Stefan Bos 500645244
Ruben Herrebrugh 500614892
Christian Aguilera 500641645

Sourcing the Singleton

Assignment 1

Catalogs and libraries

- http://en.wikipedia.org/wiki/Design_pattern_(computer_science)
- http://www.dofactory.com/Patterns/Patterns.aspx
- http://www.javacamp.org/designPattern/
- http://www.blackwasp.co.uk/GofPatterns.aspx
- http://www.allapplabs.com/java_design_patterns/java_design_patterns.htm

Normally, the information you can find on the Internet is organized in the same way. Three different group of patterns: *Creational patterns*, *Structural patterns* and *Behavioral patterns*. Patterns in the first group focus on creating and storing new objects, easing the programmer this task. Patterns in the second group handle relationships between objects. Patterns in the last group help to identify common communication (and interaction) between objects. These groups were first named by Gang of Four.

A fourth group appears in some catalogs, although it wasn't mentioned by GoF. This is the *Concurrency patterns* group, to deal with programming that requires multi-threading.

Singleton pattern

Singleton pattern is used when only an instance of a class is needed. As the user of the class is provided only with one instance of the class, it guarantees that only one instance of the class is created. The only instance is a global object that can be invoked from every point of the code.

This pattern is a *Creational pattern* since avoids the user to create instances of a particular class.

For instance, this pattern can be used when an object behaves like a database. This database would be implemented as a *Singleton* class that only contains and manages references of other class of objects. Let's name this database as *ImageDatabase* – a database that contains references to objects of the class *Image* and provides the user with methods to store and retrieve images. Only an instance of the *ImageDatabase* class is created in the application using the *Singleton pattern*. When a certain image is required, the only instance of the *ImageDatabase* class would be invoked to get the reference to the image.