Exercise TDT4240: Patterns

The goal of this exercise is to illustrate how proper design facilitates an easier life. In this assignment, you are required to modify one of the code pieces you have written in the first exercise (introduction of technology), in order to practically learn how to use design/architecture patterns. You should select a program from the ones you have written, and try to improve the design with pattern techniques learnt in the course. Please follow these 4 steps below and submit your answer afterwards.

Step 1: Implementation of a program

Firstly, you should choose a program from first exercise--Introduction technology exercise. If it is too hard to use existing code, you can also write new code.

Step 2: Implement the Singleton pattern

Secondly, implement the Singleton pattern (http://en.wikipedia.org/wiki/Singleton_pattern) in the chose program. You can choose yourself what you should use the Singleton pattern for.

Step 3: Implementation of pattern(s) from the list

Modify the program to make use of AT LEAST ONE pattern choosing from the following list (choose appropriate functionality that fits the pattern chosen).

Patterns:

1) Observer http://en.wikipedia.org/wiki/Observer_pattern

2) State http://en.wikipedia.org/wiki/State_pattern

3) Template Method http://en.wikipedia.org/wiki/Template_method_pattern

4) Model View Controller http://en.wikipedia.org/wiki/Model_view_controller

5) Abstract Factory http://en.wikipedia.org/wiki/Abstract_factory_pattern

6) Pipe and filter http://en.wikipedia.org/wiki/Pipe_and_filter_architecture

Step 4: Theory

3.a) For the patterns listing in Step3, which are architectural patterns and which are design patterns? What are the relationships and differences of architectural patterns and design patterns?

3.b) How is the pattern you chose realized in your code? (Which class(es) works as the pattern you chose?)

3.c) Is there any advantages in using this pattern in this program? (What are the advantages/disadvantages?)