PROJECT REPORT

CS 586; Spring 2016

Deadlines:

Final Project: **May 2, 2016**

Late submissions: 50% off

After **May 5** the final project will not be accepted.

This is an **individual** project not a team project.

The hardcopy of the project must be submitted. Electronic submissions are not acceptable. Notice that the Blackboard project submissions are only considered as a proof of submission on time (before the deadline).

Outline of the Report & Deliverables

1. MDA-EFSM model for the *ACCOUNT* components
   i. A list of events for the MDA-EFSM
   ii. A list of actions for the MDA-EFSM with their descriptions
   iii. A state diagram of the MDA-EFSM
   iv. Pseudo-code of all operations of Input Processors of *ACCOUNT-1* and *ACCOUNT-2*
2. Class diagram(s) of the MDA of the ACCOUNT components. In your design you MUST use the following OO design patterns:
   i. State pattern
   ii. Strategy pattern
   iii. Abstract factory pattern
3. For each class in the class diagram(s) you should:
   a. Describe the purpose of the class, i.e., responsibilities.
   b. Specify responsibility of each operation supported by each class.
   c. Describe the purpose of main attributes of the class.
4. Dynamics. Provide two sequence diagrams for two Scenarios:
   a. Scenario-I should show as to how the deposit is made in the *ACCOUNT-1* component, i.e., the following sequence of operations is issued: *open(abc,xyz,100.5), login(xyz), pin(abc), deposit(400), balance(), logout()*
   b. Scenario-II should show as to how an incorrect pin is entered three times in the *ACCOUNT-2* component, i.e., the following sequence of operations is issued: *OPEN(123,111,1000), LOGIN(111), PIN(112), PIN(222), PIN(333)*
5. Source-code and patterns
   In this part of the report you should clearly indicate which parts of the source code are responsible for the implementation of the three required design patterns:
   - state pattern
   - strategy pattern
   - abstract factory pattern.
6. Well documented (commented) source code. **Printed hardcopy of the source code is required and should be a part of the report.** Otherwise, **10 POINTS** will be automatically deducted from the project.

A CD (or flash drive) with the executables of the *ACCOUNT* components with detailed instructions explaining the execution of the program should be submitted. If the executable(s) is not provided (or not easily available), **20 POINTS** will be automatically deducted from the project grade. The CD (or flash drive) should also contain the source code of your implementation. Note that the source code may be compiled during the grading and then executed. Electronic submission of the project will not be accepted.