마인드 맵을 이용한 소프트웨어 요구사항 명세

강화, Scott Uk-Jin Lee
한양대학교 컴퓨터공학과
jh19880208@163.com, scottlee@hanyang.ac.kr

Extended Mind Map for Software Requirement Specification

Jiang Hua, Scott Uk-Jin Lee
Department of Computer Science & Engineering, Hanyang University

ABSTRACT

Requirement elicitation, analysis, and specification have been proven to be very important stages in a software development process. Software requirements are first gathered from different stakeholders through series of meetings and discussions, then analyzed and specified. Currently, requirements are often specified in natural language which is prone to produce ambiguity, misunderstanding and misinterpretation. This problem is one of well-known reasons for a software project to fail. Therefore, in this paper, we extend Mind Map to adequately specify software requirements that are easily comprehensible for both developers and various stakeholders in the project.

1. INTRODUCTION

In a large scale software development, one of the first and the most important stages are requirement elicitation, analysis and specification[1]. In the requirement elicitation stage, software developers gather various functional and non-functional requirements of the software to be built by conducting series of meetings and discussions with users, customers, and other stakeholders. Then the gathered requirements are analyzed and specified before actual software design and development commence.

The initial process of accurately specifying software requirements is critical to the success of a software project. According to the Standish CHAOS Report[2], one of the main reasons for project failure is 'incomplete requirements' as listed below.

Table 1 Project success factors and Reasons for project failure

Project success factors	Reasons for project failure
User Involvement	Incomplete Requirements
Management support	Lack of user Involvement
Clear statement	Moving Targets
Proper planning	Lack of Resources
Realistic expectations	Insufficient Communication

In a software project, it is often very difficult to extract and specify accurate and complete requirements because software developers are often not familiar with problem domain, user activities, and business environments while various stakeholders of the project are not familiar with technical possibilities and other software related issues. In order to resolve this matter, it is critical to provide an effective medium for communication between software developers and project stakeholders. Without an adequate common language, it will also be difficult to improve 'lack of user involvement' which is another main reason for possible project failure as indicated in the table 1.

Currently, natural languages and different UML models[3] are commonly used as means for requirement specification. But, requirements specified in a natural language can have ambiguity and produce possible misunderstanding and misinterpretation. In order to resolve such drawbacks, Friedrich et al.[4] and Chioaşcă[5] have developed automatic transformations for requirements specified in a natural language to requirement models. However, these approaches still suffer from various shortcoming of natural language process such as low matching rate. In addition, the generated models, such as OMS and BPMN, as well as the UML models are too technical and complex for various stakeholders of the projects to understand. Hence, with such a model, it is still difficult to produce accurate and complete requirement specification. Furthermore, this can also reduce the user involvement in the project and drive project to failure as indicated in the table 1.

Therefore, this paper proposes to extend Mind Map[6] for requirement specification. The idea is to adopt Mind Map as a cognitive and effective communication medium for both developers and stakeholders from requirement elicitation to specification. The Mind Map was chosen, as it is easily understood by all parties involved in the project. The use of the extended Mind Map to express requirements will in turn ease the accurate extraction of complete requirements and

increase the user involvement in the project.

The rest of the paper is organized as follows. The following section presents required background knowledge. Then, in section 3, the proposed extensions to Mind Map are discussed in detail. Finally, section 4 concludes the paper and introduces possible future works.

2. BACKGROUND

This section presents a brief overview of software requirements specification (SRS) and the Mind Map to provide background knowledge on the related field.

2.1 Software Requirements Specification (SRS)

The software requirements specification (SRS) describes the entire behavior of a software system to be developed for specifying complete requirements. The Use Cases can also be included in SRS to better describe user interaction with system. Also, SRS consists of various non-functional requirements in the form of constraints and verification to impose different constraints on the design or implementation of the software system. To illustrate this concept, the SRS of an online shopping system is presented in the below table as an example.

Table 2 Register User SRS

Use Case Name	Register user
Use Case No.	C01
Use Case Description	Register a new user before providing various functionalities of the system
Pre-Condition	N/A
Post-Condition	N/A
Basic Flow	1. Click register button 2. Go to register page 3. Input required information 4. Click register button 5. Success
Constraints	1. Name: String, length 20, not null Verification: a name to be registered does not exist already 2. Password: String, length: at least 6 Composed of only numbers and characters 3. Confirm Password: enter the same password again for confirmation Verification: entered password and confirm password must be identical 4. Real Name: not null 5. Phone No.: not null

2.2 Mind Map

The Mind Map is a diagram where words or texts are connected without restrictions to visualize or organize different ideas and concepts as single outline information. In other words, it is a radial diagram that cognitively represent and model a concept or a specific domain through a single word or text in a certain and correlated order[7]. According to Buzan[6], the main benefits of using Mind Maps are: ability to collect and report ideas with the use of keyword or word organizations; providing visual memory; enriching creativity and innovation. In addition, Mind Map is easily comprehensible for anyone due to its simplicity. To illustrate this concept, the below figure presents functional modules of an online shopping system modeled using Mind Map.

Figure 1 online shopping system Mind Map with function modules

3. MIND MAP FOR SRS

This section describes how Mind Map is extended to better specify the requirements. The Mind Map is extended with new naming conventions, notations, text boxes, and flowchart to enable representation of the entire elements in SRS documents. Note that the SRS example shown in the Table 2 will the used throughout this section to better illustrate the proposed extension.

3.1 Naming Convention

As an initial extension to Mind Map, the concept of Functional Module is introduced to categorize the similar functions together in a Mind Map. Also, some naming conventions are newly introduced to better represent functions and its modules.

- Function Module Name: Subject + Verb or Noun Phrase
  e.g. User Management, Goods Search, etc.
- Function Name: Verb + Object
  e.g. Register User, Login User, etc.

3.2 Use Case Description

In order to display the Use Case description of SRS, the Mind Map is extended with a highlighted text box appearing above the function node on mouse over as shown in Figure2.

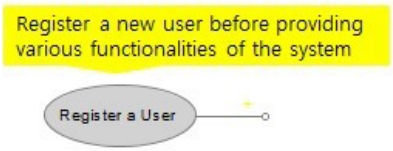
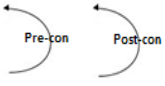


Figure 2 Use Case description in Mind Map

3.3 Pre-Condition and Post-Condition

Arrowed arc lines with text as shown below are introduced newly to Mind Map for describing Pre and Post-Conditions.



the arrows points to a functon that act as a Pre or Post-Consition of the current function

3.4 Basic Flow

In order to present the basic flow in SRS, the flowchart is added to the Mind Map where a click on a function node will allow user to view or edit the flowchart as shown in Figure 3.

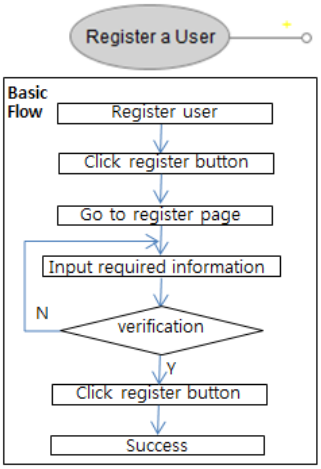


Figure 3 Mind Map showing the Basic Flow of a function

3.5 Constraints

As a last extension, each function node in the Mind Map is attached with another sub tree structure to express the constraints in SRS as shown in Figure 6. This sub tree is hidden initially and will be shown when the nodeless edge connected to the function node is clicked.

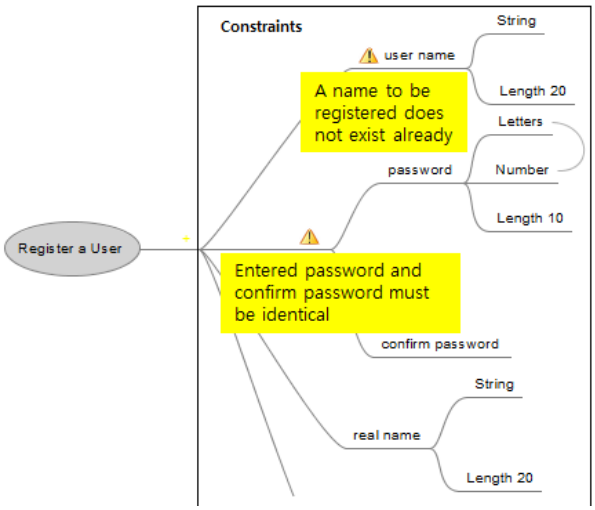


Figure 4 Constraints in Mind Map

The ⚠ icon appearing in the constraints sub tree indicates that this branch corresponds to verification specified in SRS whereas the highlighted text box below the icon presents the actual verification content.

Note that two new notations of dependency are introduced in the Mind Map to better specify constraints. These notations and their meanings are shown below.


all nodes connected with an arc line must be selected


only one nodes among multiple nodes connected with an arrowed arc line can be selected

4. CONCLUSION AND FUTURE WORKS

In a large scale software project, incomplete requirement and lack of user involvement are few of the main reasons for project failure. These problems often appear because there is no adequate means to clearly specify requirements and further use them as an effective medium to communicate with stakeholders. In order to resolve this issue, various researchers have proposed requirement models generated from SRS using natural language processing. However, these models still suffer from ambiguity, misunderstanding, and misinterpretation. Therefore, this research proposes the extended Mind Map to better specify requirements. The proposed method reduces ambiguity and improves correctness and completeness of the specified requirements by organizing and specifying necessary Use Case elements in Mind Map. At the same time, it makes specified requirement less technical and complex for stakeholders to understand when compared to conventional Use Case specifications. Furthermore, the extended Mind Map can act as an effective communication medium for developers and stakeholders, resulting in a better user involvement in software projects. Therefore, the proposed research can surely increase the project success rate by providing more complete requirements and better user involvement.

The possible future works are: first, to further improve readability and understandability of the extended Mind Map with addition of icons or notations; second, to further extend the Mind Map to display the process flow of the entire system; and third, to support automatic generation of Use Case Diagrams from the extended Mind Map.

5. REFERENCE

[1] I. Sommerville, “Software Engineering”, 9th edition, pp.94-110, Addison-Wesley, 2009

[2] I. Attarzadeh, “Project Management Practices: The Criteria for Success or Failure”, Communications of the International Business Information Management Association (BIMA), vol. 1, pp.234-241, 2008

[3] E. Hull, K. Jackson, and J. Dick, “Requirements Engineering”, 3rd edition, pp.1-20, Springer-Verleg, 2010

[4] F. Friedrich, J. Mendling, and F. Puhlmann, “Process Model Generation from Natural Language Text”, Lecture Notes in Computer Science, vol. 6741, pp.482-496, 2011

[5] E. Chioaşcă, “Using Machine Learning to Enhance Automated Requirements Model Transformation”, In the proceedings of the 34th International Conference on Software Engineering (ICSE), pp.1487-1490, 2012

[6] T. Buzan, “The Mind map Book”, BBC Active, 2006.

[7] F. Wanderley, D. S. Silveira, J. Araujo, and M. Lencastre, “Generating Feature Model from Creative Requirements using Model Driven Design”, In the proceedings of the 16th International Software Product Line Conference (SPLC), pp.18-25, 2012