Software Engineering

Assignment # 2

Assignment Submission Date: 11-Sep -2020 (Friday)

Q#1: Apply Basis Path Testing on the following code

```
public double calculate(int amount)
   {
        double rushCharge = 0;
        if (nextday.equals("yes") )
        {
                rushCharge = 14.50;
        }
        double tax = amount * .0725;
        if (amount >= 1000)
         {
         shipcharge = amount * .06 + rushCharge;
         }
         else if (amount >= 200)
                {
                shipcharge = amount * .08 + rushCharge;
                 }
        else if (amount >= 100)
                {
                shipcharge = 13.25 + rushCharge;
                }
        else if (amount >= 50)
                {
                shipcharge = 9.95 + rushCharge;
                }
        else if (amount >= 25)
                {
                shipcharge = 7.25 + rushCharge;
                }
        else
        {
        shipcharge = 5.25 + rushCharge;
         }
total = amount + tax + shipcharge;
return total;
    } //end calculate
```

Scenario: Book Order Application

This application is to be used to enable users (these are sales reps for the bookstore) to process orders for books.

Use Case: Process customer order

Actors: Sales Rep (person), Inventory Management System (software), Billing Dept (person).

Precondition: A customer has requested to order a book, and Sales rep starts Book Order Application

Main Success Scenario:

1. User (Sales Rep) is told to specify Book Title.
2. User enters Title.
3. Application queries Inventory Manager determining if Title is in catalog
4. If Title is in catalog, Application queries Inventory Manager to determine the ISBN, the price of the book, and the number of copies in stock.
5. Application reports the book price and the number of copies in stock to the User
6. Application requests the number of copies to be ordered.
7. User enters the number of copies.
8. Application requests the name and address to ship the books.
9. User enters name and address.
10. If all the needed books are in-stock, Application submits a shipping order for the books to the specified address. This order will update the inventory.
11. Application creates a book order, specifying the shipping name and address, the number of copies of the book that were backordered and the number that are being shipped, as well as the total cost of the books being shipped. This information is displayed to the user.

Alternate Scenarios:

- If Title is not in catalog in step 3, Application reports this fact to User, and terminates.
- In step 7, if the number of copies is less than or equal to 0, the Application terminates.
- In step 10, if some (but not all) of the required books are in-stock, Application submits a shipping order for just the copies of the book that are in-stock. This order will update the inventory.
- In step 10, if not all the required copies are in stock, Application submits a back order (to the Inventory Manager) for the number of additional copies that are needed. This backorder will update the inventory.

For the given scenario

Q#2: Draw Use case diagram.

Q#3: Draw a UML Sequence Diagram to illustrate the above use case. The classes in this diagram should include "Inventory Manager" (The inventory management class that has been provided for you), "Session Manager" (The class that reads and writes console messages), and "Book Order", a class that contains the information associated with this order (the book title, the ISBN, the number to be shipped, the number back-ordered, the shipping name and address, and the total charge). Add any additional classes you think appropriate.

Q#4: Draw a UML Class Diagram that illustrates the classes and their associations.