Ryerson University

Department of Electrical & Computer Engineering
Faculty of Engineering & Architectural Science

Course title:	Object Oriented Analysis Design
Course number:	COE 528
Semester/Year (e.g.F2016)	Winter 2020

Instructor:	Dr. Olivia Das

Group Number :	36

Submission Date:	April 4th, 2021
Due Date:	April 4th, 2021

Student Name	**Student Number**	**Section**	**Signature***
Abdul Mohammed	500931563	7	marz//11
Rendel Abrasia	500942743	11	Rendel Abrasia
Reza Aablue	500966944	11	Reza.Aablue
Huzaifa Ali	500901727	11	Huzaifa Ali

Use Case Description :

- **Name :** Bookstore App

- **Participating Actors :** Owner, Multiple Clients/Customers.

- **Entry Conditions :** Participating actors are logged into the application with a unique username and password. The loginVerification method (abstract method) will check and verify the entered username and password and allow the user to log-in to the application.

- **Flow of events :** The flow of events are as follows:

    1) The customer logs into the application by entering their username and password, after which they have access to the customer start screen.
    2) The start screen displays a welcome message along with their points and status, located above the table of available books. The table displays all available books in the store along with their prices. The customer then has the option to select books they choose to buy, redeem points and then buy books or log-out.
    3) If the customer chooses to add books, they can proceed to either pay with redeemable rewards points or with money. They are shown the total cost along with their updated rewards points balance and member status. The member is either a silver status or a gold status member.
    4) Then, the customer can buy another book or log-out if needed.
    5) Meanwhile, if the owner logs in, they have access to their own start screen, giving them an option between “books”, “customers” and “log-out” buttons.
    6) If the owner chooses “books”, they have access to a screen displaying a table which includes all the books in the store along with their prices. The owner has the option to add or delete a book from the table.
    7) If the owner chooses “customers”, they have access to a screen displaying all the customers and their individual data, which includes usernames, passwords, and rewards points balance. The owner has the option to add or delete a customer from the database.
    8) Finally, the owner can log-out when they are finished with their work.

- **Exit Conditions :** The Owner and customers have “back” and “log-out” buttons to exit the application.

- **Exceptions :** Owners and customers are notified immediately if connection between them is lost.

- **Special Requirements :** None

Use of Design Patterns in the Project:

The state design pattern is used in this design project. The main idea behind using this pattern is that it allows an object to change its behaviour when its internal state changes. In our program it is important to know that the customer is made of basically two different types of status : Silver or Gold. In our class diagram, classes Gold and Silver inherit the state class which depends on the reward points. The state of a customer can be changed based on the points gained when books are bought. If the customer has less than 1000 thousand reward points they have a “Silver” status, else “Gold”. This is where the state design pattern application comes in, verification of a certain condition (rewards points in this case) changes the state of an attribute(customer).