University of Sofia "St. Kliment Ohridski"

Faculty of Mathematics and Informatics, Department of Software Engineering

Thesis:

The Influence of Practise "Test-Driven Development" Over the Quality of Software Design

Overview

Student: Teodor Klisurski
MSc Program: Software Engineering
Faculty Number: M-21547
Supervisor: Sylvia Ilieva, Ph.D.
Consultant: Iva Krasteva
Date: 17.10.2007

Software development is impossible without a well defined and measured process. The design is one of the most important phases of software development process. Design quality determines the quality of resultant software. So, there are a lot of efforts to improve the quality of software design. One way to do this, is to modify the software process. The Extreme Programming is an agile methodology for software development. It contains the following practices - "Pair Programming", "Continuous Integration", "Design improvement", "Small releases", "Coding standard", "Planning game" and "Test-driven development". Test-driven development requires unit tests to be written after the architecture is done and before creation of low level design and code. With this approach the software engineer can consider the system from a different point of view - how the system should work in different use cases. Thus, many system details can be seen and the software design becomes better.

The purpose of the thesis is to import the practice "Test-driven development" into an existing software process and to examine its influence on design of software system.

The first chapter is an introduction to the problem, it resumes the advantages of the practice and postulates the purpose of the thesis.

In the second chapter, there is a theoretical overview software processes and their comparison. The design phase is observed in each process. Both traditional and modern processes are shown. The core of Extreme Programming is described and its advantages and disadvantages.

The agile practice "Test-driven development" is presented in the third chapter. Its advantages and disadvantages are shown. Basic testing terms and conceptions are defined in this chapter.

The fourth chapter describes the experiment. A similar experiment and its results are presented in the beginning. After that, there is a describtion of our experiment - inclusion of the practice into an existing software process while creating a real application.

Results are presented and analysed in the fifth chapter. Appropriate metrics are defined and applied to measure the results. Then the results are analysed and the prescriptions are given to improve the software development process.

The conclusion summarizes the thesis. Some ways for future work are shown with the purpose of improving the software development process.

The last three parts contain bibliography, the glossary of terms and abbreviations and appendixes.