USABILITY BASED RISK ASSESSMENT MODEL FOR SOFTWARE DEVELOPMENT PROCESS

JAYALETCHUMI A/P T. SAMBANTHA MOORTHY

UNIVERSITI TEKNOLOGI MALAYSIA

USABILITY BASED RISK ASSESSMENT MODEL FOR SOFTWARE DEVELOPMENT PROCESS

JAYALETCHUMI A/P T. SAMBANTHA MOORTHY

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Doctor of Philosophy

Advanced Informatics School
Universiti Teknologi Malaysia

JUNE 2016

Dedicated to my family especially my son, Kuhen who has been a strong and steadfast support in my PhD journey. They taught me the value of life and faithful love. Most of all, I cannot fully express in words for priceless love and encouragement that my son has brought in my life.

ACKNOWLEDGEMENT

Firstly, I would like to thank god for giving me determination and strength in completing this thesis.

Next, I would like to express immeasurable appreciation and deepest gratitude to my supervisors, Prof. Dr. Suhaimi Bin Ibrahim and Dr. Mohd Naz'ri Bin Mahrin for their untiring support, constructive input, advices, words of encouragement and guidance in making this study possible.

I also would like to thank my close friends, Dr. Huma Khan, Dr. Sugumaran Nallusamy, Ms. Mariayee Doraisamy and Ms. Sri Lakshmi Kanniah for their inspiring discussions and continual motivation which encouraged me further in completing this study successfully. My special gratitude goes to my parent, family members and friends for their help which has ease my PhD journey.

My special thanks also go to all the experts and respondents involved directly and indirectly in my research, especially Mr. Kathirrasan and Mr. Subramani from MAMPU for their continuous advice and support during data collection for my research. Last but not the least, I would like to thank the staff at Advanced Informatics School (AIS) and other co-researchers at who have helped me in many ways to complete this research.

ABSTRACT

Software usability is an important factor in ensuring the development of quality and usable software. Ignorance, unawareness and failure to address usability during the software development process lead to poor quality software that is associated with potential usability risks. Risk management can be used to assess and control these usability risks. However, currently knowledge on usability risks is still insufficient and model to assess these risks is also lacking, leading to ignorance in managing usability risks in the software development lifecycle process (SDLC). This thesis proposes to develop a new Usability Risk Assessment Model to assessment of usability risks during the SDLC. Initially, elements of the Usability Risk Assessment Model were identified using Systematic Literature Review (SLR) whereby five major elements, namely, Risk Identification, Risk Analysis, Risk Prioritization, Risk Classification and Risk Mitigation were included in the model. Subsequently, feedback from 270 respondents of a survey questionnaire was utilized to identify 38 possible usability risk factors, which were then used to define 42 potential usability risks. These usability risks were used as keywords in identifying 85 initial usability vulnerabilities from the literature, which were grouped into four main categories that influence software development outcomes: Institutional Context, Software Project Content, People and Action, and Development Processes. The above usability risks and their vulnerabilities were then validated by four selected experts from the Public Sector. After validation, a total of 88 distinct usability vulnerabilities for various usability risks were identified. The usability risks were analysed using the Delphi method, involving seven experts to identify the probability of occurrences, impact on SDLC phases and mitigation plans for usability risks. Aided by the probability of occurrences and impact on SDLC phases, the usability risk exposure level was quantified, and used to classify and prioritize usability risks on SDLC phases. A Web-based Usability Risk Assessment Tool as a proof-of-concept was developed using ASP.Net to automate detailed elements in order to support the implementation of the model. Using this tool, multiple case study evaluations on four software projects in the Public Sector of Malaysia had demonstrated an inverse relationship between number of usability risks and usability of software. Thus, with the proposed Usability Risk Assessment Model, usability risks can effectively identified, analysed, prioritized, classified and mitigated during software development process to reduce these risks in order to enhance the usability of software. The contributions of this research are; first, a validated list of potential usability risks, usability vulnerabilities and possible mitigation plans for the usability risks; second, classification and prioritization of usability risks on SDLC phases; and third, empirically evaluated the Usability Risk Assessment Model.

ABSTRAK

Kebolehgunaan perisian merupakan faktor penting dalam pembangunan perisian yang berkualiti dan berguna. Kejahilan, ketaksedaran dan kegagalan untuk menitikberat kebolehgunaan semasa proses pembangunan perisian telah menghasilkan perisian berkualiti rendah yang dikaitkan dengan risiko kebolehgunaan. Pengurusan risiko penting untuk menilai dan mengawal risiko kebolehgunaan. Walau bagaimanapun, kini, pengetahuan mengenai risiko kebergunaan masih tidah mencukupi dan model untuk menilai risiko ini juga masih kurang, menyebabkan kejalilan dalam mengurus risiko kebergunaan dalam proses kitaran hayat pembangunan perisian (SDLC). Tesis ini mencadangkan pembangunan *Usability Risk Assessment Model* yang baru untuk menilai risiko kebergunaan semasa SDLC. Pada awalnya, elemen dalam *Usability Risk Assessment Model* dikenalpasti menggunakan Systematic Literature Review (SLR) di mana lima elemen utama iaitu Pengenalpastian Risiko, Analisis Risiko, Keutamaan Risiko, Klasifikasi Risiko dan Mitigasi Risiko telah dikenalpasti. Maklumbalas daripada 270 responden soal selidik telah mengenalpasti 38 faktor risiko kebolehgunaan yang kemudiannya digunakan untuk menentukan 42 risiko kebolehgunaan. Risiko kebolehgunaan ini dijadikan kata kunci bagi mengenalpasti 85 kelemahan kebolehgunaan daripada kajian literatur dan dikategorikan mengikut empat kategori yang mempengaruhi hasil pembangunan perisian: Konteks Institusi, Kandungan Projek Perisian, Pengguna dan Tindakan, dan Proses Pembangunan. Segala risiko dan kelemahan kebolehgunaan disahkan oleh empat pakar daripada sektor awam. Selepas pengesahan, sebanyak 88 kelemahan kebolehgunaan telah dikenalpasti. Risiko ini dianalisis dengan kaedah Delphi bersama tujuh pakar bagi menentukan kebarangkalian risiko, impak ke atas fasa SDLC dan pelan mitigasi bagi risiko kebolehgunaan. Dengan bantuan kebarangkalian risiko dan impak ke atas fasa SDLC, tahap pendedahan risiko kebolehgunaan ditentukan dan digunakan untuk tujuan pengutamaan dan klasifikasi risiko ini pada fasa SDLC. Satu aplikasi berasaskan web, *Usability Risk Assessment Tool* telah dibangunkan dengan ASP.Net bagi mengautomasikan elemen-elemen dan menyokong pelaksanaan model yang dicadangkan. Dengan menggunakan aplikasi ini, penilaian kajian kes ke atas empat projek perisian dalam Sektor Awam di Malaysia menunjukkan hubungan songsang antara bilangan risiko kebolehgunaan dan kebolehgunaan perisian. Oleh itu, dengan *Usability Risk Assessment Model*, risiko kebolehgunaan dapat dikenalpasti, dianalisis, diutama, diklasifikasi dan dimitigasi semasa proses pembangunan untuk mengurangkan risiko ini bagi meningkatkan kebolehgunaan perisian. Sumbangan kajian ini adalah: pertama, menyediakan senarai tersah bagi risiko kebolehgunaan, kelemahan kebolehgunaan dan pelan mitigasi yang mungkin bagi risiko kebolehgunaan; kedua, pengutamaan dan klasifikasi risiko kebolehgunaan pada fasa SDLC; dan ketiga, penilaian secara empirikal *Usability Risk Assessment Model*.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ACM	-	Association for Computing Machinery
ADO.Net	-	ActiveX Data Objects
ASP.Net	-	Active Server Pages .Net
CV	-	Coefficient of Variation
ICT	-	Information Communication and Technology
IEEE	-	Institute of Electrical and Electronics Engineers
IIS	-	Internet Information Server
MAMPU	-	Malaysian Administrative Modernization and Management Planning Unit
REBOK	-	Requirement Engineering Body Of Knowledge
SDLC	-	Software Development Life Cycle
SEI	-	Software Engineering Institute
SLR	-	Systematic Literature Review
SPSS	-	Statistical Package for Social Science
SQL	-	Structured Query Language
SRS	-	Software Requirement Specification
SWEBOK	-	Software Engineering Body Of Knowledge
WWW	-	World Wide Web

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

This chapter presents introduction on development of a usability risk assessment model as a mechanism to reduce usability problems in software. The first section explains the background of the research problem, followed by the problem statement, research questions, objectives, and scope of the research. The final section presents the significance of this research, and provides a brief description on key terms applied throughout the thesis.

1.1 Background of the Problem

The demand for faster, larger-scale software with better performance has increased during the past couple of years. However, the dependency on software has created great concern and criticism on quality problems in software currently being used. Meeting users' expectations for quality software has been a tedious task for software developers (Okonta, Ojugo, Wemembu, & Ajani, 2013). This is attributable to unanticipated problems such as missing the deadlines, poorly defined software requirements, budgets being overrun and failing to deliver the expected business value (i.e., ROI) (Al-Ahmad et al., 2009; R. Kaur & Sengupta, 2013). Consequently, the existence of quality problems in software has led to many software failures (R. Kaur & Sengupta, 2013; Verner, Sampson, & Cerpa, 2008), in line with reports by Standish Group (2011), which demonstrates that only 37% of software projects developed between 2002 and 2010 were successful (Schwaber & Sutherland, 2012)

Users' expectations for quality software are related to product quality (characteristics of software) and quality in use (the interaction of different users with software to meet their needs). In order to fulfil users' expectations, *Usability* appears as a highly relevant quality attribute in achieving product quality and quality in the use of software, since usability can only be measured when the software are being used. The ISO/IEC 25010 (2011) standard defined usability as "*the extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use*." In addition, studies on existing quality models (Dromey, 1995; ISO/IEC25010, 2011; McCall, Richards, & Walters, 1977) also recognized usability as an important quality factor, and it has always been stated even in the first model of software quality, referred to as *McCall Factors Criteria Metrics (FCM)* (McCall, (1977).

However, the inability of a development team to establish this factor causes software to be plagued by usability problems that have negative impact on the overall quality of the software (Farooq & Quadri, 2012; Hua & Gong, 2013). One of the main causes behind usability problems is ignorance and unawareness about existing usability standard (S.K. Dubey & Rana, 2010). Furthermore, product design with less usability activities, inadequate usability skills and knowledge, unawareness usability engineering life-cycle activities, and inappropriately applying usability methods have led to many usability problems in software (Jokela, 2005). Examples of usabiliy problems are increase in broken links, short of interactive features and accessibility features and slower accessibility speed (Isa, Suhami, Safie, & Semsudin, 2011). Moreover, usability problems underlying operating process, interface design, and product structure have resulted in lower efficiency, effectiveness, and difficulty of use for end users (Liang, Deng, & Wang, 2009). Usability problems in software influence the overall usage level of the software (Inversini, Cantoni, & Bolchini, 2011), reaffirmed in a 2011 survey report by the Malaysian Administrative Modernization and Management Planning Unit (MAMPU), which reported that the usage of online services by the Malaysian government was merely 40%, suggesting higher usability problems in online services (MAMPU, 2011). Any effort to improve software usability after its development is not recommended, since it only increases costs, and involves, to a certain degree, in the remaking of the product (Sharma, Kalia, & Singh, 2012).

Generally, the existence of usability problems could lead to software failure; by reducing these problems, more usable and quality software could be produced (Ahmed Seffah, Donyaee, Kline, & Padda, 2006).

Numerous approaches to decrease usability problems in software have been suggested in past studies in the related literature. Usability evaluation activities such as empirical testing, inspection, and metrics for usability standards were used to evaluate the usability of software (Humayoun, 2012). This only evaluates a completed system, and does not intervene in earlier stages of the development process (Lilja, Laakso, & Palomki, 2011). Alternatively, Den Ouden (2006) discovered that most problems related to software products can be traced back to design decisions made throughout the development process. As the quality of a product is greatly influenced by the quality of the process used to develop it, it is critical to tightly integrate usability with the Software Development Life Cycle (SDLC) to develop software with fewer usability problems (Lindgaard, 1994).

A great deal of effort has been taken to incorporate formal usability standards, processes, techniques and practices into SDLC in order to improve interaction and software quality (Durrani & Qureshi, 2012; Ferre, 2003; Fischer, 2012; Heiskari, Kauppinen, Runonen, & Mannisto, 2009). However, usability practices are only integrated in certain phases of the software development process, e.g., the requirements and design phase (Carlshamre & Rantzer, 2001; A. Seffah, Djouab, & Antunes, 2001). In fact, practical implementation of this integration is lacking (Durrani & Qureshi, 2012). Furthermore, in most software development projects, usability requirements are not part of the software requirements, causing incomplete, confusing, and contradictory requirements for developers, which have resulted in difficulties for development teams to avoid usability problems (Heiskari et al., 2009). Meanwhile, ISO 13407 (1999), an international standard, proposed a framework to integrate usability in all software development phases with use of a User-Centred Design (UCD) approach. Even so, usability problems still seem to reoccur in software. This reoccurrence is possibly attributable to the presence of usability risks in all software development phases, each of which needs to be addressed independently.

Generally, a *usability risk* can be understood as the potential action or activity that leads to undesirable outcomes that impact the usability of software. Unmanaged usability risks affect the software development process, and in turn lead to various usability problems in end products. However, this term is not widely used in the literature. The majority of prior work uses the term *usability problems* and not *usablilty risks*. The term *usability risk* was first introduced in the e-commerce and World Wide Web services context (Platt, 1999). Some studies on the sources and consequences of risks related to mobile applications used this term in their research (Dey & Häkkilä, 2008; Jin, Ko, Mun, & Ji, 2007; Ketola, 2002). Further investigation also revealed that knowledge on usability risks is still lacking, whereby there is a lack of effort in identifying and unifying usability risks as a unique entity, either in the form of checklists, models or others. Consequently, inadequate knowledge on usability risks has caused constraints in managing these risks during the software development process, which hinders the effort to reduce usability problems associated with the end product. Futhermore, The approach of usability risk management can only be entirely implemented once adequate knowledge on the associated risks is acquired. Since knowledge on usability risks is still lacking, there has been great ignorance in prior work in managing usability risks compared to other risks such as those of market, technology and money (Platt, 1999). Thus, there is a need to perform risk assessment prior to risk control in order to gain adequate knowledge on these risks (Boehm, 1991).

Usability risk assessment is defined as "a systematic process to identify, analyse and prioritise usability risks that can affect the achievement of project objectives in the aspect of usability". Existing software risk assessment models were found to assess various risks in various domains such as project risks (Bazaz, Gupta, PrakashRishi, & Sharma, 2012), security risks (Mkpong-Ruffin,Umphress, Hamilton, & Gilbert, 2007), technical risks (Loutchkina, Jain, Nguyen, & Nesterov, 2013), managerial risks (Yan-qiu, Chi, & Ying, 2012), and cultural risks (Wattanapokasin & Rivepiboon, 2009). Most existing software risk assessment models focus on project risks and process risks, not on risk based on the quality attributes. Only Mofleh and Zahary (2011) proposed a framework called SPRMQ (Software Risk Management based on Quality attributes and Operational Life Cycle), which manages risk based on quality attributes such as Functionality, Reliability, Efficiency, Performance, and

Maintainability. Usability was not taken into consideration in this model. Since there is lack in models used to assess usability risks, there is a need for the development of a risk assessment model based on usability risks, which aims to reduce usability problems in software.

1.2 Problem Statement

Development teams currently face challenges in assessing potential usability risks during the SDLC for several reasons. Firstly, inadequate knowledge on usability risks has become a hindrance for development teams in reducing software usability problems. Subsequently, knowledge about usability risk factors and vulnerabilities that can further explain this concept is also lacking. Many studies have focused on risks underlying projects and processes meanwhile only a limited number of studies focused on risks underlying quality attributes. Overall, studies that focus on usability risks are still lacking. Secondly, the lack of knowledge on usability risks has created constraints in assessing them during the software development process. Without proper assessment, a risk control approach cannot be effectively implemented. In terms of risk assessment, there is a lack of models that assess usability risks during the SDLC. If development teams continue to develop software without assessing usability risks, the chances of producing software with more usability problems are higher.

Hence, we propose to develop a Usability Risk Assessment Model to assess usability risks during the SDLC. With usability-based risk assessment model, usability risks are identified, analysed and prioritized on software development phases by which the existence of usability risks during software development process could be made evident and better understood by the development team. Little effort in past studies in identifying, analysing and prioritizing potential usability risks in software development phases has provided evidence that a usability based risk assessment model is certainly needed in order to assist development teams to produce software with less usability problems, and in turn fulfil the expectation of end users. This can improve software quality and reduces the risk of software failure.

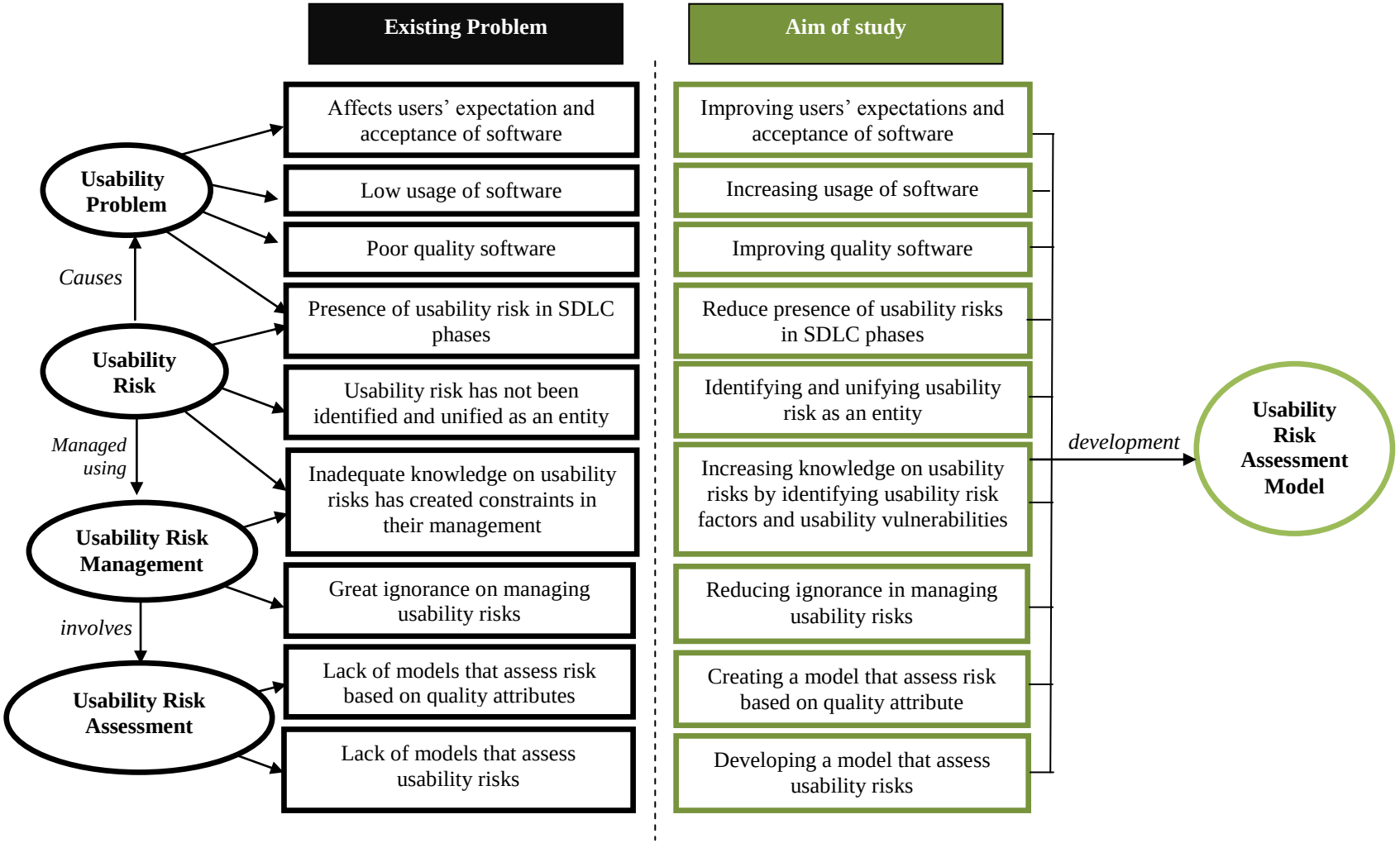

Figure 1.1 Development of problem statement

1.3 Research Questions (RQ)

This research aims to assess potential usability risks during the software development process by using the proposed Usability Risk Assessment Model. This research is expected to help development teams in identifying, analysing and prioritizing potential usability risks during the software development process to reduce usability problem. Based on the specified research problem, three research questions have been formulated, which are as follows:

RQ 1: What are the elements and methods in the Usability Risk Assessment Model?
This question answers the concern on the elements that constitutes in the proposed model and the methods to implement these elements.

RQ 2: How potential usability risks can be incorporated into SDLC phases?
This question answers the concern on how usability risks can be incorporated into SDLC phases.

RQ 3: How a suitable Usability Risk Assessment Model can be proposed based on the above findings?
This question answers the concern on how the findings from RQ1 and RQ2 can be used to develop the proposed model.

1.4 Research Objectives (RO)

The objectives of this research have been formulated based on the five research questions as follows:

RO 1: To investigate elements and methods in the Usability Risk Assessment Model
This objective determines the elements and method used in the proposed model.

RO 2: To identify potential usability risks that influences the usability of software

This objective determines the potential usability risks which effects software usability.

RO 3: To analyse and prioritize potential usability risks in SDLC phases

This objective analyses and prioritises the identified usability risks (from RO 2) in order to incorporate usability risks in SDLC phases.

RO 4: To proposed aUsability Risk Assessment Model

This objective aims to develop the proposed model using the findings from RO 1, RO 2 and RO 3.

RO 5: To evaluate the proposed Usability Risk Assessment Model

This objective evaluates the developed model (from RO 4) to ensure it achieves aim of this research

1.5 Scope of the Study

This section presents the limitations of this study, which mainly includes the software quality and risk assessment approach. A further description on the software quality and risk assessment approach is provided as follows.

- Software quality scope

  This study emphasizes the concept of software quality as explained in the quality standard: ISO/IEC 25010: 2011 Systems and software engineering -- Systems and software Quality Requirements and Evaluation (SQuaRE) -- System and software quality models. Additionally, this study focuses on usability as one of the important attributes that contribute to software quality. This study aims to improve process quality by integrating usability risks into SDLC, which directly contributes to an improvement in product quality, and subsequently improves software *quality in use*.

- Risk assessment approach scope

  Although software risk management processes involve risk assessment and risk control, this study only focuses on the risk assessment process. This is because current knowledge on usability risks is still lacking, and it is difficult to control a risk without conducting a proper assessment process. The advantage of a risk assessment process is that it identifies, analyses, and prioritizes usability risks in SDLC phases, which could potentially increase the knowledge and awareness of development teams to develop more usable software. This study proposed a usability based risk assessment model that involves three core elements: Risk Identification, Risk Analysis and Risk Prioritization. However, the elements of Risk Classification and Risk Mitigation were included in this model to add value to the proposed model.

- Since usability problem is also a common problem faced in Malaysian Public Sector, respondents and experts involved in this study were selected from Malaysian Public Sector. Furthermore, possible usability risks during software development process vary among private and public sector. Thus, focus of this study remains on potential usability risks in software development process at public sector.

1.6 Significance of the Study

Achieving the objectives of this study provides theoretical and practical significance.

- Theoretical significance

  The main aim of this study is to provide a usability based assessment model. This model focuses on usability as a quality factor that integrates well with the software development process. The proposed model contributes to the area of knowledge in the Software Engineering Body of Knowledge (SWEBOK) (P.Bourque and R.E. Fairley, 2014), particularly in Software Quality (Section 1.3-Models and Quality Characteristics and Subsection 1.3.1/1.3.2-Software

Process Quality/ Software Product Quality), Software Engineering Management (Section 3-Software Project Planning and Subsection 2.5-Risk Management), and Software Engineering Process (Section 2.2-Software Life Cycle Models). Since this study focuses on usability, it highlights knowledge on usability risks, usability risk factors and usability vulnerabilities.

- Practical significance

  The proposed model could aid software development teams, including project managers, quality managers and risk management teams to conduct usability risk assessment processes that identify, analyse, and prioritise potential usability risks that impact SDLC phases. This assessment process influences the development of quality software. Organisations could meet users' expectations on quality by ensuring the identified usability risks are handled well in each phase during the SDLC. Knowledge and understanding on potential usability risks could also proactively facilitate the development of usability requirements for software, and support the evaluation of software implementation against its requirements.

1.7 Glossary

This section explains several key terms that have been used throughout this thesis. A detailed discussion for each term is provided in Chapter 2.

(a) **Usability problem** is perceived as an aspect of the system and/or a demand on the user which makes it unpleasant, inefficient, onerous, perturbing or impossible for the user to achieve their goals in typical contexts of use (Lavery, 1997).

(b) **Usability risk factor** is a cause or characteristic that typically influences the possibility of a risk event occurrence (Islam, 2009).

(c) **Risk** is considered a function of the likelihood of a given threat-source's exercising a particular potential vulnerability, and the resulting impact of that adverse event on the organization (Stoneburner et al., 2002)

(d) **Risk event** can be certain or uncertain, and can be influenced by a single occurrence or a series of occurrences. There exists a cause for the occurrence of a risk event, which is known as a risk factor (Islam, 2009).

(e) **Usability risk** is the potential that a chosen action or activity leads to a loss or an undesirable outcome which could impact usability of a software (Naik, 2013).

(f) **Usability vulnerability** is a weakness that can be accidentally triggered or intentionally exploited, which creates the potential for harm to software (Stoneburner et al., 2002).

(g) **Threat** is a potential for a person or natural event to exercise (accidentally trigger or intentionally exploit) specific vulnerabilities (Stoneburner et al., 2002).

(h) **Probability of occurrence** refers to the probability that a potential vulnerability may be exercised within the construct of the associated threat environment (Stoneburner et al., 2002).

(i) **Impact** is the amount of potential losses that an organization could suffer from a negative or harmful risk event (Stoneburner et al., 2002).

(j) **Effectiveness** is **t**he degree to which specified users can achieve specified goals with accuracy and completeness in a specified context of use (ISO/IEC25010, 2011).

(k) **Efficiency** is the degree to which specified users expend appropriate amounts of resources in relation to the effectiveness achieved in a specified context of use use (ISO/IEC25010, 2011).

(l) **Satisfaction** is the degree to which users are satisfied in a specified context of use use (ISO/IEC25010, 2011).

1.8 Outline of the Thesis

This thesis is organised in the following chapters:

Chapter 1 presents introduction to the research, and discusses the background of the research, problem statement, goals, research questions, objectives, scope and significance of the study.

Chapter 2 provides a comprehensive review of related studies in the existing body of literature. Initially, state-of-the-art on models, processes and tools relevant to software risk assessment and software risk management are reviewed. This chapter elaborates usability as one of the significant factors in producing quality software.

Chapter 3 discusses the phases of the research design and methodology in detail. Explanation of the research phases includes related activities and deliverables. This chapter also discusses the research instruments and the evaluation criteria which were adopted in this work.

Chapter 4 documents and illustrates the data collection process using Systematic Literature Review (SLR), and the adoption of an existing mapping study. Using these, elements and methods in the Usability Risk Assessment Model are identified to achieve the first objective of the research: *To investigate elements and methods in the Usability Risk Assessment Model.*

Chapter 5 explains the processes of identifying usability risk and usability vulnerabilities, and provides respective descriptions. The process of expert validation on usability risk and its vulnerabilities are also explained.

Chapter 6 documents and illustrates the analysis of usability risks, as well as their incorporation into the SDLC phases using a Delphi based risk analysis method. Using this method, seven experts were requested to determine the probability of occurrences, impact on SDLC phases, classification and prioritization through risk exposure levels, and mitigation plan for each of the identified usability risks.

Chapter 7 describes the conceptual model of the Usability Risk Assessment Model, and explains in detail the design of the Usability Risk Assessment Tool.

Chapter 8 presents the evaluation of the Usability Risk Assessment Model using a quantitative case study.

Chapter 9 concludes this study by providing the research summary and achievements. The contributions and limitations of this research are also presented. Finally, some suggestions for future work are provided.

1.9 Chapter Summary

This chapter described the background of the problem by explaining current usability problems in software, approaches taken to reduce usability problems and the challenges associated with the implementation of these approaches. Usability problems have been considered risk factors that contribute to usability risks. By effectively managing usability risks, more usable software can be produced. The problem statement suggests that, before usability risks are managed, they should first be identified, analysed and prioritized. This chapter also described the research questions, objectives, scope and significance. The next chapter reviews the current state-of-the-art in the related literature, specifically in the areas of usability, risk management and risk assessment.

REFERENCES

Abran, Alain, Khelifi, Adel, Suryn, Witold, & Seffah, Ahmed. (2003). Usability Meanings and Interpretations in ISO Standards. *Software Quality Journal, 11*(4), 325-338.

Al-Ahmad, Walid, Al-Fagih, Khalid, Khanfar, Khalid, Alsamara, Khalid, Abuleil, Saleem, & Abu-Salem, Hani. (2009). A taxonomy of an IT project failure: Root Causes. *International Management Review, 5*(1), 93-104.

Aloini, Davide, Dulmin, Riccardo, & Mininno, Valeria. (2007). Risk management in ERP project introduction: Review of the literature. *Information & Management, 44*(6), 547-567.

Alonso-Ríos, David, Vázquez-García, Ana, Mosqueira-Rey, Eduardo, & Moret-Bonillo, Vicente. (2009). Usability: a critical analysis and a taxonomy. *International Journal of Human-Computer Interaction, 26*(1), 53-74.

Arunraj, N. S., Mandal, Saptarshi, & Maiti, J. (2013). Modeling uncertainty in risk assessment: An integrated approach with fuzzy set theory and Monte Carlo simulation. *Accident Analysis & Prevention, 55*(0), 242-255.

Ayyub, Bilal M. (2001). A Practical Guide on Conducting Expert-Opinion Elicitation of Probabilities and Consequences for Corps Facilities. Alexandria: Institute for Water Resources.

Babar, Muhammad Ali, Winkler, Dietmar, & Biffl, Stefan. (2007). *Evaluating the usefulness and ease of use of a groupware tool for the software architecture evaluation process.* Paper presented at the ESEM 2007 First International Symposium on Empirical Software Engineering and Measurement.

Babar, Muhammad Ali, & Zhang, He. (2009). *Systematic literature reviews in software engineering: Preliminary results from interviews with researchers.* Paper presented at the Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement.

Basili, Victor R, & Caldiera, Gianluigi. (2012). Improve software quality by reusing knowledge and experience. *Sloan management review, 37*.

Bazaz, Y., Gupta, S., PrakashRishi, O., & Sharma, L. (2012). *Comparative study of risk assessment models corresponding to risk elements.* Paper presented at the

2012 International Conference on.Advances in Engineering, Science and Management (ICAESM).

Bevan, Nigel. (1995). Usability is quality of use. *Advances in human factors ergonomics, 20*, 349-349.

Boehm, B. W., Brown, J. R., Kaspar, H., Lipow, M. , MacLeod, G. J., & Merrit, M. J. . (1978). *Characteristics of software quality* (Vol. 1): North-Holland Publishing Company.

Boehm, Barry W. (1989). *Software risk management*: Springer Berlin Heidelberg.

Boehm, Barry W. (1991). Software risk management: principles and practices. *Software, IEEE, 8*(1), 32-41.

Booth, Paul. (1989). *An introduction to human-computer interaction*: Psychology Press.

Boring, Ronald, Gertman, David, Joe, Jeffrey, Marble, Julie, Galyean, William, Blackwood, Larry, & Blackman, Harold. (2005). *Simplified Expert Elicitation Guideline For Risk Assessment Of Operating Events* (INL/EXT-05-00433). Washington, D.C.

Brereton, Pearl, Kitchenham, Barbara A, Budgen, David, Turner, Mark, & Khalil, Mohamed. (2007). Lessons from applying the systematic literature review process within the software engineering domain. *Journal of Systems and Software, 80*(4), 571-583.

Brooke, John. (1996). SUS-A quick and dirty usability scale. *Usability evaluation in industry, 189*(194), 4-7.

Brooks, Kenneth W. (1979). Delphi Technique: Expanding Applications. *North Central Association Quarterly, 53*(3), 377-385.

Bryant, A., & Charmaz, K. (2010). *The Sage Handbook of Grounded Theory*: Sage Publications.

Burton-Jones, Andrew, & Straub Jr, Detmar W. (2006). Reconceptualizing system usage: An approach and empirical test. *Information systems research, 17*(3), 228-246.

Carlshamre, P., & Rantzer, M. (2001). Business: Dissemination of Usability: Failure of a Success Story. *Interactions, 8*(1), 31-41.

Carr, V., & Tah, J. H. M. (2001). A fuzzy approach to construction project risk assessment and analysis: construction project risk management system. *Advances in Engineering Software, 32*(10–11), 847-857.

Chakravarti, A K, Vasanta, B, Krishnana, A S A, & Dubash, R K. (1998). *Modified Delphi methodology for technology forecasting: case study of electronics and information technology in India* (Vol.58): Technol. Forecast. Soc. Chang.

Chan, Hing Kai, & Wang, Xiaojun. (2013). Fuzzy Hierarchical Model for Risk Assessment. *London: Springer London., 10*, 978-971.

Chang, Tsung-Han, Hsu, Shu-Chen, & Wang, Tien-Chin. (2013). A proposed model for measuring the aggregative risk degree of implementing an RFID digital campus system with the consistent fuzzy preference relations. *Applied Mathematical Modelling, 37*(5), 2605-2622.

Charmaz, Kathy. (2007). *Constructing Grounded Theory* In D. S. G. College) (Eds.) (Ed.) *A Practical Guide Through Qualitative Analysis*

Chiam, Yin Kia, Staples, Mark, Ye, Xin, & Zhu, Liming. (2013). Applying a selection method to choose Quality Attribute Techniques. *Information and Software Technology, 55*(8), 1419-1436.

Choetkiertikul, M., & Sunetnanta, T. (2010). *A Risk Assessment Model for Offshoring Using CMMI Quantitative Approach.* Paper presented at the 2010 Fifth International Conference on Software Engineering Advances (ICSEA),.

Choetkiertikul, Morakot, & Sunetnanta, Thanwadee. (2012). A Risk Assessment Tool Using a CMMI Quantitative Approach. *IACSIT Int. J. Eng. Technol, 4*(4), 352-353.

Chowdhury, Abdullah Al Murad, & Arefeen, Shamsul. (2011). Software Risk Management: Importance and Practices. *IJCIT, ISSN*, 2078-5828.

Cole, Richard Stephen. (1996). *A Generic Software Library for Creating Multimedia Browse/Search Applications (Doctoral dissertation).*

Contini, Sergio, Bellezza, Furio, Christou, Michalis D., & Kirchsteiger, Christian. (2000). The use of geographic information systems in major accident risk assessment and management. *Journal of Hazardous Materials,78*(1–3), 223-245.

Corbin, J.M., & Strauss, A.L. (2008). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*: Sage Publications, Incorporated.

Corbin, Juliet M, & Strauss, Anselm. (1990). Grounded theory research: Procedures, canons, and evaluative criteria. *Qualitative sociology, 13*(1), 3-21.

Creswell, John W. (2003). Research design: Qualitative, Quantitative and Mixed Methods Approaches, 2nd edition: Sage Publications, Inc.

Dadhich, Reena, & Chauhan, Ujala. (2012). Integrating CMMI Maturity Level-3 In Traditional Software Development Process. *International Journal, 3*.

Dağdeviren, Metin, & Yüksel, İhsan. (2008). Developing a fuzzy analytic hierarchy process (AHP) model for behavior-based safety management. *Information Sciences, 178*(6), 1717-1733.

Dajani, Jarir S, Sincoff, Michael Z, & Talley, Wayne K. (1979). Stability and agreement criteria for the termination of Delphi studies. *Technological Forecasting and Social Change, 13*(1), 83-90.

Dalkey, Norman, & Helmer, Olaf. (1963). An experimental application of the Delphi method to the use of experts. *Management science, 9*(3), 458-467.

Davis, Fred D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, 319-340.

Day, Jacqueline, & Bobeva, Milena. (2005). A generic toolkit for the successful management of Delphi studies. *EJBRM, 3*(2), 103-116.

de Barros Sampaio, S. C., Barros, E. A., de Aquino, G. S., e Silva, M. J. C., & de Lemos Meira, S. R. (2010). *A Review of Productivity Factors and Strategies on Software Development.* Paper presented at the 2010 Fifth International Conference on Software Engineering Advances (ICSEA).

Delbecq, Andre L, Van de Ven, Andrew H, & Gustafson, David H. (1975). *Group techniques for program planning: A guide to nominal group and Delphi processes*: Scott, Foresman Glenview, IL.

Den Ouden, Elke. (2006). Development of a design analysis model for consumer complaints. *Technische Universiteit Eindhoven, Beta*.

Dey, Anind K, & Häkkilä, Jonna. (2008). Context-awareness and mobile devices. *User interface design and evaluation for mobile technology, 1*, 205-217.

Dromey, R. Geoff. (1995). A model for software product quality. *Software Engineering, IEEE Transactions on, 21*(2), 146-162.

Dubey, S.K., & Rana, A. (2010). Analytical Roadmap to Usability Definitions and Decompositions. *International Journal of Engineering Science and Technology, 2*(9), 4723-4729.

Dubey, Sanjay Kumar, Gulati, Anubha, & Rana, Ajay. (2012). Integrated Model for Software Usability. *International Journal on Computer Science and Engineering, 4*(3).

Dunn, William N. (2004). *Public policy analysis: An introduction* (3rd ed.). New Jersey: Pearson Prentice Hall.

Durrani, Qaiser Shehryar, & Qureshi, Samia Asloob. (2012). *Usability engineering practices in SDLC.* Paper presented at the Proceedings of the 2012 International Conference on Communications and Information Technology (ICCT).

English, J. Morley, & Kernan, Gerard L. (1976). The prediction of air travel and aircraft technology to the year 2000 using the Delphi method. *Transportation Research, 10*(1), 1-8.

Ermedahl, Andreas, & Engblom, Jakob. (2007). Execution time analysis for embedded real-time systems. *Handbook of Real-Time Embedded Systems, SHS Insup Lee, Jospeh YT. Leung, Ed. Chapman & Hall/CRC-Taylor and Francis Group*, 35.31-35.17.

Fang, Hsai-Yang, Mikroudis, George K., & Pamukcu, Sibel. (1987). A unified approach to the assessment of waste disposal sites. *Computers and Geotechnics, 3*(2–3), 129-156.

Fang, Ying Lan, Han, Bing, & Zhou, Wei Fu. (2013). Research and Design of Software Quality Control System Based on the CMMI System. *Applied Mechanics and Materials, 241*, 2837-2840.

Farooq, Sheikh Umar, & Quadri, SMK. (2012). Quality Practices in Open Source Software Development Affecting Quality Dimensions. *Trends in Information Management, 7*(2).

Feather, M. S., Cornford, S. L., Hicks, K. A., Kiper, J. D., & Menzies, T. (2008). A Broad, Quantitative Model for Making Early Requirements Decisions. *Software, IEEE, 25*(2), 49-56.

Felderer, Michael, Haisjackl, Christian, Breu, Ruth, & Motz, Johannes. (2012). Integrating manual and automatic risk assessment for risk-based testing *Software Quality. Process Automation in Software Development* 159-180: Springer.

Feng, Nan, & Li, Minqiang. (2011). An information systems security risk assessment model under uncertain environment. *Applied Soft Computing*, *11*(7), 4332-4340.

Ferre, X. (2003). *Integration of usability techniques into the software development process.* Paper presented at the International Conference on Software Engineering (Bridging the gaps between software engineering and human-computer interaction).

Fischer, H. (2012). *Integrating usability engineering in the software development lifecycle based on international standards.* Paper presented at the Proceedings of the 4th ACM SIGCHI symposium on Engineering interactive computing systems.

Folkerd, Christopher, & Spinelli, Gabriella. (2009). User exclusion and fragmented requirements capture in publicly-funded IS projects. *Transforming Government: People, Process and Policy, 3*(1), 32-49.

Fouladgar, MM, Yazdani-Chamzini, A, & Zavadskas, EK. (2012). Risk evaluation of tunneling projects. *Archives of civil and mechanical engineering*, *12*(1), 1-12.

Gardner, Christina, & Amoroso, Donald L. (2004). *Development of an instrument to measure the acceptance of internet technology by consumers.* Paper presented at the Proceedings of the 37th Annual Hawaii International Conference on System Sciences.

George, Darren, & Mallery, P. (2003). *SPSS for Windows Step by Step: A Simple Guide and Reference, 11.0 Update*: Allyn and Bacon.

Gibson, L. J. , & Miller, M. (1990). A Delphi model for planning `preemptive' regional economic diversification. *Economic Development Review, 8(2),* 4-41.

Glaser, Barney. (1967). The Discovery Of Grounded Theory: Strategies For Qualitative Research, Publisher: Al.

Glaser, Barney. (1998). *Doing Grounded Theory: Issues & Discussion*: Sociology Press.

Glaser, Barney, & Strauss, Anselm. (1992). Basics of Grounded Theory Analysis: Emergence Vs. Forcing: Sociology Press.

Grady, Robert B, & Caswell, Deborah L. (1987). Software metrics: establishing a company-wide program.

Grossman, Tovi, Fitzmaurice, George, & Attar, Ramtin. (2009). *A survey of software learnability: metrics, methodologies and guidelines.* Paper presented at the Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.

Gutierrez, Oscar. (1989). Experimental techniques for information requirements analysis. *Information & Management, 16*(1), 31-43.

Hall, Elaine M. (1998). *Managing risk: Methods for software systems development*: Pearson Education.

Han, Wen-Ming, & Huang, Sun-Jen. (2007). An empirical analysis of risk components and performance on software projects. *Journal of Systems and Software, 80*(1), 42-50.

Hanafin, Sinéad. (2004). Review of literature on the Delphi Technique. *Dublin: National Children's Office*.

Hashimi, H., Hafez, A., & Beraka, M. (2012). *A Novel View of Risk Management in Software Development Life Cycle.* Paper presented at the 12th International Symposium on Pervasive Systems, Algorithms and Networks (ISPAN).

Hasson, Felicity, Keeney, Sinead, & McKenna, Hugh. (2000). Research guidelines for the Delphi survey technique. *Journal of advanced nursing, 32*(4), 1008-1015.

Heck, Petra, Klabbers, Martijn, & van Eekelen, Marko. (2010). A software product certification model. *Software Quality Journal, 18*(1), 37-55.

Hedrick, Terry E, Bickman, Leonard, & Rog, Debra J. (1993). *Applied research design: A practical guide*: Sage.

Heiskari, J., Kauppinen, M., Runonen, M., & Mannisto, T. (2009). *Bridging the gap between usability and requirements engineering.* Paper presented at the 17th IEEE International Requirements Engineering Conference,RE'09.

Heras Saizarbitoria, Iñaki. (2006). How quality management models influence company results–conclusions of an empirical study based on the Delphi method. *Total Quality Management & Business Excellence, 17*(6), 775-794.

Hertzum, M. (2010). Images of Usability. *International Journal of Human-Computer Interaction, 26*(6), 567-600.

Hijazi, Haneen, Alqrainy, Shihadeh, Muaidi, Hasan, & Khdour, Thair. (2014). Identifying Causality Relation between Software Projects Risk Factors. *International Journal of Software Engineering and Its Applications, 8*(2), 51-58.

Hong, Zhen, & Lee, C. K. M. (2013). A decision support system for procurement risk management in the presence of spot market. *Decision Support Systems, 55*(1), 67-78.

Hoodat, Hooman, & Rashidi, Hassan. (2009). Classification and analysis of risks in software engineering. *World Academy of Science, Engineering and Technology, 56*, 446-452.

Houmb, Siv Hilde, Franqueira, Virginia NL, & Engum, Erlend A. (2010). Quantifying security risk level from CVSS estimates of frequency and impact. *Journal of Systems and Software, 83*(9), 1622-1634.

Hsu, Chia-Chien, & Sandford, Brian A. (2007). The Delphi technique: making sense of consensus. *Practical Assessment, Research & Evaluation, 12*(10), 1-8.

Hua, Lei, & Gong, Yang. (2013). Usability Evaluation of a Voluntary Patient Safety Reporting System: Understanding the Difference between Predicted and Observed Time Values by Retrospective Think-Aloud Protocols *Human-Computer Interaction. Applications and Services* , 94-100: Springer.

Humayoun, ShahRukh. (2012). Incorporating Usability Evaluation in Software Development Environments. *KI - Künstliche Intelligenz, 26*(2), 197-200.

Husnain, Muhammad, Waseem, Muhammad, & Ghayyur, S. (2009). An Interrogative Review of Requirement Engineering Frameworks. *International Journalof Reviews in Computing (IJRIC).*

Institute, Project Management. (2013). *A Guide to the Project Management Body of Knowledge (PMBOK® Guide)*: Project Management Institute, Incorporated.

Inversini, A., Cantoni, L., & Bolchini, D. (2011) Connecting usages with usability analysis through the user experience risk assessment model: A case study in the tourism domain. *Vol. 6770 LNCS*, 283-293.

Iranmanesh, Seyed Hossein, Khodadadi, Seyed Behrouz, & Taheri, S. (2009). *Risk Assessment of Software Projects Using Fuzzy Inference System.* Paper presented at the CIE 2009. International Conference on Computers & Industrial Engineering, 2009.

Isa, W.A.R.W.M., Suhami, M.R., Safie, N.I., & Semsudin, S.S. (2011). Assessing the Usability and Accessibility of Malaysia E-Government Website. *American Journal of Economics and Business Administration*, *3*(1), 40-46.

Islam, Shareeful. (2009). *Software development risk management model: a goal driven approach.* Paper presented at the Proceedings of the doctoral symposium for ESEC/FSE on Doctoral symposium.

ISO12207. (2008). ISO/IEC 12207: 2008 Systems and software engineering-Software life cycle processes. *International Organization for Standardization: Geneva*.

ISO. (2008). IEC 12207 Systems and software engineering-software life cycle processes. *International Organization for Standardization, Geneva, Switzerland*.

ISO/IEC9126. (2001). ISO/IEC 9126 -1: Software engineering – Product quality – Part 1: Quality model *9126*.

ISO/IEC25010. (2011). ISO/IEC 25010: 2011: Systems and software engineering– Systems and software Quality Requirements and Evaluation (SQuaRE)– System and software quality models. *International Organization for Standardization*.

Jairath, N, & Weinstein, J. (1993). The Delphi methodology (Part one): A useful administrative approach. *Canadian journal of nursing administration, 7*(3), 29-42.

Jiamthubthugsin, W., & Sutivong, D. (2006). *Resource Decisions in Software Development Using Risk Assessment Model.* Paper presented at the Proceedings of the 39th Annual Hawaii International Conference on System Sciences, HICSS '06.

Jianyi, Guo, Li, Zhang, Xusheng, Liu, Yuejuan, Huang, & Zhengtao, Yu. (2008). *Implementing a quantitative-based methodology for project risk assessment DSS.* Paper presented at the 27th Chinese Control Conference, 2008.

Jin, Beomsuk, Ko, Sangmin, Mun, Jaeseung, & Ji, YongGu. (2007). A Study for Usability Risk Level in Physical User Interface of Mobile Phone. In Nuray Aykin (Ed.), *Usability and Internationalization. HCI and Culture* (Vol. 4559, 327-335): Springer Berlin Heidelberg.

Jing, Xie, & Nan, Feng. (2009). *Risk evaluation process modeling in software project investment based on Bayesian networks.* Paper presented at the 16th International Conference on Industrial Engineering and Engineering Management, IE&EM '09. .

Johnson, R Burke, & Onwuegbuzie, Anthony J. (2004). Mixed methods research: A research paradigm whose time has come. *Educational researcher, 33*(7), 14-26.

Jokela, Timo. (2005). Performance rather than capability problems. insights from assessments of usability engineering processes *Product Focused Software Process Improvement* (115-127): Springer.

Jones, Capers, & Bonsignour, Olivier. (2011). *The economics of software quality*: Addison-Wesley Professional.

Jones, T. Capers. (1994). *Assessment and Control of Software Risk*. New York: Prentice-Hall.

Josang, A., Alfayyadh, B., Grandison, T., Alzomai, M., & McNamara, J. (2007). *Security Usability Principles for Vulnerability Analysis and Risk Assessment.* Paper presented at the Twenty-Third Annual Computer Security Applications Conference, ACSAC 2007. .

Joseph F. Hair, Jr., Anderson, Rolph E., Tatham, Ronald L., & Black, William C. (2005).*Multivariate data analysis (6th ed.): with readings*: Prentice-Hall, Inc.

Judd, Charles M, Smith, Eliot R, & Kidder, Louise H. (1991). Research methods in social relations.

Jurnecka, P, Hanacek, P, Barabas, M, Henzl, M, & Kacic, M. (2013). *A method for parallel software refactoring for safety standards compliance.* Paper presented at the 8th IET International System Safety Conference incorporating the Cyber Security Conference.

Jyothi, Veerapaneni Esther, Srikanth, Kaitepalli, & Rao, K Nageswara. (2012). Effective Implementation on Agile Practices-Object Oriented Metrics Tool to improve Software Quality. *International Journal of Software Engineering & Applications, 3*(4).

Kalaian, Sema A, & Kasim, Rafa M. (2012). Terminating Sequential Delphi Survey Data Collection. *Practical Assessment, Research & Evaluation, 17*(5), 2.

Kanis, H. (1997). *Usability centered research for everyday product design.* Paper presented at the Proceedings of the 13th Triennial Congress of the International Ergonomics Association, Finnish Institute for Occupational Health, Helsinki, Tampere Finland.

Kasunic, Mark. (2005). Designing an effective survey: DTIC Document.

Kaur, Kirandeep, Jyoti, Bharti, & Rani, Rekha. (2013). Analysis of Gold Plating: A Software Development Risk. *Analysis, 2*(1).

Kaur, Rupinder, & Sengupta, Jyotsna. (2013). Software Process Models and Analysis on Failure of Software Development Projects. *arXiv preprint arXiv:1306.1068*.

Keeney, Sinead, Hasson, Felicity, & McKenna, Hugh P. (2001). A critical review of the Delphi technique as a research methodology for nursing. *International journal of nursing studies, 38*(2), 195-200.

Keil, Mark, Cule, Paul E, Lyytinen, Kalle, & Schmidt, Roy C. (1998). A framework for identifying software project risks. *Communications of the ACM, 41*(11), 76-83.

Kessner, Martin, Wood, Jo, Dillon, Richard F, & West, Robert L. (2001). *On the reliability of usability testing.* Paper presented at the CHI'01 extended abstracts on Human factors in computing systems.

Ketola, P. (2002). *Integrating usability with concurrent engineering in mobile phone development*: University of Tampere.

Khan, Khalid S, Ter Riet, Gerben, Glanville, Julie, Sowden, Amanda J, & Kleijnen, Jos. (2001). *Undertaking systematic reviews of research on effectiveness: CRD's guidance for carrying out or commissioning reviews*: NHS Centre for Reviews and Dissemination.

Kitchenham, B. (2007). Guidelines for performing systematic literature reviews in software engineering. School of Computer Science and Mathematics, Keele University and Department of Computer Science, University of Durham.

Kitchenham, B, & Charters, S. (2007). Procedures for Performing Systematic Literature Reviews in Software Engineering. *Keele University & Durham University, UK*.

Kitchenham, Barbara, & Pfleeger, Shari Lawrence. (1996). Software quality: the elusive target [special issues section]. *Software, IEEE, 13*(1), 12-21.

Kitchenham, Barbara, Pickard, Lesley, & Pfleeger, Shari Lawrence. (1995). Case studies for method and tool evaluation. *IEEE software, 12*(4), 52-62.

Koh, SCL, Saad, SM, Ahmed, Ammar, Kayis, Berman, & Amornsawadwatana, Sataporn. (2007). A review of techniques for risk management in projects. *Benchmarking: An International Journal, 14*(1), 22-36.

Kontio, Jyrki. (2001). *Software engineering risk management: a method, improvement framework, and empirical evaluation.* (PhD), Helsinki University of Technology, Finland.

Kothari, Mihir. (2010). *Early warning signs in Software Projects.* AUT University.

Kristensson, David. (2004). *Quality assurance in software development & evaluation of existing quality systems.* University West.

Lai, Lotto Kim Hung, & Chin, Kwai Sang. (2014). Development of a Failure Mode and Effects Analysis Based Risk Assessment Tool for Information Security. *Industrial Engineeering & Management Systems, 13*(1), 87-100.

Laitenberger, Oliver, & Dreyer, Horst M. (1998). *Evaluating the usefulness and the ease of use of a web-based inspection data collection tool.* Paper presented at the Fifth International Proceedingson Software Metrics Symposium.

Lallemand, Carine. (2011). *Toward a closer integration of usability in software development: a study of usability inputs in a model-driven engineering process.* Paper presented at the Proceedings of the 3rd ACM SIGCHI symposium on Engineering interactive computing systems.

Lavery, Darryn and Cockton, Gilbert (1997). *Representing Predicted and Actual Usability Problems.* Paper presented at the Proceedings of International Workshop on Representations in Interactive Software Development QMW College, London

Lewis, Clayton, & Norman, Donald A. (1995). *Designing for error*. Paper presented at the Human-computer interaction.

Lewis, James R, & Sauro, Jeff. (2009). The factor structure of the system usability scale *Human Centered Design* (94-103): Springer.

Liang, Lu, Deng, Xiaoling, & Wang, Yong. (2009). *Usability Measurement Using a Fuzzy Simulation Approach.* Paper presented at the International Conference on Computer Modeling and Simulation, ICCMS'09.

Lilja, Kari K, Laakso, Kimmo, & Palomki, J. (2011). *Using the Delphi method.* Paper presented at the 2011 Proceedings of PICMET'11:Technology Management in the Energy Smart World (PICMET).

Lindgaard, G. (1994). *Usability testing and system evaluation: A guide for designing useful computer systems*: Chapman & Hall London,, UK.

Linstone, Harold A, & Turoff, Murray. (1975). *The delphi method*: Addison-Wesley Reading, MA.

Liu, Jun, Qiao, Jianzhong, & Lin, Shukuan. (2009). Risk Prediction and Measurement for Software Based on Service Oriented Architecture *Cutting-Edge Research Topics on Multiple Criteria Decision Making*, 515-522: Springer.

Liu, Yanqiu, Zhou, Chi, & Zhang, Ying. (2012). *Coordinate Preference DEA Method of Software Project Risk Assessment.* Paper presented at the International Conference on Communication Systems and Network Technologies (CSNT).

Lobato, L Lopes, do Carmo Machado, Ivan, Neto, PA da Mota Silveira, de Almeida, E Santana, & de Lemos Meira, S Romero. (2012). Risk Management in software engineering: A scoping study.

López, Cristina, & Salmeron, Jose L. (2012). Risks Response Strategies for Supporting Practitioners Decision-Making in Software Projects. *Procedia Technology, 5*(0), 437-444.

Loutchkina, I., Jain, L. C., Nguyen, T., & Nesterov, S. (2013). Systems' Integration Technical Risks' Assessment Model (SITRAM). *IEEE Transactions on Systems, Man, and Cybernetics: Systems, PP*(99), 1-1.

Lu, Haiyi, Axe, Lisa, & Tyson, Trevor A. (2003). Development and application of computer simulation tools for ecological risk assessment. *Environmental Modeling & Assessment, 8*(4), 311-322.

Lu, Shaokai, & Yan, Hong. (2013). A comparative study of the measurements of perceived risk among contractors in China. *International Journal of Project Management, 31*(2), 307-312.

Lummus, Rhonda R, Vokurka, Robert J, & Duclos, Leslie K. (2005). Delphi study on supply chain flexibility. *International Journal of Production Research, 43*(13), 2687-2708.

Makrodimitris, Georgios, Polemi, Nineta, & Douligeris, Christos. (2014). Security Risk Assessment Challenges in Port Information Technology Systems *E-Democracy, Security, Privacy and Trust in a Digital World* (24-35): Springer.

MAMPU, Modernization and Management Planning Unit. (2011). Kajian Baseline Perkhidmatan E-Kaunter & Paperless Government.

Markmann, Christoph, Darkow, Inga-Lena, & von der Gracht, Heiko. (2013). A Delphi-based risk analysis — Identifying and assessing future challenges for supply chain security in a multi-stakeholder environment. *Technological Forecasting and Social Change, 80*(9), 1815-1833.

Massacci, Fabio, Bouquet, Fabrice, Fourneret, Elizabeta, Jurjens, Jan, Lund, Mass S, Madelénat, Sébastien, . . . Piessens, Frank. (2011). Orchestrating security and system engineering for evolving systems *Towards a Service-Based Internet* (134-143): Springer.

McCall, Jim A, Richards, Paul K, & Walters, Gene F. (1977). *Factors in software quality*: General Electric, National Technical Information Service.

McFarlan, F. W. (1981). Portfolio Approach to Information System. In Harvard Business Review (Ed.), (Vol. 59(3), 142-150).

McLeod, Laurie, & MacDonell, Stephen G. (2011). Factors that affect software systems development project outcomes: A survey of research. *ACM Comput. Surv., 43*(4), 1-56.

Minglu, Li, Jianping, Li, Hao, Song, & Dengsheng, Wu. (2009). *Risk Management in the Trustworthy Software Process: A Novel Risk and Trustworthiness Measurement Model Framework.* Paper presented at the Fifth International Joint Conference on INC, IMS and IDC, NCM '09.

Misra, Subhas C., Kumar, Vinod, & Kumar, Uma. (2006). *Different techniques for risk management in software engineering: a review*. Paper presented at the ASAC 2006, Alberta.

Mkpong-Ruffin, Idongesit, Umphress, David, Hamilton, John, & Gilbert, Juan. (2007). *Quantitative software security risk assessment model.* Paper presented at the Proceedings of the 2007 ACM workshop on Quality of protection.

Mofleh, H.M., & Zahary, A. (2011). A Framework for Software Product Risk Management Based on Quality Attributes and Operational Lefe Cycle (SPRMQ). *Risk, 1*, 3.

Mohamud Sharif, Abdullahi, & Basri, Shuib. (2011). Software Risk Assessment: A Review on Small and Medium Software Projects

Software Engineering and Computer Systems. In Jasni Mohamad Zain, Wan Maseri bt Wan Mohd, & Eyas El-Qawasmeh (Eds.), (Vol. 180, 214-224): Springer Berlin Heidelberg.

Mohapatra, Sanjay. (2012). *Information Theory and Best Practices in the IT Industry*: Springer.

Mokhtari, Kambiz, Ren, Jun, Roberts, Charles, & Wang, Jin. (2012). Decision support framework for risk management on sea ports and terminals using fuzzy set theory and evidential reasoning approach. *Expert Systems with Applications, 39*(5), 5087-5103.

Mustafa, Norfazira, Sanusi, Nurazlina Md, & Jalil, Intan Ermahani A. (2010). *Software risk assessment visualization tool design using probabilistic inference models.* Paper presented at the 2010 International Symposium in Information Technology (ITSim).

Naik, Radhika. (2013). Software Risk Management. *International Journal of Advances in Engineering Sciences, 3*(1), 17-21.

Ngai, E. W. T., & Wat, F. K. T. (2005). Fuzzy decision support system for risk analysis in e-commerce development. *Decision Support Systems, 40*(2), 235-255.

Nielsen, Jakob. (1994). *Usability engineering*: Access Online via Elsevier.

Nogueira, J. C., Luqi, & Bhattacharya, S. (2000). *A risk assessment model for software prototyping projects.* Paper presented at the 11th International Workshop on Rapid System Prototyping, RSP 2000.

Nunamaker, J. F., Jr., & Chen, M. (1990). *Systems development in information systems research*. Paper presented at the Proceedings of the Twenty-Third Annual Hawaii International Conference on System Sciences.

O'Regan, Gerard. (2014). Software Process Improvement *Introduction to Software Quality* (199-209): Springer.

Okoli, Chitu, & Pawlowski, Suzanne D. (2004). The Delphi method as a research tool: an example, design considerations and applications. *Information & Management, 42*(1), 15-29.

Okonta, OE, Ojugo, AA, Wemembu, UR, & Ajani, D. (2013). Embedding Quality Function Deployment In Software Development: A Novel Approach. *West African Journal of Industrial and Academic Research, 6*(1), 50-64.

Onwuegbuzie, Anthony J, Leech, Nancy L, & Collins, Kathleen MT. (2012). Qualitative analysis techniques for the review of the literature. *The qualitative report, 17*(28), 1-28.

Organization, International Standardization. (1999). ISO 13407 : Human-centred design processes for interactive systems.

Otte, Tobias, Moreton, Robert, & Knoell, Heinz D. (2008). *Applied quality assurance methods under the open source development model.* Paper presented at the 32nd Annual IEEE International Computer Software and Applications, COMPSAC'08. .

P. Bourque and R.E. Fairley, eds. (2014). Guide to the Software Engineering Body of Knowledge. *IEEE Computer Society, Version 3.0*.

Pare, Guy, Sicotte, Claude, Jaana, Mirou, & Girouard, David. (2008). *Prioritizing clinical information system project risk factors: a Delphi study*. Paper presented at the Proceedings of the 41st Annual Hawaii International Conference on System Sciences.

Patel, Sneha D, & Pattewar, Tareek M. (2014). Automatic Detection of Unsafe Dynamic Component Loadings: A Survey.

Peansupap, Vachara, & Walker, Derek HT. (2006). Information communication technology (ICT) implementation constraints: a construction industry

perspective. *Engineering, construction and architectural management, 13*(4), 364-379.

Pfeiffer, John. (1968). Decision Making in Action. Chapter 2, New Look at Education: Systems Analysis in our Schools and Colleges.

Phyllis Tharenou, Ross Donohue, Brian Cooper. (2007). *Management Research Methods*: Cambridge University Press.

Platt, A. B. (1999). *The usability risk.* Paper presented at the Proceedings of the 18th IEEE Symposium on Reliable Distributed Systems.

Pourdarab, Sanaz, Nosratabadi, Hamid Eslami, & Nadali, Ahmad. (2011). Risk Assessment of Information Technology Projects Using Fuzzy Expert System *Digital Information and Communication Technology and Its Applications* 563-576: Springer.

Powell, Catherine. (2003). The Delphi technique: myths and realities. *Journal of advanced nursing, 41*(4), 376-382.

Qu, Xiaobo, Meng, Qiang, Yuanita, Vivi, & Wong, Yoke Heng. (2011). Design and implementation of a quantitative risk assessment software tool for Singapore road tunnels. *Expert Systems with Applications, 38*(11), 13827-13834.

Quesenbery, Whitney. (2004). Balancing the 5Es of Usability. *Cutter IT Journal, 17*(2), 4-11.

Radatz, J., Geraci, A., & Katki, F. (1990). "IEEE standard glossary of software engineering terminology *IEEE Std 1540-2001* (Vol. 610121990, 121990).

Radulovic, Filip, García-Castro, Raúl, & Gómez-Pérez, Asunción. (2015). SemQuaRE — An extension of the SQuaRE quality model for the evaluation of semantic technologies. *Computer Standards & Interfaces, 38*(0), 101-112.

Rennie, D. L., Phillips, J. R., & Quartaro, G. K. (1988). Grounded Theory: A Promising Approach to Conceptualization in Psychology? *Canadian Psychology/Psychologie Canadienne, 29*(2).

Rennie, David. (2006). The Grounded Theory Method: Application of a Variant of its Procedure of Constant Comparative Analysis to Psychotherapy Research. In In C. T. Fischer (Ed.) (Ed.), *Qualitative research: Instructional empirical studies* New York: NY: Elsevier.

Rowe, Gene, & Wright, George. (1999). The Delphi technique as a forecasting tool: issues and analysis. *International journal of forecasting, 15*(4), 353-375.

Sadiq, Mohd, & Shahid, Mohd. (2013). A systematic approach for the estimation of software risk and cost using esrcTool. *CSI transactions on ICT, 1*(3), 243-252.

Sangeetha, M, Arumugam, C, Sapna, PG, & Senthil Kumar, KM. (2011). A cooperative approach to ensure software product quality. *International Journal of Computer Science Issues (IJCSI), 8*(4).

Sauro, Jeff. (2010). How common are usability problems?, from http://www.measuringusability.com/problem-frequency.php

Schmidt, Roy, Lyytinen, Kalle, Keil, Mark, & Cule, Paul. (2001). Identifying software project risks: an international Delphi study. *Journal of management information systems, 17*(4), 5-36.

Schreiber, Rita Sara, & Stern, Phyllis Noerager. (2001). *Using grounded theory in nursing*: Springer Publishing Company.

Schwaber, Ken, & Sutherland, Jeff. (2012). *Software in 30 Days: How Agile Managers Beat the Odds, Delight Their Customers, and Leave Competitors in the Dust*: Wiley. com.

Seffah, A., Djouab, R., & Antunes, H. (2001). *Comparing and reconciling usability-centered and use case-driven requirements engineering processes.* Paper presented at the Australian Computer Science Communications.

Seffah, Ahmed, Donyaee, Mohammad, Kline, Rex B, & Padda, Harkirat K. (2006). Usability measurement and metrics: A consolidated model. *Software Quality Journal, 14*(2), 159-178.

Seffah, Ahmed, & Metzker, Eduard. (2004). The obstacles and myths of usability and software engineering. *Communications of the ACM, 47*(12), 71-76.

Shackel, B. (1986). *Ergonomics in design for usability*. Paper presented at the Proceedings of the Second Conference of the British Computer Society, human computer interaction specialist group on People and computers: designing for usability, York, United Kingdom.

Shahriar, Hossain, & Haddad, Hisham. (2014). *Risk assessment of code injection vulnerabilities using fuzzy logic-based system.* Paper presented at the Proceedings of the 29th Annual ACM Symposium on Applied Computing.

Sharif, Abdullahi Mohamud, & Basri, Shuib. (2011). Software risk assessment: a review on small and medium software projects *Software Engineering and Computer Systems* (214-224): Springer.

Sharif, Abdullahi Mohamud, Basri, Shuib, & Ali, Hassan Osman. (2014). Strength and Weakness of Software Risk Assessment Tools. *International Journal of Software Engineering & Its Applications, 8*(3).

Sharif, Abdullahi Mohamud, & Rozan, Mohd Zaidi Abd. (2010). Design and Implementation of Project Time Management Risk Assessment Tool for SME Projects using Oracle Application Express. *World Academy of Science, Engineering, and Technology (WASET), 65*, 1221-1226.

Sharma, Aman Kumar, Kalia, Arvind, & Singh, Hardeep. (2012). An Analysis of Optimum Software Quality Factors. *IOSR Journal of Engineering, 2*(4), 663-669.

Sherer, S. A. (1995). *The three dimensions of software risk: technical, organizational, and environmental.* Paper presented at the Proceedings of the Twenty-Eighth Hawaii International Conference on System Sciences.

Shneiderman, B. , & Plaisant, C. . (2010). *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (Fifth Edition ed. ed.). Reading: MA Addison-Wesley Publ. Co.

Simes-Marques, Mário, & Nunes, Isabel. (2012). Usability of interfaces. *Ergonomics-A Systems Approach*, 155-170.

Stevens, Jim, & Sprague, Alex. (2011). Risk Assessment in Software Development Projects.

Stoneburner, Gary, Goguen, Alice, & Feringa, Alexis. (2002). Risk management guide for information technology systems. *Nist special publication, 800*(30), 800-830.

Story, Victoria, Hurdley, Louise, Smith, Gareth, & Saker, James. (2000). Methodological and practical implications of the Delphi technique in marketing decision-making: A re-assessment. *The Marketing Review, 1*(4), 487-504.

Sulaman, Sardar Muhammad, Weyns, Kim, H, Martin, #246, & st. (2013). *A review of research on risk analysis methods for IT systems*. Paper presented at the

Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering, Porto de Galinhas, Brazil.

Tang, Ai-guo, & Wang, Ru-long. (2010). *Software project risk assessment model based on fuzzy theory.* Paper presented at the International Conference On Computer and Communication Technologies in Agriculture Engineering (CCTAE).

Tao, Ye. (2008). *A study of software development project risk management.* Paper presented at the FITME'08. International Seminar on Future Information Technology and Management Engineering, 2008.

Thangaratinam, Velayutham, & Selvambigai, Shakila. (2011). *Health technology assessment in maternal and perinatal medicine: delphi survey of practice, systematic reviews of evidence and meta analyses.* University of Birmingham.

Tharenou, Phyllis, Donohue, Ross, & Cooper, Brian. (2007). *Management research methods*: Cambridge University Press New York.

Tianyin, Pu. (2011). *Development of software project risk management model review.* Paper presented at the 2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC).

Uzzafer, M. (2011). *A Novel Risk Assessment Model for Software Projects*. Paper presented at the 2011 International Conference on Computer and Management (CAMAN).

Vairavamoorthy, K., Yan, Jimin, Galgale, Harshal M., & Gorantiwar, Sunil D. (2007). IRA-WDS: A GIS-based risk analysis tool for water distribution systems. *Environmental Modelling & Software, 22*(7), 951-965.

Van Manen, Sipke, Brandt, Ed, Van Ekris, Jaap, & Geurts, Wouter. (2013). TOPAAS: An Alternative Approach to Software Reliability Quantification. *Quality and Reliability Engineering International*.

Verner, June, Sampson, Jennifer, & Cerpa, Narciso. (2008). *What factors lead to software project failure?* Paper presented at the Second International Conference on Research Challenges in Information Science, RCIS 2008.

Vilbergsdottir, Sigurbjorg Groa, Hvannberg, Ebba Thora, & Law, Effie Lai-Chong. (2014). Assessing the reliability, validity and acceptance of a classification

scheme of usability problems (CUP). *Journal of Systems and Software, 87*, 18-37.

Vuolle, Maiju, Aula, Anne, Kulju, Minna, Vainio, Teija, & Wigelius, Heli. (2008). Identifying usability and productivity dimensions for measuring the success of mobile business services. *Advances in Human-Computer Interaction.*

Wang, Ping, Chao, Kuo-Ming, Lo, Chi-Chun, Huang, Chun-Lung, & Younas, Muhammad. (2007). A fuzzy outranking approach in risk analysis of web service security. *Cluster Computing, 10*(1), 47-55.

Wattanapokasin, W., & Rivepiboon, W. (2009, 15-17 May 2009). *Cross-Cultural Risk Assessment Model.* Paper presented at the 2009 International Conference on Signal Processing Systems.

Westfall, Linda. (2011). *Software Risk Management*. Paper presented at the International Conference on Software Quality, San Diego, California.

Wheeler, Evan. (2011). Chapter 5 - Formulating a Risk. In Evan Wheeler (Ed.), *Security Risk Management* (87-103). Boston: Syngress.

Williams, Patricia L, & Webb, Christine. (1994). The Delphi technique: a methodological discussion. *Journal of advanced nursing, 19*(1), 180-186.

Williams, Ray C, Pandelios, George J, & Behrens, Sandra G. (1999). *Software Risk Evaluation (SRE) Method Description: Version 2.0*. Pittsburgh, PA, USA.: Carnegie Mellon University, Software Engineering Institute.

Winograd, Terry A, & Flores, Carlos Fernando. (1986). *Understanding computers and cognition: A new foundation for design*: Intellect Books.

Woolridge, R. W., McManus, D. J., & Hale, J. E. (2007). Stakeholder Risk Assessment: An Outcome-Based Approach. *Software, IEEE, 24*(2), 36-45.

Wulan, Muqi, & Petrovic, Dobrila. (2012). A fuzzy logic based system for risk analysis and evaluation within enterprise collaborations. *Computers in Industry, 63*(8), 739-748.

Xenos, Michalis. (2001). *Usability perspective in software quality.* Paper presented at the Usability Engineering Workshop, The 8th Panhellenic Conference on Informatics with International Participation, Southern Cyprus.

Xie, Gang, Yue, Wuyi, Wang, Shouyang, & Lai, Kin Keung. (2010). Dynamic risk management in petroleum project investment based on a variable precision rough set model. *Technological Forecasting and Social Change, 77*(6), 891-901.

Yacoub, Sherif M, Ammar, Hany H, & Robinson, Tom. (2000). *A methodology for architectural-level risk assessment using dynamic metrics.* Paper presented at the Proceedings of 11th International Symposium on Software Reliability Engineering, ISSRE 2000.

Yan-qiu, Liu, Chi, Zhou, & Ying, Zhang. (2012, 11-13 May 2012). *Coordinate Preference DEA Method of Software Project Risk Assessment.* Paper presented at the 2012 International Conference on Communication Systems and Network Technologies (CSNT).

Yong, Hu, Jiaxing, Huang, Juhua, Chen, Mei, Liu, & Kang, Xie. (2007). *Software Project Risk Management Modeling with Neural Network and Support Vector Machine Approaches.* Paper presented at the Third International Conference on Natural Computation, ICNC 2007.

Zimmermann, H-J. (2010). Fuzzy set theory. *Wiley Interdisciplinary Reviews: Computational Statistics, 2*(3), 317-332.