Requirements Gathering Methods in System Engineering

Radek SILHAVY, PETR SILHAVY, ZDENKA PROKOPOVA
Department of Computer and Communication Systems
Faculty of Applied Informatics
Tomas Bata University in Zlin
nám. T. G. Masaryka 5555, 760 01 Zlín
Czech Republic
rsilhavy@fai.utb.cz, psilhavy@fai.utb.cz, prokopova@fai.utb.cz   http://fai.utb.cz

Abstract: - The requirements engineering is mandatory phase which all development process start with. Mistakes in requirements elicitation therefore take very important role in a project success. In these article requirements elicitation methods are described in context of the system development and finally the generic requirements engineering process is described.

Key-Words: - requirements elicitation, requirements engineering, empirical methods.

1 Introduction

The system engineering is important discipline, which takes key 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 is more than important.

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

According to [2,3,4] the system design projects have very often important issues. 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.

The organization of this contribution is as follows. Chapter 2 describes the requirements classification. Chapter 3 discusses the methods of requirements documentation. Chapter 4 describes the gathering of requirements. In the chapter 5 is the discussion of the generic requirements engineering process. Finally chapter 6 is a conclusion.

2 Requirements definition

The requirement [1] is a description of the functionality or condition which stakeholders define for the system. The requirements should be classified under the several criteria. Firstly is talked about the raw requirements 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.

The requirements have to be verifiable. The implementation of the requirement can be determined through inspection or in form of some tests cases.

3 Requirements documentation

The requirements should be documented in form of free text, in form of structured text and in graphical notation.

The form of free text is common, but brings a many of misinterpretation issues [5]. In the free text description the technical jargon or acronyms have to be omitted.

The structured text and graphical form are commonly used together.

For user goals or for overall overview of the user needs, technique which is adopted from the agile methodologies (clanek 1 zdroj 22) is used. In the extreme programming, which belongs to the agile methodology group, user stories are used. In the extreme programming user takes important role in the software development.

User stories are forms or cards which contains goal of selected task. The tasks are described in free language without any internal structure. When user stories are written, vague subjects, adjectives, prepositions, verbs and subjective phrases are avoided.

3.1 Graphical Method - SysML

A requirement as graphical entity [6] is based on class model node, which its own stereotype, called requirement. This can be seen on the figure No. 1.

The requirements modeling constructs [6] (all examples and definition are adopted from this standard - [6]) are intended to provide a bridge between traditional requirements management tools and the other SysML models.

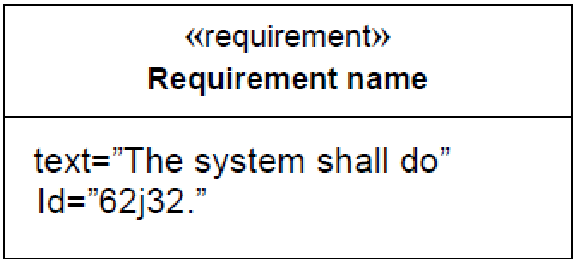

Fig 1: Individual Requirement Entity

A requirement [6] is defined as a stereotype of UML Class subject to a set of constraints. A standard requirement includes properties to specify its unique identifier and text requirement. Additional properties such as verification status, can be specified by the user.

Several requirements relationships are specified. These include relationships [6] for defining a requirements hierarchy, deriving requirements, satisfying requirements, verifying requirements, and refining requirements.

A composite requirement can contain subrequirements in terms of a requirements hierarchy. This relationship enables a complex requirement to be decomposed into its containing child requirements. A composite requirement may state that the system shall do A and B and C, which can be decomposed into the child requirements that the system shall do A, the system shall do B, and the system shall do C.

An entire specification can be decomposed into children requirements, which can be further decomposed into their children to define the requirements hierarchy.

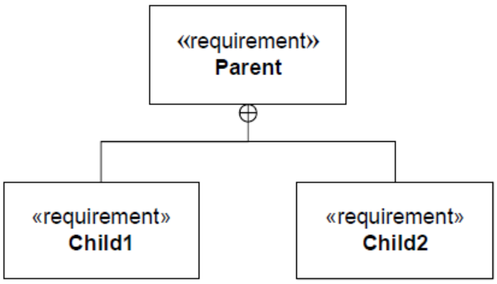

Fig 2: Containment Relationship

The “derive requirement” relationship relates a derived requirement to its source requirement. This typically involves analysis to determine the multiple derived requirements that support a source requirement. The derived requirements generally correspond to requirements at the next level of the system hierarchy.

The verify relationship defines how a test case or other model element verifies a requirement. In SysML, a test case or other named element can be used as a general mechanism to represent any of the standard verification methods for inspection, analysis, demonstration, or test. There can be tagged values – for example verification status.

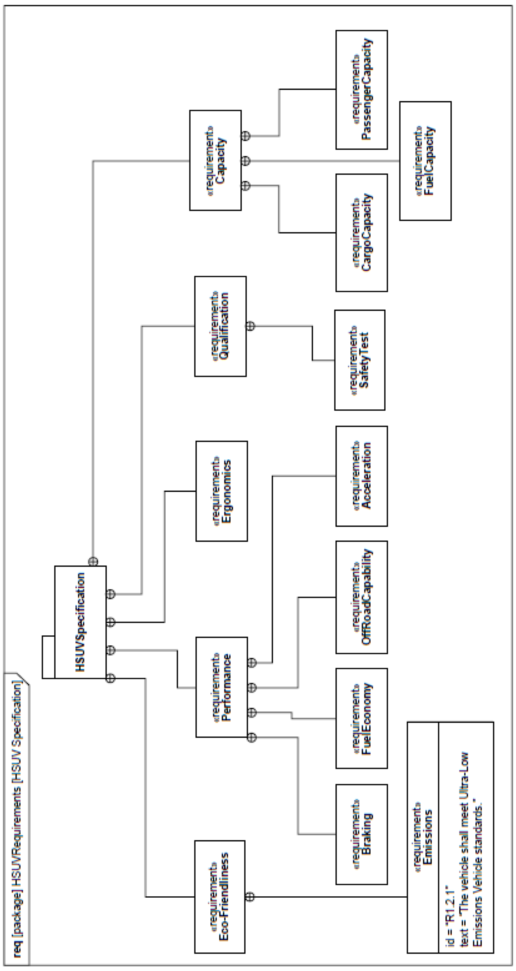
req [package] HSUVRequirements [HSUV Specification]
HSUVSpecification
«requirement» Capacity
«requirement» PassengerCapacity
«requirement» FuelCapacity
«requirement» CargoCapacity
«requirement» Qualification
«requirement» SafetyTest
«requirement» Ergonomics
«requirement» Acceleration
«requirement» Performance
«requirement» OffRoadCapability
«requirement» FuelEconomy
«requirement» Braking
«requirement» Eco-Friendliness
«requirement» Emissions
id = "R1.2.1"
text = "The vehicle shall meet Ultra-Low Emissions Vehicle standards."

Fig 3: SysML Requirements model

4 Requirements gathering

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 [7].

The Interviews; very common in either in the scope of system and software engineering [1,7] 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 tree 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 [7]. 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 [7] 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 [7] 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 which is based on task decomposition [7,8]. 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 to observe users in their own environment. This methods 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.

5 Requirements engineering process

The requirements engineering process represents a formal view of all necessary activities to achieve a complete and balanced system requirements.

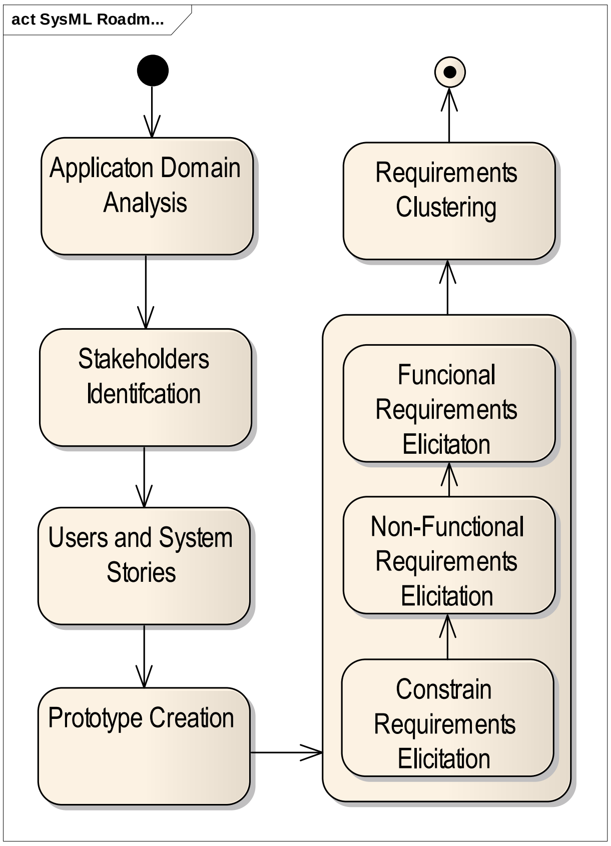
act SysML Roadm...
Applicaton Domain Analysis
Stakeholders Identifcation
Users and System Stories
Prototype Creation
Funcional Requirements Elicitaton
Non-Functional Requirements Elicitation
Constrain Requirements Elicitation
Requirements Clustering

Fig 4: Requirements gathering process

The requirements engineering process 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.

In the phase No. 5,6,7 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.

6 Conclusions

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.

Acknowledgments
This work was supported by the Ministry of Education, Youth and Sports of the Czech Republic under the Research Plan No. MSM 7088352102.

References:

[1] SOMMERVILLE, Ian. Software Engineering. Eight Edition. Harlow : Pearson Education Limited, 2007. 824 s. ISBN 978-0-321-31379-9.

[2] The Standish Group, "CHAOS Chronicles v3.0." The Standish Group, Tech. Rep., 2003. [Online]. Available: http://standishgroup.com/chaos/toc.php

[3] M. van Genuchten, "Why is Software Late? An Empirical Study of Reasons For Delay in Software Development." IEEE Transactions on Software Engineering, vol. 17, no. 6.1991.

[4] H. F. Hofmann and F. Lehner, "Requirements Engineering as a Success Factor in Software Projects." IEEE Software,vol. 18, no. 4.2001.

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

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

[7] Engineering and Managing Software Requirements. Berlin : Springer Berlin Heidelberg, 2005. Requirements Elicitation: A Survey of Techniques, Approaches, and Tools, s. 19-49. ISBN 978-3-540-28244-0.

[8] 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.