

Phân tích các vấn đề về Design Principle: S.O.L.I.D và đề xuất cải thiện

Modules	Nguyên lý vi phạm	Lý do vi phạm	Đề xuất cải thiện	Trạng thái
Cart	SRP	Lớp chịu nhiều trách nhiệm, bao gồm quản lý giỏ hàng, tính toán tổng số lượng, kiểm tra tính khả dụng, và xử lý ngoại lệ.	Đã tạo ra CartService để định nghĩa và xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện
Cart	OCP	Lớp không dễ mở rộng vì logic kiểm tra tính khả dụng được viết cứng trong phương thức checkAvailabilityOfProduct.	Đã tạo ra CartService để định nghĩa và xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện
Cart	DIP	Lớp phụ thuộc trực tiếp vào Media và CartMedia, làm tăng sự phụ thuộc cứng vào các lớp chi tiết.	Đã tạo ra CartService để định nghĩa và xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện
Book	SRP	Lớp Book thực hiện các chức năng không liên quan đến nhau: lưu trữ dữ liệu sách và thao tác database (phương thức getMediaById.	Tách logic làm việc với DB liên quan tới book sang BookDAO	Đã thực hiện
Book	OCP	Lớp Book bị phụ thuộc chặt chẽ vào SQL query trong phương thức getMediaById, khi thay đổi dữ liệu hoặc query thì cần sửa lớp.	Tách logic làm việc với DB liên quan tới book sang BookDAO	Đã thực hiện
CD	SRP	Thực hiện logic truy vấn dữ liệu từ database thay vì thuộc về DAO.	Tách logic làm việc với DB liên quan tới CD sang CDDAO	Đã thực hiện
DVD	SRP	Phương thức getMediaById và getAllMedia thực hiện truy vấn cơ sở dữ liệu, không phù hợp với trách nhiệm của lớp DVD, vốn chỉ đại diện cho một thực thể dữ liệu.	Tách logic truy vấn cơ sở dữ liệu ra một lớp DAO riêng như DVDDAO. Lớp này sẽ chịu trách nhiệm quản lý dữ liệu liên quan đến DVD.	Đã thực hiện

Media	SRP	Các phương thức chứa truy vấn cơ sở dữ liệu làm cho lớp chịu trách nhiệm quá nhiều việc	Tách logic làm việc với DB liên quan tới Media sang mediaDAO	Đã thực hiện
Order	SRP	Lớp Order chịu trách nhiệm quá nhiều: quản lý thông tin đơn hàng, tính toán phí và số tiền thanh toán.	Tách chức năng tính toán (getAmount()) ra một lớp dịch vụ riêng như OrderService.	Chưa thực hiện
Order	SRP	Danh sách lstOrderMedia được quản lý trực tiếp trong lớp Order, dẫn đến trách nhiệm trùng lặp khi thêm/xóa OrderMedia.	Sử dụng lớp trung gian quản lý danh sách OrderMedia, ví dụ OrderMediaManager.	Chưa thực hiện
BaseController	SRP	Lớp BaseController không chỉ chịu trách nhiệm điều hướng (controller logic), mà còn trực tiếp xử lý logic liên quan đến Cart.	Đã tạo ra CartService để định nghĩa và xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện
BaseController	OCP	Lớp phụ thuộc trực tiếp vào Cart, khiến việc mở rộng hoặc thay đổi logic của Cart yêu cầu sửa đổi trong BaseController.	Đã tạo ra CartService để định nghĩa và xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện
BaseController	DIP	Lớp phụ thuộc trực tiếp vào một implementation cụ thể (Cart) thay vì một abstraction (interface hoặc lớp trừu tượng).	Đã tạo ra CartService để định nghĩa và xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện
HomeController	SRP	Lớp HomeController chịu trách nhiệm điều khiển luồng sự kiện trong màn hình chính, nhưng cũng trực tiếp gọi đến phương thức lấy dữ liệu Media từ cơ sở dữ liệu.	Tách logic truy vấn dữ liệu Media ra một lớp riêng (ví dụ: MediaService), giúp HomeController chỉ tập trung vào điều khiển luồng dữ liệu và sự kiện.	Chưa thực hiện
HomeController	OCP	Lớp phụ thuộc trực tiếp vào lớp Media để lấy dữ liệu, do đó nếu cần thay đổi cách truy vấn dữ liệu (ví dụ: đổi từ SQL sang NoSQL), lớp này sẽ cần sửa đổi.	Tạo một lớp service (như MediaService) để xử lý việc truy vấn, và HomeController chỉ gọi qua service này, giúp mở rộng dễ dàng mà không phải sửa đổi lớp controller.	Chưa thực hiện
HomeController	DIP	Lớp này phụ thuộc trực tiếp vào implementation cụ thể của Media thay vì một abstraction. Điều này dẫn đến sự phụ thuộc chặt chẽ vào cách thực thi hiện	Tạo một interface (ví dụ: IMediaService) để trừu tượng hóa việc truy vấn dữ liệu Media, giúp giảm sự phụ thuộc của HomeController vào lớp cụ	Chưa thực hiện

		tại của lớp Media.	thể.	
PaymentController	SRP	Lớp PaymentController vừa xử lý luồng thanh toán, vừa chịu trách nhiệm làm trống giỏ hàng bằng cách gọi trực tiếp Cart.emptyCart(). Điều này vi phạm SRP vì lớp đảm nhận quá nhiều trách nhiệm.	Tách logic liên quan đến giỏ hàng (xử lý việc làm trống giỏ hàng) vào một service riêng như CartService để tăng tính tập trung của PaymentController.	Đã thực hiện
PaymentController	OCP	Vi phạm nhẹ. Nếu cần thay đổi cách làm trống giỏ hàng (ví dụ: chuyển từ xử lý cục bộ sang xử lý trên server), cần sửa đổi trực tiếp trong PaymentController.	Đã tạo ra CartService để định nghĩa và xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện
PaymentController	DIP	Lớp này phụ thuộc trực tiếp vào lớp Cart (implementation cụ thể) để làm trống giỏ hàng, dẫn đến phụ thuộc chặt chẽ.	Đã tạo ra CartService để xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện
PlaceOrderController	SRP	Lớp này chịu trách nhiệm xử lý luồng sự kiện đặt hàng, kiểm tra tính khả dụng, tạo hóa đơn, tính phí vận chuyển và validate thông tin giao hàng, dẫn đến đảm nhận quá nhiều trách nhiệm.	Tách các trách nhiệm này thành các lớp hoặc service riêng, chẳng hạn: ValidationService (validate thông tin), ShippingService (tính phí giao hàng), và OrderService (xử lý đặt hàng).	Chưa thực hiện
PlaceOrderController	OCP	Không vi phạm rõ ràng. Nhưng các phương thức như validatePhoneNumber, validateName và calculateShippingFee khó mở rộng vì chúng được triển khai trực tiếp trong controller.	Sử dụng chiến lược hoặc dịch vụ bên ngoài để hỗ trợ việc validate và tính phí, giúp mở rộng hoặc thay đổi logic dễ dàng hơn mà không cần chỉnh sửa controller.	Chưa thực hiện
PlaceOrderController	DIP	Lớp này phụ thuộc trực tiếp vào Cart và các chi tiết implementation của nó. Điều này làm giảm khả năng thay đổi hoặc thay thế Cart.	Đã tạo ra CartService để xử lý các tác vụ liên quan tới giỏ hàng	Đã thực hiện