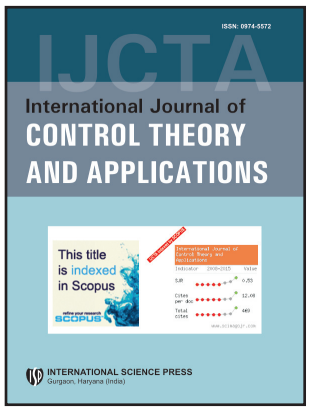

International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 9 • Number 43 • 2016

Software Requirements Review Framework: A Case Study of Resource Tracking Project

Siti Osnita Mokhtar[a], Rosmawati Nordin[b], Zalilah Abd Aziz[c] and Rashidah Md Rawi[d]

[a-d]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia. Email: [a]osnita@gmail.com, [b]roswati@tmsk.uitm.edu.my, [c]zalilah@tmsk.uitm.edu.my, [d]rrawi@tmsk.uitm.edu.my

Abstract: Software requirements specification is a crucial document which describes the stakeholders' requirements for the system to be developed. The requirements must be validated earlier to avoid the risk of system design and programming rework, which can lead to additional effort and higher cost to fix the defects at the later stage of the project. One of the most common means of requirements validation technique is requirements review, a widely used method in the industry. However, the lack of proper standard for review process has resulted in multiple obstacles to achieve requirements validation goal. This paper recommends a framework for software requirements review. A case study using a real-world software development project involving two IT organizations in Malaysia, was conducted to evaluate the proposed framework. Positive research findings from the case study provide useful insights on the practicality of the framework within a real-life software development project setting in the industry.

Keywords: Software requirements review, framework, requirements validation, requirements engineering, software development project, case study.

1. INTRODUCTION

According to the 2015 CHAOS report released by the Standish Group, a primary research advisory organization, 71% of the software projects end in partial or total failure. Only 29% of the projects fulfill the success factors of, on time and on budget with satisfactory results, in terms of the stakeholders' satisfaction and meet target requirements [1]. This accentuates the importance of requirements validation, one of the activities in requirements engineering phase, which aims to ensure the correctness and conformance of the proposed system against the stakeholders' requirements [2].

Previous researchers [3], [4] have proposed the requirements review process, one of the most common means of validation technique [5], [6], which is intended to identify the requirements defects at the early stage of the software development life cycle. Requirements defects need to be fixed earlier, in order to avoid the risk of rework at the later stage of the project, which can lead to the risk of additional effort and higher cost to the project [7].

However, this technique encounters multiple challenges which hinders it from achieving the requirements validation goal. Thus, there is a need to improve and develop a proper standard for requirements review process, which is the aim of this research study.

The purpose of this paper is to propose a framework for software requirements review and validate it against a real-world software development project. The following sections of this paper will highlight the related studies on requirements review, describe the proposed framework, present the research methodology, and discuss the findings before offering a conclusion.

2. REQUIREMENTS REVIEW

Kotanya and Sommerville [3] have suggested the earliest version of requirements review process, which comprises the following six activities: (1) Review planning and preparation, (2) Document distribution, (3) Pre-review preparation, (4) Conduct the review meeting, (5) Follow-up actions, and (6) Document revision. Subsequently, Sommerville [4] has proposed an improved version of the review process, which separates the activities into three phases, mainly, (1) pre-review activities, (2) review meeting, and (3) post-review activities.

Even though, requirements review is a long-established validation technique [8], practitioners are still encountering various obstacles, which have caused many pitfalls to the organizations such as, poorly defined software requirements specification [9], incorrect developers' assumptions based on the defective requirements [10], unprepared reviewers [5], less receptive document's author [2], and difficulty to detect incomplete and ambiguous requirements [11].

According to Swarnalatha and Srinivasan [12], these challenges occur due to the limited regulatory guidelines on how to conduct the requirements review process. Due to that, requirements review activity receives lesser attention, as the team's focus is more on software development and testing stages [13]. Therefore, the project team might be tempted to either spend minimal time or skip the review activity entirely [10].

The current review process needs to be improved further to cater for the organizations' practical needs. Thus, these impediments are the motivation for this research study, which is to develop a framework for software requirements review.

3. SOFTWARE REQUIREMENTS REVIEW (SORE) FRAMEWORK

In this paper, we have adapted the requirements review process based on the previous research works [3], [4]. The review process has been enhanced to include the regulatory guidelines and provide explicit representation of the review phase. As illustrated in Figure 1, the proposed Software Requirements Review (SORE) Framework consists of two main sections: guidelines and review phase.

The description of the framework is described as follows:

A. Guidelines

The first section of the SORE framework is the guidelines. The objective of the guidelines is to influence the execution success of the review process based on four critical aspects: entry criteria, review meeting, reviewer and exit criteria. Each aspect postulates the recommendations to address the review challenges highlighted earlier. These recommendations are derived based on various literature reviews [2], [4], [10], [14]–[17], as shown in Figure 2.

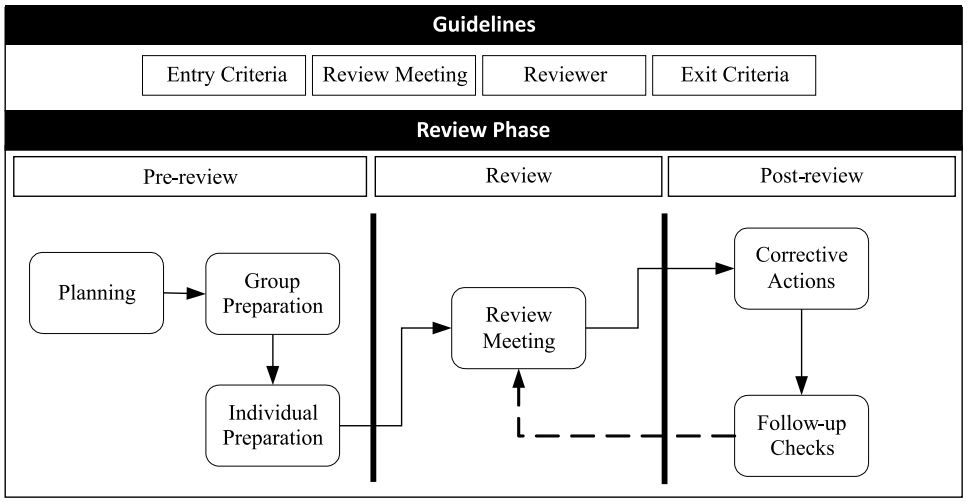

Figure 1: Software Requirements Review (SORE) Framework

Guidelines

Entry Criteria	Review Meeting	Reviewer	Exit Criteria
• Requirements specification must adhere to the acceptable format, contain no major spelling / grammatical mistakes and free from any formatting issues. • Set a clear review objective, which is to improve the software requirements quality, not to evaluate the performance of the project team members. • Provide the review scope about which requirements aspects to scrutinise and what issues to discover. • Reviewers must perform initial review of requirements specification and identify the list of defect on their own.	• Schedule for the date and time which all the reviewers can attend. • Separate the review meeting into multiple short sessions, which should not exceed from 2-hour time frame for each session. • Assign adequate time for the reviewers to perform the reviews. • Restrict from re-reviewing the same section of the document more than three times.	• Optimum number of reviewers for each review meeting should be fewer than 7 people. • Appoint the right project stakeholders for the review session, other than the document author, who have knowledge about the requirements.	• All issues highlighted in the review meeting have been addressed and resolved accordingly. • Corrective actions for the requirements specification have also been correctly done.

Figure 2: SORE Framework guidelines

B. Review Phase

In the second section, SORE framework emphasizes the three pillars of review phase: pre-review, review and post-review, which refer to the key stages involved during a software requirements review. Each pillar under the review phase is described as follows:

1. *Pre–review:* The objective of this initial review stage is to do the planning and preparation to ensure effective review activity. The planning activity consists of proposing the meeting schedule, setting

up the review team and distributing software requirements specification to the reviewers. Pre-review preparation will be done in a group and individual basis. Group preparation refers to the group activity which provides the initial understanding and overview of the requirements specification. Subsequently, individual reviewer will read and identify the defects in the specification prior to the review meeting.

2 *Review:* All the reviewers will meet based on the agreed time and place for the review meeting. The document's author will 'walk-through' the specification together with the reviewers. The aim of this stage is to consolidate the identified defects and recommend possible actions to address the issues.

3 *Post-review:* After the review stage completes, the document's author will revise and perform the correction to the specification. Once the corrective action is done, the specification will be redistributed to the members of the review team for their verification. If required, subsequent review meeting may be conducted to address any unresolved issues. This stage is concerned with the finalization of the requirements specification for the stakeholders' acceptance.

4. METHODOLOGY

To evaluate whether our proposed framework is viable for the organizations' needs, we conducted a case study, a dominant research method for software engineering research studies [18]. Case study is an observational research method to investigate how a certain task is conducted within its real-life context [19]. For this paper, the objective of the case study was to evaluate the practicality of the SORE framework for the industry practitioners using a real-world software development project.

The case study involved two organizations, Company C, one of the leading IT organizations in Malaysia and Company S, a business partner of Company C. Company C would like to have a centralized company-wide resource tracking system to monitor its engineers' time spent for the respective IT projects. It has appointed Company S, a software engineering company located in Selangor, Malaysia, to develop the resource tracking system.

The project was carried out in a period of five months, from July 2016 until November 2016 and the requirements review sessions were specifically conducted from 25th July 2016 until 9th August 2016. There were three face-to-face review meetings and three offline review activities via email communications, which involved six reviewers as presented in Table 1.

Table 1
Review Team Members for Resource Tracking Project

Team Member	*Company*	*Designation/Department*	*Role in the Project*
Person A	Company C	General Manager/Project Management	Project Sponsor/Business User
Person B	Company C	Manager/Project Management	Project Manager/Business User
Person C	Company C	Senior BA Advisor/Commercial Management	Business Analytics Advisor
Person D	Company C	Team Leader/Process and Information Office	Systems Integration Advisor
Person E	Company S	Senior Systems Analyst/Software Engineering	Systems Analyst
Person F	Company S	Senior Systems Analyst/Software Engineering	Systems Analyst

5. RESULT AND DISCUSSION

There were 53 requirements specified in the software requirements specification, out of which, 24 contained defects. These defects were categorized into several types, such as incomplete, ambiguous and inconsistent

requirements. The case study was conducted in accordance to the SORE framework guidelines as outlined in Table 2.

Table 2
Compliance of the SORE Framework Guidelines

Guidelines	*Case Study Execution*
Entry Criteria	• Requirements document was prepared according to the organization's format. • Person E conducted an initial review meeting to explain about the review scope and content of the document. • All the reviewers attended the initial review meeting, except for Person C, who had other project commitment during that particular session. • Reviewers had performed their own preparations prior to the review meeting.
Review Meeting	• Review phase was conducted within 12 working days. • Meetings were short and within a two-hour period for each session.
Reviewer	• Appointed reviewers were the right people selected for the review team. • They have extensive experience in various software development projects.
Exit Criteria	• All the requirements defects had been rectified prior to the sign-off of the review phase.

Based on the case study activities, it is observed that the proposed SORE framework guidelines have positively influenced the outcome of the review phase. The following observations were recorded according to the three pillars of review phase:

A. **Pre-review:** The planning and preparation were performed according to the proposed framework. The right project stakeholders were selected as the reviewers. They had the common understanding regarding the objective and scope of the review process. Due to their extensive experience in the software industry, they were familiar with the terminologies and technical areas described in the requirements specification. Thus, they managed to conduct their own assessment of the requirements specification independently.

B. **Review:** Review meetings duration was appropriate since the reviewers had performed their individual preparation prior to the review meeting. This had encouraged the reviewers' commitments, which contributed to the fruitful review sessions. Additionally, Person A, as the project sponsor, had played a significant role in managing the conflicts among the reviewers, whenever there was disagreement about certain defects.

C. **Post-review:** Reviewers were clear about the objective of the review process and recognized its importance to influence the project success. They refused to sign off the requirements specification until all the defects were fixed. Consequently, the specification was finalized once all the defects had been rectified correctly. As a result, the review phase had ended successfully within the target project timeline.

D. **Issues Discovered:** Despite the positive findings of the case study, there were few issues recorded. The issues are described as follows:

1 There was no validation checklist provided to the reviewers. Reviewers performed the requirements review based on their knowledge and experience. Thus, there might be a possibility of certain defects were overlooked.

2. Person C did not join the first review meeting. Due to his absence, some of the defects were re-examined in the subsequent review meetings. Unnecessary time was spent reviewing the same defects which had been discussed and agreed earlier.

3. There was a change of systems analyst during the second review meeting. Certain defects were repeatedly highlighted and discussed throughout the subsequent review sessions, due to insufficient knowledge transfer from Person E to Person F, who had a weak understanding of the overall requirements and inadequate knowledge about the previous highlighted defects.

There was no validation checklist provided to the reviewers. Reviewers performed the requirements review based on their knowledge and experience. Thus, there might be a possibility of certain defects were overlooked.

Person C did not join the first review meeting. Due to his absence, some of the defects were re-examined in the subsequent review meetings. Unnecessary time was spent reviewing the same defects which had been discussed and agreed earlier.

There was a change of systems analyst during the second review meeting. Certain defects were repeatedly highlighted and discussed throughout the subsequent review sessions, due to insufficient knowledge transfer from Person E to Person F, who had a weak understanding of the overall requirements and inadequate knowledge about the previous highlighted defects.

6. CONCLUSION

This paper presents a research result of the SORE framework implementation based on a real-world project in its real-life setting involving two organizations. The case study findings confirm that the SORE framework guidelines have positive impact on the requirements review phase. The review sessions have been effectively conducted and completed within the scheduled project timeline. The final software requirements specification has been signed off by the reviewers, which becomes the basis for the next stage of the software development project.

Additionally, this study provides useful insights on the practicality of the SORE framework within a specific environment, which can be a reference for other practitioners to make an informed decision for their own particular environment.

The framework will be enhanced further based on the findings of this paper. Further research study will be conducted to acquire the feedback for SORE framework from the industry experts. A tool may be developed to facilitate the framework execution.

REFERENCES

[1] S. Hastie and S. Wojewoda (2015) InfoQ webpage on Standish Group 2015 Chaos Report. [Online]. Available: http://www.infoq.com/articles/standish-chaos-2015.

[2] K. Pohl and C. Rupp, *Requirements Engineering Fundamentals - A Study Guide for the Certified Professional for Requirements Engineering Exam : Foundation Level – IREB compliant*, 2nd ed. California: Rocky Nook Inc., 2015.

[3] G. Kotonya and I. Sommerville, *Requirements Engineering: Processes and Techniques*. Chichester: John Wiley & Sons, Inc., 1998.

[4] I. Sommerville, *Software Engineering*, 9th ed. Boston: Addison-Wesley, 2011.

[5] S. B. Saqi and S. Ahmed, "Requirements Validation Techniques Practiced in Industry : Studies of Six Companies," M. Software Engineering thesis, Blekinge Institute of Technology, Sweden, Oct 2008.

[6] L. H. Sulehri, "Comparative Selection of Requirements Validation Techniques Based on Industrial Survey," M. Computer Science thesis, Blekinge Institute of Technology, Sweden, Dec 2009.

[7] K. Pohl, *Requirements Engineering: Fundamentals, Principles, and Techniques*. Springer-Verlag Berlin Heidelberg, 2010.

[8] K. Khan, P. V. V. Kumar, A. Ahmad, T. Riaz, W. Anwer, M. Suleman, O. Ajmal, T. Ali, and A. V. K. Chaitanya, "Requirement Development Life Cycle: The Industry Practices," in *9th International Conference on Software Engineering Research, Management and Applications (SERA)*, 2011, p. 12–16.

[9] C. J. Satish and M. Anand, "Software Documentation Management Issues and Practices: A Survey," *Indian J. Sci. Technol.*, Vol. 9, No. 20, pp. 1–7, May 2016.

[10] K. Wiegers and J. Beatty, *Software Requirements*, 3rd ed. Washington: Microsoft Press, 2013.

[11] D. Aceituna, "Addressing Off-nominal Behaviors in Requirements for Embedded Systems," Ph.D. dissertation Software Engineering, North Dakota State University, Dec 2014.

[12] K. S. Swarnalatha and G. N. Srinivasan, "A Survey on Emerging Trends in Requirement Engineering for a Software Development Life Cycle," *Int. J. Adv. Res. Comput. Commun. Eng.*, Vol. 2, No. 1, pp. 950–957, Jan 2013.

[13] U. A. Raja, "Empirical Studies of Requirements Validation Techniques," in *Proceedings of 2nd International Conference on Computer, Control and Communication, IC42009*, 2009, Vol. 1, No. 19.

[14] K. Wiegers, *More About Software Requirements: Thorny Issues and Practical Advice*. Redmond, Washington, 2006.

[15] K. Wiegers, *Peer Reviews in Software: A Practical Guide*. Boston: Addison-Wesley, 2002.

[16] K. Wiegers, "The Seven Deadly Sins of Software Reviews," *Softw. Dev.*, Vol. 6, No. 3, pp. 44–47, 1998.

[17] S. Ow and M. Yaacob, "A Study on the Requirements Review Process in Software Development: Problems and Solutions," in *Proceedings of 21st International Computer Software and Applications Conference*, 1997, pp. 542–547.

[18] P. Runeson, M. Host, A. Rainer, and B. Regnell, *Case Study Research in Software Engineering: Guidelines and Examples*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2012.

[19] R. K. Yin, *Case Study Research: Design and Methods*, 4th ed. Thousand Oaks, California: Sage Publications Inc, 2009.