Group 1
Initial Identification of Design Patterns

At this point you are starting to do a decomposition of the system description aiming to ultimately create a class diagram for the software system that you will design and build. Something that will help you to design in flexibility, clean separation of concerns, and evolvability will be to start thinking about use of design patterns also.

As you study each of the design patterns, you should start developing in your design toolbox tip offs or triggers for each of the design patterns. When you see that tipoff or trigger in a system description or statement of requirements, you start considering that the associated pattern may be appropriate.

This exercise, which should evolve as your design emerges, can provide the basis for the description of your design pattern information in your project releases.

Design Patterns Identification

Identify the standard design patterns that you will consider applying to solve this design problem. For each pattern that you are considering provide the following information:

- The generic Gang of Four name for the design pattern
- A name in the context of this application that clearly indicates to any knowledgeable reader exactly where it fits into this system
- List all the participant roles from the generic Gang of Four pattern. You get this directly from the class structure on your design pattern information sheet. To the right of each participant role, indicate all the nouns that would cover that role. If you have a participant role that does not have a noun playing that role, explain why not.
- Provide a short (several sentences) description of how the design pattern is being used in this application. This will most likely include information from the system description to show the connection back to this application.

Use one copy of the following template for each design pattern that you would use. Make as many copies of the template as you need.

Generic GoF Design Pattern Name: Strategy	
Pattern Name in terms of system context: System Algorithms / Market simulation algorithm	
List one per line the participants from the GoF class structure (add rows, if needed)	**Nouns from your table that take on that participant role. If there is no noun for a participant role explain why.**
Client	User
Context	Porfolio/ Button
Strategy	market simulation algorithm
ConcreteStrategy1	no-growth-market simulation algorithm
ConcreteStrategy2	bull-market simulation algorithm
ConcreteStrategy3	bear-market simulation algorithm
Market Simulation algorithm will be an interface that define a algorithm function. ConcreteStrategy 1, 2, and 3 will then implement that function. User can interact with the portfolio or some kind of buttons to call the algorithms.	

Generic GoF Design Pattern Name: Observer	
Pattern Name in terms of system context: System Menu Bar	
List one per line the participants from the GoF class structure (add rows, if needed)	**Nouns from your table that take on that participant role. If there is no noun for a participant role explain why.**
Subject/Observable	Portfolio
Observer	Buttons
ConcObserver	Open button
ConcObserver	Log out button
ConcObserver	Save button
…	…
User can click the “Open” button to open a file. The button will change the state of the portfolio. Then, the portfolio will send notification to all observers and update them with the new states.	