Class Project 1

CIS 476/566 Software Architecture and Design patterns
Posted: 02/03/2020
Due: 2/17/2020

A software company MicroOffice has produced four generations of Word Processing Applications, called Word90, Word00, Word10, and Word20. Suppose you are writing a program to test their GUIs. The GUI components we are interested in are **Panel, Button, and Textbox**. These GUI components look a little different in different generation. Each generation has it own program for testing these GUI components. To know which test to run, you will need to instantiate objects that correspond to each one of the GUI components.

We assume that generations of the Word to be tested are stored in a configuration file (text file). Because this situation fits the Abstract Factory pattern so well, you can use that pattern to organize the creation of objects that correspond to GUI components. You will also need to use the variation of singleton pattern to ensure that at most two instances of each generation in each test run. Please note finishing running all generations specified in the configuration file is considered as one test run.

Here is an example of the configuration file content. You can create your own.

```
Word90
Word00
Word90
Word20
Word10
Word00
Word20
Word90
Word00
```

Questions

1) Give the UML diagram. You should integrate singleton into abstract factory pattern.

2) Give the code (in any language) based on the UML class diagram given in 1). As output, you need to display three different messages (e.g., “Panel Word90”, Button Word90”, and “Textbox Word90”) for the generation specified in configuration file. You should give a warning message if the same generation are asked to run more than twice.

3) Zip your UML diagram, source code, output screen shot in one .zip file and upload to class project I folder in Canvas before due.