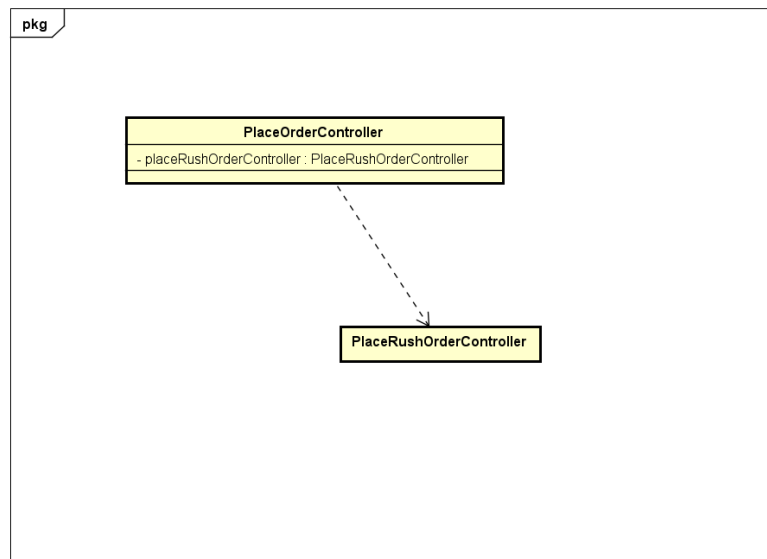


Design concept

Coupling: Data coupling

Module: PlaceOrderController



Mô tả: lớp PlaceOrderController gọi đến PlaceRushOrderController khi khách hàng muốn giao hàng nhanh.

Cải tiến: không

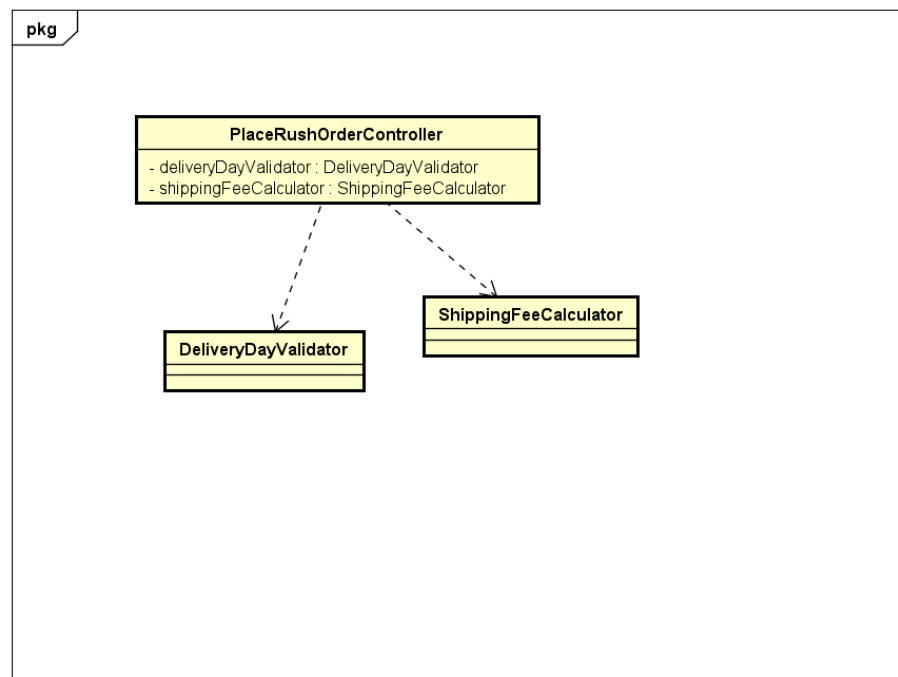
Cohesion: Procedural cohesion

Module: PlaceRushOrderController



Mô tả: Trong lớp PlaceRushOrderController thì có hai phương thức là validateRushDeliveryDay và calculateShippingFee, chúng được để trong cùng một lớp vì nó liên quan đến thứ tự thực hiện việc đặt hàng nhanh, chức năng của chúng là khác nhau.

Cải tiến: Có thể tách thành hai lớp, một lớp chịu trách nhiệm kiểm tra thông tin người dùng việc đặt hàng, một lớp chỉ để tính tiền.



Design Principles

Single Responsibility Principle

Related Modules	Description	Improvement
Class utils.Utils	Lớp này đang phụ trách các chức năng không liên quan đến nhau như getLogger, getCurrencyFormat, getToday, md5	Tách riêng ra thành các class riêng như utils.MyLogger (phụ trách getLogger), utils.CurrencyFormatter (phụ trách getCurrencyFormat), utils.MyDateTime (phụ trách getToday), utils.Encryptor (phụ trách md5)

Từ

```
public class Utils {  
  
    public static DateFormat DATE_FORMATER = new SimpleDateFormat("yyyy/MM/dd HH:mm:ss");  
    private static Logger LOGGER = getLogger(Utils.class.getName());  
    static { ...  
  
    public static Logger getLogger(String className) { ...  
  
    public static String getCurrencyFormat(int num) { ...  
  
    /** ...  
    public static String getToday() { ...  
  
    /** ...  
    public static String md5(String message) { ...  
  
}
```

thành

```
public class MyLogger {  
  
    private static Logger LOGGER = getLogger(MyLogger.class.getName());  
    static { ...  
  
    public static Logger getLogger(String className) { ...  
  
}
```

```
public class CurrencyFormatter {  
    public static String getCurrencyFormat(int num) { ...  
}
```

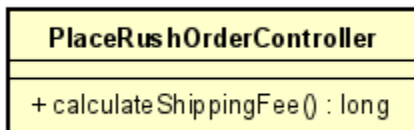
```
public class MyDateTime {  
    public static String getToday() { ...  
}
```

```
public class Encryptor {  
    public static String md5(String message) { ...  
}
```

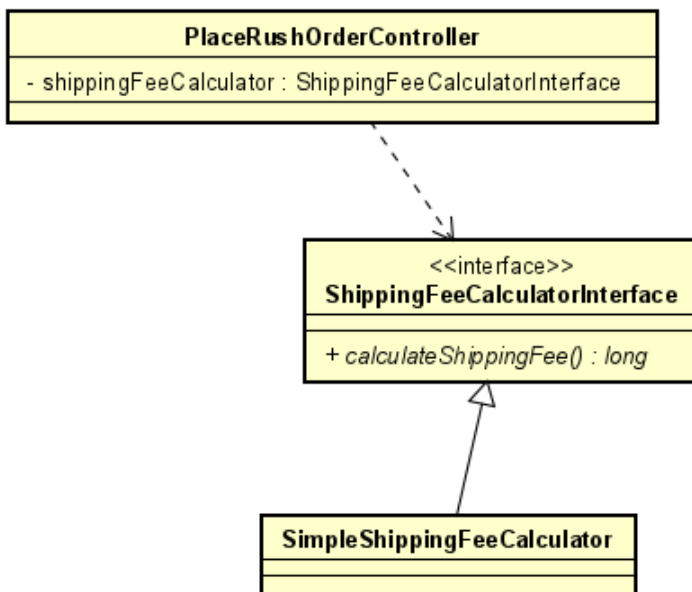
Open/Closed Principle

Related Modules	Description	Improvement
Method calculateShippingFee của class controller. PlaceRushOrderController	Phương thức tính phí calculateShippingFee hiện tại khó có thể thay đổi nếu không trực tiếp thay đổi code trong chính class controller.PlaceRushOrderController.	Tạo một interface ShippingFeeCalculatorInterface có phương thức trừu tượng calculateShippingFee, khi muốn cài đặt một cách tính phí mới thì tạo một class giả dụ SimpleShippingFeeCalculator thực thi ShippingFeeCalculatorInterface, và class controller. PlaceRushOrderController có một thuộc tính là thể hiện của interface ShippingFeeCalculatorInterface (class SimpleShippingFeeCalculator) phụ trách việc tính phí này

Từ

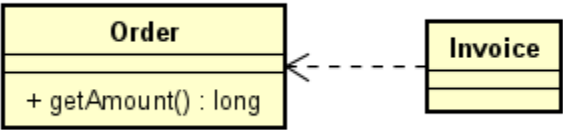


thành

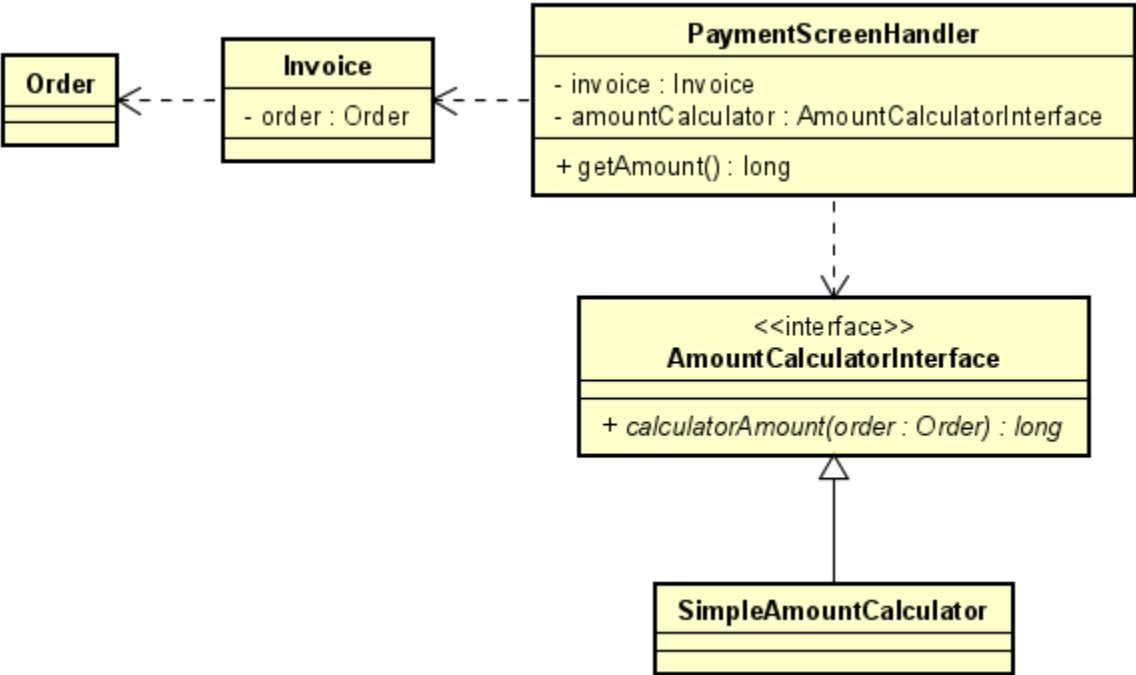


Related Modules	Description	Improvement
Class entity.order.Order	Khi muốn thay đổi phương thức tính tiền, ta phải thay đổi trực tiếp phương thức getAmount trong class này.	Tạo một interface AmountCalculatorInterface có phương thức tương tự getAmount, một class SimpleAmountCalculator thực thi interface trên, đồng thời class PaymentScreenHandler có một thể hiện của interface (class thực thi này) để phụ trách tính tiền

Từ

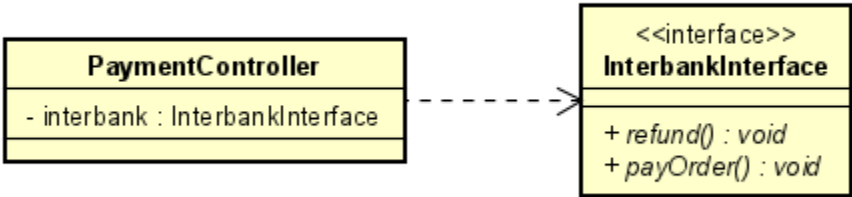


thành



Related Modules	Description	Improvement
Interface subsystem.interbank. InterbankInterface	Khi ta muốn sử dụng một Interbank khác, chưa chắc Interbank này đã cung cấp đủ method refund.	Tạo một interface RootInterbankInterface có method payOrder, InterbankInterface sẽ kế thừa interface này.

Từ



thành

