Visualizing Software Risks in Software Engineering Projects using Risk Sensitivity Analysis Approach

1N.A.H. Ghazali, 2S.S.M.Fauzi*, 3R.A.JM.Gining, 4W.A.W.M.Sobri, 5A.J. Suali
[1-5]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Malaysia

Abstract. Project risk management involves a systematic process of identifying, analyzing and responding to software risk in the project. Unfortunately, there were too many software risks involved in software engineering projects. Thus, it took a lot of time to investigate all the software risks that potentially occur in the project. Most of the project managers were only aware about the software risks when the software risk has occurred during the project. Therefore, an overview to visualize and prioritize software risks using sensitivity analysis need to be developed to help project managers determine software risks that has occurred in software engineering projects. This study adopts Systematic Literature Review (SLR) as the methodology to identify the software risks that have occurred in the project and a web-based integrated with risk sensitivity analysis approach to visualize and prioritize the software risks. The study suggests that a web-based application integrated with risk sensitivity analysis can be an alternative technique to assist project managers to identify high potential software risks that might occur in software engineering projects.

Key words: Risk management, Software risks, Sensitivity Analysis, Visualisation, Risk Sensitivity Analysis

1. Introduction
Project risk management plays a critical role in a managed software projects. It can determined the direction of the project either successful or fail (Fauzi & Sanim, 2014). The management of project risks involves a systematic process which is to identify, analyse and respond to the project risks (Marchewka, 2003). The software risks that were faced in the past are analysed and specific actions are implemented to prevent the occurrences in the future (Fauzi, Ramli, & Nasir, 2008). Software risks are uncertainty event that gives negative effects on the project objectives if it occurs (Marchewka, 2003). Unfortunately, there are several common mistakes that are always done by the project manager in managing software risk in projects. As a result, the project managers are late to identify the software risks that have high priority to occur. These mistakes can be reduced by visualizing the software risks by using the risk sensitivity analysis approach. Risk sensitivity analysis is a technique used to identify the important uncertainties in the purpose of visualizing data collection (Cullen & Frey, 1999). Risk sensitivity analysis is not a technique to avoid risks at all cost. It is a technique to produce a theoretical foundation and provide well-informed visualization of project risks to the project managers.

This study proposed a web-based application which integrate with a risk sensitivity analysis approach. The application of web-based application would assist the project manager to visualize and prioritize software risks that potentially occurs in software engineering projects. The first section of this paper will introduce about the software risks in software engineering projects. Next, the methodology used is discussed in the study. The result of this study is presented in the Result Section. Then, the result is discussed. The last Section concludes the study, which is the Conclusion Section.

2. Research Background
Software engineering project involves many software risks, and it took a lot of time to investigate all the risks that can potentially occurred during the project (Kremljak & Kafol, 2014; Makhani, Khan, & Soomro,
2010). Generally, there are two assessments on software risks which are internal risks and external risks. Internal risks are software risks that come from the inside of the organization and can be a threat to the project. On the other hand, external risks are software risks that come from the outside of the organization and it is difficult to be controlled (Asif, Ahmed, & Hannan, 2014). Management risks, people risks, and financial risks are types of internal risks. While, technology risks, requirement risks and estimation risks are type of external risks.

Management risk is a risk which is crucial to the project manager. This is because management risks tests the leadership skill of the project manager. In fact, a good project manager actually is also a good risk manager (Makhani et al., 2010). If the project manager can manage software risks properly and formulate effective risk management strategies, the project manager can prevent software risk from affecting the project progress and project success (Bannerman, 2008; Wallace, Keil, & Rai, 2004). The examples of management risks are poor internal communication, ill-defined project scope, and inexperienced project manager (Kremljak & Kafol, 2014).

People risk is an essential matter in project development and decision making. People that involve in project development are expected to be economical, and humane, but their nature also allow them to fulfill different drives in business decisions (Kremljak & Kafol, 2014). The examples of people risks are team member who are lack of necessary skill-set, losing critical staff at crucial point of the project and unmotivated team members.

Financial risks are related to the possibility of events in terms of its general financial viability that affect the project negatively (Kremljak & Kafol, 2014). The examples of financial risks are cost of resources which are used to develop the increasing project, budget overruns and increase in payment of the staff. Technology risks are the exposure to loss because of technology activities. This means that technology risks is any risks that caused by software and hardware failure, software security issues, and the improper use of the software and hardware technologies during the system development (Youin, Malaiya & Ray, 2016). The examples of technology risks are inappropriateness of technology and tools, software error loses key data and processor memory insufficient.

Requirement risks are negative events that are related to ambiguous requirements and the change in the user requirements during the project development process. Requirement risks usually involve directly with specification of the project. Besides that, requirement risks can affect the financial, the completion of time and the quality of the project if it occurred. The examples of requirement risks are stakeholder who does not engage properly, new requirement from the customer and change in the project requirement (Arnaut, Ferrari, & Souza, 2016).

Estimation risks can be defined as expressing the duration, intensity and magnitude and the reaching of the potential consequences of a software risk in quantitative (Lewis, 2001). The examples of estimation risks are unrealistic deadlines, improper budget and improper use of resources.

Therefore, the project manager needs to be prepared with the most potential software risks that can occur in the project. This is to ensure that the potential software risks can be manage properly and does not impact the project. There was evidence that showed the project manager did not manage software risk in their project systematically even though they claim that they did manage the software risks.

A literature mentioned that most project managers are only aware about the software risks when the software risks have occurred during the projects (Bannerman, 2008). These problems have arisen because there are only a few method that can visually identified and analysed the software risks at the initial stage of the project (Sharif, Basri, & Ali, 2014).

Risk sensitivity analysis is an approach that helps the project manager identifies which software risks are important and which software risks are not (Smith et al., 2008). Therefore, by performing risk sensitivity analysis, the project manager can get earlier prediction for susceptibility of the project to the software risks (Makhani et al., 2010). Despite that, little study applies risk sensitivity analysis approach in managing software risks in software engineering projects.

This study proposed a web based application to provide an overview to the project managers which software risks that might potentially affect software engineering project progress using risk sensitivity analysis approach.
3. Methodology

The proposed study is alienated into two phases which includes the Systematic Literature Review (SLR) and the development of the application.

| Phase 1 (Software risks identification) | Method | Activities |
|----------------------------------------|--------|------------|
| Systematic Literature Review (SLR)    |        | - Identification of research |
|                                        |        | - Selection of primary studies |
|                                        |        | - Study quality assessment |
|                                        |        | - Data extraction and monitoring |
|                                        |        | - Data synthesis |

| Phase 2 (Application design and implementation) | Method | Activities |
|------------------------------------------------|--------|------------|
| Construct interface and database development |        | - Analysing the requirement |
|                                                 |        | - Design interface |
|                                                 |        | - Create database |
|                                                 |        | - Contract coding |
|                                                 |        | - Test application |

SLR was used in the first phase of the study to synthesize and determine software risks that occurred in software engineering projects. In second phase, web-based application integrate with a risk sensitivity analysis approach is developed. Following is one of the equation used to calculate and prioritize software risks in software engineering projects:

\[
RE = \text{Probability of occurrences (UE)} \times \text{Probability of consequences (UE)}
\]

Where, UE = Uncertainty event

4. Results

Software Risks in Software Engineering Projects: Table 2 illustrates software risks that occurred in software engineering projects.

| Types of Software Risks | Software Risks                          | Percentage (%) | Frequency |
|-------------------------|-----------------------------------------|----------------|-----------|
| Management              | Ill-defined project scope               | 8              | 6         |
|                         | Project milestone cannot be defined      | 3              | 2         |
|                         | Ill-defined project requirements         | 8              | 6         |
|                         | Requirements change                      | 5              | 4         |
|                         | Incomplete requirements                  | 1              | 1         |
|                         | Ill-defined project planning             | 8              | 6         |
|                         | Ill-defined project completion time      | 8              | 6         |
|                         | Lack of agile progress tracking mechanism| 5              | 4         |
|                         | Too many projects                        | 1              | 1         |
|                         | Lack of customer presence                | 5              | 4         |
|                         | Ill-defined customer role                | 3              | 2         |
|                         | Failure to manage end-user expectation    | 1              | 1         |
|                         | Lack of quality focus                    | 5              | 4         |
|                         | Lack of resources                        | 7              | 5         |
|                         | Inexperience project manager             | 4              | 3         |
|                         | Lack of management commitment             | 1              | 1         |
|                         | Lack of project management competence    | 5              | 4         |
|                         | Top manager make decisions without consulting the others | 4 | 3 |
|                         | No control over staff priorities         | 1              | 1         |
Improper definition of roles and responsibilities
Poor internal communication

| People                          |       |       |
|--------------------------------|-------|-------|
| Lack of necessary skill-set    | 22    | 6     |
| Experience and training problem| 7     | 2     |
| Lack of team work              | 7     | 2     |
| Unmotivated team member        | 19    | 5     |
| Pressure to deliver project on an accelerated schedule | 7 | 2 |
| Resistance from groups or individuals | 11 | 3 |
| Conflicts between team member  | 11    | 3     |
| Losing critical staff at crucial point of the project | 7 | 2 |
| Bad customer relationship      | 7     | 2     |

| Technology                     |       |       |
|--------------------------------|-------|-------|
| Team member unfamiliar with the technology | 16 | 2 |
| Users unfamiliar with technology and require additional training | 8 | 1 |
| Lack of key technology         | 8     | 1     |
| Inappropriateness of technology and tools | 31 | 4 |
| Staining computer capabilities | 8     | 1     |
| Software error kills experiment | 8     | 1     |
| Software error loses key data  | 8     | 1     |
| Processor memory insufficient  | 8     | 1     |
| Database management software uses derived data | 8 | 1 |

Visualisation of software risks based on sensitivity analysis: Figure 1 and Figure 2 illustrates the web-based application used to visualize and prioritize software risks that occurred in software engineering projects.

Figure 1. Visualise software risks in software engineering projects.
5. Discussion
Results from Table 1 suggested that the identified software risks occur in software engineering projects. The results suggest that management risks is the highest to occur compared to people risks and technology risks. The percentage of occurrence of management risks is 66% while people risks are 25% and technology risks with only 12%. This is because management risks are very crucial in software engineering projects. Any mistakes in managing software engineering project can lead into the occurrences of management risks. Result in Table 1 also illustrate that people risks and technology risks are less to occur compared to management risks. This is due to people risks and technology risks that can be reduce by applying a mitigation plan such as sending the team members for training and always making sure that the team members have good teamwork. These results are crucial to assist the project manager to be aware about the potential software risks that has occurred in software engineering projects.

Subsequently, Figure 1 illustrates the identified software risks that has been visualized based on its importance to the software engineering project. The identified software risks are visualized in risk impact table using risk sensitivity analysis technique. By this way, the project manager can estimate the probability of occurrences and consequences for each software risks. The probability of occurrences and consequences of each software risks will determine the value of risk exposure and percentage of impact if the software risks occur in the software project. The higher the value of risk exposure and percentage of software risk, the higher the impact that the software risk will affect the software project if it occurs. Specifically, the visualizations of software risks are used to provide the project manager an overview on which software risks have the highest impact and which has the lowest impact on the software engineering project. Thus, the project manager can be well prepared and formulate the effective risk management strategies to prevent the software risks from affecting the project progress.

6. Conclusion
This study explores the software risks that occur in software engineering projects. Ten research papers used to synthesize the software risks that have occurred in software engineering projects. The identified software risks were classified into three types of software risks which include management risks, people risks and technology risks. This study also has identified 39 software risks that have occurred in software engineering projects. 21 of the software risks were from the management risks, another 9 software risks were from people risks and the last 9 software risks were from technology risks. From the result, it suggests that management risks have the highest occurrences in software engineering projects compared to people risks and technology risks. This study is important because it adopts risk sensitivity analysis to highlight and visualize the potential software risks that might occur in software engineering projects. Besides that, this study also assists project managers to be more aware with the potential software risks in software engineering projects.

This study has also provided risk impact table to give project managers an overview of the potential software risks in software engineering projects. Thus, the project manager can have adequate time to formulate the risk management strategies and prevent the potential software risks from affecting the project progress.

In the future, this study can be improved by providing various ways to visualize software risks. The visualization can be include by using pie chart, line bar graph and SWOT analysis graph instead of only by using risk impact table and horizontal bar graph. Various ways of visualization can increase the
effectiveness and helps project manager to get clearer overview of software risks in software engineering project.

Acknowledgements
The authors gratefully acknowledge the help of the Ministry of Higher Education for supporting this research, through the Research Acculturation Grant Scheme (RAGS) grant (600-RMI/RAGS 5/3 (4/2014).

References
[1] Arnaut, B. M., Ferrari, D. B., & e Souza, M. L. D. O. (2016). A requirements engineering and management process in concept phase of complex systems. IEEE International Symposium on Systems Engineering (ISSE). IEEE.

[2] Asif, M., Ahmed, J., & Hannan, A. (2014). Software Risk Factors: A Survey and Software Risk Mitigation Intelligent Decision Network Using Rule Based Technique. International MultiConference of Engineers and Computer Scientists (IMECS).

[3] Bannerman, P. L. (2008). Risk and risk management in software projects: A reassessment. Journal of Systems and Software, 81(12), 2118-2133.

[4] Cullen, A. C., & Frey, H. C. (1999). Probabilistic techniques in exposure assessment: a handbook for dealing with variability and uncertainty in models and inputs: Springer.

[5] Fauzi, M., & Sanim, S. (2014). Mapping of software configuration management issues with the area of responsibility. IEEE 8th International Conference on Application of Information and Communication Technologies (AICT).

[6] Fauzi, S. S. M., et al. (2010). Software Configuration Management in Global Software Development: A Systematic Map. 17th Asia Pacific Software Engineering Conference (APSEC). IEEE.

[7] Fauzi, S. S. M., Ramli, N., & Nasir, M. (2008). Assessing Software Risk Management practices in a small scale project. International Symposium on Information Technology (ITSim).

[8] Kremljak, Z., & Kafol, C. (2014). Types of Risk in a System Engineering Environment and Software Tools for Risk Analysis. Procedia Engineering, 69(0), 177-183.

[9] Lewis, J. P. (2001). Limits to software estimation. ACM SIGSOFT Software Engineering Notes, 26(4), 54-59.

[10] Makhani, S., Khan, A., & Soomro, S. (2010). Project management risk sensitivity analysis. Journal of Information & Communication Technology, 4(Spring), 38-48.

[11] Marchewka, J. T. (2003). Information Technology Project Management–Providing Measurable Organisational Value, 2003: John Wiley and Sons Inc., USA.

[12] Sharif, A. M., Basri, S., & Ali, H. O. (2014). Strength and Weakness of Software Risk Assessment Tools. International Journal of Software Engineering & Its Applications, 8(3).

[13] Smith, E. D., Szidarovszky, F., Karnavas, W. J., & Bahill, A. (2008). Sensitivity analysis, a powerful system validation technique. Open Cybernetics & Systems Journal, 2, 39-56.

[14] Wallace, L., Keil, M., & Rai, A. (2004). Understanding software project risk: a cluster analysis. Information & Management, 42(1), 115-125.

[15] Younis, A., Malaizya, Y. K., & Ray, I. (2016). Assessing vulnerability exploitability risk using software properties. Software Quality Journal, 24(1), 159-202.