

## I. Coupling:

### 1. PlaceOrder

#### 1.1 Content Coupling:

STT	Related Module	Description	Improvement
1	PlaceOrderController(createOrder)	PlaceOrderController sử dụng phương thức 'getlstOrderMedia' rồi dùng phương thức 'add' để thêm trực tiếp 1 order vào lớp Order	Sử dụng phương thức "addOrderMedia" thay vì sử dụng 'getlstOrderMedia' + 'add'

#### 1.2 Common Coupling

STT	Related Module	Description	Improvement
1	Cart(getCart)	Sử dụng phương thức getCart ở các lớp khác nhau để truy cập và thay đổi giá trị của thuộc tính static 'cartInstance'	Chuyển phần code sử dụng static ở lớp Cart thành các phương thức ở chính lớp này, khi các lớp khác cần thì chỉ gọi đến các phương thức đó

#### 1.3 Stamp Coupling

STT	Related Module	Description	Improvement
1	PlaceOrder(calculateShippingFee)	Phương thức chỉ cần sử dụng thuộc tính getAmount() của đối tượng order nhưng lại truyền cả đối tượng order vào	Thay tham số truyền vào "order" bằng tham số 'amount'

### 2. PlaceRushOrder

#### 2.1 Common Coupling

STT	Related Module	Description	Improvement
1	Cart(getCart)	Sử dụng phương thức getCart ở các lớp khác nhau để truy cập và thay đổi giá trị của thuộc tính static 'cartInstance'	Chuyển phần code sử dụng static ở lớp Cart thành các phương thức ở chính lớp này, khi các lớp khác cần thì chỉ gọi đến các phương thức đó

### 2.3 Stamp Coupling

STT	Related Module	Description	Improvement
1	PlaceRushOrder(calculateShippingFee)	Phương thức chỉ cần sử dụng thuộc tính getAmount() của đối tượng order nhưng lại truyền cả đối tượng order vào	Thay tham số truyền vào "order" bằng tham số 'amount'

## II. Cohesion

### 1. PlaceOrder

STT	Related Module	Description	Improvement
1	PlaceOrderController(validateName, validatePhoneNumber, validateDeliveryInfo,...)	Validate các trường dữ liệu lần lượt với các phương thức này, tuy nhiên chúng không liên quan với nhau về mặt chức năng nên vẫn thiếu tính kết dính và khó tái sử dụng	Tách các component này thành các module khác nhau và đặt ở vị trí phù hợp

### 2. PlaceRushOrder

STT	Related Module	Description	Improvement
1	PlaceRushOrderController(validateDate, validateDeliveryInfo,...)	Validate các trường dữ liệu	Tách các component này

		lần lượt với các phương thức này, tuy nhiên chúng không liên quan với nhau về mặt chức năng nên vẫn thiếu tính kết dính và khó tái sử dụng	thành các module khác nhau và đặt ở vị trí phù hợp
--	--	--	--

### III. SOLID

#### 1. PlaceOrder

##### 1.1 Single responsibility:

STT	Related module	Description	Improvement
1	PlaceOrderController	Có nhiều chức năng trong cùng 1 lớp (xử lý yêu cầu đặt hàng, tính phí, xử lý thông tin đặt hàng)	Tách phần xử lý thông tin(validate,..) thành 1 lớp mới
2	InterbankSubsystemController	Thực hiện 2 nhiệm vụ: điều khiển luồng dữ liệu, chuyển đổi dữ liệu api sang dạng controller yêu cầu	Tách thành 2 lớp riêng cho 2 nhiệm vụ trên

##### 1.2 Open/closed:

STT	Related module	Description	Improvement
1	PlaceOrderController (phương thức calculateShippingFee)	Khi muốn thêm một cách tính phí giao hàng khác, ta phải sửa đổi lại code cũ	Tạo ra 1 interface với phương thức trừu tượng là calculateShippingFee. Khi muốn thêm cách tính phí mới thì implement interface trên
2	Cart (phương thức calSubtotal)	Khi muốn thêm một cách tính phí giao hàng khác, ta phải sửa đổi lại code cũ	Tạo ra 1 interface với phương thức trừu tượng là calSubtotal. Khi muốn thêm cách tính phí mới thì

			implement interface trên
--	--	--	--------------------------

### 1.3 Liskov substitution

STT	Related module	Description	Improvement
1	Media	Phương thức getAllMedia ở lớp cha kì vọng trả về một List, tuy nhiên các lớp con kế thừa nó lại đang trả về NULL	Xóa bỏ các lớp con vì các lớp này không cần thiết phải override lại (do không phải phương thức abstract)

### 1.4 Interface Segregation

STT	Related module	Description	Improvement
1	InterbankInterface	Có những hệ thống Interbank không hoàn tiền lại cho khách hàng nên lúc đó, phương thức refund sẽ trở nên dư thừa	Tách phương thức refund vào 1 interface mới và chỉ thực thi nó khi cần

### 1.5 Dependency Inversion

STT	Related module	Description	Improvement
1	PaymentTransaction	Khi muốn thêm 1 phương thức thanh toán mới vào hệ thống sẽ gặp khó khăn do class PaymentTransaction đang phụ thuộc nhiều vào class CreditCard	Tạo ra class abstract PaymentCard, và lớp PaymentTransaction chỉ quan tâm đến lớp này

## 2. PlaceRushOrder

### 2.1 Single responsibility:

STT	Related module	Description	Improvement
-----	----------------	-------------	-------------

1	PlaceOrderController	Có nhiều chức năng trong cùng 1 lớp (xử lý yêu cầu đặt hàng nhanh, tính phí, xử lý thông tin đặt hàng nhanh)	Tách phần xử lý thông tin(validate,..) thành 1 lớp mới
2	InterbankSubsystemController	Thực hiện 2 nhiệm vụ: điều khiển luồng dữ liệu, chuyển đổi dữ liệu api sang dạng controller yêu cầu	Tách thành 2 lớp riêng cho 2 nhiệm vụ trên

## 2.2 Open/closed:

STT	Related module	Description	Improvement
1	PlaceOrderController (phương thức calculateShippingFee)	Khi muốn thêm một cách tính phí giao hàng khác, ta phải sửa đổi lại code cũ	Tạo ra 1 interface với phương thức trừu tượng là calculateShippingFee. Khi muốn thêm cách tính phí mới thì implement interface trên
2	Cart (phương thức calSubtotal)	Khi muốn thêm một cách tính phí giao hàng khác, ta phải sửa đổi lại code cũ	Tạo ra 1 interface với phương thức trừu tượng là calSubtotal. Khi muốn thêm cách tính phí mới thì implement interface trên

## 2.3 Liskov substitution

STT	Related module	Description	Improvement
1	Media	Phương thức getAllMedia ở lớp cha kì vọng trả về một List, tuy nhiên các lớp con kế thừa nó lại đang trả về NULL	Xóa bỏ các lớp con vì các lớp này không cần thiết phải override lại (do không phải phương thức abstract)

## 2.4 Interface Segregation

STT	Related module	Description	Improvement
1	InterbankInterface	Có những hệ thống Interbank không hoàn tiền lại cho khách hàng nên lúc đó, phương thức refund sẽ trở nên dư thừa	Tách phương thức refund vào 1 interface mới và chỉ thực thi nó khi cần

## 2.5 Dependency Inversion

STT	Related module	Description	Improvement
1	PaymentTransaction	Khi muốn thêm 1 phương thức thanh toán mới vào hệ thống sẽ gặp khó khăn do class PaymentTransaction đang phụ thuộc nhiều vào class CreditCard	Tạo ra class abstract PaymentCard, và lớp PaymentTransaction chỉ quan tâm đến lớp này