**Tên Yorn Bunthoeurn**

**Mssv: 20170287**

**Lớp CNTT 09 – K62**

**Thực hành 4:**

**a. Thiết kế các testcase cho UC Place Rush Order bằng phương pháp TDD như đã học và được hướng dẫn thực hành**

**b. Thực hành thêm javadoc cho các class, method và attribute cho UC Place Rush Order. Sau đó tiến hành review code và refactoring lại những điểm code chưa hợp lý (lưu ý cần chỉ rõ cần refactoring ở điểm nào và tại sao)**

**1. Thực hành thiết kế Unit test**

**a.**

ta sẽ thực hành việc thiết kế các testcases cho phương thức validateDeliveryInfo nằm trong controller PlaceOrder

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên tham số** | **Yêu cầu** |
| 1 | name | chỉ bao gồm chữ cái, không chứa ký tự đặc biệt, không được phép null |
| 2 | phone | chỉ bao gồm chữ số, độ dài 10 ký tự và bắt đầu là số 0 |
| 3 | address | không được phép null, không chứa ký tự đặc biệt |

**A screenshot of a computer

Description automatically generated with medium confidence**

b.Tạo UnitTest bằng Eclipse

Graphical user interface

Description automatically generated

Kết quả:

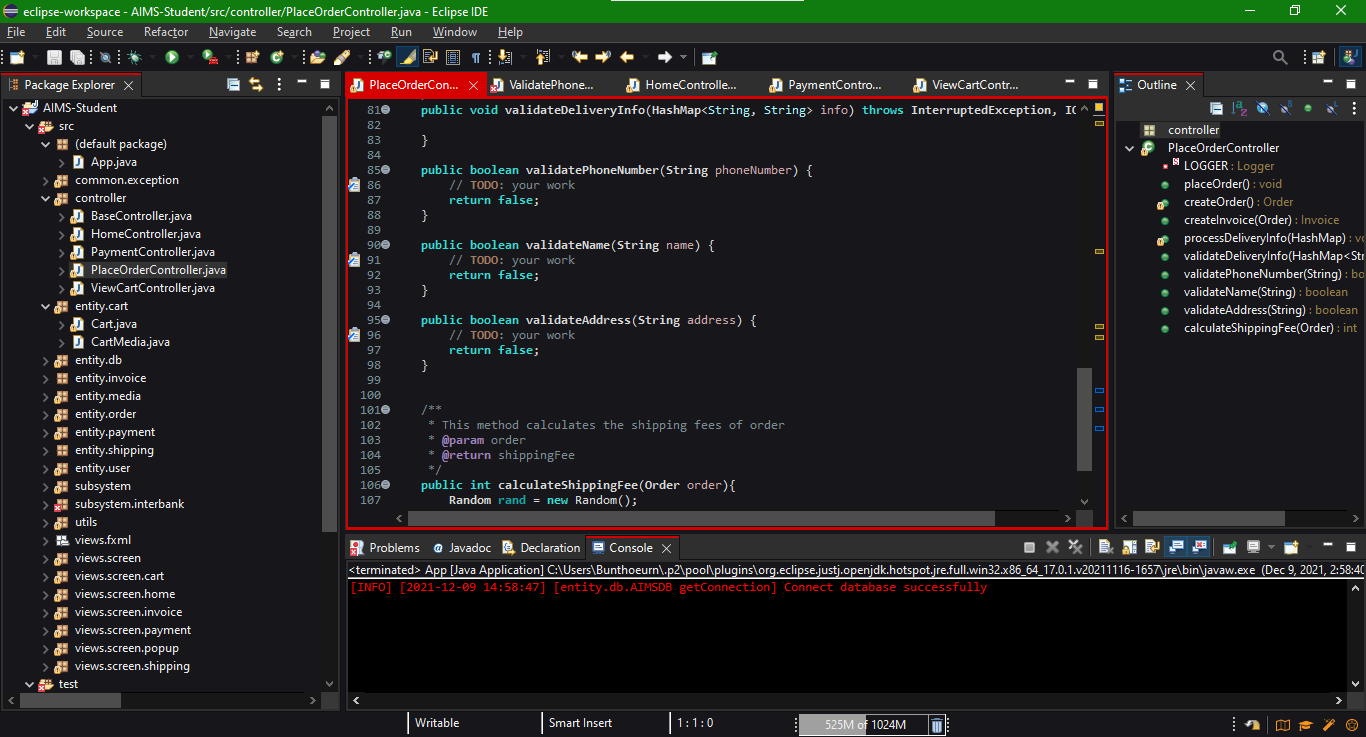
Text

Description automatically generated

c.Thực hành xây dựng các phương thức theo TDD

A screenshot of a computer

Description automatically generated



* Kết quả sử dụng antation @Parameterized và @CsvSource

A screenshot of a computer

Description automatically generated with medium confidence

-Phần Chạy Hàm App.Java

Graphical user interface, PowerPoint

Description automatically generated

d. Thực hành tạo test suite

Graphical user interface, text

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Bắt đầu với code sample

* Trước khi đến với bài học các bạn hãy xem qua một class có tên ***API.java*** khoảng 15-20p và hãy suy ngẫm các câu hỏi:
  + Class này mục đích
  + Class này do
  + Sử dụng class này như

Graphical user interface, text, application

Description automatically generated

Text

Description automatically generated

#### Thực hành tạo Javadoc với Eclipse

1. **Import project vào Eclipse**

* Mở Eclipse và import Project đã clone về máy, hoặc nếu bạn đã clone thì sử dụng git pull để lấy code mới nhất về nhánh master
* Mở class src/utils/API.java
* Tiếp theo chúng ta sẽ tiến hành thêm javadoc cho class và method ở trong class API.java này

1. **Thực hành thêm doc cho API class**

* Để thêm doc vào trong Java code thông qua Eclipse thì chúng ta có 3 cách:
  + Gõ thủ công
  + Dùng thanh công cụ
  + Dùng các shortcuts
* Để thêm javadoc cho class, method hoặc attribute thông qua thanh công cụ thì ta sẽ click con trỏ chuột lên ngay bên trên class, method hoặc attribute đó. Sau đó click chuột phải -> Source -> Generate Element Comment
* Cách nhanh nhất để thêm java doc chính là thông qua shortcurt. Để thêm javadoc thì ta click chuột lên phía trên của class, method hoặc attribute sau đó gõ /\*\* và ấn Enter, sau đó Eclipse sẽ tạo Javadoc tự động cho chúng ta
* Ví dụ với class API

Graphical user interface, text, application, email

Description automatically generated

* Việc thêm javadoc ở trên mới cho class API chỉ là bước đầu và khá sơ khai, vậy nên chúng ta cần bổ sung thêm các thông tin khác như: mô tả, @version, ngày viết chương trình
* Việc thêm mô tả cho một class, method hoặc attribute nên để lên đầu của documentation. Sau khi thêm các thông tin ta sẽ có thông tin documentation như sau:

Graphical user interface, text, application, website

Description automatically generated

* Sau khi thêm documentation cho class API thì chúng ta đã có một cái nhìn tổng quan về mục đích của class này . Tiếp theo chúng ta sẽ tiếp tục thêm documentation cho các attribute và method
* ***Attributes***

A picture containing graphical user interface

Description automatically generated

* ***get method***

Graphical user interface, text, application

Description automatically generated

* ***post method***

Text

Description automatically generated

* ***allowMethods***

Graphical user interface, text, application

Description automatically generated

1. **Thực hành thêm comment cho các phương thức**
   * Để làm rõ hơn các câu lện trong từng method thì chúng ta có thể thêm comment (//) vào trong các method như sau

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

* ***Extract method:*** phương pháp này có nghĩa là nếu như bạn thấy đoạn code nào đó lặp đi lặp lại nhiều lần trong các phương thức khác thì hãy tách đoạn code đó ra thành một phương thức riêng
* ***Extract class***: đây là cách làm tương tự như extract method nhưng ở mức độ class, những phương thức nào có liên quan tới nhau và hay được sử dụng thì chúng ta có thể tách ra một class khác và tái sử dụng bằng cách kế thừa hoặc kết tập

#### Thực hành refactoring code với class API

1. **Nhận diện vấn đề**

* Đây là thời điểm chúng ta sẽ quay lại với class API phía trên, nếu chúng ta để ý kỹ chúng ta sẽ thấy 2 phương thức get và post có nhiều phần chung được phân tách như sau:

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

* Sau khi nhìn lại 2 methods và có sự phân cách giữa các dòng chúng ta sẽ thấy giữa 2 phương thức post và get này có các đoạn code chung như setup connection và đọc dữ liệu trả về từ server,
* Ta có thể tiến hành refactor đoạn mã nguồn này bằng cách chọn đoạn mã nguồn cần extract, ấn chuột phải, chọn Refactor 🡪 Extract method.

1. **Thực hành refactoring**

* Chúng ta sẽ tiến hành trích xuất 2 phương thức đặt tên là setupConnection và readResponse
* Code chúng ta sau khi refactoring sẽ như sau
  + ***setupConnection()***

Graphical user interface, text, application

Description automatically generated

* + ***ReadResponse()***

Graphical user interface, text

Description automatically generated

* Khi đó 2 phương thức get và post sẽ còn lại như sau

Graphical user interface, text

Description automatically generated

Text

Description automatically generated