1. Coupling
   1. Content coupling

|  |  |  |
| --- | --- | --- |
| Related modules | Description | Improvement |
| PlaceRushOrderController | Phương thức createRushOrder() đang thêm trực tiếp một đối tượng orderMedia vào trong thuộc tính lstOrderMedia của đối tượng Order | Sử dụng phương thức addOrderMedia() thay vì lấy ra thuộc tính lstOrderMedia và thêm trực tiếp đối tượng orderMedia bằng phương thức add() của lớp HashMap |

* 1. Stamp coupling:

|  |  |  |
| --- | --- | --- |
| Related modules | Description | Improvement |
| PlaceRushOrderController | Phương thức updateShippingFee() truyền vào đối tượng order nhưng chỉ sử dụng mỗi phương thức getAmount() | Chỉ truyền vào phương thức updateShippingFee() một tham số có kiểu dữ liệu int biểu diễn cho tổng chi phí đơn hàng |

1. Cohesion

Các module của thiết kế cho “UC Place Rush Order” đã đạt được mức độ Functional Cohesion khi mà các subcomponent đều thực hiện một công việc nào đó hướng tới mục đích chung của component

1. Nguyên lý thiết kế SOLID
   1. Single Responsibility: Các module của thiết kế cho “UC Place Rush Order” đã thoả mãn nguyên tắc Single Responsibility
   2. Open/Closed:

|  |  |  |
| --- | --- | --- |
| Related modules | Description | Improvement |
| PlaceRushOrderController: Phương thức updateShippingFee() | Trong tương lai, nếu muốn tính phí ship theo các khác hoặc cần lưu nhiều kiểu tính ship khác nhau thì sẽ phải sửa trực tiếp vào phương thức | Tạo ra một interface ShippingFeeCalculator và implement 2 cách tính tiền ship khác nhau của Place Order và Place Rush Order bằng 2 class implements interface vừa tạo |

* 1. Liskov Subtitution: Các module của thiết kế cho “UC Place Rush Order” đã thoả mãn nguyên tắc Liskov Subtitution
  2. Interface Segregation: Các module của thiết kế cho “UC Place Rush Order” đã thoả mãn nguyên tắc Interface Segregation
  3. Dependency Inversion:

|  |  |  |
| --- | --- | --- |
| Related modules | Description | Improvement |
| CreditCard và PaymentTransaction | Hiện tại, PaymentTransaction đang phụ thuộc chặt chẽ vào lớp CreditCard. Nếu như sau này muốn thêm hoặc thay đổi phương thức thanh toán từ CreditCard sang một loại hình thẻ nào khác sẽ vi phạm Dependency Inversion | Tạo một lớp abstract là PaymentCard và lớp PaymentTransaction chỉ quan tâm tới lớp PaymentMethod này |