UNIFIED SOFTWARE REQUIREMENTS ELICITATION PROCESS OVERVIEW

SILHAVY, P[etr]; SILHAVY, R[adek] & PROKOPOVA, Z[denka]

Abstract: The requirements are essencial need for all of the software development process. The quality of the requirements elicitation therefore takes significant role in a software project success. In these article the Unified Requirements Elicitaion process is described in as a generic process
Key words: requirements elicitation, requirements process

1. INTRODUCTION

The software engineering takes role in the scientific and engineering area. Because of the system are more and more complicated.

The set of the system requirements describes the functionality of a system, its goals and constrains, which are applicable to the future system. The requirements engineering process takes very important role in the system engineering. Many systems are become software centric and in this background appropriate design of the functionality are more than important.

The aim of this contribution is to introduce and discuses benefits of employing the Unified Requirements Elicitaion engineering techniques in the system engineering.

According to (Genuchten, 2001; Hofnann, 1991) the system design projects have very often-important issues, which are connected to requiremetns analysis. The factors, which are the most problematic, are cost of the project, delays in terms and technical issues. The requirements engineering is the phase, which takes probably the most responsibility of named issues in the projects.

2. REQUIREMENTS

The requirement (Sommerville, 2007) is a description of the functionality or condition, which stakeholders define for the software. The requirements should be classified under the several criteria. Firstly is talked about the raw requirements (Silhavy, 2011) are list of functionality or condition for the proposed system, which is unanalyzed yet. The most important in this phase is to establish the project goals, which should be achieved.

The next group of the requirements is non-functional requirements. The purpose of these requirements is familiarized system designers with problem domain and conditions in the domain. Well-known examples of these are reliability, performance, safety or security. These non-functional requirements are critical in the system evaluation very often.
The next group is so-called system characteristics. The system characteristics are commonly prepared in negative way. It means, that system design specifies what irrelevant system behavior is.

Constraint requirements are used for set limits upon the design alternatives the systems. No matter how the problem is solved the constraint requirements must be adhered to.

The appropriate requirement is unitary and atomic; it means describes strictly individual part. Secondly the requirement has to be complete and consistent. This is very important in context of requirements contradiction.

3. REQUIREMENTS ELICITAION METHODS

The selected methods of the system requirements gathering are described in this chapter. There are many research methods, which were adopted or modified for requirements gathering or elicitation (Zowgi, 2005).

The Interviews; very common in either in the scope of system and software engineering (Sommerville, 2007; Zowgi, 2005). The interview is origin for social science research. It is human based activity. The requirements engineer has to be able to discuss with stakeholders or with future system users.

The interview has three basic types; structured, semi-structured and un-structured. The most valuable for the requirements definition is the un-structured interview. Contrastingly, un-structured interview have to be held by some experienced engineer. The structured interview commonly leads to missed some of important requirements. The set of questions is not well prepared and structured.

As well-working compromise; the semi structured interview, where basic set of the question is prepared and used.

The brainstorming is very useful addition to the semi-structured interview (Zowgi, 2005). If there are more then on stakeholders, the whole group is questioned in same time. The answers and discussions of the all group are noticed.

The laddering (Zowgi, 2005) is a technique, which is used as part of the brainstorming. It allows moderating the debate and brings hierarchical structures of the stakeholder (or stakeholders) answers.

The questionnaires; is probably the most important impersonal method. It is very useful in preliminary (Zowgi, 2005) requirements gathering. The questionnaires lack in discovering new facts or dimensions of the proposed system. Therefore have to be prepared by an experienced requirement engineer with huge knowledge of the problem domain.

The task analysis; an activity that is based on task decomposition (Carlshamre, 1996). The top-level tasks are designed first and then by top-down approach all sub-task are derivate and described. In the task analysis, users' tasks and system's tasks are at same level of importance. These tasks can be documented in form user or system stories, which were already described in the chapter 3. The individual task should be resulted in more than one story, even further in more than one requirement.

The observation; a method, which comes to system engineering from the ethnography and other sociological research. The observation allows observing users in their own environment. This method is valid for human-centric system design. For observation are valuable hidden cameras, with

respect to privacy. Users very often change their behavioral, if they are informed about observation.

The prototyping is one of the most valuable solutions for requirements capturing. The prototyping has no value for early phase of the requirements engineering. It allows determining very concrete and detailed requirements, in the time, when introductory requirements are already collected.

4. REQUIREMENTS ELICITAION METHODS

The Unfied Requirements Elicitation Process (Fig. 1.) contains these steps:

1. Application Domain Analysis
2. Stakeholders Identification
3. Users and System Stories
4. Prototype Creation
5. Functional Requirements Elicitation
6. Non-Functional Requirements Elicitation
7. Constrain Requirements Elicitation
8. Requirements Clustering.

The application domain analysis is very important phase. During this phase all of the environmental aspects are determined. Understanding of the domain concepts is very important for next steps.

The stakeholders Identification is next logical step, which is based on the domain analysis. In this step is necessary to resolve budget holder and all users of the modeled system. The list of stakeholder is created and then used for interview or questionnaires.

The domain analysis and stakeholder's analysis are then used for creation of user and system stories.

These basic stories are then reworked in form of prototype. This prototype is used in our proposed methodology for requirements gathering. We expected that prototype is appropriate for functional, non-functional and for constrain requirements elicitation.

The methodology offers interview in the form of brainstorming, which is moderate by laddering.

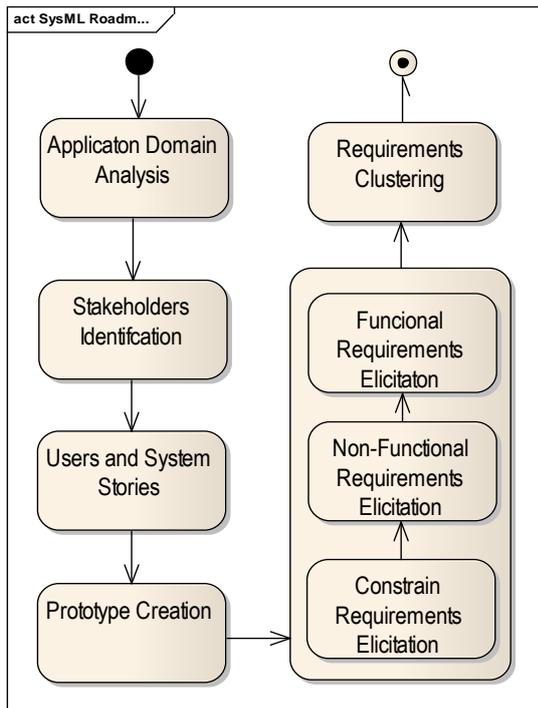The final phase of proposed approach is requirements clustering, in which system modules are designed.

Fig. 1. Unified Requirements Elicitation Process

5. CONCLUSION

The aim of the contribution was to describe requirements engineering as a mandatory phase which all development process start with. The requirements elicitation takes very important role in a project success.

In these article requirements gathering methods were described in context of the system development and finally the generic requirements engineering process were described. Further research will dealing with preparing of the modification of graphical notation for generic requirements engineering process.

6. ACKNOWLEDGEMENTS

7. REFERENCES

Carlshamre P, Karlsson J.A usability-oriented approach to requirements engineering. In: *Proceedings of the 2nd International conference on Requirements Engineer-ing*, April 15–18, Colorado Springs, CO. 1996

Hofmann H. F. et al., Requirements Engineering as a Success Factor in Software Projects. *IEEE Software*, vol. 18, no. 4.2001

Kamsties E., "Requirements Engineering: Dealing with the Complexity of Sociotechnical Systems Development." in Engineering and Managing Software Requirements, A. Aurum and C. Wohlin, Eds. Springer-Verlag, 2005

Object Management Group. Systems Engineering Domain Special Interest Group [online]. 2007-2011 [cit. 2011-03-20]. *Available on:* WWW: http://syseng.omg.org/

Silhavy R., Silhavy P. & Prokopova, Z. (2011). Clustered Requirements in System Engineering Project Estimation. *Internatinal Journal of Mathematical models and Methods in Applied Sciences,* Vol. 5, No. 1, (June 2011) page numbers (1052-1059), ISSN 1998-0140

SOMMERVILLE, Ian. *Software Engineering*. Eighth Edition. Harlow: Pearson Education Limited, 2007. 824 p. ISBN 978-0-321-31379-9

Zowgi et al. Requirements Elicitation: A Survey of Techniques, Approaches, and Tools, pp. 19-49. ISBN 978-3-540-28244-0