Exercise TDT4240: Patterns

This exercise aims to illustrate how proper design facilitates an easier life. In this assignment, you are required to modify one of the code pieces you wrote in the first exercise (introduction of technology) to practically learn how to use design/architecture patterns.

Deliveries:

Whether you complete the exercise individually whether in a pair of two students, you should submit the following:

1. A Zip file consists of the following programs codes with all of the files needed for running them:

- **Implementation program:** a program from the first exercise--Introduction technology exercise that you choose. You can also write new code if it is too hard to use existing code.
- **Singleton pattern implemented program:** A program in which you implement the Singleton pattern (http://en.wikipedia.org/wiki/Singleton_pattern) in the chosen program. You can choose for yourself what you should use the Singleton pattern for.
- **Singleton pattern implemented a program that is modified by pattern(s) from the following list:** A program that you modify the singleton pattern implemented a program by using AT LEAST ONE pattern choosing from the following list (choose appropriate functionality that fits the pattern chosen).
  Patterns:
  a) Observer http://en.wikipedia.org/wiki/Observer_pattern
  b) State http://en.wikipedia.org/wiki/State_pattern
  c) Template Method http://en.wikipedia.org/wiki/Template_method_pattern
  d) Model View Controller http://en.wikipedia.org/wiki/Model_view_controller
  e) Abstract Factory http://en.wikipedia.org/wiki/Abstract_factory_pattern
  f) Entity Component System https://en.wikipedia.org/wiki/Entity_component_system
  g) Pipe and filter http://en.wikipedia.org/wiki/Pipe_and_filter_architecture

2. Runnable Files

- APK files or a runnable JAR file (for LibGDX) of three programs

3. A report in a pdf format that you include these parts:

- **Part 1: Speak about the implemented program:** The chosen program from the first exercise--Introduction technology exercise or the new code.
- **Part 2: Speak about the implementation of the Singleton pattern:** Secondly, Speak about the implementation of the Singleton pattern (http://en.wikipedia.org/wiki/Singleton_pattern) in the chosen program. Comparing the original code and the singleton pattern implemented code in your report is recommended.
- **Part 3: Speak about the implementation of the pattern(s) from the lists of the pattern:** Speak about how you modify the program by using pattern(s).
- **Part 4: Answer these theoretical questions in your report:**
  4.a) For the patterns listed in Step 3, which are architectural patterns, and which are design patterns? What are the relationships and differences between architectural patterns and design patterns?
  4.b) How is your chosen pattern realized in your code? (Which class(es) works as the pattern you chose?)
  4.c) Are there any advantages to using this pattern in this program? (What are the advantages/disadvantages?)
- **Part 5: Comment:** Speak about if you completed the exercise individually or with a pair. If in the pair, say his/her name.