SOEN 6441 Winter 2023 section WW: Assignment 1

Due: as indicated on Moodle

You must continue with the same group you developed part 1 with; no credit will be given otherwise.

Objectives:

- Unit testing
- 3-tier architecture/ 3-layer design
- Design patterns
- Mock objects
- GUI using JavaFX
- Generics
- Threading

Requirements for phase 2:

1. All the requirements of phase 1 are required in phase 2.
2. In phase 2, you need to provide a GUI using JavaFX.
3. The functionality of displaying all properties must run on its own thread.
   You also need to make sure the thread that is being used by the GUI is not used by any business logic process.

4. You were asked by the Architect to develop the following class since it will be needed in phase 17 that will be developed in 2026.

Develop a Set class

A set data structure is a collection of **unique** items with no specific order. For example, the sets A=(1,2,3) and B=(2,1,3) are the same. We need our Set class to be generic. That means, we should be able to create a set of identifiable objects. An Identifiable object must have a method called getID that returns an integer. The set will accept only elements with unique IDs. For example, if the set already has an element with ID=17, the set will not accept adding another element with ID=17.

The operation of the set are:

1) Add an element
2) Remove an element. This method takes an ID as a parameter and returns the element with that ID and removes it from the set.
3) Peek takes and ID and return true if there is an element with that ID in the set, false otherwise.
4) Size returns the number of elements in the set
5) Override Equals to test if two sets are equal or not. Two sets are said to be equal if they have the same number of elements and these elements have the same ID values.
6) Display all elements in the set.

When you demo your code, you need to provide a driver class that shows that all these functionalities of the set class work. The marker will ask you to carry out some operations.

Submission:

You must submit:

- A proper **detailed UML class diagram** in **pdf format** that shows
  * names and access modifiers of all attributes,
  * names, parameters, and access modifiers of all methods including constructors
  * proper relations between classes and interfaces
  * the design patterns you used
- **Sample runs** of the program that show all the functionalities required in **pdf format**
- **The code** itself

You will give a demo of your project to the TA. All group members must be present during the demo. No credit will be given without a demo. Any student should be able to answer questions about any part of the project.

Grading Rubric

GUI design and implementation	30 marks	Appealing, user friendly, designed well with suitable GUI elements
GUI is decoupled from the business logic	10 marks	
GUI thread is not used by any other business model process	10 marks	
The display properties runs on its own thread	20 marks	
The generic set class	20 marks	
Overall code quality	10 marks	Indentation/ comments / method names / variable names / class names

0 credit will be given if:

1- No UML is submitted. It is a must to submit the detailed UML class diagram OR
2- The code does not compile OR
3- No demo

One submission per group please. In the Moodle submission comment, please indicate the names and the ID's of the team members worked on the project. -2 if this information is not given accurately.