Issues and Challenges of Requirements Review in the Industry

Siti Osnita Mokhtar, Rosmawati Nordin, Zalilah Abd Aziz and Rashidah Md Rawi

Department of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia; osnita@gmail.com, roswati@tmsk.uitm.edu.my zalilah@tmsk.uitm.edu.my, rrawi@tmsk.uitm.edu.my

Abstract

Background/Objectives: This paper highlights various issues and challenges of requirements review in the industry, based on existing literature reported by fellow researchers and practitioners. **Methods/Statistical Analysis**: A literature review method was adapted. The literature search was conducted based on several keywords, namely, requirements validation, requirements review and validation techniques. Subsequently, critical analysis of the available literatures was conducted using the thematic analysis approach. **Findings:** Overall, the issues and challenges of requirements review in the industry are categorized into six perspectives: review process, reviewer team, requirements specification, validation support tool, organizational culture, and governance. These six categorical barriers complicate successful executions of requirements review, which result in cost overrun and delayed project implementation. It is aspired for these findings to serve as a reference for researchers and industry key players possessing interest in requirements validation research area. Industry practitioners can also use these findings to make an informed decision regarding the requirements review process in their organizations. **Application/Improvements:** Future work of this research is to propose a framework which provides guidelines for better requirements review execution. Further research is also necessary to address the challenges highlighted.

Keywords: Requirements Engineering, Requirements Review, Requirements Validation, Software Development

1. Introduction

In a software development project life cycle, requirements engineering is one of the most important early phases. Requirements engineering phase is divided into five main activities, which are requirements elicitation, requirements analysis, requirements specification, requirements validation and requirements management[1,2].

The importance of requirements-related activities for a software development project has been widely acknowledged. Requirements engineering typically focuses mainly on the system identification and stakeholders requirement[3,4]. The success or failure of the project is attributed to the quality of the requirements[5].

Requirements validation activity aims to certify that the software requirements specification is the acceptable description of the proposed system, before it is used as the basis in the next phase of software development life cycle. Any requirements defects will affect the subsequent stages of system design and implementation[6]. Additional resources, time and effort will be required to fix such error, negatively impacting the project cost and schedule[7,8].

Various techniques can be employed to detect defective requirements, such as requirements review, prototyping, testing-based validation, model-based validation and viewpoint-oriented validation[2,4,6,8,9]. These techniques can be used individually or combined depending on the complexity of the requirements for the system to be developed.

Requirements review in particular is the most common, widely accepted and utilized means of requirements validation by industrial key players[5,8,10]. However, a num-

ber of obstacles caused by this technique complicate the task of achieving requirements validation goal.

The purpose of this paper is to highlight the major issues and challenges of requirements review in the industry. This 'Introduction' section touches on the background details, while the rest of the paper is organized as follows: Section 2 provides a brief overview about the requirements review, while Section 3 presents the various issues and challenges it poses. Finally, Section 4 concludes this paper and extends the recommendations for future works related to the research area.

2. Requirements Review

Requirements review is a manual validation technique, which can be grouped into several types. They include commenting, walkthrough, inspection and reading techniques[4,8,10]. The review is normally performed through a multiple stakeholders' collaboration, which typically includes the users, customers and software development project team[1,11].

As illustrated in Figure 1, requirements review is accomplished by performing the following tasks:

1. Plan review: Identification of relevant stakeholders to setup the review team and arrangement of the time and place for the review session.
2. Distribute document: Distribution of the software requirements specification to the members of review team.
3. Pre-review: Initial preparation by individual reviewers for error identification in the software requirements specification prior to the review meeting.
4. Hold review meeting: Conduct the review session to discuss and recommend possible actions to address the requirements defects.
5. Follow-up actions: Amendment of the software requirements specification and follow-up actions to rectify the defect.
6. Revise document: Finalize the software requirements specification or plan for subsequent review iteration.

Three major types of requirements documented in software requirements specification are business requirements, user requirements and functional requirements. Firstly, business requirements describe the project's high-level objective, while conversely, user requirements specify the tasks that the system is expected to be able to

accomplish, by the user. On the other hand, functional requirements provide the description of the proposed system's behaviour[6,12,13].
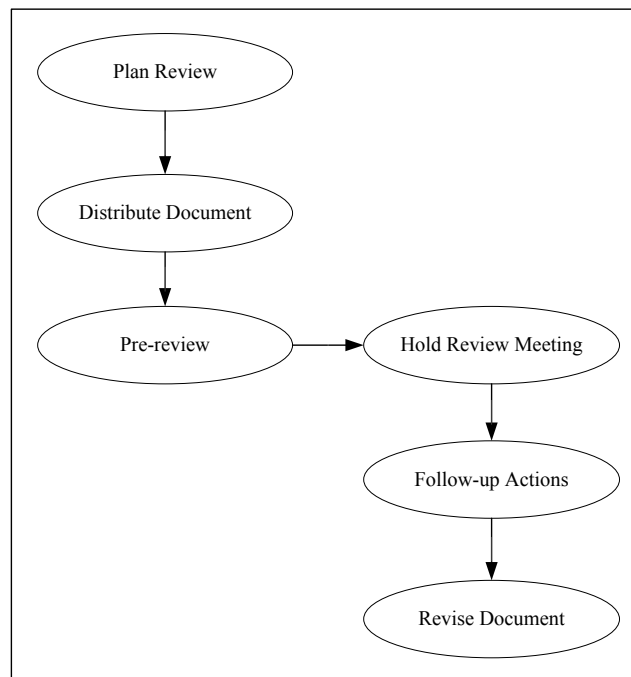

Figure 1. Requirements Review Process.

Requirements review is a significant task that ensures the accuracy and completeness of the software requirements specification, in order to conform to both the business needs and user requirements[11]. However, there are various issues and challenges encountered, as described in the next section.

3. Issues and Challenges of Requirements Review

Ow and Yaacob[14] have identified several problems related to requirements review. They mentioned these various difficulties: lack of ability to recognize the defect in the requirements; unrealistic requirements by the end users; lack of commitment from reviewer team; conflict among reviewers; and lack of support tools for review activity.

Some of these issues have remained unresolved for almost two decades. They contribute to the adverse effects impacting on the success of software development project. This paper intends to revisit the problems within this research area. Major issues and challenges of requirements review, as highlighted by industry experts and researchers can be categorized into several perspectives:

3.1 Review Process

Swarnalatha and Srinivasan[15] have highlighted that there is limited guidelines on the requirements specification's validation process. Requirements engineers are unsure of the requirements aspects that require reviewing with the project stakeholders[16]. This will inevitably lead to the risk of overlooking any missing requirements[17].

According to Wiegers and Beatty[6], searching for any error in the requirements is a challenge for the reviewers, due to the lack of a standard checklist. Additionally, Lee et al.,[17] have argued that the criteria for the validation process are vague. A reviewer may interpret the requirements based on his/her own understanding, while another might understand the same item differently based on his/her interpretation. As a result, there is a risk of rework in the system design or source codes modification if the error in the requirements is belatedly discovered at a later phase of the project6.

Requirements review is an abstract analysis of the textual requirements. It is difficult to detect whether the requirements specification is according to the user's needs. Sommerville[2] has asserted that this hindrance encountered by the requirements engineers is even harder for the users during the review process.

Reviewers must know of the reading technique and understanding of the software requirements specification[8,10,18]. This is usually a technical aspect of an expertise that users usually lack[16]. Therefore, error in the requirements specification may not be fully discovered during the review meetings. This may cause additional efforts for error rectification, even though the requirements specification has been signed off by the users[2].

3.2 Reviewer Team

Organizing a requirements review session with a large team of reviewers is typically ineffective. Wiegers and Beatty[6] have stated that scheduling a group meeting requires involvement from all the reviewers. This is both challenging and inconvenient for a geographically dispersed team.

The availability of individual reviewers will also pose as an issue as review meetings are normally conducted in several sessions[8,19]. All the reviewers may not be able to meet at the same time due to work commitments. Longer time and additional costs may be required to conduct the requirements review activity.

Moreover, reviewers usually are unprepared for the requirements review session. They fail to read the requirements specification beforehand, and thus, are unable to comprehensively compile the list of defects. As a result, the review session will require more allocated time. The reviewers may only start reading the document and identifying the defects during the review session itself. Eventually, they will fall victim to boredom and lose interest in the review session[6,8].

The reviewers can also get side-tracked by other aspects. For example, they may show more interest in discussing the user interface design of the proposed solution or expressing their diverging views regarding the work's project scope. They might also have disputes over any issues found, or arguments on whether the issue discovered is really a defect, or debates on the solutions of an identified problem[6,20]. These distractions render the review sessions ineffective.

For complex software development projects like embedded software or safety-critical system, reviewing the requirements specification is prone to errors as reviewers may not be able to detect the documentation's shortcomings. Consequently, the project team will need to employ a validation technique to minimize human error. However, this technique necessitates for the reviewers to have some technical knowledge of the field. Therefore, it can only be performed by qualified technical people, such as the requirements engineers, while the non-technical reviewers will not be able to participate[21].

Yousuf et al.,[22] have reported that the global software development team also encounters a number of issues compared to the co-located development team. As the team members are normally dispersed across different locations, this has caused miscommunication and mistrust issues among them. The lack of informal communication is also one of the reasons that contribute to these issues.

The team also highlights other issues, such as the lack of quality control, inadequate knowledge sharing and delays in problem resolution. These issues typically occur due to the shortcomings associated with distance, time zone and cultural differences between members of the team.

3.3 Requirements Specification

A vast majority of the requirements specifications are written in the natural language text. Due to the complexity

and abstract nature of the text, their validation process is a labour-intensive and error-prone task[21,23]. Moreover, the size of the requirements also impacts the review activity. Reviewing a thick requirements specification containing a large amount of information is a tedious, costly and time consuming task[8,17]. According to Swarnalatha and Srinivasan[15], standard guidelines and clear criteria on reviewing such lengthy documents are also lacking.

Satish and Anand[24] have explained that the low quality of requirements can be attributed to poorly defined software requirements specification. Usually, common defects found are due to requirements being either incomplete, ambiguous, inconsistent or incorrect[6,21,25].

Out of all, incomplete and ambiguous requirements are the most challenging defect to be detected. Unless reviewers are knowledgeable of the requirements, these requirements defects might go unnoticed or ignored. There are instances in which the requirements engineer has overlooked to explicitly specify the pre-condition and post-condition of a particular requirement[21].

Zowghi and Gervasi[26] have suggested that an internal contradiction in the requirements specification causes the occurrence of inconsistent requirements. Software developers will consequently make incorrect assumptions, based on the defective requirements6. Thus, the resulting error requires additional effort for its rectification, increasing the cost of the project and causing delay in project delivery16,27.

3.4 Validation Support Tool

Requirements review is normally performed by reviewing the requirements documentation manually[11]. According to Aceituna[21], the manual method is employed as IT personnel are reluctant to use a tool-based solution. They usually necessitate for investments in the context of time, effort and money. The employees, especially junior staffs who are unfamiliar with the tool, will require training. This creates the need and effort to get the training staffs. For organizations, the engagement of a consultant or training provider is compulsory if there is no internally qualified person to conduct the training.

Hence, lack of support tool for the review session may cause certain errors in the requirements specification to be missed out[16,25]. Additionally, budget constraints may also hinder IT practitioners from implementing researchers' recommendations for the improvement of the requirements review process[21].

3.5 Organizational Culture

Organizations normally conduct requirements validation activity on an ad-hoc basis as they are short of experienced and knowledgeable requirements engineers. Thus, lesser attention is directed to the requirements validation activity, as the focus is more on software development and testing stages[18].

Requirements review requires requirements engineers to be open-minded with their colleagues' feedbacks[4,6]. This does not sit well with some, as it can be difficult to acknowledge that requirements specification they have written contains errors, requiring further amendment. Wiegers and Beatty[6] have postulated that it will take time to instil the characteristics of a receptive requirements engineer within the organizational culture. Organizational promotion and awareness within the organization might be necessary to ensure that the requirements review activity can be implemented effectively.

3.6 Governance

Requirements review activity is typically deemed as insignificant and the cause for delays to the project schedule. Reviewing thick requirements documents is a tedious task, so software engineers are inclined to focus on programming tasks and software testing, rather than spending time on the requirements validation activity. Therefore, the project team might be tempted to either spend minimal time or skip the review activity entirely[6,12,13]. Thus, monitoring the organizational implementation of requirements review activity requires good governance. This will ensure that the requirements specification contains the right requirements for the software development project. Aceituna[21] suggests for revision of policies and procedures in order to promote effective requirements validation activity in the organization.

4. Conclusion and Future Work

The paper has highlighted the issues and challenges of requirements review in the industry, which affecting project resources, schedule and budget negatively. Some major obstacles faced during requirements review are due to the lack of regulatory guidelines for the review process, vague criteria for defect detection and lack of support tool in assisting the manual review process.

It is aspired for this paper to serve as a reference for researchers and industry key players possessing inter-

est in requirements validation research area. The subject matter can be extended by conducting a case study about the issues and challenges of requirements review in the industry for further insight. A future work of this research proposes for a framework that will provide comprehensive process guidelines, for an improved execution of requirements review activity. Further research study is also indispensable to address the challenges highlighted in this paper.

5. References

1. Bourque P, Fairley RE (Dick). eds. Guide to the Software engineering body of knowledge version 3.0. IEEE Computer Society. 2014.

2. Sommerville I. Boston: Addison-Wesley: Software engineering. 9th ed. 2011.

3. Glinz M. A glossary of requirements engineering terminology, version 1.6 May 2014. Zurich: International Requirements Engineering Board (IREB). 2014.

4. Pohl K, Rupp C. California: Rocky Nook Incorporation: Requirements Engineering fundamentals - a study guide for the certified professional for requirements engineering exam : foundation level – IREB compliant. 2nd ed. 2015.

5. Khan K, Kumar PVV, Ahmad A, Riaz T, Anwer W, Suleman M, Ajmal O, Ali T, Chaitanya AVK. Requirement development life cycle: The industry practices. Proceedings of 9th International Conference on Software Engineering Research, Management and Applications (SERA). 2011; 12-16.

6. Wiegers K, Beatty J. Washington: Microsoft Press: Software requirements. 3rd ed. 2013.

7. Pohl K. Requirements engineering: fundamentals, principles, and techniques. California: Berlin Heidelberg: Springer-Verlag. 2010.

8. Saqi SB, Ahmed S. Requirements validation techniques practiced in industry : Studies of six companies. Sweden: Blekinge Institute of Technology. 2008.

9. Stephens R. Indiana: John Wiley & Sons, Incorporation: Beginning software engineering. 2015.

10. Sulehri LH. Comparative selection of requirements validation techniques based on industrial survey. Sweden: Blekinge Institute of Technology. 2009.

11. Pandey D. The research roadmap of requirement validation. International Journal of Advanced Research in Computer Science. 2013; 4(4):52-4.

12. Wiegers K. Boston: Addison-Wesley: Peer reviews in software: A practical guide. 2002.

13. Wiegers K. The seven deadly sins of software reviews. Software Development. 1998, 6(3):44-7.

14. Ow S, Yaacob M. A study on the requirements review process in software development: Problems and solutions. Proceedings of 21st International Computer Software and Applications Conference. 1997; p. 542-47.

15. Swarnalatha KS, Srinivasan GN. A survey on emerging trends in requirement engineering for a software development life cycle. International Journal of Advanced Research in Computer and Communication Engineering. 2013; 2(1):950-57.

16. Aragon G, Escalona MJ, Hilera JR, Fernandez-Sanz L, Misra S. Applying model-driven paradigm for the improvement of web requirement validation. Acta Polytech Hungarica. 2012; 9(6):211-32.

17. Lee YK, In HP, Kazman R. Customer requirements validation method based on mental models. Proceedings of 21st Asia-Pacific Software Engineering Conference. 2014; 199-206.

18. Raja UA. Empirical studies of requirements validation techniques. Proceedings of 2nd International Conference on Computer, Control, Communication. IC42009. 2009.

19. Condori-Fernandez N, Espana S, Sikkel K, Daneva M, Gonzalez A. Analyzing the effect of the collaborative interactions on performance of requirements validation. Proceedings of 20th International Working Conference on Requirements Engineering: Foundation for Software Quality. 2014; 216-31.

20. Sabatucci L, Ceccato M, Marchetto A, Susi A. Ahab's legs in scenario-based requirements validation: An experiment to study communication mistakes. Journal of Systems and Software. 2015; 109:124-36.

21. Aceituna D. Addressing off-nominal behaviors in requirements for embedded systems. North Dakota: North Dakota State University. 2014.

22. Yousuf F, Zaman Z, Ikram N. Requirements validation techniques in GSD: A survey. Proceedings of IEEE International Multitopic Conference. 2008; 553-57.

23. Li W. Consistency checking of natural language temporal requirements using answer-set programming. Kentucky: University of Kentucky. 2015.

24. Satish CJ, Anand M. Software documentation management issues and practices: A survey. Indian Journal of Science and Technology. 2016; 9(20):1-7.

25. Kamalrudin M, Hosking J, Grundy J. MaramaAIC: Tool support for consistency management and validation of requirements. Automated Software Engineering. 2016; p. 1-45.

26. Zowghi D, Gervasi V. The three Cs of requirements: Consistency, completeness, and correctness. Proceedings of 8th International Workshop on Requirements Engineering: Foundation for Software Quality. 2002; p. 155-64.

27. Kamalrudin M, Sidek S. A review on software requirements validation and consistency management. International Journal of Software Engineering and its Applications. 2015; 9(10):39-58.