fetchProductDetails`: Gọi `productRepository.findById`, ánh xạ sang `ManagerProductDTO`.
-

1.17 ProductDetailFetcherFactory

- **States**: Không có (là factory).
- **Attributes**: Không có.

Operations:

- `getFetcher(role: String)`: `ProductDetailFetcher` - Trả về fetcher theo vai trò.
Methods:
 - `getFetcher`: Trả về `CustomerProductDetailFetcher` hoặc `ManagerProductDetailFetcher` dựa trên role.
-

2. Package: order

- **Nhiệm vụ**: UC004 (Place Order), UC005 (Place Rush Order), UC007 (Cancel Order).

Class Diagram: Vẽ biểu đồ cho package order.

2.1 OrderController

- **States**: Không có (là controller).

Attributes:

- `orderService`: `OrderServiceImpl` - Gọi các thao tác đặt/hủy đơn.
Operations:
- `placeOrder(orderDTO: OrderDTO)`: `ResponseEntity<OrderDTO>` - Đặt đơn thường.
- `placeRushOrder(rushOrderDTO: RushOrderDTO)`: `ResponseEntity<RushOrderDTO>` - Đặt đơn gấp.
- `cancelOrder(orderId: String)`: `ResponseEntity<Void>` - Hủy đơn hàng.

- getOrderDetails(orderId: String): ResponseEntity<OrderDTO> - Lấy chi tiết đơn hàng.
- setDeliveryInfo(orderId: String, deliveryInfoDTO: DeliveryInfoDTO): ResponseEntity<OrderDTO> - Cập nhật thông tin giao hàng.
- getInvoice(orderId: String): ResponseEntity<InvoiceDTO> - Lấy hóa đơn.
- initiatePayment(orderId: String, paymentMethod: String): ResponseEntity<String> - Khởi tạo thanh toán.

Methods:

- placeOrder: Gọi orderService.placeOrder, trả về OrderDTO.
 - placeRushOrder: Gọi orderService.placeRushOrder, trả về RushOrderDTO.
 - cancelOrder: Gọi orderService.cancelOrder, trả về 200 OK hoặc lỗi.
 - getOrderDetails: Gọi orderService.getOrderDetails, trả về OrderDTO.
 - setDeliveryInfo: Gọi orderService.setDeliveryInfo, trả về OrderDTO.
 - getInvoice: Gọi orderService.getInvoice, trả về InvoiceDTO.
 - initiatePayment: Gọi paymentService.payOrder, trả về URL thanh toán.
-

2.2 OrderService (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- createOrder(cartId: String): OrderDTO - Tạo đơn từ giỏ hàng.
- placeOrder(orderDTO: OrderDTO): OrderDTO - Đặt đơn thường.
- placeRushOrder(rushOrderDTO: RushOrderDTO): RushOrderDTO - Đặt đơn gấp.
- cancelOrder(orderId: String): void - Hủy đơn hàng.
- getOrderDetails(orderId: String): OrderDTO - Lấy chi tiết đơn hàng.
- checkAvailability(cartId: String): Boolean - Kiểm tra tồn kho.
- validateDeliveryInfo(deliveryInfoDTO: DeliveryInfoDTO): Boolean - Kiểm tra thông tin giao hàng.
- setDeliveryInfo(orderId: String, deliveryInfoDTO: DeliveryInfoDTO): OrderDTO - Cập nhật thông tin giao hàng.
- isRushOrder(orderId: String): Boolean - Kiểm tra đơn gấp.
- setPending(orderId: String): void - Đặt trạng thái chờ xử lý.
- checkRushDeliveryAddress(address: String): Boolean - Kiểm tra địa chỉ giao gấp.
- checkRushProductEligibility(cartId: String): Boolean - Kiểm tra sản phẩm đủ điều kiện giao gấp.
- getInvoice(orderId: String): InvoiceDTO - Lấy hóa đơn.
- updatePaymentStatus(orderId: String, status: String): void - Cập nhật trạng thái thanh toán.

Methods: Không có.

2.3 OrderServiceImpl

- **States:** Không có (là service).

Attributes:

- orderRepository: OrderRepository - Thao tác với database.
- paymentService: PaymentServiceImpl - Xử lý thanh toán.
- calculatorFactory: ShippingFeeCalculatorFactory - Tính phí giao hàng.
- cartService: CartServiceImpl - Xử lý giỏ hàng.
- auditLogService: AuditLogServiceImpl - Ghi log.

Operations: (Như interface OrderService).

Methods:

- createOrder: Gọi cartService để lấy giỏ, kiểm tra checkAvailability, tạo Order, lưu qua orderRepository.
 - placeOrder: Validate orderDTO, lưu qua orderRepository, ghi log.
 - placeRushOrder: Validate rushOrderDTO, kiểm tra checkRushProductEligibility, tính phí qua calculatorFactory, lưu.
 - cancelOrder: Kiểm tra điều kiện hủy, cập nhật trạng thái qua orderRepository, ghi log.
 - getOrderDetails: Gọi orderRepository.findById, ánh xạ sang OrderDTO.
 - setDeliveryInfo: Validate qua validateDeliveryInfo, cập nhật qua orderRepository.
 - getInvoice: Tạo InvoiceDTO từ Order, trả về.
-

2.4 OrderDTO

States:

- "PENDING": Chờ xử lý.
- "APPROVED": Đã duyệt.
- "CANCELLED": Đã hủy. (*Dựa trên status*)

Attributes: (Kế thừa từ BaseOrderDTO).

Operations: Không có.

Methods: Getter/setter.

2.5 RushOrderDTO

- **States:** (Kế thừa từ OrderDTO).

Attributes:

- Kế thừa từ BaseOrderDTO.

- rushDeliveryTime: Date - Thời gian giao gấp.
Operations: Không có.
Methods: Getter/setter.
-

2.6 BaseOrderDTO

States:

- "PENDING": Chờ xử lý.
- "APPROVED": Đã duyệt.
- "CANCELLED": Đã hủy. (*Dựa trên status*)

Attributes:

- orderId: String - Mã đơn hàng.
- customerId: String - ID khách hàng.
- items: List<OrderItemDTO> - Danh sách sản phẩm.
- status: String - Trạng thái.
- deliveryInfo: DeliveryInfoDTO - Thông tin giao hàng.
- subtotal: int - Tổng phụ.
- shippingFee: int - Phí giao hàng.
- total: int - Tổng tiền.
- creationDateTime: Date - Ngày tạo.

Operations: Không có.

Methods: Getter/setter.

2.7 OrderItemDTO

- **States:** Không có (là DTO phụ).

Attributes:

- productId: String - Mã sản phẩm.
- quantity: int - Số lượng.
- unitPrice: int - Đơn giá.
- lineTotal: int - Tổng dòng.

Operations: Không có.

Methods: Getter/setter.

2.8 DeliveryInfoDTO

- **States:** Không có (là DTO phụ).

Attributes:

- recipientName: String - Tên người nhận.
 - phoneNumber: String - Số điện thoại.
 - email: String - Email.
 - provinceCity: String - Tỉnh/thành phố.
 - address: String - Địa chỉ.
 - shippingInstructions: String - Hướng dẫn giao hàng.
- Operations:** Không có.
Methods: Getter/setter.
-

2.9 OrderRepository (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- findById(orderId: String): Optional<Order> - Tìm theo ID.
 - findByStatus(status: String): List<Order> - Tìm theo trạng thái.
 - save(order: Order): Order - Lưu đơn hàng.
 - deleteById(orderId: String): void - Xóa đơn hàng.
 - updateOrderStatus(orderId: String, status: String): void - Cập nhật trạng thái.
 - updateShippingFee(orderId: String, fees: int): void - Cập nhật phí giao hàng.
- Methods:** Định nghĩa giao diện JPA.
-

2.10 Order

States:

- "Pending Processing": Chờ xử lý sau khi đặt.
 - "Approved": Được duyệt bởi quản lý.
 - "Rejected": Bị từ chối.
 - "Cancelled": Đã hủy bởi khách hàng.
- Attributes:**
- orderId: String - Mã đơn hàng.
 - customerId: String - ID khách hàng.
 - items: List<OrderItem> - Danh sách sản phẩm.
 - status: String - Trạng thái.
 - deliveryInfo: DeliveryInfo - Thông tin giao hàng.
 - subtotal: float - Tổng phụ.
 - shippingFee: float - Phí giao hàng.
 - total: float - Tổng tiền.

- creationDateTime: Date - Ngày tạo.
- discount: float - Giảm giá.

Operations:

- getSubtotal(): float - Tính tổng phụ.
- getVAT(): float - Tính VAT.
- getTotal(): float - Tính tổng tiền.

Methods:

- getSubtotal: Tổng lineTotal từ items.
 - getVAT: Tính VAT dựa trên subtotal (giả định).
 - getTotal: Cộng subtotal, shippingFee, trừ discount.
-

2.11 OrderItem

- **States:** Không có (là entity phụ).

Attributes:

- productId: String - Mã sản phẩm.
- quantity: int - Số lượng.
- unitPrice: float - Đơn giá.
- lineTotal: float - Tổng dòng.

Operations:

- getLineTotal(): float - Tính tổng dòng.

Methods:

- getLineTotal: Nhân quantity với unitPrice.
-

2.12 DeliveryInfo

- **States:** Không có (là entity phụ).

Attributes:

- recipientName: String - Tên người nhận.
- phoneNumber: String - Số điện thoại.
- email: String - Email.
- provinceCity: String - Tỉnh/thành phố.
- address: String - Địa chỉ.
- shippingInstructions: String - Hướng dẫn giao hàng.

Operations: Không có.

Methods: Getter/setter.

2.13 ShippingFeeCalculator (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- calculateShippingFee(dto: BaseOrderDTO): int - Tính phí giao hàng.
Methods: Không có.
-

2.14 StandardShippingFeeCalculator

- **States:** Không có (là triển khai cụ thể).
- **Attributes:** Không có.

Operations:

- calculateShippingFee(dto: BaseOrderDTO): int - Tính phí giao thường.
Methods:
 - calculateShippingFee: Tính dựa trên subtotal và deliveryInfo (giả định).
-

2.15 RushShippingFeeCalculator

- **States:** Không có (là triển khai cụ thể).
- **Attributes:** Không có.

Operations:

- calculateShippingFee(dto: BaseOrderDTO): int - Tính phí giao gấp.
Methods:
 - calculateShippingFee: Tính dựa trên subtotal, rushDeliveryTime (giả định).
-

2.16 ShippingFeeCalculatorFactory

- **States:** Không có (là factory).

- **Attributes:** Không có.

Operations:

- getCalculator(dto: BaseOrderDTO): ShippingFeeCalculator - Trả về calculator theo loại đơn.
Methods:
 - getCalculator: Trả về StandardShippingFeeCalculator hoặc RushShippingFeeCalculator dựa trên dto.
-

2.17 InvoiceDTO

- **States:** Không có (là DTO).

Attributes:

- orderId: String - Mã đơn hàng.
 - shippingFee: float - Phí giao hàng.
 - total: float - Tổng tiền.
- Operations:** Không có.
Methods: Getter/setter.
-

3. Package: payment

- **Nhiệm vụ:** UC006 (Pay Order).

Class Diagram: Vẽ biểu đồ cho package payment.

3.1 PaymentController

- **States:** Không có (là controller).

Attributes:

- paymentService: PaymentServiceImpl - Gọi các thao tác thanh toán.
Operations:
- payOrder(orderId: String): ResponseEntity<PaymentResult> - Thanh toán đơn hàng.
- processRefund(orderId: String): ResponseEntity<RefundResult> - Xử lý hoàn tiền.
Methods:

- `payOrder`: Gọi `paymentService.payOrder`, trả về `PaymentResult`.
 - `processRefund`: Gọi `paymentService.processRefund`, trả về `RefundResult`.
-

3.2 PaymentService (Interface)

- **States**: Không có (interface).
- **Attributes**: Không có.

Operations:

- `payOrder(orderId: String)`: `PaymentResult` - Thanh toán đơn hàng.
 - `validatePayment(orderId: String)`: `Boolean` - Kiểm tra hợp lệ trước thanh toán.
 - `processRefund(orderId: String)`: `RefundResult` - Xử lý hoàn tiền.
 - `checkCancellationValidity(orderId: String)`: `Boolean` - Kiểm tra hủy đơn.
Methods: Không có.
-

3.3 PaymentServiceImpl

- **States**: Không có (là service).

Attributes:

- `paymentRepository`: `PaymentRepository` - Thao tác với database.
 - `paymentGateway`: `PaymentGateway` - Gọi cổng thanh toán.
 - `orderService`: `OrderService` - Cập nhật trạng thái đơn hàng.
 - `auditLogService`: `AuditLogService` - Ghi log.
Operations: (Như interface `PaymentService`).
Methods:
 - `payOrder`: Validate qua `validatePayment`, tạo Invoice, gọi `paymentGateway.processPayment`, cập nhật trạng thái qua `orderService`, ghi log.
 - `processRefund`: Kiểm tra `checkCancellationValidity`, gọi `paymentGateway.processRefund`, cập nhật trạng thái, ghi log.
 - `validatePayment`: Kiểm tra đơn hàng tồn tại, trạng thái hợp lệ.
 - `checkCancellationValidity`: Gọi `orderService.checkCancellationValidity`.
-

3.4 PaymentRepository (Interface)

- **States**: Không có (interface).

- **Attributes:** Không có.

Operations:

- save(payment: Payment): Payment - Lưu giao dịch.
 - findById(orderId: String): Optional<Payment> - Tìm giao dịch theo orderId.
Methods: Định nghĩa giao diện JPA.
-

3.5 PaymentGateway (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- processPayment(invoice: Invoice): PaymentResult - Xử lý thanh toán.
 - processRefund(orderId: String): RefundResult - Xử lý hoàn tiền.
Methods: Không có.
-

3.6 VNPayClient

- **States:** Không có (là triển khai cụ thể).
- **Attributes:** Không có.

Operations:

- processPayment(invoice: Invoice): PaymentResult - Gọi API VNPay thanh toán.
 - processRefund(orderId: String): RefundResult - Gọi API VNPay hoàn tiền.
Methods:
 - processPayment: Gọi API VNPay, trả về PaymentResult.
 - processRefund: Gọi API VNPay, trả về RefundResult.
-

3.7 Payment

States:

- "Completed": Thanh toán thành công.
- "Failed": Thanh toán thất bại.

- "Refunded": Đã hoàn tiền.
Attributes:
 - transactionId: String - Mã giao dịch.
 - orderId: String - Mã đơn hàng.
 - totalAmount: float - Tổng tiền.
 - transactionContent: String - Nội dung giao dịch.
 - transactionDate: Date - Ngày giao dịch.
 - paymentMethod: String - Phương thức thanh toán.
 - paymentStatus: String - Trạng thái thanh toán.
 - refundStatus: String - Trạng thái hoàn tiền.
 - refundDate: Date - Ngày hoàn tiền.**Operations:** Không có.
Methods: Getter/setter.
-

3.8 Invoice

- **States:** Không có (là entity phụ).

Attributes:

- orderId: String - Mã đơn hàng.
 - amount: float - Số tiền.
 - content: String - Nội dung.
Operations: Không có.
Methods: Getter/setter.
-

3.9 BasePaymentResult

- **States:** Không có (là DTO).

Attributes:

- orderId: String - Mã đơn hàng.
 - transactionId: String - Mã giao dịch.
 - errorMessage: String - Thông báo lỗi.
Operations: Không có.
Methods: Getter/setter.
-

3.10 PaymentResult

States:

- "Completed": Thanh toán thành công.
 - "Failed": Thanh toán thất bại. (*Dựa trên paymentStatus*)
Attributes:
 - Kế thừa từ BasePaymentResult.
 - totalAmount: float - Tổng tiền.
 - transactionContent: String - Nội dung giao dịch.
 - transactionDate: Date - Ngày giao dịch.
 - paymentStatus: String - Trạng thái thanh toán.**Operations:** Không có.
Methods: Getter/setter.
-

3.11 RefundResult

States:

- "Refunded": Đã hoàn tiền.
 - "Failed": Hoàn tiền thất bại. (*Dựa trên refundStatus*)
Attributes:
 - Kế thừa từ BasePaymentResult.
 - refundStatus: String - Trạng thái hoàn tiền.
 - refundDate: Date - Ngày hoàn tiền.**Operations:** Không có.
Methods: Getter/setter.
-

4. Package: notification

4.1 NotificationService (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- sendEmail(orderId: String): void - Gửi email thông báo.
Methods: Không có.
-

5. Package: cart

5.1 CartController

- **States:** Không có (là controller).

Attributes:

- cartService: CartServiceImpl - Gọi các thao tác giỏ hàng.
Operations:
 - viewCart(): CartDTO - Xem giỏ hàng.
 - updateCart(productId: String, quantity: int): CartDTO - Cập nhật giỏ hàng.
 - removeFromCart(productId: String): CartDTO - Xóa sản phẩm khỏi giỏ.
 - addProductToCart(userId: String, productId: String, quantity: int): ResponseEntity<CartDTO> - Thêm sản phẩm vào giỏ.**Methods:**
 - viewCart: Gọi cartService.viewCart, trả về CartDTO.
 - updateCart: Gọi cartService.updateCart, trả về CartDTO.
 - removeFromCart: Gọi cartService.removeFromCart, trả về CartDTO.
 - addProductToCart: Gọi cartService.addProductToCart, trả về CartDTO.
-

5.2 CartService (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- addProductToCart(userId: String, productId: String, quantity: int): CartDTO - Thêm sản phẩm vào giỏ.
 - viewCart(): CartDTO - Xem giỏ hàng.
 - updateCart(productId: String, quantity: int): CartDTO - Cập nhật giỏ.
 - removeFromCart(productId: String): CartDTO - Xóa sản phẩm khỏi giỏ.
 - emptyCart(cartId: String): void - Xóa toàn bộ giỏ.
Methods: Không có.
-

5.3 CartServiceImpl

- **States:** Không có (là service).

Attributes:

- **cartRepository:** CartRepository - Thao tác với database.
 - **productService:** ProductServiceImpl - Kiểm tra sản phẩm.
Operations: (Như interface CartService).
Methods:
 - **addProductToCart:** Kiểm tra sản phẩm qua productService, cập nhật Cart, lưu qua cartRepository.
 - **viewCart:** Gọi cartRepository.findById, ánh xạ sang CartDTO.
 - **updateCart:** Cập nhật quantity trong CartItem, lưu lại.
 - **removeFromCart:** Xóa CartItem khỏi Cart, lưu lại.
 - **emptyCart:** Xóa toàn bộ items trong Cart, lưu lại.
-

5.4 CartRepository (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- **findById(userId: String):** Optional<Cart> - Tìm giỏ theo user.
 - **save(cart: Cart):** Cart - Lưu giỏ hàng.
Methods: Định nghĩa giao diện JPA.
-

5.5 Cart

- **States:** Không có (trạng thái phụ thuộc vào items).

Attributes:

- **id:** Long - Mã giỏ hàng.
 - **userId:** String - ID người dùng.
 - **items:** List<CartItem> - Danh sách sản phẩm trong giỏ.
Operations: Không có.
Methods: Getter/setter.
-

5.6 CartItem

- **States:** Không có (là entity phụ).

Attributes:

- id: Long - Mã mục giỏ.
 - productId: String - Mã sản phẩm.
 - quantity: int - Số lượng.
Operations: Không có.
Methods: Getter/setter.
-

5.7 CartDTO

- **States:** Không có (là DTO).

Attributes:

- id: Long - Mã giỏ hàng.
 - userId: String - ID người dùng.
 - items: List<CartItemDTO> - Danh sách sản phẩm.
Operations: Không có.
Methods: Getter/setter.
-

5.8 CartItemDTO

- **States:** Không có (là DTO phụ).

Attributes:

- productId: String - Mã sản phẩm.
 - quantity: int - Số lượng.
Operations: Không có.
Methods: Getter/setter.
-

6. Package: audit

6.1 AuditLogService (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- logSearchAction(userId: String, keyword: String, role: String): void - Ghi log tìm kiếm.
 - logViewProduct(productId: String, userId: String, role: String): void - Ghi log xem sản phẩm.
 - logAddAction(userId: String, productId: String, role: String): Boolean - Ghi log thêm sản phẩm.
 - logDeleteAction(userId: String, productId: String, role: String): Boolean - Ghi log xóa sản phẩm.
 - checkDeleteLimit(userId: String): Boolean - Kiểm tra giới hạn xóa.
 - logUpdateAction(userId: String, productId: String, role: String): Boolean - Ghi log cập nhật sản phẩm.
 - logOrderAction(userId: String, orderId: String, role: String): Boolean - Ghi log đặt đơn.
 - logPayment(transaction: PaymentTransactionDTO): void - Ghi log thanh toán.
- Methods:** Không có.
-

6.2 AuditLogServiceImpl

- **States:** Không có (là service).

Attributes:

- auditLogRepository: AuditLogRepository - Thao tác với database.
Operations: (Như interface AuditLogService).
Methods:
 - logSearchAction: Tạo AuditLog, lưu qua auditLogRepository.
 - logViewProduct: Tương tự, ghi log xem chi tiết.
 - logAddAction: Ghi log thêm sản phẩm, trả về thành công/thất bại.
 - logDeleteAction: Ghi log xóa, kiểm tra checkDeleteLimit.
 - checkDeleteLimit: Gọi auditLogRepository.countDeletesByUserId, so sánh với giới hạn.
 - logUpdateAction: Ghi log cập nhật sản phẩm.
 - logOrderAction: Ghi log đặt đơn hàng.
 - logPayment: Ghi log thanh toán.
-

6.3 AuditLogRepository (Interface)

- **States:** Không có (interface).
- **Attributes:** Không có.

Operations:

- save(auditLog: AuditLog): AuditLog - Lưu log.
 - countDeletesByUserId(userId: String): int - Đếm số lần xóa của user.
Methods: Định nghĩa giao diện JPA.
-

6.4 AuditLog

- **States:** Không có (trạng thái phụ thuộc vào actionType).

Attributes:

- id: Long - Mã log.
- actionType: String - Loại hành động (VD: "SEARCH", "ADD", "DELETE").
- userId: String - ID người dùng.
- productId: String - Mã sản phẩm.
- keyword: String - Từ khóa (nếu có).
- role: String - Vai trò người dùng.
- timestamp: DateTime - Thời gian thực hiện.
Operations: Không có.
Methods: Getter/setter.
-