

## Assignment 3

### CSCI 45000 - Principles of Software Engineering

**Due Date : 12/8/2020**

**Overview:** For Assignments 2 and 3, you will design and implement a very basic online store. The online store should handle the following basic events:

- Users/Administrators Login & Registration
- Browsing Items
- Updating Items
- Removing Items
- Adding Items
- Purchasing Items

**Goal:** The goal of this project is to help teach good object-oriented software design and implementation practices. Please find below the overall project description that we will be implementing through assignment 2 and 3.

In this **Assignment 3**, your job will be to build upon and improve your RMI based framework/skeleton and class diagram for the online store from Assignment 2. For this assignment, you will be looking at Application Control Patterns that can be applied to the Model-View-Control architecture that you defined in the previous assignment. For this you will be focus on implementing a simple **Front Controller** pattern with login and authorization process for both administrators and customers for your online store. Your design should also demonstrate at least one application of using **Command** and **Factory** or **Abstract Factory** patterns. For this project we will consider these to be two distinct roles – meaning an administrator account cannot act like a customer account – if the same individual wants to play both roles he/she must have two separate accounts. Any updates to your design should be reflected in an updated domain model and any updated discussion of design decisions should be made. These will be your tasks for assignment 3.

**All of the other requirements from Assignment 2 are still valid.** Any updates to your design should be reflected in an updated domain model and any updated discussion of design decisions should be made.

#### General requirements

Your code should be properly documented and should make use of common language standards and implementation practices.

**Deliverables for Assignment 3:** You are expected to submit the following items as part of Assignment #3. The following files should be submitted on Canvas in a zipped folder (e.g. StudentFirstname\_StudentLastName.tar.gz):

- Source Code (e.g. \*.java files) without any compilation or runtime error. [ 50 points]
  - A README file that outlines all of the files contained in the ZIP submission.
  - A MAKEFILE that can be used to compile the code
  - A JAR file to execute the code.

- **Assignment #3 Report** – an outline regarding any new classes you have created and any updated discussion of the transition from Assignment #2 to Assignment #3. Your code should run without errors for full credit. You should include sample runs demonstrating implementation of each required functionality [ 50 points].

## **Requirements**

**Requirements Description:** The following is a description of the clients request for an Online Store.

The client desires an online store where they can sell goods (and possibly services) to customers geographically dispersed around the world. Think Amazon but on a smaller scale and budget. Their desire is to have a system that is constructed in a portable language (Java) and makes use of their existing network. The system itself should present a view for the customer to interact with as well as a view for the employees or administrators of the company to interface with. For the customer there is a need for them to be able to browse available products – this should present the customer with the type, description and price of the item with the options to add to their shopping cart. If the customer attempts to add a quantity of the item more than the current supply the system should prevent the customer from adding these and prompt them with a message on the availability of the item. The customer should be able to also purchase their items from the shopping cart. The administrators should be able to update an item's description within the system, update its price, and update its quantity. The administrator should also be able to remove items from the system if so desired. Administrators should be able to add other administrators as well as add/remove customer accounts. A customer should be able to initially register for their account by themselves. The system should handle any faults or unexpected scenarios gracefully. It should be reliable and should allow for multiple customer requests during the course of execution.

Specific actions/events that should be handled as part of the system:

- Users/Administrators Login & Registration
- Browsing Items
- Updating Items
- Removing Items
- Adding Items
- Purchasing Items