Innopolis University | Software Architecture | Spring 2016

Agile Techniques: Test Driven Development (TDD)

Néstor Cataño | **Homework – Week 04** | Due: 15/Feb/2016

Description

The goal of this laboratory is to conduct an experiment in which Test Driven Development (TDD) techniques are used for coding a class Task that adheres to the 3 software requirements below. Class Task encodes a task that is used in the planning of project activities. For instance, a project can be composed of several tasks that need to be realized to complete the project. Tasks have a planned start (completion) date and an actual start (completion) date. More concretely, class Task includes attributes "planned start date" (psd), "planned completion date" (pcd), "actual start date" (asd), and "actual completion date" (acd). The class implementation sets any of these attributes to -1 to mean that the attribute has not yet been given an initial value.

In the spirit of TDD, you should first design Java Unit (JUnit) Tests for class Task, and then evolve (change) its functionality (its methods) based on the errors produced by Eclipse during the execution of the Unit Tests that you wrote. The writing of the JUnit tests must be based on the description of the software requirements below[1].

Software Requirements

(i) The "planned start date" for a task is smaller than or equal to its "planned completion date".

(ii) If a task has an "actual completion date", then it also has an "actual start date".

(iii) The "actual completion date" for a task is bigger than its "actual start date" or it has no "actual completion date".

What to Hand In?

- An Eclipse Eclipse project in the format <YOUR-NAME>.ZIP that includes the code of class Task and the Java Unit tests that you wrote. The code of class Task is the evolved (modified) code of class Task that successfully passes the JUnit tests.
- The written JUnit tests must ensure test coverage. That is, you must test your Java implementation for all the possible **combination** of positive, negative, and zero values for psd, pcd, asd, and acd. And, you must write Java Unit tests for the 3 software requirements above.

[1] Check the slides of the Lab for a step-by-step description of this.

```
public class Task {
  private int psd, pcd;
  private int asd, acd;

  public Task() { psd = pcd = asd = acd = -1; }

  public int getPsd() { return psd; }

  public void setPsd(int psd) { this.psd = psd;}

  public int getPcd() { return pcd; }

  public void setPcd(int pcd) { this.pcd = pcd; }

  public int getAsd() { return asd; }

  public void setAsd(int asd) { this.asd = asd; }

  public int getAcd() { return acd; }

  public void setAcd(int acd) { this.acd = acd; }
}
```