Identifying Requirements Risks: The Approaches

Ow Li Lee[1], Massila Kamalrudin[2] and Safiah Sidek[3]
[1]Faculty of Information and Communication Technology,UniversitiTeknikal Melaka Malaysia,
Jalan Hang Tuah Jaya 76100 Durian Tunggal, Melaka Malaysia
lilyow@hotmail.com

[2, 3, 4] Institute of Technology Management and Entrepreneurship, Center of Advance Computing & Technology, University Teknikal Malaysia Melaka,

Article Info
Volume 83
Page Number: 938 - 943
Publication Issue:
March - April 2020

Article History
Article Received*: 24 July 2019*
Revised*: 12 September 2019*
Accepted*: 15 February 2020*
Publication*: 13 March 2020*

Abstract

Requirements risks are inherent in the requirements elicited from the stakeholders. It is important to access and mitigate the requirement risk as early as possible to avoid a poor quality of requirements engineering. The poor quality of requirements has a direct effect on the quality of software development. Hence, our paper aims to investigate on the approaches to identify requirement risks.There are a few available papers found on the topic of requirements risk in our literature review. We discussed on the results in the discussion section. In conclusion, we found that most of the researches prefer interview session, questionnaire and survey in identifying requirement risk.

Keyword: Requirement Risks, Requirement Engineering

Introduction

We are always been told or reported that requirements engineering is the main issue contributed to the failure of system projects. To be exact, the poorly conducted requirements capturing had comprised 90% amongst other issues [1] and resulting in requirements defects. Requirements engineering is one of the issue that hinders the process of a quality software development life cycle, it is important to focus on the requirements phase to identify and propose countermeasure to improve the quality of requirements engineering. In defining the project scope or understanding the requirements, there are a number of problems addressed [2] including requirements engineering is not properly implemented, underestimated for effort on coding phase, underestimated for testing effort, underestimated or overestimated for the skill of the team and lastly project delivery date delayed due to modification from customer side. The root cause of the requirement issues mentioned above most probably from requirement risk inherent in requirement. Requirement risk always results in requirements defects and hence affects the quality of requirements. In this paper, we investigate the methods or techniques used by the researcher or industry in identifying requirement risk. Requirements risk identification is vital to determine the number of requirements defects. The number of requirements defects are avoidable and its number can be greatly reduced if the requirements risks are identified in prior during requirements engineering.

This paper is organized into four sections. The first section elaborates on introduction on requirement risk and the second section describes the previous works in literature review. We discussed on the techniques found to identify requirement risk in the third section of our paper. Lastly, we conclude our study on the techniques to identify requirement risks in the last section.

Literature Review

The most conventional and well recognized method to identify requirements risk is requirement inspection[3]. Requirement inspection is commonly used by the requirement engineers to check on the requirement defects in manual way. Sadia H., Rizwan Beg Md. and Faisal Md. [4] advocate the practice of inspection in identifying requirement risks. There are five different roles involved in their inspection technique including moderator, author, reader, inspector and recorder. The roles mentioned are required to follow five steps of requirements inspection process to identify requirement risks. The identified requirement risk will serve as an input for their requirement risk log.

Appukkutty K. et al. [5] realized there are numerous number of software risk model proposed but there is scarce study on requirements risk assessment.Hence, they presented a methodology to assess software risk at the requirement engineering level adopting Unified Modelling language (UML). They agreed that it is more efficient in terms of cost and time to detect risk at the early stage of software development life cycle instead of later stage. In their study, the risks are identified by using failure mode in the classes of UML diagram.

UML is popular to be adopted in assessing the risk, Naeem M. R. et al. [6] identified and monitored risk at all of the phases in software development using V-model methodology. In their work, they assumed that risks are being highlighted in use case diagram and evaluated. Besides, they also assumed that requirement engineers are trained to visualize requirements risks based on the use case diagram provided. The assumptions made by them in their proposed approach might be invalidity for the result of their experiment. Use case diagrams are used to illustrate functions of the systems in an easier way for understanding and they didn’t portray exactly which one is considered as requirement risk. The second assumption regarding requirement engineers might be false as requirement engineers are not trained to spot requirements risk as the purpose.

There are a series of problems identified by study conducted from author Amber S., Shawoo N. and Begum S. [2] in determining risk during requirements engineering process. Their study proposed a UML oriented approachto model, reason and manage risk in requirements engineering instead of the design phase. Despite of their detail explanation on the proposed approach, we focused on the way they identify requirements risk. Generally, their risks are identified through a series of activity of, questionnaire and brainstorming session with the stakeholders to identify risks. Their approach proposal to determine risk during requirements engineering seemed to be promising yet it is not proven with any kinds of experiment.

A study conducted by Li X. and Liu Q. [7]to propose a requirement risk assessment focused on stakeholder risk analysis. The stakeholders are divided into requirement provider and project team member with their own risk factors. The degrees of the risk factors included are defined by the stakeholders. They claimed that risk analysis done by the stakeholders is easier in identifying risk cause.

Likewise, the study conducted is merely a approach proposal to identify requirement risk based on stakeholders' preference. There is no evidence provided from their study to confirm their proposed approach.

Chadani P. and Gupta C. identified and classified a set of risky requirements in their study [8] employing AHP method. AHP is a pair wise comparison method to decide on requirement implementation between two requirements. Apart from the solution proposed to deal with risky requirements, they assessed the risk of requirements by questionnaire. The finding obtained from the questionnaire is used to compute the level of risk.Likewise, Saini C. and Jyoti A. [9] introduced a new approach for short term projects risk analysis using AHP method. The project risks are identified using pair wise comparison matrices tailored by project developer or experts.

Despite of risk identification in requirements engineering, Roy B. and Dasgupta R. [10]attempted to identify risk types and risk factors in each of the phase in software development life cycle. They identify requirement risk based on generic sense. There is no detail explanation included in their study on how to identify risk based on common sense. We have only found a small part regarding requirements risk from their study, which is related to our topic of research.

Shrivastava S. V. and Rathod U.[11] conducted a study to create a set of risk factors affecting the performance of distributed agile development projects. In addition, they had mentioned on the risk management approaches that are frequently practiced in the context of distributed agile development. They conducted an in depth interview followed by questionnaire with practitioners to answer their research questions including identifying risk factors.

Author Islam S. and Stoica A. J. [12] also adopts the same techniques, which refer to interview or questionnaire, to identify software risk. They proposed a goal driven approach to be integrated in requirements engineering activity with aim to manage software risk in the early stage.Further, the researcher Islam S. also collaborated with other researchers to propose a off shore model to track and control software risk [13]. The model proposed shows in details how they track and control the software risk in requirements engineering. They adopted the similar approach to identify software risk which is questionnaire and brainstorming.

We found another work from author Wallace L. and Keil M. [14] also identify the potential software risk from interview and survey session with stakeholders. The identified risks from the interview are mapped into a four quadrants categorization framework based on the perceived importance from the manager's opinion.There is another work found from the researcher Lin A. Y. and Parinyavuttichai N. [15] to identify project risk by semi structured interview session and project documentation.

Besides, Gupta D and Sadiq M. [16] proposed a software risk assessment and estimation model (SRAEM) to predict possible software risks. Their proposed model of SRAEM considers nine critical risk elements including software complexity, staff involved, targeted reliability, requirement for product, estimation method, monitoring method, adopted development process, software usability and lastly tools. SRAEM considered function point concept in identifying the software risk. Their work focused on the software risk assessment only without including any software risk mitigation activity.

Another similar work focusing on software risk assessment by author Foo S. W. and Muruganantham A. named Software Risk Assessment Model (SRAM) [17]. Their proposed work is very similar with SRAEM since they are also focusing on the nine critical risk elements. However, the software risks are identified by questionnaire for their proposed SRAM model.

Software Fault Tree Analysis is well known for identifying potential causes resulting in defect. A study from Sadiq M. et al. [16] adopted software fault tree to identify and analyze risk into two different fields in requirements phase and design phase. Their work is similar with SRAEM and SRAM mentioned above. They discussed the weaknesses in SRAEM and others approaches in prior to proposing a new model named Software Risk Evaluation (SRE). They claimed that their software risk assessment model is better since they consider requirements risk in their model while others don't. In fact, we have mentioned above that a few models proposed to tackle the risk in requirements.

Likewise, Hoodat H. and Rashidi H. adopting software fault tree in identifying, classifying and analyzing software risk [18]. They classified the software risk based on risk attributes and then analyzed them by using risk tree. However, their work focused on risk assessment and risk analysis only.

Researcher Patil S. and Ade R. [19] predicted software risk including requirements risks by proposing a model. They used three correlation matrix including development matrix, client requirements and project schedule parameters as the input for risk prediction. In our opinion, the requirements are not well focused to tackle the inherent risk since this model is proposed to handle software risk in global setting.

In another situation, Uzzafer M. has his own scheme called Uzzafer's scheme to identify software risk [20]. In Uzzafer's scheme for identifying software risk, it is actually relies on the questionnaire from Software Engineering institutes.

Discussion

There are many available methods in capturing requirements or analyzing requirementsand yet we can hardly find study focused on identifying requirements risk.We summarized the techniques used to identify requirements risk based on the study found in our literature.The Table 1 below shows the summarized approaches to capture requirements risks.

Requirements Risk Identification	
Interview	[11][2][14][15]
Questionnaire / Survey	[11][17][12][4][20][14][13]
Function Point	[21]
Measured by Stakeholders	[7]
Identified on Generic Sense / Common Sense	[10]
Software Fault Tree	[16][18]
Uzzafer's Scheme	[20]
UML Specifications	[6][5]
Brainstorming	[12][4][13]
Matrix	[19][8][9]
Inspection Technique	[4]

Table 1: Approaches on How to Capture Requirements Risks

Based on Table 1 above, risks in requirements are mostly identified by questionnaire or survey. Interview sessions are as common as well in capturing the requirements risks.Most importantly, interview session is always more reliable and efficient since communication between face to face always produces better interpretations especially on the results and facial expression. The input and comment provided by that stakeholder is very important as it expresses their thoughts on the specific requirements risks.However, the key focus for a reliable and efficient interview is to find the most suitable candidate in related field. Irrelevant candidate for the interview may results in false interpretations.

The others techniques to identify requirements risk including function point, defined by stakeholders, based on common sense, software fault tree, based on UML specifications, matrix and lastly inspection technique. These methods are considered uncommon in identifying requirement risks due to their reliability of the method since it does not involve the input from the stakeholders.

Nevertheless, existing risk managements are typically implemented during the later stage of software development life cycle including design phase or coding phase [2]. Requirement risk management is encouraged to be implemented since the early stage of requirements to ensure a better quality of requirements and software development.

Conclusion

Requirements engineering is regarded as one of the main reason for system development failure. It is important to focus on the activity of requirement engineering in order to ensure the quality of requirements before development phase. In this case, we believe that requirement risk is the root cause of the poor quality of requirements since requirement risks always lead to requirement defects.This paper investigates the techniques used to identify requirement risks. Based on our conducted literature review, we found that most of researches adopt interview session, survey and questionnaire to identify requirement risks. The face to face communication to identify requirement risks is more reliable and efficient since interpretations can also be done on the facial expression. In our future research, we would like to adopt the reliable methods to identify requirement risk based on our review.

Acknowledgement

We would like to thank UTeMZamalah and UniversitiTeknikal Malaysia Melaka and grant FRGS No: FRGS/1/2016/ICT 01/FTMK-CACT/F00325 for funding this research.

References:

[1] B. F. Davey and C. Cope, “Requirements Elicitation – What ’ s Missing ?,” *Informing Sci. Inf. Technol.*, vol. 5, no. September, 2008.

[2] S. Amber, N. Shawoo, and S. Begum, “Determination of risk during requirement engineering process,” *J. Emerg. Trends Comput. Inf. Sci.*, vol. 3, no. 3, pp. 358–364, 2012.

[3] S. K. Nayak, R. A. Khan, and M. R. Beg, “Evaluation of Requirement Defects: An Implementation of Identification Technique,” *Int. J. Inf. Educ. Technol.*, vol. 2, no. 5, pp. 448–450, 2012.

[4] H. Sadia, M. R. Beg, and M. Faisal, “Requirement Risk Identification : A Practitioner ’ s Approach,” *Int. J. Comput. Appl.*, vol. 102, no. 15, pp. 13–15, 2014.

[5] K. Appukkutty, H. H. Ammar, and K. G. Popstajanova, “Software Requirement Risk Assessment Using UML.” The 3rd ACS/IEEE International Conference onComputer Systems and Applications, 2005., 2005.

[6] M. R. Naeem, W. Zhu, A. A. Memon, and A. Khalid, “Using V-Model methodology, UML process-based risk assessment of software and visualization,” in *Proceedings of 2014 International Conference on Cloud Computing and Internet of Things, CCIOT 2014*, 2014, no. Cctot, pp. 197–202.

[7] X. Li and Q. Liu, “Requirement Risk Assessment Focused-On Stakeholder Risk Analysis,” in *Proceedings - International Computer Software and Applications Conference*, 2009, vol. 1, pp. 640–641.

[8] P. Chandani and C. Gupta, “Requirement Risk Prioritization Using Analytic Hierarchy Process: A

Gateway to Identify Risky Requirements," *2018 11th Int. Conf. Contemp. Comput. IC3 2018*, pp. 1–6, 2018.

[9] C. Saini and A. Jyoti, "Analysis for Short Term Projects (STP ' s) using Fuzzy AHP Method," *Int. J. Adv. Reseach Comput. Sci. Softw. Eng.*, vol. 3, no. 11, pp. 392–396, 2013.

[10] B. Roy and R. Dasgupta, "A study on software risk management strategies and mapping with SDLC," *Advances in Intelligent Systems and Computing*, vol. 396. pp. 121–138, 2016.

[11] S. V. Shrivastava and U. Rathod, "Categorization of risk factors for distributed agile projects," *Inf. Softw. Technol.*, vol. 58, pp. 373–387, 2015.

[12] S. Islam and A.-J. Stoica, "Software Risk Management Modeling Using Goal-Driven Approach from Early Requirements Engineering," *Risk Assess. Manag. Acad. Publ. 2012.*, 2012.

[13] S. Islam, S. H. Houmb, D. Mendez-Fernandez, and M. M. A. Joarder, "Offshore-outsourced software development risk management model," *ICCIT 2009 - Proc. 2009 12th Int. Conf. Comput. Inf. Technol.*, pp. 514–519, 2009.

[14] L. Wallace and M. Keil, "Software Project Risks and Their Effect on Outcomes," *Commun. ACM*, vol. 47, no. 4, pp. 68–73, 2004.

[15] A. Y. Lin and N. Parinyavuttichai, "Is project risks as emergent phenomena: Towards a model of risk escalation and its management," *Australas. J. Inf. Syst.*, vol. 19, pp. 1–22, 2015.

[16] M. Sadiq, M. W. Ahmad, M. K. I. Rahmani, and S. Jung, "Software Risk Assessment And Evaluation Process (SRAEP) using model based approach," *ICNIT 2010 - 2010 Int. Conf. Netw. Inf. Technol.*, no. May 2015, pp. 171–177, 2010.

[17] S. W. Foo and A. Muruganantham, "Software risk assessment model," *Proc. 2000 IEEE Int. Conf. Manag. Innov. Technol.* , vol. 2, pp. 536–544, 2000.

[18] H. Hoodat and H. Rashidi, "Classification and analysis of risks in software engineering," *World Acad. Sci. Eng. Technol.*, vol. 56, no. 8, pp. 446–452, 2009.

[19] S. Patil and R. Ade, "Software Requirement Engineering Risk Prediction Model," *Int. J. Comput. Appl.*, vol. 102, no. 2, pp. 1–6, 2014.

[20] M. Uzzafer, "A novel risk assessment model for software projects," *2011 Int. Conf. Comput. Manag. CAMAN 2011*, no. December, 2011.

[21] D. Gupta and M. Sadiq, "Software risk assessment and estimation model," *Proc. Int. Conf. Comput. Sci. Inf. Technol. ICCSIT 2008*, pp. 963–967, 2008.