

Generation of test cases from use cases. A research line.

Javier Gutiérrez / javierj@lsi.us.es

1. The problem.

The goal of the software industry is to develop software system that satisfied the needs of their users and clients. The right implementation of the expected functionality of the system is an important amount of this satisfaction. Thus, it is vital to test the system under test to verify that it commits its functional requirements. This test phase is usually called system testing. System testing verifies the system through its external interfaces, from the point of view of external actors. System testing might involved, not only functionally, but everything that can be expressed as a requirement, like reliability, security, performance, navigability, usability, etc. However, our work is focused only in functionality.

2. Our quest.

The functionality of a future software system is stored in its functional requirements. Thus, functional system testing is mainly based on execute test cases for verifying that system do what their functional requirements said that it must do.

We focus our efforts over web and information systems. Nowadays, templates in natural language and UML diagrams are the most widely used artefacts to store and design functional requirements in information and web systems. The questions found are: How can we drive this process?. How can we design test cases from use cases?. Our thesis will be the answer in a formal, systematic and realistic way. Results have not only to be academic-valid, but they also have to be applicable in software industry.

3. Previous work.

We have performed for two years a survey about the state of the art in approaches for generation of test cases from use cases. Results concluded that there is a poor study of the problem. Many approaches only describe the problem in a superficial way, without offering documentation, models, techniques and tools.

However, our survey reveals the process (a set of workflows, activities and tasks) that we adopted for generation of test cases. This process could be based in the identification of use scenarios and the combination among scenarios. It is also important to derive test cases, not only from use cases in isolation, but sequences of use cases. There are classic test techniques, like domain partitioning, hat can be successfully applied to complement the generation process.

3. Goals.

We have identified the concrete goals of our thesis from the results of our survey. These goals are:

- Identify information needed for generate test cases and extends UML Use Case model to store that information.

- Define a complete process for generate test cases. In the scope of our work a test case is defined as actions among test case and system, test data and expected results. A complete process means that our work have to describe how to generate executable test scripts.
- Define a UML-based test model to represent the information needed by our process, and define the transformations between requirements model to test model.
- Include related aspects like measure of coverage and test case optimization.
- Allow the application of our process in early stages of the development process. Before the system is built.
- Build a supporting tool.

These goals are original due we have not found any approach that covers all goals expressed. In fact, we have not found any approach that proposed any UML-based model for testing or study related aspects.

4. The state of the art now.

Nowadays, we have a set of studies and results related to the goals listed above. The main result is our survey, cited before. We have also studied many open-source tools for system and web testing focused in Java platform. We have fund interesting ideas about how to define a language to express events among actors and system under test and how to design and build supporting test tools.

We have started from a concrete elicitation process called NDT and developed from one of the director of this thesis. This process defines several models for requirement elicitation. We have identified the role of every model in the testing process, and, nowadays, we are studying the improvements needed, as mentioned in goal 1.

We have developed a method and tool to generate test objectives automatically from NDT use cases. This method is only a part of our process for generating test cases.

5. Future work

The advantage of the thesis is slow, manly due the lack of academic previous works. We are investigating in several related fields due system testing, GUI testing, requirements elicitation and web testing.

At the time to write this text, we are formalizing our approach for generating activity diagrams from use cases and generate usage scenarios from the activity diagrams. Next steps will continue defining and formalizing the activities and task of our process.