Using FMEA for early robustness analysis of Web-based systems

Jianyun Zhou, Tor Stålhane
Department of Computer and Information Science, Norwegian University of Science and Technology, 7491 Trondheim, Norway
{jianyun,staalhane}@idi.ntnu.no

1. Introduction

Time pressure and quality issues are two main challenges facing today's web development professionals. To achieve quick development of high-quality systems, a lot of methods and techniques have been proposed. A widely recognized strategy in current practice is to emphasize early quality assurance techniques, as the late detection of defects are well known to be expensive and time-consuming. In this paper we take robustness as a critically important quality attribute, and propose a general framework for conducting early robustness analysis for web-based systems, based on Jacobson's analysis method [1] and FMEA (failure mode and effect analysis) [2].

2. Robustness concept

Giving a clear definition to robustness is not easy, especially when it is mentioned together with reliability. An elegant way to distinguish between them is to look at the types of faults that caused a problem. While reliability problem is often caused by internal faults within the system or component, robustness failure is caused by external faults rising from the operational environment, such as an unexpected input.

To formalise the difference, let's introduce a partition model for all the system operational conditions. It can be divided into three parts: SD, FD, and UD. SD (standard domain) refers to the set of all operational conditions for which a system satisfies its specification. FD (failure domain), refer to the set of all operational conditions for which the behavior of the system contradicts the specification. UD (unanticipated domain) contains the set of all operational conditions which are not included in the specification.

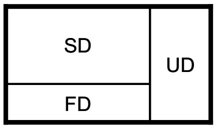

Figure 1 A partition over all operational conditions

Reliability is related to the failure domain. The smaller the failure domain is, the more reliable is the system. When FD = {}, the system is said to be correct regardless of whether UD is empty or not. Robustness concerns unanticipated domain. To improve robustness, measures should be taken to either enlarge the range of specification (completeness), or ensure system stability under unexpected environment.

This feature is very important for a successful web-based system. For the first, Web-based systems are accessed via the HTTP protocol, which has made such systems available everywhere. It is difficult, if not impossible, to control the input profile of end users. Web-based systems must have tolerance to errors and unexpected interactions from user environment. Secondly, Web-based systems are often not developed separately, integrating with existing systems (components or legacy systems) that are not produced specifically for the Web-based system. They must therefore be able to tolerate errors and unexpected interactions caused by these subsystems.

3. Robustness analysis method

As errors and misconceptions found in later phases of the system development process are expensive and time-consuming to fix, it is evident that a meticulous analysis of the system and its behavior should be carried out as early as possible in the development process. In this section we will present a robustness analysis framework that can be applied during analysis and preliminary design phases, based on Jacobson's analysis method and FMEA (failure mode and effect analysis method). The main purpose is to find robustness-critical elements of the system, and to identify preventive actions.

3.1. The role of Jacobson's analysis method

To propose such a framework, what are needed are two essential tools: one is a principled approach to modeling, and the other is a systematic method for carrying out analysis. For the purpose of modeling, we use Jacobson's analysis method. Like other modeling techniques, it captures the essential knowledge of the system. Unlike other techniques, it captures system's behavioral aspects, while not structural properties. This is the key feature that makes the method feasible in our framework. At early stages of the development cycle little information about system structure is available.

Jacobson's analysis method is based on Use Cases. By analyzing each use case, it identifies a set of objects that will participate in the use case, and classifies them into one of the following three stereotypes: Boundary objects, which the actors use when communicating with the system; Entity objects, which are usually objects from the domain model; Control objects, which server as the "glue" between boundary objects and entity objects.

Correspondingly, in a web-based system, boundary objects refer to the objects that the users will use to interact with the system. These are elements that compose a web page, such as hypertext, forms, menus, buttons, and so on. Entity objects often map to the database tables and elements in legacy systems. They represent resources required by use case execution. Control objects embody mostly application logic. They serve as mediator between the users and the stored data. This is where one captures the frequently changing business rules and policies.

Some rules are also defined to restrict interactions among these three types of objects:

1. Actors can only talk to boundary objects
2. Boundary objects can only talk to Control objects and Actors.
3. Entity objects can only talk to Control objects.
4. Control objects can talk to boundary objects, other Control objects and Entity objects, but not to Actors.

The contribution of Jacobson's method to our robustness analysis framework is two-fold. Firstly, it provides a practical and feasible way to model the system during analysis phase, decomposing the system into objects. Secondly, as Control objects capture application logic and manage all interactions between Boundary objects and Entity objects, they serve as natural placeholders for carrying out robustness analysis.

3.2. The role of FMEA

FMEA is a useful technique in risk management. It considers how each component of a system might fail and determine their effects on the required system functions. It is relatively time-consuming and requires detailed system information. Typically the FMEA is conducted late in the design process. Our purpose is to explore the possibility of using it earlier during analysis and preliminary stages. As a structural composition model is not available at that time, the analysis models from Jacobson's method can serve this purpose.

FMEA is carried out for each control objects. It considers all failure modes of each control objects, possible external causes rising from the operational environment, chance of occurrence, chance of detection, and severity of the effects on the system robustness requirement. The results are typically presented in a specially designed worksheet.

By combining these results, we can achieve following objectives:

1. Identify robustness critical elements, that is, the part of application logic that is prone to failure due to external errors or abnormal conditions
2. Rank & prioritize the possible causes of robustness failures
3. Develop preventive actions, to either eliminate possible causes or reduce the effects on system robustness

4. Conclusions

Including quality assurance activities early in the development process is an efficient way to enhance system quality. In this paper we have presented a general framework to carry out early robustness analysis for web-based systems. The proposed approach integrates Jacobson's analysis method and a light-weighted version of FMEA. As a result, we can identify robustness critical elements, rank and prioritize possible causes, and develop preventive actions.

However, the issue of using these methods in the proposed framework is not settled by merely outline such agendas. Many of the steps require research and effort to show their feasibility. This opens directions for future work.

5. References

[1] D. Rosenberg and K. Scott, *Use Case Driven Object Modeling with UML: A Practical Approach*, Addison-Wesley, 1999.

[2] J.D. Andrews and T.R. Moss, *Reliability and Risk Assessment,* Professional Engineering Publishing Limited, London and Bury St Edmund, UK, 2002.

0730-3157/04 $20.00 © 2004 **IEEE**