

# Requirements Elicitation and Analysis: Towards the Automation of Software Project Risk Management

Azlin Nordin, Lili Marziana Abdullah, Farihin Diayana Mohamad Fadzil, Nor Aqila Syamira Roselan

Department of Computer Science  
Kulliyyah of ICT  
International Islamic University Malaysia  
Gombak, Malaysia  
{azlinnordin, lmarziana}@iium.edu.my,  
{diyanafadzil, noraqilasyamira}@gmail.com

**Abstract**—Risk management (RM) has always been the focus of software project management studies. The abilities to monitor and control the software project risks are among the critical issues examined by various researchers. This paper discusses steps in the RM process based on Boehm model. A survey was conducted to know the perceptions of respondents on their practices of managing project risks. This paper also reports the results of the survey concerning the desired features for risk management tool (RMT). In addition, a comparison with other RM process model is also included. Requirements were gathered from domain experts and were analyzed. Based on the results, the authors proposed, designed and developed the tool to automate the Boehm's RM process.

**Keywords**—risk management tool; risk management; software project; Boehm model

## I. INTRODUCTION

The management of risks in projects is one of the main topics of interests for researchers and practitioners working in the area of project management. In general, a risk can be defined as a possibility of loss, the loss itself, or any characteristics, object or action that is associated with the possibility [1]. In this work, we focus only on software risks. Whilst, RM is the process to identify and respond to risks throughout the project life cycle.

Boehm [2] stated that RM is a series of steps, which goals are to identify and eliminate software risks before they become either threats or a major source of expensive rework. The main purpose of RM is to acknowledge all possible risks to a project, assess their likelihood and consequence, and then determine resolution steps depending on the nature of the risks [3]. The idea is to minimize any unforeseen and unexpected issues arising during the course of the project by properly planning for contingencies.

Proper planning leads to minimizing uncertainties, which might lead to more critical problem in the development. This shows that RM is essential to be implemented in projects. Amongst the benefits of RM are the ability to (1) identify alternative actions, (2) increase confidence in achieving

project objectives, (3) improve potential of success, (4) increase estimation accuracy and many more [4].

Hence, the objectives of the study are to (1) investigate the existing RM models; (2) compare the models; (3) elicit requirements for RM process; and (4) validate the requirements through prototyping technique. For objectives (1) and (2), we investigated the existing RM models from the literature. This research also focuses on the study of RM techniques applied to software projects based on Boehm's model.

In order to achieve objective (3), we designed a survey to get the respondents' feedback on their current RM practices. Finally, from a thorough examination of the literature and the results from the survey, we designed and developed the prototyping tool i.e. *e-Risk* to validate the RM process requirements. This includes the best practices, techniques and processes that are likely to contribute to the effectiveness of the management of risks. Detailed discussion on requirements of each of the processes will be included in Section III. It is important to highlight that the scope of this work is only on functional requirements.

The paper is structured as follows: In the next section, the literature is discussed. A comparison with existing RM models is included. Following this, the research methodology is elaborated. Then, the empirical study is described; the analysis and the main findings are presented before conclusions are reached. Finally, conclusions are drawn on directions for future research and practices.

## II. LITERATURE REVIEW

A common approach in RM is the process model approach that specifies the stepwise tasks for managing risks. Influential RM process models in software engineering are associated with Boehm [2, 5], Charette [6] and Kontio [1, 7], and are also found in various industry and national standards for example AS/NZS ISO 31000:2009 [8], CMU/SEI-2006-TR-008 [9] and ISO/IEC 16085:2006 [10]. These models include a generic set of processes for example risk identification, risk analysis, risk evaluation and risk treatment

[8]. The major contribution of process models is that they specify the individual activities to guide and direct development teams to effectively manage risks. These steps are usually planned to be executed sequentially and iteratively to manage known and new risk factors as project progresses to completion.

This research revisited two early, dominant RM process models - the classic Boehm model [2, 5] and Kontio's RM Process (Riskit) [1, 7]. This section discusses the steps of RM through the two RM process models. The following subsections briefly discuss each of the models.

#### A. Boehm Model

The first formal RM process in software development was defined by Barry Boehm in 1989 [2, 5]. Boehm model [5] focuses on the concept of "risk exposure" as defined by the relationship, where the probability of an unsatisfactory outcome and the loss due to the unsatisfactory outcome determine the valence of the risk event.

Boehm [5] focused on software risks and identified 10 top software risk items to be addressed by software development projects. Boehm introduced a RM process model to manage risks that covers all software development phases as stated in [11].

The first step is *risk identification*, where the process entails listing all conceivable risks to the project. Boehm [5] identified 10 top risks in software projects i.e. (1) Personnel shortfalls; (2) Unrealistic schedules and budgets; (3) Developing the wrong functions and properties; (4) Developing the wrong user interface; (5) Gold plating; (6) Continuing stream of requirements changes (7) Shortfalls in externally furnished components; (8) Shortfalls in externally performed tasks; (9) Real-time performance shortfalls; and (10) Straining computer-science capabilities. Boehm then listed software risk question items relevant to the top 10 risks.

The second process is *risk analysis*, in which risks are evaluated and then followed by *risk prioritization* [5]. This involves determining the probability of occurrence and the possible negative effects for each risk. The risk-prioritization stage is used to specify the sequence in which the risks are to be dealt with. After the risks are prioritized, the next process is *risk resolution* or known as *risk treatment*; in which the activities are to plan, mitigate, monitor, and communicate – as discussed in the following section.

Once a risk has been analyzed and prioritized, strategies should be defined as to how the user should handle the risk [5]. The objective of the *risk treatment* process is to develop cost effective options for treating the risks. Treatment options, which are not necessarily mutually exclusive or appropriate in all circumstances, are driven by outcomes that include either *avoiding the risk*, *reducing (mitigating) the risk*, *transferring (sharing) the risk*, or *retention (accept) the risk*.

The subsequent step is *risk monitoring*. It is essential for the user to regularly monitor the progress of the project, track the progress toward resolving high-risk items and taking corrective action where appropriate [2]. Hence, after risks

have been identified, analyzed, and prioritized, the recommended actions are established.

#### B. Riskit Model

Another RM process model is Riskit [1, 7], which was developed by Jyrki Kontio in 1997. Riskit is applied mainly in large organizations and particularly used for IT projects. The Riskit process model has seven main processes i.e. (1) RM definition, (2) goal review, (3) risk identification, (4) risk analysis, (5) risk control planning, (6) risk control and (7) risk monitoring [1]. The detailed discussion on each of the processes can be referred at [1, 7].

Kontio [1] claimed that Riskit provides (1) accurate risks definitions, (2) explicit objectives, constraints and other project drivers, (3) modeling and documenting risk qualitatively, (4) ratio and ordinal scale ranking to prioritize risks, (5) the concept of utility loss to rank the loss associated with risk, (6) explicit model for different perspectives of stakeholders, and (7) operational definition and training support.

#### C. Comparison between Boehm's Model and Riskit Model

In general, both the models cover all the relevant RM processes. Stern and Arias [11] stated that the effectiveness of the Riskit model is that it can generate accurate results of risks by using the *probability theory*. Probability theory is a branch of mathematics concerned with the analysis of random phenomena. However, it fails to cover small to medium sized organizations. Stern and Arias [11] also stated that this model does not bridge the gap between risk estimation and risk metrics. In the end, Boehm model is chosen as the main reference for developing *e-Risk* because of its comprehensive coverage of the RM process.

### III. RESEARCH AND DEVELOPMENT METHODOLOGY

The methodology that is applied for the requirements engineering process in order to derive requirements for the RM process has the combination characteristics of (1) an iterative and incremental methodology and (2) prototyping technique. By using the iterative and incremental methodology i.e. Pandey Requirements Engineering Framework [12, 13] as the main reference for the project, the project is implemented in repeated iterations until the requirements are stable for the implementation phase.

The justification for the iterative and incremental approach is to accommodate integration of changes whilst refining the requirements. Based on the prototype implementation, validation is conducted in order to find errors or faults that have occurred. Sets of test scenario were provided to the users in order to test the system. Results from the validation were gathered by enquiring users' feedbacks. Relevant documentation was also prepared throughout the processes.

Requirements elicitation process was performed by brainstorming, interview, and comparison with existing RMT

and survey techniques. Each of these techniques is described in the following subsections.

### A. Interview and Brainstorming

Initially, brainstorming sessions were conducted with three domain experts in the area of risk management. The domain experts clarified (1) the risk question items in the checklist and (2) understanding about RM process. In addition, an interview session with an industry practitioner was also conducted. The interview sessions were conducted in open-ended questions to allow the interviewee to respond more spontaneously.

Based on the brainstorming and interview sessions, there is evidence that RM is important for software projects as uncertainties can moderate the effects of project planning on process performance. However, based on the interview session, the interviewee stated that some organizations do not implement RM process correctly and effectively. This may be due to the RM culture that is not developed within organizations or the project managers are not familiar with the processes of managing risk.

### B. Comparison With Existing RMT

In addition to the interview technique, several existing RMTs have been studied to investigate the main features that should be included and the changes that can be made to improve the quality of the tool. These tools are SimpleRisk [14], Iris Intelligence [15] and Modulo Risk Manager Software [16] (see Table I).

TABLE I: COMPARISON OF EXISTING RMT

| RMT                                      | ID | AL | PR | MG | RE | MO |
|--|----|----|----|----|----|----|
| <b>Simple-Risk [14]</b>                  | √  | √  | √  |    |    |    |
| <b>Iris Intelli-gence [15]</b>           | √  | √  | √  |    |    | √  |
| <b>Modulo Risk Manager Software [16]</b> | √  | √  | √  |    | √  |    |

Legend: ID: Identification, AL: Analysis, PR: Prioritization, MG: Management, RE: Resolution, MO: Monitoring

SimpleRisk is an open source tool to perform RM activities. It has the ability to submit risks, plan mitigations, facilitate management reviews, prioritize for project planning, and track regular reviews. Whilst, Iris Intelligence embeds practices of RM methodology in a fully automated system that can be instantly customized to match specific customer preferences and reporting requirements. Modulo Risk Manager Software helps organizations to automate processes required for risk assessment by collecting and centralizing data relating to technology assets.

Based on the comparisons, it can be concluded that the existing RMTs under investigation do not include all RM processes in order for users to manage risks. Thus, it is

necessary for our work to include all of the RM processes in the mission to automate them.

### C. Survey

Besides interview and brainstorming sessions, in order for the main goal to be accomplished, survey was also conducted to gather requirements. The purpose of conducting the survey is to collect information from respondents in order to achieve the purpose of the study that is to investigate their awareness of RM and importance of having RMTs in managing software risks. This survey was conducted for those who have knowledge in risk domain.

The instrument used in this study was a set of questionnaire, which was divided into two sections. The first part was the demographic information that presents information about the respondents' profiles. The second part on the other hand, assessed respondents' perception on how their companies manage risks and the importance of having RMT. The questionnaire consisted of 16 question items that were 8 yes-no questions and 8 open-ended questions [17].

The statements were open ended so that respondents can answer in detail, and can qualify and clarify responses. Extensive demographic information, however, was not considered relevant to the findings of the study. Through the questionnaire, it was possible to explicitly observe the respondents' awareness level in managing project risks and their perceptions on having RMTs.

#### i. Participants

Copies of the questionnaire were distributed to selected communities in International Islamic University Malaysia (IIUM). The questionnaires were not only distributed to domain experts in risk management, but also to the IT practitioners who have knowledge about risk management. The focus of the survey is to elicit beliefs and perceptions regarding the importance of RM and needs on having RMTs. Gender bias was intended to be avoided but inevitably occurred due to the difference in ratio between male and female participants available at the time of questionnaire distribution. The participants were aware that all the data and measurements obtained from the research will be stored confidentially.

#### ii. Data Collection Procedure

The questionnaires were distributed to IIUM community i.e. academicians, administrative staff by sending the questionnaire online form using Google Docs. The questionnaire's link was sent to the respondents' emails. Using Google Docs can help to reduce cost and time to distribute the questionnaires. The researchers were also able to effectively contact large numbers of respondents by using the online questionnaire.

Besides, it can also ease the respondents to provide their feedback, as it does not require any time restriction as the

questionnaire can be answered online; in which it can be also accessible anytime.

### iii. Data Analysis

After the data was collected, it was organized and analyzed. As the questionnaires were developed in Google Docs, all responses were assembled automatically into a Google Docs spreadsheet, where it is saved as an Excel spreadsheet. For analysis of closed-ended questions, the data were analyzed using descriptive statistics. Frequency tables were also drawn and from these, the data collected were presented in pie diagrams and bar graphs.

### iv. Survey Results

The survey results will be elaborated based on four perspectives i.e. (a) adoption of risk management, (b) RM responsibilities, (c) implementation of requirements management tool (RMT) and (d) necessity of RMT.

a) *Adoption of Risk Management.* Fig. 1 shows the respondents' responses whether their companies follow a particular RM methodology and/or standards. As can be seen, the respondents who answered yes only 13% whereas 88% stated that their companies did not follow any RM methodology and/or standards. The fact that majority of them answered no was may be due to the companies only rely on historical data in order to estimate and manage risks.

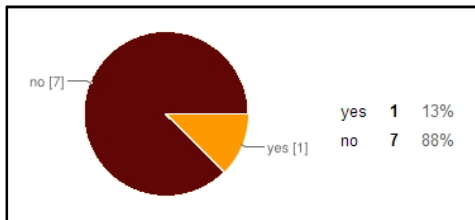


Fig. 1: Company's RM methodology and/or standards

b) *RM Responsibilities.* Respondents were also asked whether their companies have a strategic management or board-level responsible person to look after RM portfolio as shown in Fig. 2. The responses were predictable that majority of them admitted that their companies did not have any dedicated person to look after RM portfolio. This may be due to their companies' project manager only takes project's risk for granted. Thus, this can actually cause the companies to overlook the concealed and even known risks as the companies do not report any risks.

Other than that, the respondents also claimed that they usually manage the project's risk through meeting session among the staff. In this session, the responsible staff will be assigned to resolve and monitor the risk. Nonetheless, the respondents stated that this method has its own weakness as they cannot monitor specific risks that occurred.

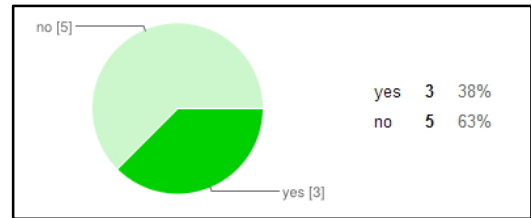


Fig. 2: Strategic management or responsible person for RM portfolio

c) *Use of RMT.* Another important finding as shown in Fig. 3 is on whether the respondents have ever used RMT for their project or not. It is interesting to note that for all respondents, 88% of them have not use RMT for managing risk in their projects. This may be either they were not familiar with the practice or the extent of RMT was indeed very low. This is also because the technique used to manage risk is manual that is by documenting the risks and also having meeting sessions.

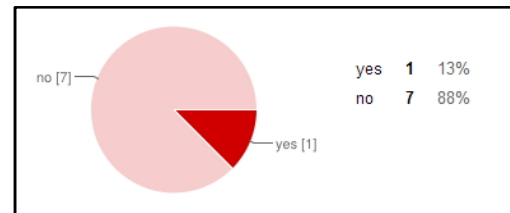


Fig. 3: Use of RMT in project

d) *Necessity of RMT.* Last part of the questionnaire was to collect the respondents' opinions on the needs of RMT for managing risks in their projects. The findings were in agreement with Dhlamini, Nhamu, and Kachepa's [18] study that shows that in order to provide more effective ways of risk management, software tools, which are intelligent and adaptive to RM strategies, are needed. The analysis showed that the respondents cannot agree more that the main reason that they need RMT is because to ensure that risks can be resolved effectively and efficiently.

By digesting and summarizing all the findings from the survey, we conclude that the use of RM practices (tools) is relevant in managing software project. Majority of the respondents were aware that RMTs could help their companies to systematically identify, resolve and monitor risks that they encountered.

### D. Application of Boehm's Model

This subsection discusses our selection of RM techniques to be applied in the Boehm's model. As elaborated in Section II, the processes in Boehm's model will be further explained focusing on techniques for each process.

i) *Risk Identification.* In order to ensure that all risks areas are comprehensively addressed, we further mapped the Boehm's risk items to SEI Taxonomy [19]. A checklist is the most simple yet one of the effective techniques that has been used for risk identification because it allows for exposure identification and documentation [20]. Besides, having a checklist to identify risks provides a clearer picture for users

regarding their projects and also makes it easier to identify user's risk exposure.

In this work, the checklists were forwarded to the right candidates and in the right manner to produce optimum results. The main benefit of the checklist technique to RM is that it provides a quick, low cost way of identifying and assessing the risk exposure of a project against the major factors found by others to be important in determining the outcome of software projects.

ii) *Risk Analysis and Prioritization*. As for risk analysis process, the technique that has been used as a reference is a 5x5 risk rating matrix technique. According to [22], a risk matrix is used to identify the risks levels. This is a simple mechanism to increase visibility of risks and assist management decision making. The x-axis for risk matrix is the consequences of the project whereas y-axis is the likelihood of risks that can occur in a project. The consequences are referred to the findings of study conducted by Adisson and Vallabh [22] in the most important risk factors based on project manager experience.

Based on the findings in [22], it can be summarized that the most important consequences are in terms of scope, schedule, budget, staffing and competencies, and also quality. The consequences are ranked from insignificant as the *lowest*, *negligible*, *moderate* and *extensive* to significant as the *highest*. The likelihood or can be known as the frequency of the risk occurrence is ranked from rare, unlikely, possible, likely to almost certain. The consequences and likelihood that have been developed were adapted from [22].

After the risks have been organized into a risk table, the risks were prioritized by ranking them into 5 levels that are *very low*, *low*, *medium*, *high* and *very high*. Some of them have very low consequences or very low likelihood of occurring – or both. If numerical values were given for likelihood and consequence, the risk exposure can be calculated. Risk exposure is calculated as follows [5]:

$$\text{Risk Exposure (RE)} = L(\text{UO}) \times C(\text{UO});$$

where L = likelihood of occurrence of an unsatisfactory outcome (UO) and C is the consequence of the loss to the project should the risk occur.

iii) *Risk Control (RM Planning, Risk Resolution and Risk Monitoring)*. Based on the risk Items, the prototype automatically generates a chart for the risk resolution whether the risk is supposed to undergo mitigation, retention, transfer or avoidance and it also gives recommendations on how to respond to those risks. For the risk monitoring technique, a risk report that consists of all information related to the identified risks is generated in order for the user to keep track or record the risks [23].

The risk reporting module provides the ability to generate in depth management reports and statistics. Detailed reports and statistics including graphs can be generated for each individual module. The report covers all areas from risk identification process until risk resolution process. Print supports include export feature to Microsoft Word and Adobe PDF thus assisting the user to further the analysis and tracking.

## E. Requirements Result

There are 50 functional requirements that can be concluded based on analysis of the findings and also from the literature review. Those requirements have been categorized according to 9 modules that are: (1) identification (4 requirements); (2) analysis (4 requirements); (3) prioritization (3 requirements); (4) resolution (4 requirements); (5) documentation (6 requirements); (6) database (6 requirements); (7) communication (7 requirements); (8) authentication (6 requirements); (9) profile (10 requirements).

In this paper, because of space constraint, we only choose to include the risk identification module (see Table II). Software requirements specification is included in the project documentation.

TABLE II: RISK IDENTIFICATION MODULE

| REQ        |                      | DESCRIPTION   |
|------------|----------------------|---|
| <b>ID1</b> | view_category        | User shall be able to view the risk categories. The risk categories are requirement (stability), requirement (validity), design (interface), design (non-development), design (performance), engineering specialties, development, management method, resources and contract. |
| <b>ID2</b> | select_category      | User shall be able to select the risk categories. Risk categories refer to ID1.   |
| <b>ID3</b> | mark_checklist       | User shall be able to provide feedback for each of the checklist items provided at each category after choosing the certain risk categories.  |
| <b>ID4</b> | view_identified_risk | User shall be able to view the possible risk based on the checklist answered before on each risk categories. Requirement referred to ID3.   |

## F. Requirements Analysis and Design.

Those 50 requirements were analyzed by conducting requirement validation. In this phase, the requirements were reviewed by domain expert who is knowledgeable in RM area. Other than that, a prototype was also developed in order for the authors to have more understanding and clarification about the tool. The prototype was developed using Mockup prototyping tool [24]. It supports easy and inexpensive modification to design, which makes this method useful in the early phases of design.

After users choose the risk category, there will be a checklist that is related to each category. Users are able to tick the questions in the checklist that are assumed to be related to their projects. By conducting requirement analysis, it is easier to identify the missing and incomplete requirements for the tool development. In design process, relevant diagrams (i.e. use case, activity, entity-relationship diagram and architecture) were derived in order to represent and design the requirements.

## G. Evaluation

Requirements validation was conducted in order to find missing requirements or errors in the RM requirements. Sets of test scenario were provided to the users in order to evaluate the system. Lastly, the problems were documented. The result

from the validation was evaluated in this phase by enquiring users' feedbacks. All the feedbacks were studied and analyzed before conducting any changes or updates.

#### IV. CONCLUSION AND RECOMMENDATION

This work should help the academicians, researchers, and practitioners to understand the requirement engineering process in gathering the requirements for the RM process. In addition, software developers may also use the gathered requirements to further build the complete RMT.

It is recommended that in future work, the automated process should incorporate knowledge-based and intelligence properties. This could increase its effectiveness in software development projects. Such tools could have the ability to learn and adapt its behavior depending on what exactly transpires during a project's life cycle. Furthermore, use of such a tool in future projects ensures continuity in the use of experience in RM from current projects. Above all, this study has contributed in understanding software risk management.

#### ACKNOWLEDGMENT

We take this opportunity to express our profound gratitude to the university, faculty and to those who have directly or indirectly contributed to this work for their support.

#### REFERENCES

- [1] J. Kontio, "The riskit method for software risk management, Version 1.00," Computer Science Technical Reports. University of Maryland, College Park, MD, USA, 1997.
- [2] B. W. Boehm and R. Ross, "Theory-w software project management: Principles and examples," *IEEE Trans. Softw. Eng.*, vol. 15, no. 7, pp. 902–916, Jul. 1989. [Online]. Available: <http://dx.doi.org/10.1109/3229489>
- [3] S. C. Misra, V. Kumar, U. Kumar, "Different Techniques for RMin Software Engineering: A Review", Eric Sprott School of Business Carleton University, Banff, Alberta, 2006.
- [4] P. L. Bannerman, "Risk and risk management in software projects: A reassessment," *Journal of Systems and Software*, vol. 81, no. 12, pp. 2118–2133, 2008.
- [5] B. Boehm, "Software risk management: principles and practices," *IEEE Software*, vol. 8, no. 1, pp. 32–41, Jan. 1991.
- [6] R. N. Charette, *Software engineering risk analysis and management*. Intertext Publications, 1989.
- [7] J. Kontio et al., *Software engineering risk management: A method, improvement framework, and empirical evaluation*. Helsinki University of Technology, 2001.
- [8] M. Leitch, "ISO 31000: 2009: The new international standard on risk management," *Risk Analysis*, vol. 30, no. 6, pp. 887–892, 2010.
- [9] C. P. Team, "CMMi for development, version 1.2, CMMi-dev v1. 2, continuous representation," CMU/SEI-2006-TR-008, Technical Report, Software Engineering Institute (August 2006), <http://www.sei.cmu.edu/pub/documents/06.reports/pdf/06tr008.pdf>, Tech. Rep., 2006.
- [10] "ISO/IEC 16085:2006 standard, systems and software engineering – life cycle processes - risk management," 2006.
- [11] R. S. MBA and J. C. Arias, "Review of risk management methods," Volume 4-Number 1-January 2011-Semi annual Publication, vol. 4, no. 1, pp. 59, 2011.
- [12] D. Pandey, U. Suman, and A. Ramani, "An effective requirement engineering process model for software development and requirements management," in *Advances in Recent Technologies in Communication and Computing (ARTCom)*, 2010 International Conference on. IEEE, pp. 287–291, 2010.
- [13] D. Pandey, U. Suman, A. Ramani, and D. AhilyaVishwavidyalaya, "A framework for modelling software requirements," *International Journal of Computer Science Issues*, vol. 8, no. 3, 2011.
- [14] SimpleRisk: Enterprise Risk Management Simplified, [Online]. Available: <http://www.simplerisk.org/> [Accessed: May 2, 2014].
- [15] Iris Intelligence, [Online]. Available: <http://www.irisintelligence.com/> [Accessed: May 2, 2014]
- [16] Modulo Risk Manager Software, [Online]. Available: <http://modulo.com/vendor-risk-management/> [Accessed: May 1, 2014]
- [17] Risk Management Questionnaire, [Online]. Available: [https://docs.google.com/forms/d/1\\_bTX7r0sfm6l5dBCGnx54MB50a7o2DY\\_2-Zx6Y2EloA/viewform](https://docs.google.com/forms/d/1_bTX7r0sfm6l5dBCGnx54MB50a7o2DY_2-Zx6Y2EloA/viewform) [Accessed: May 3, 2014]
- [18] J. Dhlamini, I. Nhamu, and A. Kaihepa, "Intelligent risk management tools for software development," in *Proceedings of the 2009 Annual Conference of the Southern African Computer Lecturers' Association*. ACM, pp. 33–40, 2009.
- [19] M. J. Carr, S. L. Konda, I. Monarch, F. C. Ulrich, and C. F. Walker, "Taxonomy-based risk identification," DTIC Document, Tech. Rep., 1993.
- [20] "Risk Assessment Questionnaires and Coverage Checklist" Big I Advantage, Inc., and Swiss Re Corporate Solutions, (Ed.08/12-1). [Online]. Available: [https://www.michagent.org/eweb/PDF/Member\\_Services/EO/Questionnaire\\_Checklists.pdf](https://www.michagent.org/eweb/PDF/Member_Services/EO/Questionnaire_Checklists.pdf) [Accessed: May 1, 2014].
- [21] L. Simpleman, P. McMahon, B. Bahnmaier, K. Evans, and J. Lloyd, "Risk management guide for DOD acquisition," DTIC Document, Tech. Rep., 1998.
- [22] T. Addison and S. Vallabh, "Controlling software project risks: An empirical study of methods used by experienced project managers," in *Proceedings of the 2002 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology*. South African Institute for Computer Scientists and Information Technologists, pp. 128–140, 2002.
- [23] E. Wallmüller, "Risk management for it and software projects," in *Business continuity*. Springer, pp. 165–178, 2002.
- [24] The Online Vector Based Mockup and Wire framing Tool, [Online]. Available: <https://moqups.com/> [Accessed: May 1, 2014]