AN ENHANCED RISK IDENTIFICATION AND ASSESSMENT MODEL TO IMPROVE SOFTWARE RISK MANAGEMENT

AHDIEH SADAT KHATAVAKHOTAN

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2017

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: **Ahdieh Sadat Khatavakhotan** (Passport No: [redacted])

Registration/Matric No: **WHA090009**

Name of Degree: **Doctor of Philosophy (PhD)**

Title of Project Paper/Research Report/Dissertation/Thesis (“this Work”):

An Enhanced Risk Identification and Assessment Model to Improve Software Risk Management

Field of Study: **Software Engineering**

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya (“UM”), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate’s Signature Date:

Subscribed and solemnly declared before,

Witness’s Signature Date:

Name:

Designation:

ABSTRACT

In software development, inability to define software requirements correctly, underestimating project cost and schedule often result in project failure. These causes are indeed among the risks that are often overlooked or underestimated and their negative impact should they occur. Although there are many risk identification and assessment (RI&RA) process models available today, these models have some weaknesses such as the inability to identify the potential risks and assessing their impact accurately. Hence, this research proposes an enhanced risk identification and assessment model, E-RIAM to address those weaknesses. E-RIAM incorporates five main enhancements that makes it able to: i) identify a maximum of 20 potential major and moderate risks in each software development phase; ii) identify a maximum of 20 potential common major and moderate risks in the entire project; iii) prepare a list of potential software risks of each development phase; iv) provide a risk database that stores the potential, most serious, and common software risks; and v) A Dynamic Verifier Core (DVC) team (i.e., a risk team with more than 20 years of experience in software risk management) to verify the list of risks that had been identified and assessed by the risk analysts. A support tool, Res-DVC, was also developed to facilitate the RI&A processes. To evaluate whether E-RIAM can improve the efficiency of the RI&A processes, two case studies were carried out on 40 medium-sized software projects to collect the data needed for the evaluation process. Two independent groups comprising one control group (i.e., Risk Team 1 and Risk Team 2 of the two case studies) and one treatment group (i.e., DVC1 and DVC2 of the two case studies) was used. Two hypotheses were formulated to evaluate E-RIAM. Hypothesis 1 tests the efficiency of the risk identification process, while hypothesis 2 tests the accuracy of the risk assessment process. Hypothesis 1 was tested using Wilcoxon Signed Ranks Test. The results of the test show that E-RIAM can affect significant improvement to the risk identification process. Two approaches were used to test hypothesis 2. The first approach compares the severity level (i.e., major, moderate, and minor) of the identified, and materialised risks which had been assessed by the risk teams against the severity level of the corresponding risks (i.e., data given by the software company). The total number of matching risks distributed according to the three severity levels were compiled and analysed. The outcomes show that the DVC teams were able to identify and assess more risks correctly when compared to the number of risks that were identified and assessed by the risk teams. The second approach compares and analyses the total number of risks that had materialised in both the case studies (i.e., data given by the software company), but failed to be identified by both the risk teams and DVC teams. The results show that the DVC teams were able to identify and assess more risks correctly than the risk teams.

ABSTRAK

Dalam pembangunan perisian, ketidakupayaan untuk menakrifkan keperluan perisian dengan betul, anggaran kos projek dan jadual yang di bawah anggaran, sering mengakibatkan kegagalan projek. Sebab-sebab ini sesungguhnya adalah di antara risiko yang selalu terlepas pandang atau impak negatifnya dianggar di bawah anggaran sekiranya mereka menjadi nyata. Walaupun kini terdapat banyak model proses pengenalian dan penilaian risiko (RI&A), model-model ini mengandungi beberapa kelemahan yang menyebabkan ketidakupayaan untuk mengenalpasti risiko-risiko potensi dan menilai impak mereka dengan tepat. Oleh yang demikian, penyelidikan ini bertujuan untuk mencadangkan satu model penambahbaikan pengenalian dan penilaian risiko (E-RIAM) bagi menangani kelemahan tersebut. Lima penambahbaikan utama telah diperkenalkan dalam E-RIAM. Ini termasuk i) mengenalpasti sejumlah maksima 20 risiko potensi yang utama dan sederhana bagi setiap fasa pembangunan perisian; ii) mengenalpasti sejumlah maksima 20 risiko potensi umum yang utama dan sederhana bagi keseluruhan projek; iii) menyediakan satu senarai risiko perisian yang berpotensi bagi setiap fasa pembangunan; iv) menyediakan satu pangkalan data risiko yang menyimpan risiko berpotensi, yang paling serius dan biasanya dijumpai dalam risiko perisian; dan v) memperkenalkan satu pasukan “Dynamic Verifier Core (DVC)“ (iaitu, satu pasukan risiko yang berpengalaman lebih daripada 20 tahun dalam pengurusan risiko perisian), bagi mengesahkan senarai risiko yang telah dikenalpasti dan dinilai oleh para penganalisis risiko. Satu alatan sokongan, Res-DVC, telah dibangunkan bagi memudahkan proses RI&A. Untuk menilai sama ada E-RIAM boleh meningkatkan kecekapan proses RI&A, dua kajian kes telah dijalankan terhadap 40 buah projek pembangunan perisian bersaiz sederhana bagi mengutip data yang diperlukan untuk proses penilaian. Rekabentuk penyelidikan berupa dua buah kumpulan tidakbersandaran yang terdiri daripada satu kumpulan kawalan (iaitu, Pasukan Risiko 1 dan Pasukan

Risiko 2 bagi kedua-dua kajian kes) dan satu kumpulan rawatan (iaitu, Pasukan DVC1 dan Pasukan DVC2 bagi kedua-dua kajian kes) telah digunakan. Dua hipotesis telah dirangka bagi menilai E-RIAM. Hipotesis 1 menguji kecekapan proses pengenalian risiko manakala hipotesis 2 menguji ketepatan proses penilaian risiko. Hipotesis 1 telah diuji dengan menggunakan Ujian Wilcoxon Signed Ranks. Keputusan bagi ujian tersebut menunjukkan bahawa terdapat peningkatan yang ketara dalam proses pengenalian risiko. Dua pendekatan telah digunakan untuk menguji hipotesis 2. Pendekatan pertama membandingkan tahap kekerasan (iaitu, utama, sederhana, dan kurang penting) bagi risiko-risiko yang telah dikenalpasti, menjadi nyata dan dinilai oleh pasukan-pasukan risiko dengan tahap kekerasan bagi risiko-risiko yang bersetandingan (iaitu, data yang diberi oleh syarikat perisian). Jumlah risiko-risiko yang padan, telah disusun dan dihurai berdasarkan agihan ketiga-tiga kategori tahap kekerasan. Keputusan menunjukkan bahawa pasukan DVC dapat mengenalpasti dan menilai lebih banyak risiko dengan betul berbanding dengan risiko yang dikenalpasti dan dinilai oleh pasukan risiko. Pendekatan kedua membanding dan menghuraikan jumlah risiko yang menjadi nyata dalam kedua-dua kajian kes (iaitu, data yang diberi oleh syarikat perisian), tetapi gagal dikenalpasti oleh kedua-dua pasukan risiko dan pasukan DVC. Keputusan menunjukkan bahawa pasukan DVC dapat mengenalpasti dan menilai lebih banyak risiko dengan betul berbanding dengan pasukan risiko.

ACKNOWLEDGEMENTS

First and above all, I praise **God, the Almighty** for helping me finish this crucial stage of my academic life. I have always passed the difficult stages of my personal and academic life by trust in Him and with the aid of His boundless mercy. I hope I can follow His command regarding acquisition of knowledge to serve mankind all through my life. I wish I am deserving of His benevolence.

During the years of working on this research, Associate Professor **Dr. Ow Siew Hock** directed me throughout the research path, both as the supervisor and an insightful and compassionate guide. Her meticulous nature and attentiveness on multiple revisions throughout the present thesis is admirable. She has my everlasting gratitude.

My sincere gratitude goes to Associate Professor **Dr. Noor Haroon Abdul Karim**, Head of Department of Library and Information Science who offered valuable suggestions on choice of appropriate statistical tests. I further wish to express appreciation to **Mr. K H Teh** who proofread the thesis patiently and professionally, thanks to his patience together with his technical knowledge regarding the correction of English sentences structure.

I would like to give my special appreciation to Pofessor **Dr. Abdullah Gani**, Dean of the Faculty of Computer Science & Information Technology (FCSIT), and **Dr. Siti Hafizah**, Head of Software Engineering Department. I also acknowledge the University of Malaya Postgraduate Research Grant (PPP) A/C Number PS027-2012A for the support of the research in its final stage.

Words alone cannot fully express my wholehearted gratitude to **my dear honorable father**, **Seyed Ali**, who has been inspiring me to study since my childhood. His liberal and modest attitude towards materialistic issues and his observance on the quality of the construction projects he has successfully completed over half a century, will be my guiding light on my path forever. I also wish to honor **my mother, Mahin**, who has always been my unshakable support and has encouraged and revitalized me through the hardships. Her sincere and boundless self-sacrifice for her children has enabled us to blossom and pursue our personal goals in life. I can never make up for even a small part of what they did for me. My further thanks go to **my brother, Mehdi**, who has always been a compassionate companion to me and has provided comfort and peace in the paternal house, in spite of the many personal responsibilities he has to shoulder, and help me finish my education abroad with much less concern.

I would like to express my profound gratitude to **my beloved wise husband, Dr. Navid**, who has been my supporter and guide throughout my postgraduate education. He has been holding the family together during long periods of my education and research, and has been a constant encouragement by reminding me of short-lived problems resulting in sweet results. I could not have achieved my goals without his valuable encouragement. He also provided me with the opportunity of attending several international conferences and seminars, for which I am especially grateful.

My children, Niki and Hafez, were born and grew up during my years of research effort. I spent many nights awake on researching in their warm presence and invigorating energy. I wish they will also follow the path of research to serve society, and hope they will forgive me for dedicating so much time to my research instead of spending it with them.

Last but not the least, I would like to express my deep gratitude to my mother-in-law, **Ashraf Zomorrodian**, whose kind residence abroad for some months gave me peace of mind as well as the chance to spend more time in the Research and Development centre.

I wish the results of this research will advance the corpus of knowledge in science, particularly in the qualitative enhancement of software projects which have been constantly facing challenges for decades. In this way, I may be able to make a little contribution to human society by publishing the acquired knowledge. “***Ahdieh Khotan”***

TABLE OF CONTENTS

LIST OF FIGURES

`

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

E-RIAM	:	Enhanced Risk Identification and Assessment Model
DVC	:	Dynamic Verifier Core
RT	:	Risk Team
RV	:	Risk Value
RC	:	Risk Coordinator
RIE	:	Risk Identification Efficiency
L	:	Likelihood of occurrence of a risk
I	:	Impact of a risk
Res-DVC	:	The support tool of E-RIAM
Req	:	Requirements Analysis phase
Des	:	Design Phase
PnT	:	Programming and Testing phase
Imp	:	Implementation and Release phase
EnP	:	Entire Project
TNRM	:	Total Number of Risks that had Materialised
TNRNM	:	Total Number of risks that had Not Materialised
SL	:	Severity Level (Major, Moderate, or Minor)

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

1.1 Research Background

In the past three decades, with the rapid advancement in information technology and its widespread application in every aspect of our life, it becomes increasingly important that all software projects must be successful to produce software that are of quality, reliable and robust (Parthasarathy & Sharma, 2016; Frey, 2014). Software project risk management is crucial in ensuring that these projects are completed successfully with little or no cost or time over-run (Jowah, 2015). Despite the availability of many risk management models and related tools, there is still no effective method of predicting and eliminating threats to software projects (Wanderley et al., 2015). As a result, a large percentage of software and IT projects failed. The Standish Group (2015) reported that a large percentage of completed software projects had deviated drastically, and resulted in costs and time over-runs.

There are four major phases in risk management – risk identification, risk assessment, design and implementation of risk response plans, and evolution of activities (PMI, 2015).

ISO 31000 is one of the most reputable and widely-used standards for project management (Purdy, 2010; Klipper, 2015). The first two phases of risk management - risk identification and risk assessment – are very important phases, and any risk response, mitigation and remedial plans rely on the success of these first two phases (Choetkiertikul et al., 2015). Hence, this research is focused on improving risk identification and risk assessment in software projects. These two phases of risk management involve human-ware activities, hence, the development of a proper structure for exploiting the competence and experience of risk management experts, is critical in these phases (Parthasarathy & Sharma, 2016; Beaver, & Schiavone, 2006). Past researchers have not proposed any model for designing suitable structures, and effective ways of exploiting the skills and experience of risk experts, within or outside of their organisations (Judith & Kate, 2007; Poth, 2014; Parthasarathy & Sharma, 2016).

1.2 Problem Statement

The main problem investigated in this research is the inefficiency of existing software projects risk identification and assessment models. This problem was recognized through studies conducted by reputable specialized groups (Standish Group, 2015; Lindholm, 2015). Project risk management will improve along with any enhancement to the risk identification procedures, especially in being able to identify the more important risks that could result in project failure (Justin, 2006). Improvement in project risk management can also lead to the production of better software systems and products because any improvement will positively affect performance, costs reduction (the cost price), and project completion schedule (Hoermann et al., 2012).

Many studies have also reported on inaccuracy in assessing important risks, which pertains to their likelihood of occurrence and the impact of materialised risks (Antinyan1 et al, 2014). By conducting risk assessment systematically, it is possible to focus on the important likely-to-occur risks and risks that can potentially result in disastrous consequences. Accurate assessment accelerates the risk assessment process and allows a software company to give more firm commitment as to the budget, completion, and delivery of the software, as defined in the software project plan (Wu et al., 2014; Basile et al., 2015).

1.3 Objectives of Research

Besides the many sub-goals and benefits stated in the research problems section above, the main objectives of this research are as follows:

i) To determine whether the identification of a maximum of 20 potential (moderate and major) risks is sufficient in each development phase of a software project;

ii) To determine whether the identification of a maximum of 20 common (moderate and major) risks is sufficient for the entire software project;

iii) To propose an Enhanced Risk Identification and Assessment Model (E-RIAM); and

iv) To evaluate whether E-RIAM can improve accuracy in the risk identification and risk assessment processes.

1.4 Research Question

The primary research objective is to improve identification and assessment of software project risks, and thus, to enhance the risk management of software projects. Various standards and models have been proposed for identifying and assessing software project risks, but many studies still report that the high failure rate of software projects is due to the inefficiency of these models in practice (Lindholm, 2015). Many standards have provided a general framework for the risk identification and the risk assessment processes. Other models provide recommendations for problem-solving, as well as improvement of the risk management process, but they only offer partial solution to the problem.

This research is aimed at proposing the structures, methods, and mechanisms for risk identification and risk assessment. It will emphasise the participation of internal risk analysts and external experts (recommended by reliable sources) in the processes, and provide answers to the following questions:

i) How should the framework and recommendations in the ISO 31000/31010:2009 standard be enhanced to improve risk identification in software projects?

ii) How should the framework and recommendations in the ISO 31000/31010:2009 standard be enhanced to improve risk assessment in software projects?

iii) How should the proposed enhanced model for risk identification and risk assessment be assessed?

The answers to these questions will be very useful to managers who have always strived to ensure that their projects achieve the predetermined performance level, and are completed on time and within the approved budget. These answers will also be useful to software developers who have always aimed at developing reliable software products, and delivering them on-time and within costs (Yahav, Kenett, & Bai, 2014). In addition, software project risk management will increase the quality of software products, and satisfy users’ requirements (Pozzebon et al, 2014).

1.5 Research Scope

ISO 31000 2009 Edition 5 (henceforth referred to as ISO 31000) is the latest official standard for project management. There are other standards such as DOD 2015 for project risk identification and risk assessment, which also share key features with ISO 31000. Many studies have been conducted in recent years on ways of improving the risk identification and risk assessment processes in software projects (Purdy, 2010). A majority of these studies had focused on using surveys to gather expert opinions on risks and then rank the risks using Analytical Hierarchical Process (AHP) or fuzzy techniques. However, these methods tend to be project-specific, and are often irrelevant to the actual projects where risk management is considered a seamless process. Moreover, those approaches require extensive disclosure of confidential corporate data on every project, which will give rise to security issues. Hence, these models and techniques have not been widely applied and they are mostly of academic research interest, rather than for actual industrial application (Reifer, & Boehm 2007; Boehm, 2007). That is reason for excluding such approaches from the scope of this study.

The enhancement applied to the risk identification and risk assessment processes of software projects involves several development phases, as specified in every project.

Relative improvement can be achieved in overall risk identification and risk assessment of projects by focusing on these two processes.

This research focuses only on the first two phases of the risk management - risk identification and risk assessment. This is because both phases are important in a software development lifecycle, and can affect the success of subsequent phases. This research concentrates on risk identification and risk assessment because risk management is too extensive to be covered completely in this thesis. Another major reason for focusing on the two phases is the seamless link between risk identification and risk assessment in terms of operation and success. This research also focuses on commercial software projects. The success of any risk management effort is dependent on the accurate prediction of candidate risks (Berghe et al. 2015; Khan & Khan, 2013). One advantage the proposed model has over the other models is the comprehensive classification of candidate risks in the various phases of commercial software projects. Although there is a wide range of commercial software, which are oriented towards customer interaction, any concentration only on such programs and failure to cover other aspects such as drivers, software systems or embedded software, could restrict the scope of this research project.

A five-member risk team was formed to validate the models used in previous case studies. Although the number of members in the risk team was based on the number reported for risk management in similar software projects, it could also be a limitation to the current research. If the number of team members varies, the research scope may expand or shrink. Thus, more detailed analysis of the results of the study can be made on this aspect. Another limitation to consider in future studies is the appointment of three external experts to the verifier team. One more limitation of this research is the dual case studies conducted in parallel during the two case studies, which involved a total of 40 projects. There are two reasons for this: i) the short or limited time allocated to this academic research, and ii) concerns over the privacy issues by project managers. In view of these limitations, different time limits were considered for the risk analysts and risk experts during the risk identification and risk assessment processes. In this way, there will be a consistent procedure for risk identification and risk assessment with respect to team control and treatment to ensure that valid analytical results are obtained. The above limitations define the project scope, but can be modified in further studies.

1.6 Methodology

Choosing an appropriate research methodology is crucial to achieve valid results in any academic research (Creswell, 2012). This research started with a review of relevant literature that includes academic books, journal papers, conference papers, standards, etc., published in recent years. The literature review, presented in Chapter 2 of this thesis, had provided useful information on the problems and weaknesses of the risk management models adopted in software projects, particularly, in the risk identification and risk assessment phases. The proposed enhanced model addresses all the problems identified during the literature review. At the same time, the frequently-occurring risks in commercial projects and those widely cited in official reports and papers, were classified according to the software development phases. This research takes into consideration the major inefficiencies of current software risk management models and introduces a new comprehensive classification of project risks at the requirements analysis, design, programming and testing, implementation and release phases. The risk teams conducting the risk identification and risk assessment, used the ISO 31000/31010:2009 and the proposed method, at the same time. The data collected from the case studies were statistically analysed to prove the hypotheses stated in Chapter 3. The features of the proposed model for risk identification and risk assessment in software projects were evaluated by comparing them to the features of ISO 31000.

A tool was designed and implemented using the Rapid Application Methodology (RAD) method in an SQL server for the risk identification and risk assessment of software projects using the proposed model. This tool incorporates various features and functions of the proposed model, and provides a method for employing systematic data required for assessing the performance of risk teams using the new model and ISO 31000/31010:2009, in parallel. Chapter 3 discusses the hypotheses for this research together with a more detailed description of the methodology.

1.7 Thesis Structure

This thesis consists of six chapters. Chapter 1 discusses the background of the research, the research problem, scope, limitations and methods (Figure 1.1). Chapter 2 presents a review on relevant works pertaining to risk identification and risk assessment, published in journals, reference books, standards, etc. It also discusses the problems and shortcomings of the existing models on risk management in software projects, particularly in the two early phases of risk management – risk identification and risk assessment. This chapter also discusses various risks identified or addressed in previous studies and compares the findings and analyses from academic papers published in recent years.

Chapter 3 covers the methodology adopted in this thesis. It discusses the features of the methodology and its suitability for use for the proposed model. It also illustrates the most important research activities of this thesis in figures, discusses the reliability, and validity of the findings of the case studies to evaluate the proposed model. This chapter also presents the hypotheses of this thesis, and discusses how the independent variables and metrics are used to reject or prove each hypothesis.

Chapter 4 covers the model which was designed and implemented to identify and assess risks in software projects. This model addresses the shortcomings of existing models discussed in Chapter 2. It also describes the structure and composition of the risk teams, and how the risk experts interact with internal risk analysts. The chapter also describes a comprehensive classification of candidate risks in different phases of software development. It also presents the Res-DVC tool to facilitate the implementation of the model and to collect data needed to assess the model performance.

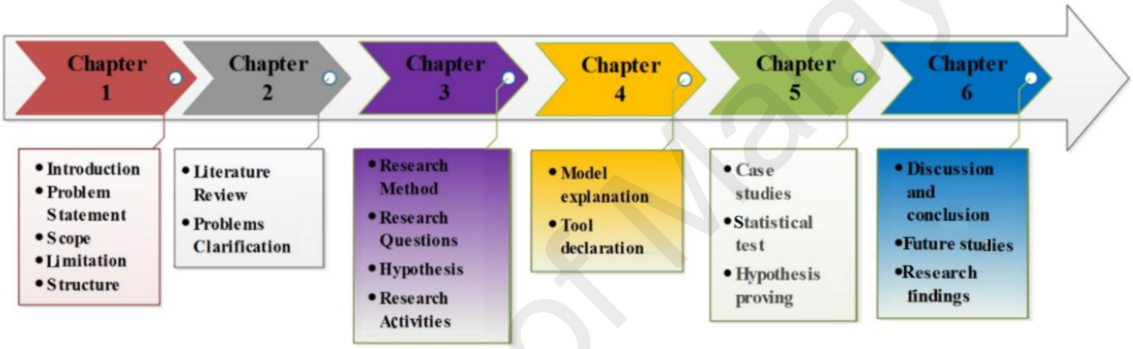


Figure 1.1: Thesis Structure

Chapter 5 discusses the two case studies conducted on 40 commercial medium-sized projects from two departments of a large commercial company. It discusses the statistical tests that were applied on the data collected and analysed using SPSS 22. This chapter lists the major risks identified in each project phase separately, and the frequency and ranking of each risk. Moreover, this chapter discusses and compares the control teams (ISO 31000/31010:2009) and the treatment teams (E-RIAM), in terms of efficiency. Finally, the hypotheses were evaluated based on the metrics and variables reviewed in the methodology.

Chapter 6 focuses on the problems covered by the proposed model, and addresses issues connecting the reliability, and internal and external validity of the research. It also discusses the problems and shortcomings encountered in completing this thesis as well as the reasons, limitations and constraints. The last section of Chapter 6 presents the conclusion of the research, together with several recommendations for future research on various perspectives.

CHAPTER 2: LITERATURE REVIEW

Risk management is a common and vital process in all engineering disciplines (Liu, Wang & Xiao, 2009; Larson, 2014). Software risk management is an important topic not only for software engineers but also for all organisations (Hydari, 2015). It enables project managers to find out the potential threats to their projects, as well as defects and to take appropriate pre-emptive measures (Persson et al., 2009). This chapter presents a literature review on issues relating to risk management, which include project risk management process, risk identification, assessment models, and related researches.

2.1 Definitions

The risk-related terms used in this section are based on the definition in the Software Engineering Glossary and risk management standards vocabulary (ISO Guide 73:2009; Burke, & Barron, 2015; Wanderley et al., 2015; DOD, 2015).

i) Risk: the probability that an asset has been affected badly by an event, which could be a threat or a mistake.

ii) Threat: a danger that is a destructive factor and usually targets the system security.

iii) Vulnerability: a failure or weakness in system security that can be caused by an attacker and can be part of the system such as control, implementation, and design.

iv) Counter measures: some executive, managerial and technical controls, which are used to ensure system integrity and privacy.

v) Impact on the organisation: when a risk happens, defects in software development can also affect the reputation, policy, and the contracts of the organisation.

vi) Probability: The probability of any accident occurring is merely an estimation and is usually expressed in percentage.

vii) Project risk: Project risk is defined by PMI as 'an uncertain event or condition that if it occurs, it can have a positive or negative effect on a project's objectives'.

viii) Risk management process: Project Risk management is the identification, assessment, and prioritization of risks followed by coordinated and economical application of resources to minimize, monitor, and control the probability of occurrence of risks and/or impact of unfortunate events or to maximize the realization of opportunities.

ix) Risk identification: Process of finding, recognizing and describing risks that involves identification of risk sources, events, causes and potential consequences. It involves historical data, theoretical analysis, informed and expert opinions, and stakeholder's needs. Comprehensive list of risks based on events that might create, enhance, prevent, degrade, accelerate or delay achievement of objectives.

x) Risk assessment: A structured process for organizations to identify how objectives may be affected. It is synonymous with analysing risks in terms of the consequences and their probability of occurrences, before further action is taken. It provides a better understanding on risks affecting achievement of objectives, as well as adequacy and effectiveness of controls already in place.

xi) Risk mitigation: A course of action taken to reduce the probability of occurrence and/or potential loss (consequences) from a risk factor which includes executing contingency plans when a risk metric crosses a predetermined threshold (when a risk factor becomes a problem).

xii) Technical risk: Technical risks generally lead to failure of functionality and performance.

xiii) Schedule risk: Project schedule does not meet planned milestone when project tasks and schedule release risks are not addressed properly. Schedule risks mainly affect the project, and consequently on the financial implication to the company and might lead to project failure.

xiv) Cost risk: The cost exceeds the planned project budget.

xv) Risk response control: Executing and evaluating the effectiveness of risks response plans. It is essential to have a well-defined schedules to ensure the success of this activity in order to enhance opportunities and minimize loss.

2.2 Software Project Risk Factors

Risk factors are used to prioritise the risks and determine the potential loss using some assessment methods. The main classes of risk factors include process maturity, technological newness, and application size and complexity. Risk factor represents uncertainty as well as the undesired consequences of a risk from a particular aspect.

2.2.1 Taylor's Risk Factors

Taylor (2006) introduced the main risk factors to be considered by information system developers and customers in selecting outsourcing approaches, and suggested appropriate strategies to deal with them. Table 2.1 shows that even after conducting some control and risk reduction activities or mitigation measures, serious threats still remain.

Some risks such as time, budget management, personnel and technology problems must be taken seriously. The two types of risks that are not noteworthy in any risk prediction plans include unforeseen and intractable risks. However, they can seriously affect the success of an IT project. Unforeseen risks are those that are either not recognized or their results and occurrence probabilities can be ignored. Therefore, they are not considered in risk evaluation. Intractable risks are those that hinder the project in different ways, despite plans to control and reduce the effects of those risks.

Table 2.1: Intractable risks or problems that still exist after mitigation

Software Risks	Percentage of projects where risks was anticipated	Percentage of projects where problems still exist after mitigation
Schedule and budget management	61	21
Vendor staffing	42	13
Newness of technology	31	13
Client organisation culture	18	10
Client expectations	20	7
Multiple sites; multiple countries	28	3

2.3 Process of Project Risk Management

Project Management Body of Knowledge (PMBOK) defines project risk management (PRM) as a systematic approach for identifying, analysing and controlling project risks in a way that maximizes positive outcomes of events from the effectiveness and probability aspects, and minimizes the negative consequences of events from the effectiveness and probability aspects (PMI, 2013). Figure 2.1 shows the project risk management process (Marchewka, 2015).

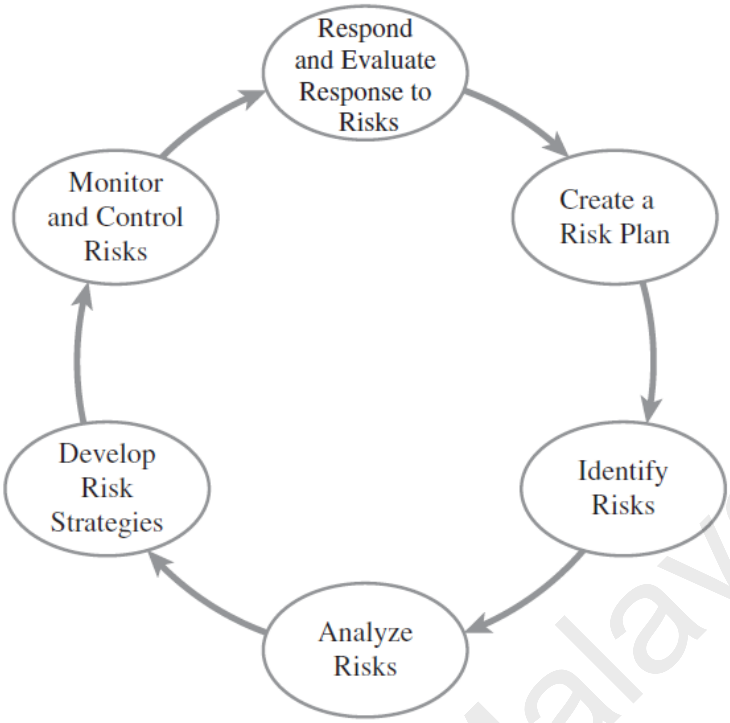


Figure 2.1: Project risk management process

According to Marchewka (2015), Project Risk Management (PRM) is an iterative process that starts from risk planning, as explained briefly below.

i) Risk Planning: In this step, the required resources to respond to different risks are identified. These resources include professional and unskilled personnel, the time needed, tools and technologies. Usually, the threats and opportunities are considered generally, and the people involved in the risk management process for the identification and analysis aspects of the risk management process, are identified. Hence, the most important task of this step is to prepare the resources and the tools so that they will be available, when needed.

ii) Risk Identification: This step involves the exact identification of opportunities, threats, and the cause and effect of each risk.

iii) Risk Assessment: After identifying the causes and effects of the risks, the occurrence probability and their adverse effects are determined. To assess the risks, some qualitative and quantitative approaches are used and several useful tools are available for this purpose. Accurate risk assessment helps the project managers in prioritising the risks based on number of threats and opportunities identified.

iv) Risk Strategies: Usually, project risk strategies include adopting various approaches - ignoring the risk, removing the risk fundamentally, decreasing the risk occurrence probability or its destructive consequences, and transferring the risk to third parties such as insurance companies. Often, some metrics are defined to determine when a risk can occurred. Also, some systems and approaches are required to ensure the availability of the resources that are needed. These strategies, the risk triggers, and the respective risk control actions are all incorporated in the risk response plan.

v) Risk Monitoring and Control: In a project lifecycle, permanent control and monitoring is carried out on the project environment, which includes all the components and sections which could be affected by the risks. Permanent control and monitoring helps in recognising the risk occurrence continuously, and the most appropriate method will be selected to mitigate its consequences.

vi) Risk Response and Evaluation: Risk control needs some practical actions before any decision is made. These actions are performed as a risk response measure and the most important action is resource allocation based on the planned risk strategy. On the other hand, risk evaluation involves determining the success rate of the performed actions, as well as documenting the experience and knowledge gained to manage the risks of similar IT projects in the future. Hence, risk documentation is done as part of the risk management process, as shown in Figure 2.2 (DOD, 2015). The solid line in the figure that starts from risk monitoring indicates the flow of the risk management process, and the feedback lines indicate the decision-making in the risk analysis and risk handling (strategies) steps. The dashed lines indicate the feedback from “risk monitoring” of the early risk planning and risk identification steps.

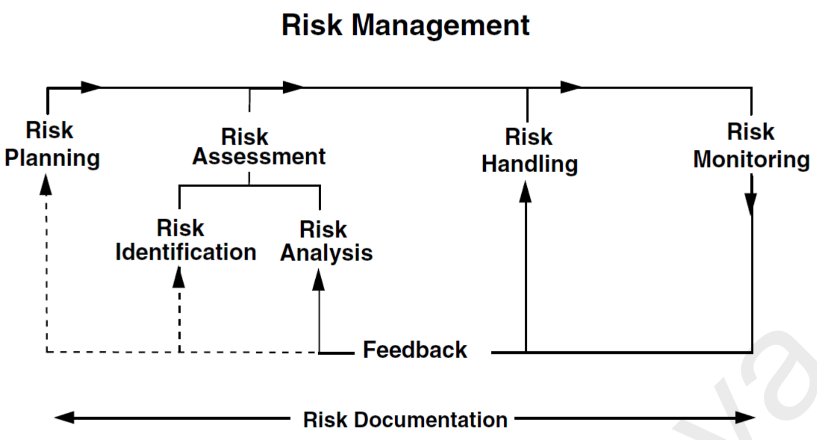


Figure 2.2: Risk Management Flow (DOD, 2015)

2.3.1 Risk Handling

Risks are handled by using of the right strategies to bring them to an acceptable level, as shown in Figure 2.3. In this connection, the workforce, costs, and timing must all be specified, the resource constraints identified, and major risk handling objectives must be set to identify the options, and to select and implement the most appropriate actions.

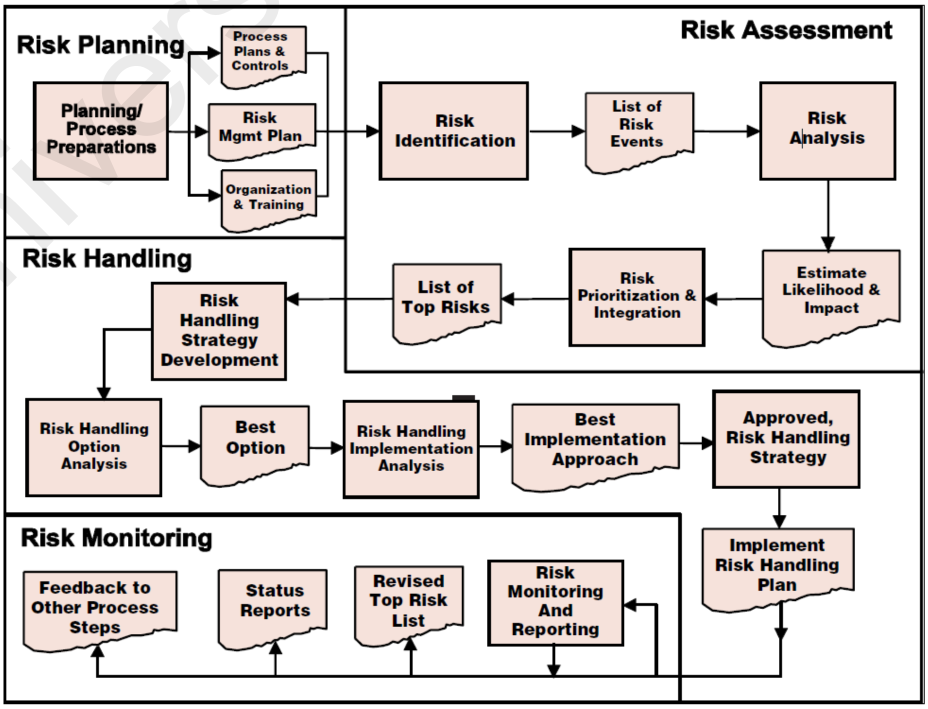

Risk Planning
Process Plans & Controls
Planning/ Process Preparations
Risk Mgmt Plan
Organization & Training
Risk Assessment
Risk Identification
List of Risk Events
Risk Analysis
Estimate Likelihood & Impact
Risk Prioritization & Integration
List of Top Risks
Risk Handling
Risk Handling Strategy Development
Risk Handling Option Analysis
Best Option
Risk Handling Implementation Analysis
Best Implementation Approach
Approved, Risk Handling Strategy
Implement Risk Handling Plan
Risk Monitoring
Feedback to Other Process Steps
Status Reports
Revised Top Risk List
Risk Monitoring And Reporting

Figure 2.3: Risk Handling in Risk Management Process (PMI, 2013)

2.4 Risk management models

This section presents the current software risk management models and their attributes.

2.4.1 Spiral model

The spiral model, proposed by Boehm (1988), emphasises risk analysis and management in the software development process, which involves identifying the non-deterministic aspects of project risks. It tries to follow an efficient economic strategy to deal with the resources of the risk, as illustrated in Figure 2.4.

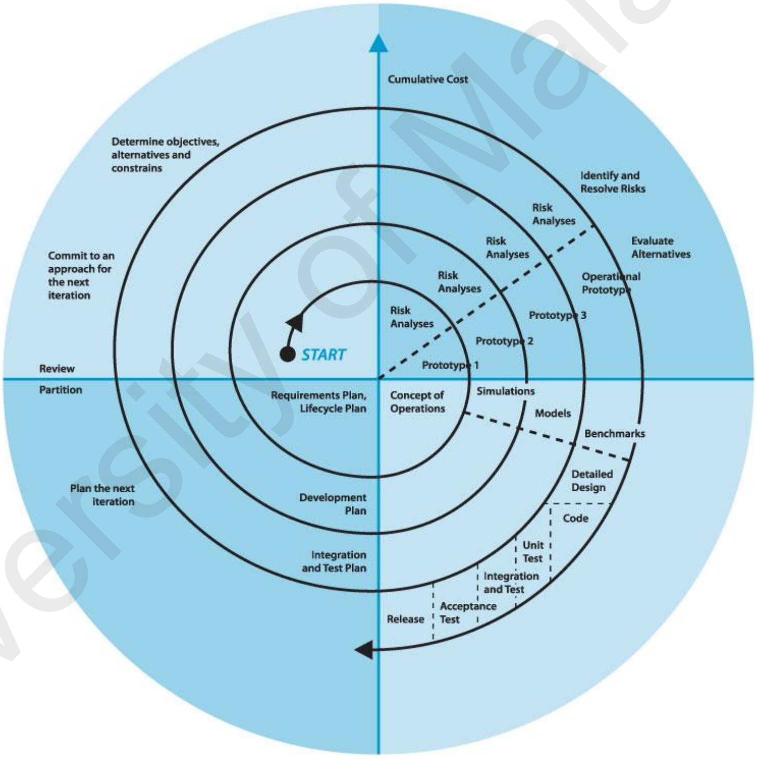


Figure 2.4: Spiral Model: A Risk-Based Software Development Model

2.4.2 Constructive Cost Model (COCOMO) II

Boehm (2000) stated that the COCOMO II model is efficient in reducing the risks related to the cost of software projects. In addition, risk analysis plays an important role in increasing software project efficiency.

2.4.3 Persson's Model

Implementing software risk management enables project managers to identify the potential threats to projects, the potential defects, and to take appropriate counter-measures against the threats. Persson et al. (2009) proposed an integrated framework for risk management in distributed projects. They categorised fields of risks and risk factors into six levels, as shown in Table 2.2.

Table 2.2: Risk fields and corresponding risk factors for software projects

Risk Field	Risk Factors		
Task Distribution	Task Uncertainty	Task Equivocality	Task Coupling
Knowledge management	Knowledge Creation	Knowledge Capture	Knowledge Integration
Geographical Distribution	Spatial Distribution	Temporal Distribution	Goal Distribution
Collaboration Structure	Collaboration Capability	Collaboration Mechanisms	Process Alignment
Cultural Distribution	Language Barriers	Work Culture	Cultural Bias
Stakeholder Relations	Stakeholder Commitments	Mutual Trust	Relationship Building
Communication Infrastructure	Personal Communication	Interaction Media	Teleconference Management
Technology Setup	Network Capability	Tool Capability	Configuration Management

2.4.4 Structured risk management model

Thomas and Bhasi (2011) proposed a structured model for software project risk management based on data gathered from 527 projects from of 95 companies. The model was developed based on four factors: executive management, Human Resource management, user coordination, and project planning.

2.4.5 The Risk Ranking and Filtering Method (RRF)

The risk ranking and filtering method (RRF) developed by Kwan and Leung (2011), facilitates potential failure detection and the auditing processes. The proposed risk management methodology supports risk dependencies. They also proposed a set of metrics for project risks that supports risk dependencies (Kwan & Leung, 2010). In their proposed risk response plans, they emphasized the importance of selecting a suitable risk response strategy, and the use of risk ranking tools. The most common way is to use tables. Table 2.3 shows their proposed risk response actions. They emphasized that it is not necessary to manage all the risks of a project, but to concentrate on only the top 20 risks in big projects.

Table 2.3: Risk response actions

Severity Level	**Probability**	**Purpose of Response Action**	**Risk (I>0)**	**Opportunity (I<0)**
High	High	High	Reduces Impact & Probability	Exploits Opportunity
Medium	High	Low	Reduces Probability	Enhances Impact
Medium		High	Reduces Impact	Enhances Probability
Low	Low	Low	Monitors Risk	Ignores Opportunity

The impact for risks and opportunity concurrently are denoted by capital I, respectively. When the value of impact is greater than zero, there will be a risk, and when it is less than zero, there will be an opportunity.

2.4.6 Cigital Risk Analysis Process

Figure 2.5 illustrates Cigital's continuous risk-analysis process developed by Verdon, and McGraw (2004). This process loops constantly at many levels of description through several phases. In the Cigital's approach, business goals determine the risks, which drive the methods, which yield the measurement, which drives decision support, and which in turn drives the fix/rework and application quality.

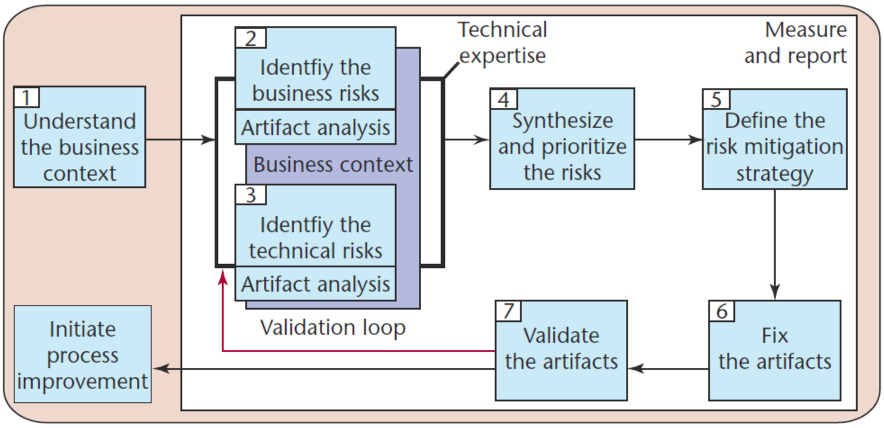

1
Understand the business context
2
Identfiy the business risks
Artifact analysis
Business context
3
Identfiy the technical risks
Artifact analysis
Technical expertise
4
Synthesize and prioritize the risks
Measure and report
5
Define the risk mitigation strategy
6
Fix the artifacts
7
Validate the artifacts
Validation loop
Initiate process improvement

Figure 2.5: Cigital's Continuous Risk-Analysis Process (Verdon & McGraw, 2004)

2.4.7 The Incremental Commitment Model Process Framework

The Incremental Commitment Model (ICM) is a process framework for improving decision-making and the project management processes. As this framework creates a balance between the risks and opportunities, Boehm (2008) proposed an ICM-based framework to identify the risks related to developing component-based systems.

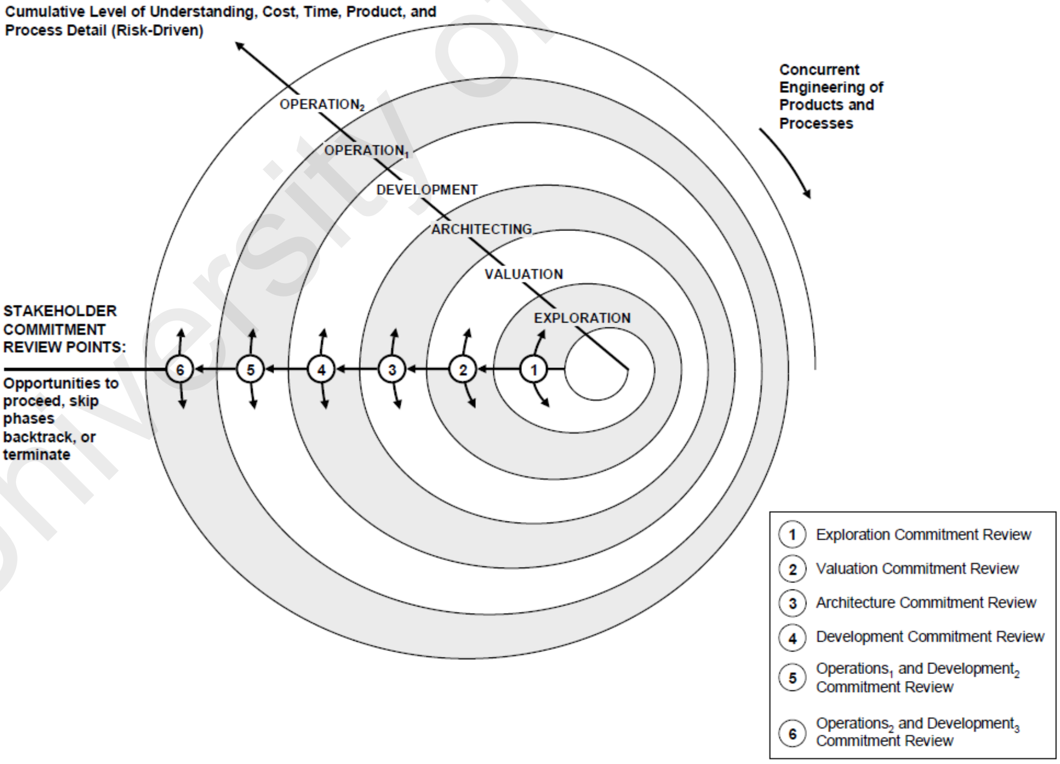

Cumulative Level of Understanding, Cost, Time, Product, and Process Detail (Risk-Driven)
Concurrent Engineering of Products and Processes
OPERATION₂
OPERATION₁
DEVELOPMENT
ARCHITECTING
VALUATION
EXPLORATION
STAKEHOLDER COMMITMENT REVIEW POINTS:
Opportunities to proceed, skip phases backtrack, or terminate
6
5
4
3
2
1
1 Exploration Commitment Review
2 Valuation Commitment Review
3 Architecture Commitment Review
4 Development Commitment Review
5 Operations₁ and Development₂ Commitment Review
6 Operations₂ and Development₃ Commitment Review

Figure 2.6: ICM Rotational Steps

Figure 2.7 illustrates the concurrent activities of the project lifecycle together with the ICM effort levels. The ICM mechanism which compares one or more commitment reviews is initiated before the start of the project lifecycle to perform synchronization, stabilisation and risk evaluation. Basically, there are five types of commitment reviews as shown in Figure 2.7 and explained briefly below:

i) Exploration Commitment Review (ECR): includes scheduling, scope, resources, and focuses on the exploration phase plan. The plan documents the risk-based contents and the risk levels, as these details are needed when managing and evaluating the risks.

ii) Valuation Commitment Review (VCR): records the results of the exploration phase, and the plan for the valuation phase.

iii) Foundation Commitment Review (FCR): or architecture commitment review highlights the most probable risks that might pose a threat to the plans, architectures, requirements satisfaction, and operational concepts.

iv) Development Commitment Review (DCR): this is performed by the software developer using special tools and products to ensure that the developer meets the identified feasibility factors. Both the FCR and DCR are carried out based on the Architecture Review Board (ARB) procedures as determined by AT&T.

v) Operations, Development Commitment Review (1): The OCR focuses on evidence of the adequacy of plans and preparations with respect to doctrine, organization, training, material, leadership, personnel, and facilities (DOTMLPF). The operations needs focus on the details of plans, budgets, and schedules.

vi) Operations, Development Commitment Review (2): The second ODCR addresses the often much higher operational risks of fielding an inadequate system.

At the end of each phase, the developer must provide evidences for the performed activities. For example, evidences for the first development phase - the requirements elicitation phase - can be the requirements specifications obtained by simulation, modeling or prototyping.

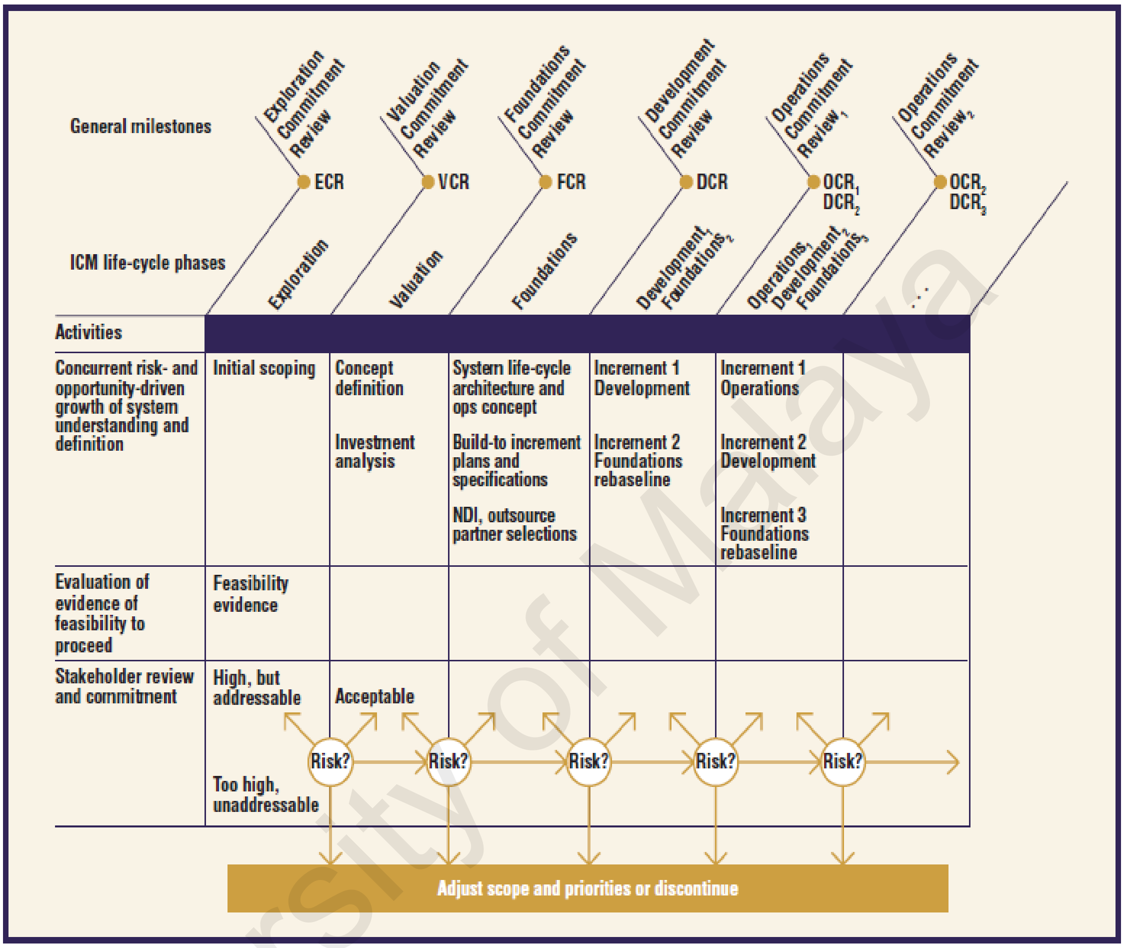

General milestones
Exploration Commitment Review
Valuation Commitment Review
Foundations Commitment Review
Development Commitment Review
Operations Commitment Review₁
Operations Commitment Review₂
ECR
VCR
FCR
DCR
OCR₁, DCR₂
OCR₂, DCR₃
ICM life-cycle phases
Exploration
Valuation
Foundations
Development₁ Foundations₂
Operations₁ Development₂ Foundations₃
. . .
Activities
Concurrent risk- and opportunity-driven growth of system understanding and definition
Initial scoping
Concept definition
Investment analysis
System life-cycle architecture and ops concept
Build-to increment plans and specifications
NDI, outsource partner selections
Increment 1 Development
Increment 2 Foundations rebaseline
Increment 1 Operations
Increment 2 Development
Increment 3 Foundations rebaseline
Evaluation of evidence of feasibility to proceed
Feasibility evidence
Stakeholder review and commitment
High, but addressable
Acceptable
Too high, unaddressable
Risk?
Risk?
Risk?
Risk?
Risk?
Adjust scope and priorities or discontinue

Figure 2.7: Processes in the Lifecycle of ICM

2.5 Risk classification

2.5.1 Risk Breakdown Structure

One of the common ways of identifying the potential risks and classifying them is to use the Risk Breakdown Structure (RBS) (Tanimoto et al., 2012; Ruhe & Saliu, 2005). RBS prepares a class hierarchy of potential risks of the software and IT projects. Using the RBS structure, the potential risks are classified into common or specific class, respectively. The main classes of IT project risks include the commercial, technical, organizational, and project management risks (Ruhe & Saliu, 2005). The technical risks of IT projects are classified into hardware, software, and network sub classes (Figure 2.8), although there might be other classifications of technical risks. The project management risks are divided into three subclasses: resources, connections, and estimations. Some risks that are related to quality might impact on project management as well. The classification of a risk may vary from one project to another depending on its specific features. Some of these features focus on security or delivery time.

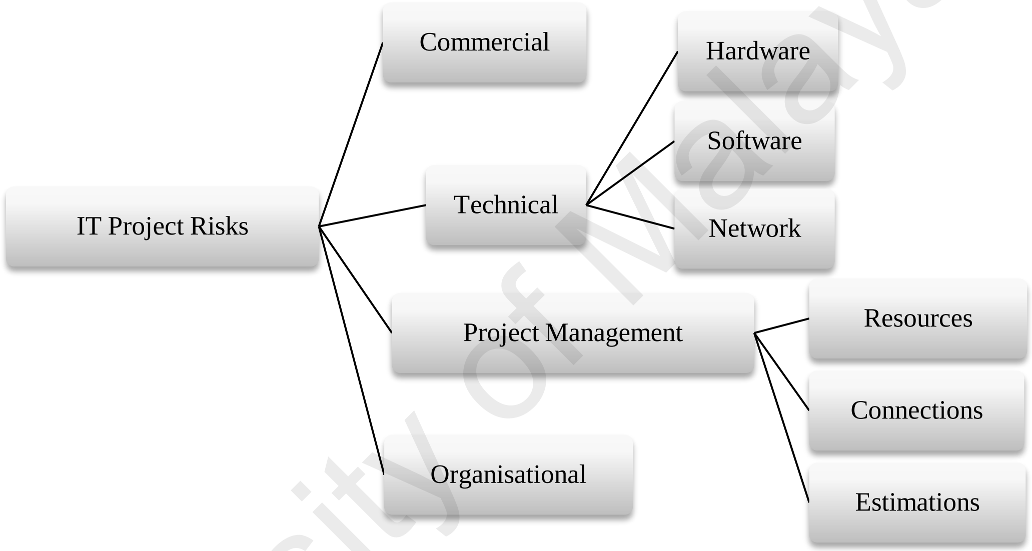


Figure 2.8: A Standard Sample of RBS for IT Project Risks

If a proper RBS is not used, this will result in incomplete classes, and there will not be a suitable strategy for risk management.

2.5.2 Reifer's Risk Category

The following risk categories are defined by Reifer (2002) for risks of Internet-based projects:

i) Personnel shortfalls;

ii) Misalignment with business goals;

iii) Unrealistic customers and schedule expectations;

iv) Volatile technology;

v) Unstable software releases;

vi) Constant changes in software functionality;

vii) Newer methods and unstable tools;

viii) High turnover;

ix) Friction within the team; and

x) Unproductive office space.

2.5.3 U.S. Air Force Risk Classification

Another classification of software risks is based on the Handbook of the U.S. Air Force, as shown in Table 2.4. The risks are divided into four types – requirements, personnel, reusable software, and tools and environment (AFPAM 908003, 2013). Each type of risk is sub-divided into four to five categories. For requirements type of risk, it is divided into five categories: size, resource constraints, application, technology, and requirements stability. The categories for the other three types of risks are shown in Table 2.4 The risk occurrence probability of each category is divided into three classes: impossible (up to 30%), possible (40% -70%), and frequent (more than 70%).

Table 2.4: U.S. Air Force classification of software risks (AFPAM 908003, 2013)

No	Risk	Categories
1	Requirements	Size
		Resource constraints
		Application
		Technology
		Requirements stability
2	Personnel	Availability
		Mix

		Experience
		Management environment
3	Reusable software	Availability
		Modifications
		Language
		Rights
		Certification
4	Tools and environment	Facilities
		Availability
		Rights
		Configuration management

2.5.4 Smith and Politowski Risk Classification

Acording to Smith and Politowski, (2013), software project risks can be divided into nine categories: i) financial risk, ii) technology risk, iii) security risk, iv) information risk, v) people risk, vi) business process risk, vii) management risk, viii) external risk, and ix) risk of success (which occurs when a project is so successful that it attracts more transactions than expected, but fails to scale the overload requirements).

2.6 Project Risk Identification

2.6.1 IT Project Risk Identification Framework

During a project lifecycle, risks can occur in any phase, and they can have serious impact on many aspects of the project, work products, and project goals. Hence, it is crucial to identify the various types of risks at the starting phase of the project lifecycle.

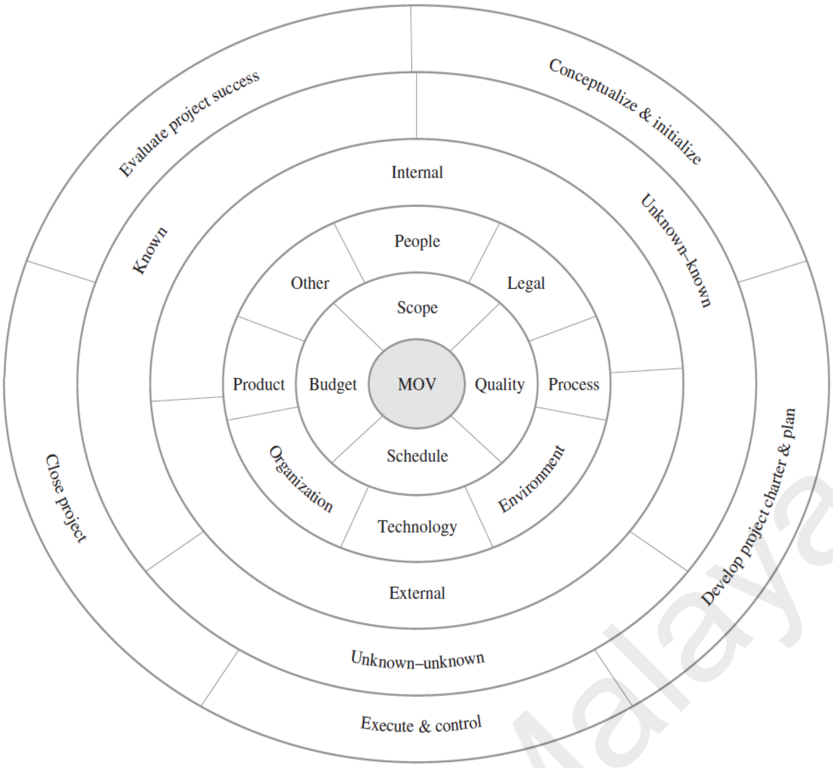


Figure 2.9: The Framework for IT Project Risks

Figure 2.9 illustrates a framework to identify IT project risks. The Measurable Organisational Value (MOV) is placed at the centre of the framework to indicate the main goals of the organisation which will be achieved through the projects implemented by the organisation. It is, thus, used as a yardstick to measure the success or failure of a project. The next layer of the IT project risks framework includes the objectives that outline the vital role to be played to meet the MOV. In the third layer, the IT project resources and their risk potential are located. Risk resources include the people involved, organisational, national and international rules, all types of processes, technologies, products, and matters related to the project environment. The fourth layer categorises risk resources into the internal and the external classes. Usually project managers are responsible for the internally assigned resources. Hence, if a risk comes from an external resource, the risk mitigation approach is often not within the control of project manager, and it will be handled at the organisational level. The fifth layer divides the risks into three categories: known, unknown-known, and unknown-

unknown. Known (identified) risks are those that show similar pattern of occurrence probabilities, and their impacts are known or already recorded. Unknown-known (predictable) risks are those in which their occurrences are expected, but there is no estimation about their impact and consequences. Unknown-unknown (unidentified) risks are threats in which there is no pre-information, and as a result, it is necessary to apply reactive strategies after their occurrence. The last layer shows the five different phases of the project lifecycle, and three classes of the risk impact to a project - low, medium, or high. Therefore, risk management should be performed dynamically and systematically.

2.6.2 Project Risk Identification Tools

Several tools and techniques for risk identification have been proposed by IT project experts over the last three decades. Many of these tools and techniques are used for identifying risks of other types of projects. The common approach to identifying risks is as follows: the project manager will first study the risk identification processes, project documents, and project information, and log those cases which may either lead to a threat or an opportunity. To obtain correct, updated and valid information, organisational officials, internal and external experts, project risk experts and other stakeholders need to meet to exchange views on all related issues. The common risks identification methods may be a combination of the following approaches (Karadsheh et al., 2008):

i) Brainstorming: In this approach, the risks are first identified by each of the experts assigned to identify potential project risks. These experts could be project risk experts, stakeholders, other experts from within or outside the organization. A brainstorming session is held and each expert presents his/her ideas and the probable solutions based on their knowledge, and experiences. The other experts will give their feedback and seek further clarification on any proposed solutions. Researches have shown that some experts in such brainstorming sessions do not welcome discussion about their ideas. Therefore, socio-psychological issues should also be considered before organising any brainstorming session.

ii) Delphi: This method involves the use of questionnaires to obtain feedback. In this method, each expert is briefed on a specific scenario, and the expert's ideas about the scenario will be collected and compiled by a coordinator. All the views expressed by the experts of this round will then be discussed and necessary modifications will be made as suggested. In the next round, the outcome of the first round will be passed to the experts. This procedure is repeated in order to obtain the experts' views in every round to identify the different classes of risks more accurately.

iii) Interview: Interviewing people who were involved in similar projects - successful or unsuccessful - in the past will greatly help to anticipate potential risks. Interviews can be conducted through the telephone, Google talk, and other instant messaging services.

iv) Root-cause analysis: To identify the risks, all actions and resources of the project as well as the common projects risks must be considered, in order to determine their relevance to the project.

v) SWOT Analysis: In many organisations, strategic plans are prepared based on SWOT analysis. Such analyses are suitable for risk identification, especially external risks identification. The opportunities identified following the SWOT analysis are useful to project risk managers for performing project opportunity management.

vi) Checklists: Different kinds of checklists have been prepared and released by various organisations, companies and professional groups for risk management of IT projects. These checklists can be customised for use as a standard approach for risk identification.

vii) Project assumption analysis: Project assumptions can be the cause of many risks. The project trend, resources, actions, scheduling and even the goals are set based on assumptions. Wrong assumptions can skew the results, giving rise to several risks. Therefore, project assumption analysis can make the assumptions to be more accurate, thus, making risk identification more efficient.

2.7 Project Risk Assessment

Having identified the risks that could threaten the successful outcome of a project, risk assessment will concentrate on the effect of each risk. Specifically, risk assessment is conducted from two perspectives - events and consequences. Risk assessment provides a basis for budget allocation, policy mitigation, and risk responses. The main steps in risk assessment stated in the National Institute of Standards and Technology (NIST, 2012 - re approval, 2015) standard, are shown in Figure 2.10. In this standard, identification is considered a part of assessment.

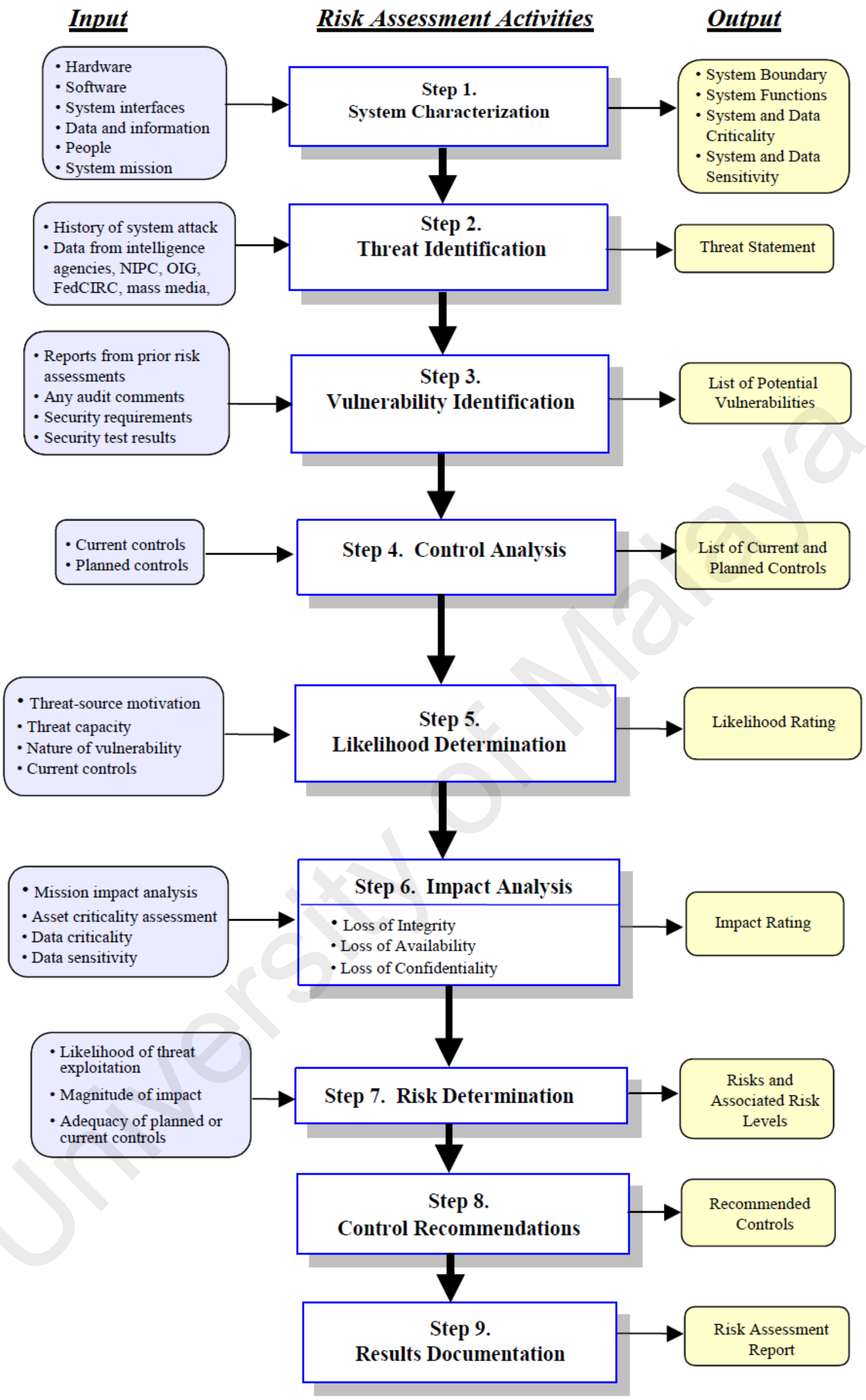


Figure 2.10: Steps in Risk Assessment (NIST, 2015)

Risk assessment is a vital process in any software risk management plan that is aimed at fullfiling the product specifications. Sound decision-making and good financial management depend very much on a good understanding of the risks. Hence, reliable risk management can be realised only through an accurate risk assessment process, otherwise, there would always be two major problems to resolve: i) major budget risks and hindrance to mitigation/deterrence mechanisms, and ii) budget is hindered and wasted.

2.7.1 Importance of Project Risk Assessment

Risk assessment is very important because budget allocations cannot be properly decided if the potential adverse effects of risks and their consequences are not fully understood. Undertaking risk management would also give a clear indication of the risks to manage.

2.7.2 Project Risk Assessment Techniques

There are various risk assessment techniques that can be used depending on the type and scope of the software. Risk assessment is a subjective technique as it is conducted by relevant personnel, hence, it is error prone. Risk assessment can be done in different phases of software development. Yaqoub and Ammar (2002) proposed a method for risk assessment at the software architecture design phase, and it focuses on reliability-based risks.

Goseva-Popstojanova et al. (2003) proposed a risk assessment method that uses Rational Rose Real-Time, and generating the risk factors scenarios using the Markov model. Their approach is more efficient in estimating the risk factors, as well as identifying more levels of risk severity. It also provides additional information for risk analysis. In the research, severity analysis is performed to assess the risk factors and their components, besides considering the potential consequences of defects.

The researchers proposed four levels of risk severity, and each level is assigned a severity index, as follows:

1-Catastrophic: errors that may lead to death;

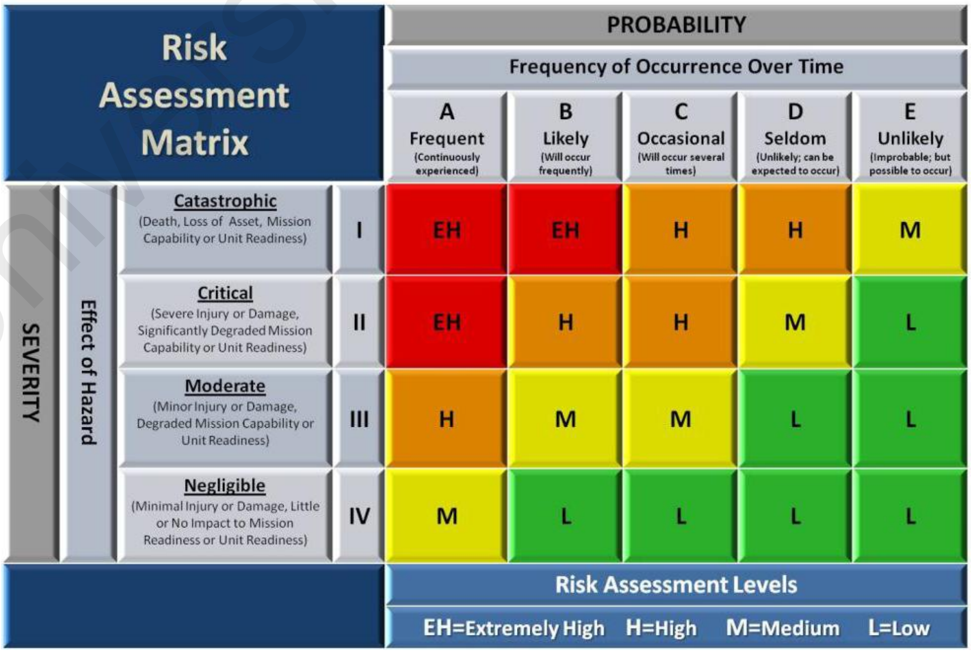
2-Critical: errors that may lead to serious injuries;

3-Moderate (Marginal): errors that may lead to partial injuries; and

4-Neglegible (Minor): errors that may not lead to any injury, but may still need to be managed or resolved.

Combining the severity level and complexity metrics can produce the heuristic factors of the risks, and also rank the risks for use case and scenario lists, based on the risk factors

Figure 2.11 shows the risk assessment matrix. Combining the severity level and the frequency of the probability of occurrence of the risks over time will produce the rankings (EH-Extremely High, H-High, M-Medium, L-Low) of the impact on a software project.



Risk Assessment Matrix			PROBABILITY				
			Frequency of Occurrence Over Time				
			A Frequent (Continuously experienced)	B Likely (Will occur frequently)	C Occasional (Will occur several times)	D Seldom (Unlikely; can be expected to occur)	E Unlikely (Improbable; but possible to occur)
SEVERITY Effect of Hazard	**Catastrophic** (Death, Loss of Asset, Mission Capability or Unit Readiness)	I	EH	EH	H	H	M
	Critical (Severe Injury or Damage, Significantly Degraded Mission Capability or Unit Readiness)	II	EH	H	H	M	L
	Moderate (Minor Injury or Damage, Degraded Mission Capability or Unit Readiness)	III	H	M	M	L	L
	Negligible (Minimal Injury or Damage, Little or No Impact to Mission Readiness or Unit Readiness)	IV	M	L	L	L	L

Risk Assessment Levels

EH=Extremely High H=High M=Medium L=Low

Figure 2.11: Risk Assessment Matrix (NIST, 2015)

In their research on risk evaluation, Tiwana and Keil (2004) collected some information on more than 700 IT projects from 60 large companies. They proposed a simple model for risk evaluation and named it “1 minute risk assessment tool”.

Tiwana and Keil (2004) identified six main risk drivers for software projects together with their relative risks, as shown in Figure 2.12. The figure indicates that the most common risk that could lead to project failure is the use of an inappropriate methodology.

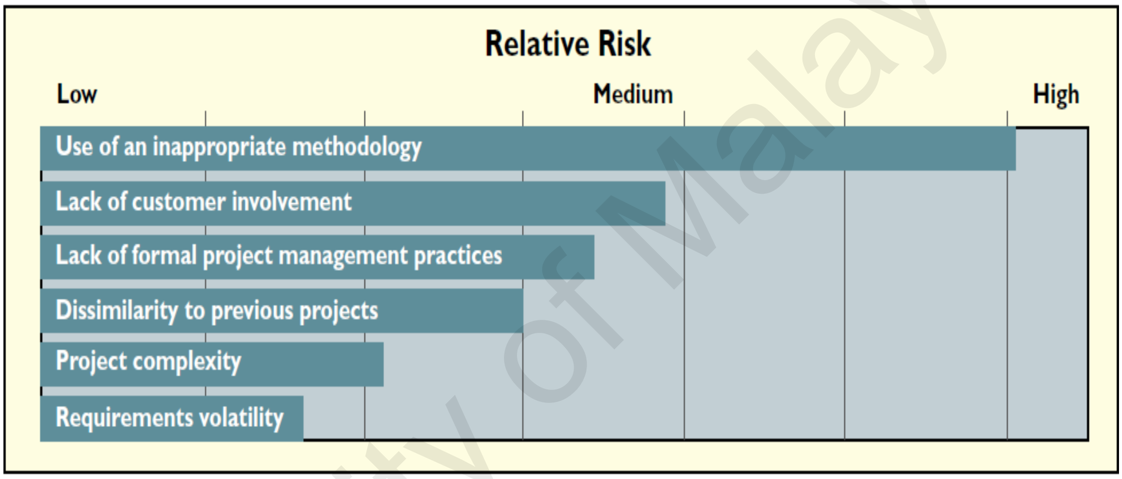


Figure 2.12: Hierarchical Holographic Model (Risk Drivers and The Relative Severity)

Benaroch and Appari (2010) proposed the software cost risks model based on the relationship between project cost changes and risk factor changes. Project sensitivity and excessive costs per unit are the two main parameters that this model uses to estimate each risk. They emphasised that if the estimation is done correctly and the result is combined with that of another estimation model such as COCOMO, the eventual outcome is beneficial to project risk management. Tables 2.5 and 2.6 show the most popular techniques as well as tools for risk assessment phase, respectively (Annex B of ISO/IEC 31010.2009).

Table 2.5: Popular techniques for risk assessment (Annex B of ISO/IEC 31010.2009).

Recommended Risk Assessment Techniques	
Brainstorming	Cause-and-effect analysis
Structured or semi-structured interviews	Layer protection analysis (LOPA)
Delphi method	Decision tree
Checklist	Human reliability analysis (HRA)
Preliminary hazard analysis (PHA)	Bow tie analysis
Hazard and operability study (HAZOP)	Reliability centered maintenance
Hazard analysis and critical control points (HACCP)	Sneak circuit analysis
Toxicity assessment	Markov analysis
Structured What If Technique (SWIFT)	Monte Carlo simulation
Scenario analysis	Bayesian statistics and Bayes nets
Business impact analysis	FN curve
Root cause analysis	Risk index
Failure mode and effects analysis (FMEA)	Consequence/probability matrix
Fault tree analysis	Cost/benefit analysis
Event tree analysis	Multi-criteria decision analysis (MCDA)
Cause and consequence analysis	

Table 2.6: Suitable tools for risk assessment (Annex B of ISO/IEC 31010.2009).

Recommended Risk Assessment Tools	
Altova MetaTeam	RiskAoA/Supervene
Capital asset pricing model	Risk Radar Enterprise (RRE)
EPRI Risk and Reliability Workstation (CAFTA)	Risk register
Event chain methodology	Systems Analysis Programs for Hands-on Integrated Reliability Evaluations (SAPHIRE)
Probabilistic risk assessment (also called Probability Consequence or Probability Impact Model)	SimpleRisk (based on NIST 800-30 risk management framework)
The RIMS Risk Maturity Model (RMM) for enterprise risk management	

2.8 Role of Software Risk Management

The most important role of software risk management is ensuring critical success factors (CSFs) (Daojin, 2010). The steps involved in risk management are more complicated than that of other activities in the project management process. They involve some complicated technological and human factors aside from the need for critical judgments (Boehm, 2008). Boehm emphasised that the success of projects is attributed to having skilled personnel to make correct judgments. He opined that risk management allows project managers to assign suitable people to critical positions such as the risks analyst. This would lead to better decision-making in risk identification.

2.9 Software Management Standards

There are various standards which can be adopted for conducting a software risk management process. Almost all of these standards provide a framework with some recommendations aside from the instructions. The most common and important standards are described below.

2.9.1 ISO 31000-2009

ISO 31000-2009 (approved by ISO/IEC/IEEE 16326-2009) is a standard for risk management process. Even though it mainly provides principles and guidelines for a specific framework, it is applicable to organisations of any size. This standard encompasses two references:

2.9.1.1 ISO Guide 73:2009

ISO Guide 73:2009 **i**ncludes the vocabulary of terms and definitions concerning risk management, aimed at providing uniform and transparent communication among those involved in the risk management process.

2.9.1.2 ISO/IEC 31010: 2009

ISO/IEC 31010: 2009 was developed in partnership with the International Electrotechnical Commission (IEC), focuses mainly on the concepts, processes and techniques for risk assessment.

2.9.2 IEEE Standard 1490-2011

The last IEEE standard for software project management is based on PMBOK version 4. Chapter 11 of the standard discusses project risk management. Figure 2.13 shows project risk management process based on this standard. It consists of Risk Plan, Identification, Response, and Control. According to this standard, Risk Identification consists of input, tool & techniques, and output. Figure 2.14 shows the 11 risk identification inputs.

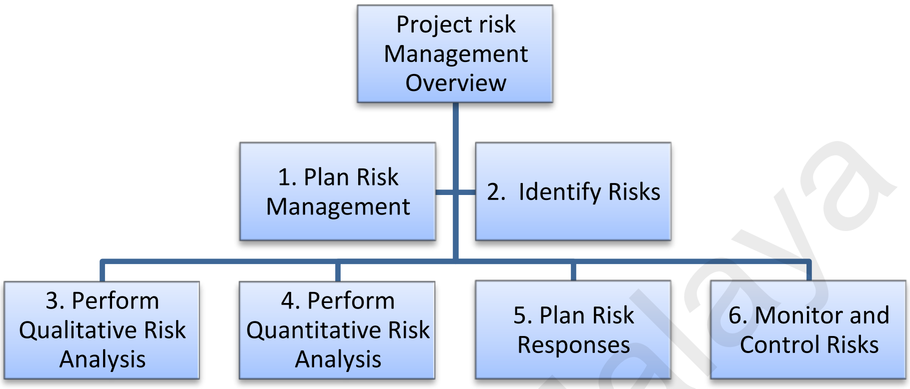
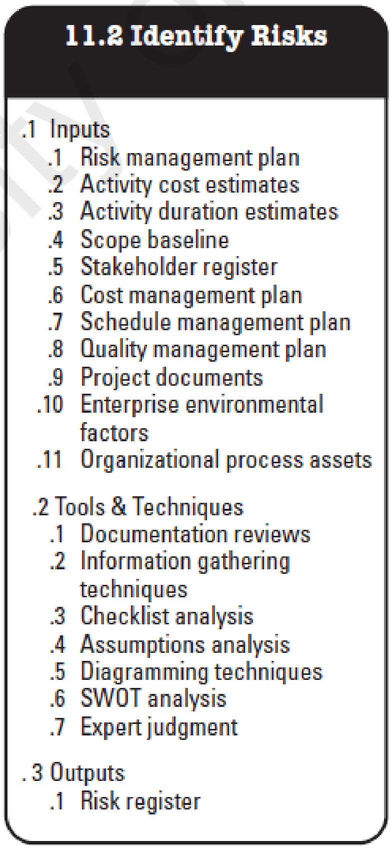

Project risk Management Overview
1. Plan Risk Management
2. Identify Risks
3. Perform Qualitative Risk Analysis
4. Perform Quantitative Risk Analysis
5. Plan Risk Responses
6. Monitor and Control Risks

Figure 2.13: Project Risk Management Process Based on IEEE Standard 1490-2011



11.2 Identify Risks

.1 Inputs
- .1 Risk management plan
- .2 Activity cost estimates
- .3 Activity duration estimates
- .4 Scope baseline
- .5 Stakeholder register
- .6 Cost management plan
- .7 Schedule management plan
- .8 Quality management plan
- .9 Project documents
- .10 Enterprise environmental factors
- .11 Organizational process assets

.2 Tools & Techniques
- .1 Documentation reviews
- .2 Information gathering techniques
- .3 Checklist analysis
- .4 Assumptions analysis
- .5 Diagramming techniques
- .6 SWOT analysis
- .7 Expert judgment

. 3 Outputs
- .1 Risk register

Figure 2.14: Risk Identification Process Based on IEEE Standard 1490-2011

2.9.2.1 Information Gathering Technique

Figure 2.14 shows the seven tools and techniques used to identify the risks. According to this standard, the outputs of identified risk list are associated with cause and effect. In the IEEE standard 1490, Risk Assessment is both a qualitative and quantitative risk analysis which includes Assessment, Probability, and Impact with regard to the Probability and Impact Matrix. This standard advises risk categorization in risk qualitative analysis, although specific classification has not been provided.

2.9.3 ISO 31000-2009

This standard along with ISO Guide 73-2009 is the vocabulary of a reputable international standard for risk management which took four years to become a legislation. The standard can be implemented in various industries and it advises IEEE. As a result, it has been cited in various standards such as 27001 (IEEE website). Figure 2.15 shows the ISO 31000- 2009 Risk Management Process:

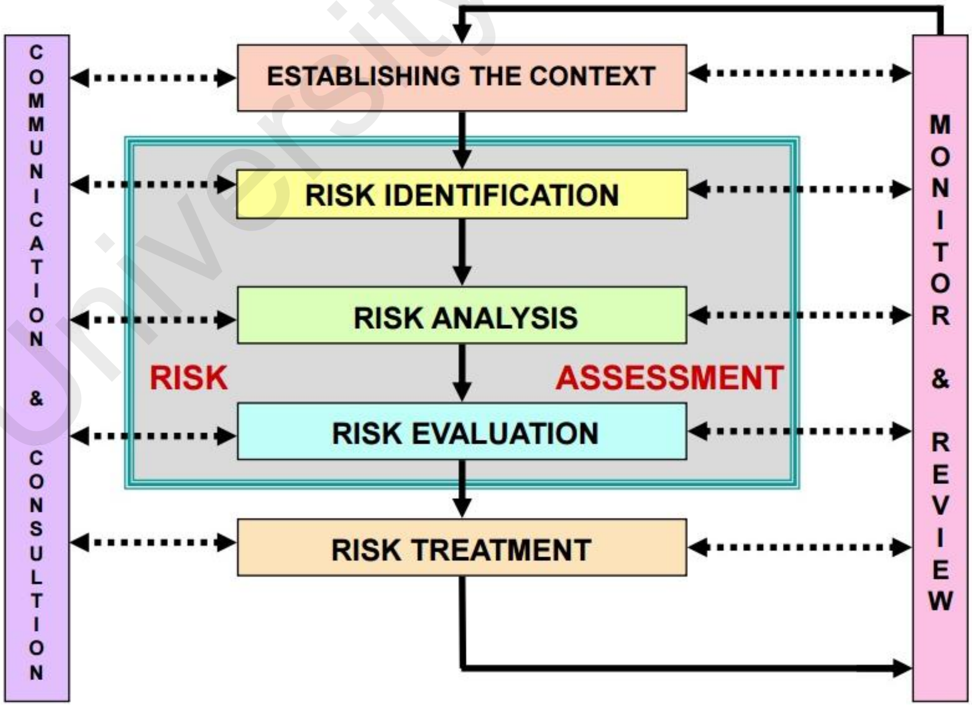


Figure 2.15: ISO 31000- 2009 Risk Management Process

The process has two interconnected elements: Consultation and communication with internal and external stakeholders, and continuous reviewing and monitoring of risk changes. ISO 31000 has no preference for either qualitative or quantitative risk analysis, and considers the combination of likelihood and impact to achieve risk information function rank. Risk assessment in ISO 31000 involves Risk Identification as the first activity followed by Risk Analysis and Risk Evaluation. The process begins with the definition of the organization (project) objectives known as Establish the Context. The last step is Risk Treatment, which aims at reducing the likelihood of risks and consequences. ISO 31000-2009 is the revised standard of AUSNZ43602004. In this standard, risk management process is considered a commercial part of the organization. Independence is the main weak point. Continuous consultation with internal and external stakeholders is another weak point of this standard, which slows down the implementation. The standard discusses the identification of all risks, including the not important risks, thus, making the implementation inefficient and costly. Due to the above-mentioned weaknesses, researchers have proposed different solutions to enhance the standard, but progress has been minimal. ISO 31000-2009 is not clear in some aspect and this might lead to incorrect decisions. Risk Assessment techniques of IEC 31010 are within this standard. The guides for implementing ISO 31000 are provided in a technical report TR31004-2013. A practical guide was published in 2013 for ISO 31000. According to this document, Small and Medium-sized Enterprises (SMEs) are not as fully equipped as large companies, therefore, they require certain guidelines to keep pace with ISO 31000. Despite all the known shortcomings of ISO 31000, it is the most comprehensive and only accepted international standard for risk management endorsed by the 28 countries, and it encompasses the experience of hundreds of experts concerning risk management.

2.9.4 AFPAM 908003 US Air Force Department Standard for Risk Management

The standard provides the last approval procedure for completing the Risk Assessment process. The AFPAM 908003 Guide and Tools, approved on February 11th, 2013 concerns the risk assessment process. Figure 2.16 shows the five-step Risk Management process in AFPAM 908003 standard. The first step is Hazard Identification which includes task analysis, factor determination, and hazard conditions which can lead to risks. Here, identifying risks is performed along with the causes and effects. The second step is Technical and Qualitative Measurement of hazards. In order to reach an assessment, this step consists of five key activities:

i) Exposure evaluation of hazards;

ii) Determining the severity via the adverse effects on activities, equipment, individuals, etc.;

iii) Determining hazard chance event;

iv) Determining risk level according to the severity and reliability ranging from extremely high to low; and

v) Combining the severity and probability using assessment matrix.

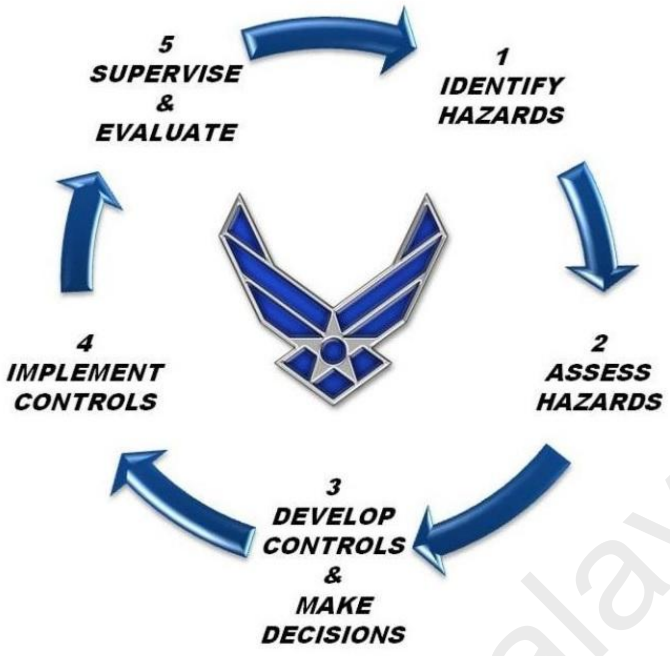
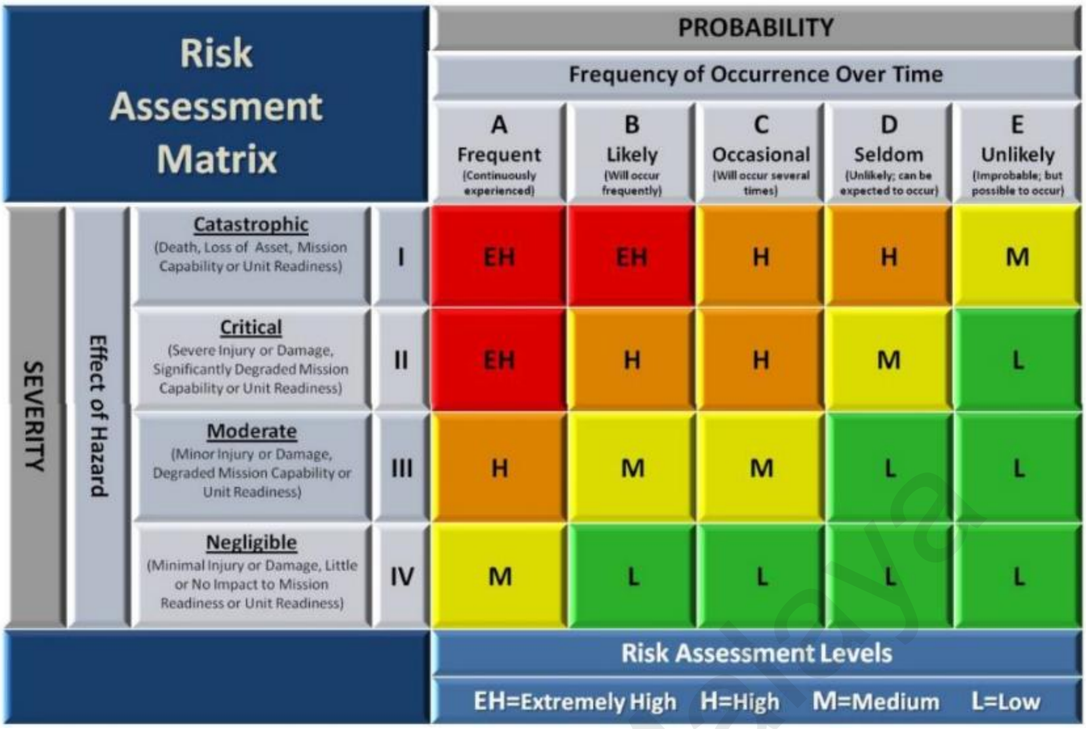


Figure 2.16: Five-Step Risk Management Process in the AFPAM 908003 Standard

The third step is Decision-Making and Control Development which involves selecting certain strategies to reduce or eliminate the destructive effects of risks. The fourth step, Implementation Control Strategies, indicates the various responsibilities for risk-decision implementation. The last step is evaluation where risk management managers and leaders are responsible for supervision at every level in order to ensure timely and correct control of the process. The standard recommends continuous monitoring of activities and operations. A feedback system is essential for a successful evaluation, and this includes generating systemic reports, and benchmarking. Figure 2.17 shows the risk assessment matrix in this standard.



Risk Assessment Matrix				PROBABILITY				
				Frequency of Occurrence Over Time				
				A Frequent (Continuously experienced)	B Likely (Will occur frequently)	C Occasional (Will occur several times)	D Seldom (Unlikely; can be expected to occur)	E Unlikely (Improbable; but possible to occur)
SEVERITY	Effect of Hazard	Catastrophic (Death, Loss of Asset, Mission Capability or Unit Readiness)	I	EH	EH	H	H	M
		Critical (Severe Injury or Damage, Significantly Degraded Mission Capability or Unit Readiness)	II	EH	H	H	M	L
		Moderate (Minor Injury or Damage, Degraded Mission Capability or Unit Readiness)	III	H	M	M	L	L
		Negligible (Minimal Injury or Damage, Little or No Impact to Mission Readiness or Unit Readiness)	IV	M	L	L	L	L

Risk Assessment Levels

EH=Extremely High H=High M=Medium L=Low

Figure 2.17: Risk Assessment Matrix in the AFPAM 908003 Standard

2.9.5 NIST 80037 Standard

NIST 80037 standard, released on May 6th, 2014, provides a framework for risk management in information systems in special applications. Risk management in this standard covers a wide range of fields from strategic to tactic risks. Figure 2.18 shows an overview of the risk assessment process in this standard. The process begins with information system classification followed by security controls, implementation, and evaluation. Steps 5 and 6 are information system authorization for operating and monitoring the security of the imposed controls. In other words, the standard merges software development and risk assessment which is mainly due to the sensitivity, known as Life Cycle Security Approval. This standard delegates methodology selection of risk assessment methodology to organizations informally, known as Synonym Risk Analysis.

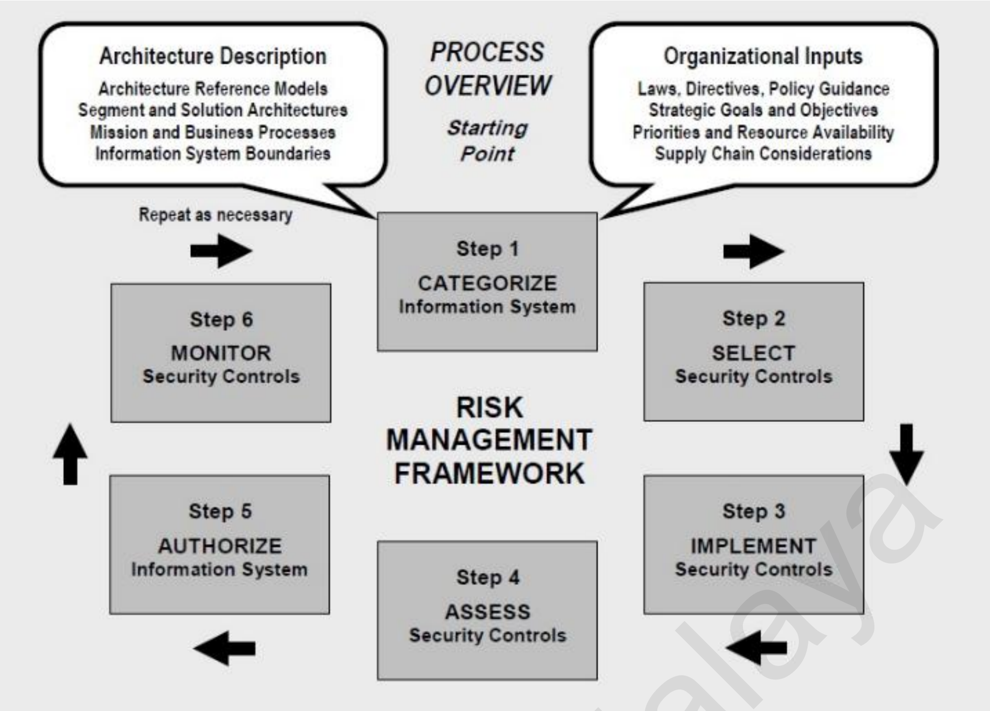

Architecture Description
Architecture Reference Models
Segment and Solution Architectures
Mission and Business Processes
Information System Boundaries
PROCESS OVERVIEW
Starting Point
Organizational Inputs
Laws, Directives, Policy Guidance
Strategic Goals and Objectives
Priorities and Resource Availability
Supply Chain Considerations
Repeat as necessary
Step 1 CATEGORIZE Information System
Step 6 MONITOR Security Controls
Step 2 SELECT Security Controls
RISK MANAGEMENT FRAMEWORK
Step 5 AUTHORIZE Information System
Step 3 IMPLEMENT Security Controls
Step 4 ASSESS Security Controls

Figure 2.18: Overview of the Risk Assessment Process in the NIST 80037 Standard

In this standard, the risk identification process which begins from the highest level of the process to the lowest level is considered a part of risk assessment. NIST published Risk Assessment guidelines within the 80030 Special Publication, in September 2012. Figure 2.19 shows the position of risk assessment in this standard.

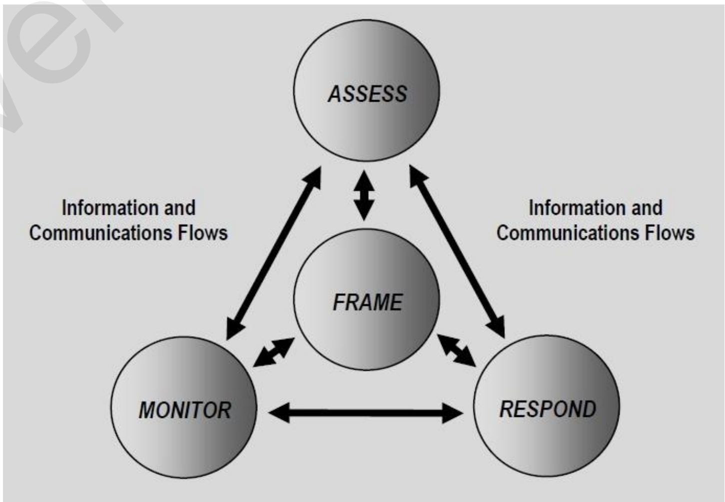

ASSESS
Information and Communications Flows
FRAME
Information and Communications Flows
MONITOR
RESPOND

Figure 2.19: Risk Management Process in the NIST 80037 Standard

Figure 2.20 shows the risk assessment process of the NIST 80037 standard. There are four steps: i) Preparing for risk assessment, ii) Performing assessment, iii) Exchanging opinions concerning the results, and iv) Assessment keeping. In the first step, involves determining related goal, scope, and concretes, and information sources required for risk analysis. The standard does not recommend any model for the assessment, and should be done on a case by case basis. The second step which covers the most important assessment tasks begins with the identification of threats and vulnerability conditions. Risks are finalized with likelihood of occurrence and impact also covered in this step. The third step is to share risk-related information and assessment results with project managers. Continuous consultation and opinion exchange with project managers, stakeholders, information security officers, etc. is considered a part of the process in this assessment. However, no information is provided with regard to this. The fourth step is keeping risk assessment, in which essential information is provided for risk monitoring and response. Risk monitoring and response are two major and important components of the risk management process.

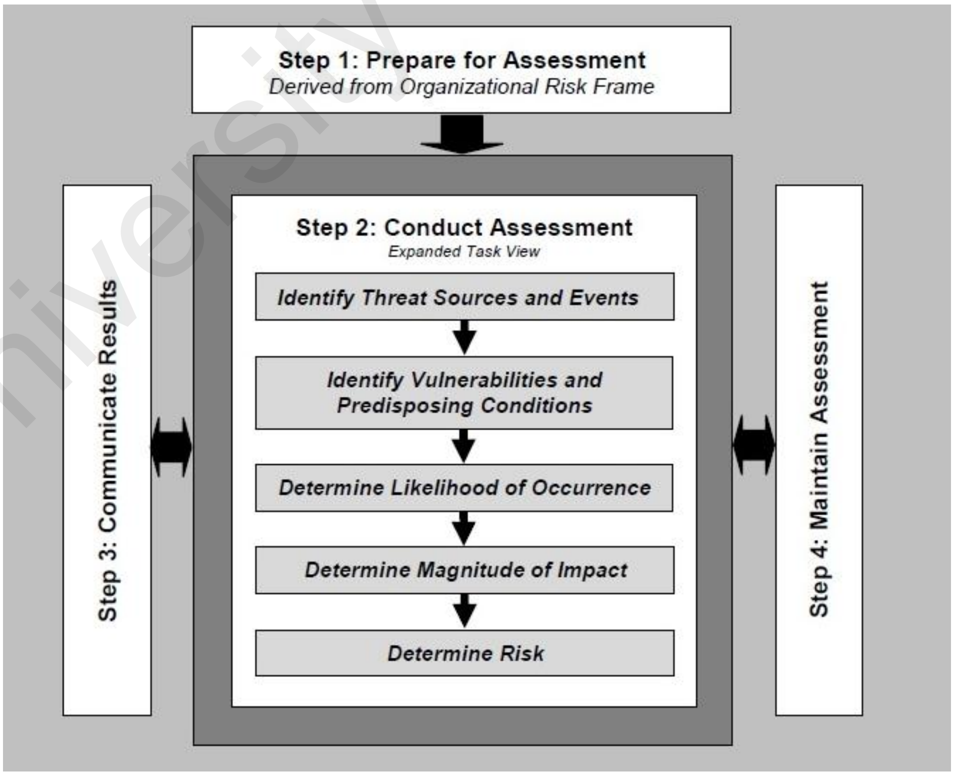

Step 1: Prepare for Assessment
Derived from Organizational Risk Frame
Step 2: Conduct Assessment
Expanded Task View
Identify Threat Sources and Events
Identify Vulnerabilities and Predisposing Conditions
Determine Likelihood of Occurrence
Determine Magnitude of Impact
Determine Risk
Step 3: Communicate Results
Step 4: Maintain Assessment

Figure 2.20: Risk Assessment Process in the NIST 80037 Standard

2.9.6 DOD 2015

The DOD 2006 was revised for risk and opportunity management. Previously, DOD 2006 had been applied in various fields, especially software and information systems, as an acceptable standard. DOD 2015 risk management was developed for systems engineering and related areas, particularly software development. In DOD 2015 standard, the risk management process is a five-step process (Figure 2.21).

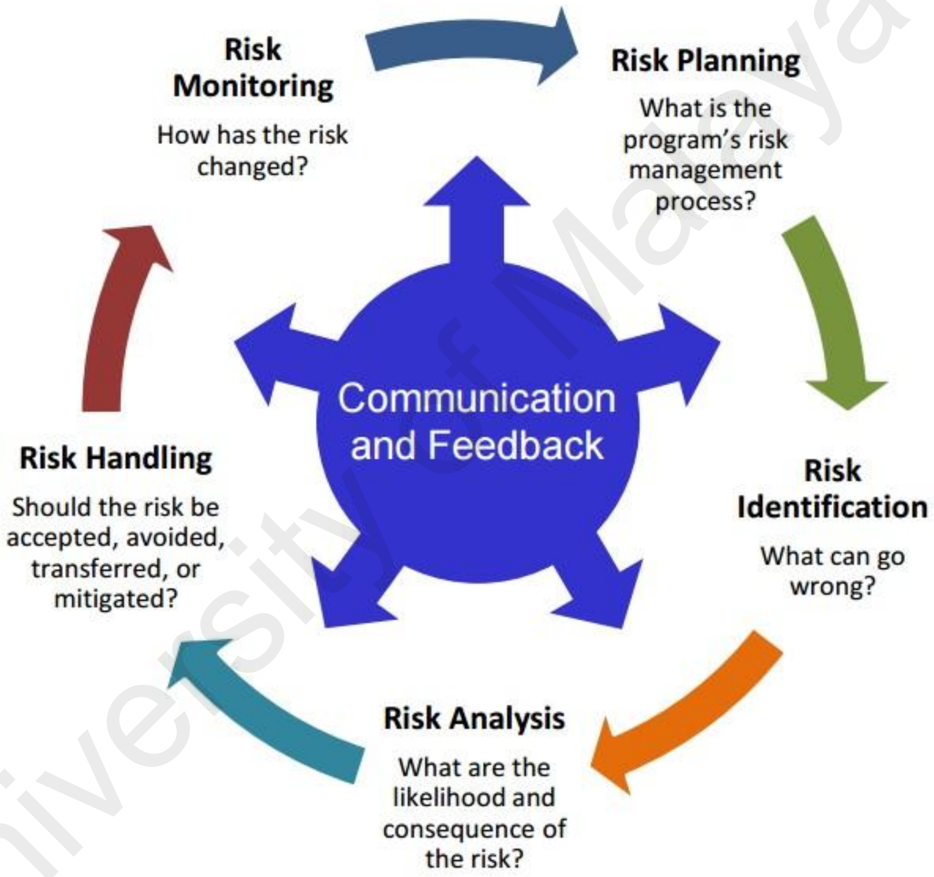

Risk Monitoring
How has the risk changed?
Risk Planning
What is the program's risk management process?
Communication and Feedback
Risk Identification
What can go wrong?
Risk Analysis
What are the likelihood and consequence of the risk?
Risk Handling
Should the risk be accepted, avoided, transferred, or mitigated?

Figure 2.21: Five-Step Risk Management Process in DOD 2015 Standard

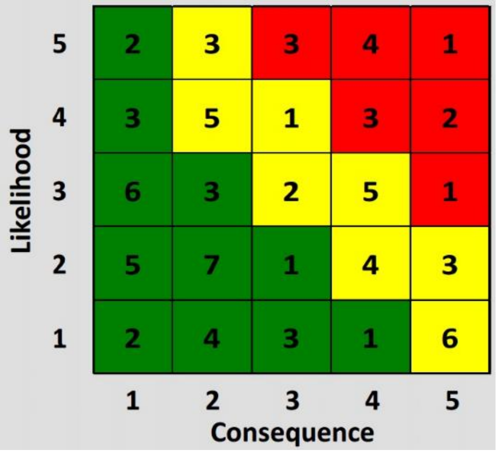
The first step is risk planning where the risk management process is developed, implemented, and documented. To do so, responsibilities are defined and risk analysis criteria (consequence and likelihood) together with procedures are specified. Resources, timing, and budget will also be determined. In the second step - risk identification - as the methods appropriate for each case (brainstorming, Delphi, diagramming) are considered as lower level methods. Table 2.7 shows the documents that can be used for risk identification (DOD, 2015).

Table 2.7: Documents for Risk Identification in the DOD 2015 Standard

Analysis of Alternatives (AoA)	Life-Cycle Mission Data Plan (LMDP)
Acquisition Strategy	Integrated Master Plan (IMP)
Acquisition Program Baseline (APB)	Integrated Master Schedule (IMS)
Systems Engineering Plan (SEP)	Contract structure and provisions
Analysis of Alternatives (AoA)	Government technical requirements and specifications documents
Systems Engineering Management Plan (SEMP)	Joint Capabilities Integration and Development System (JCIDS) documents
Test and Evaluation Management Plan (TEMP)	Integrated Master Plan (IMP)
Technology Readiness Assessment (TRA)	Integrated Master Schedule (IMS)
Program Protection Plan (PPP)	Contract structure and provisions
Life-Cycle Sustainment Plan (LCSP)	Government technical requirements

The next activity of this step is risk categorization which considers external risks pertaining to resources and market, and technical risks such as technology and requirements, as two separate categories.

Figure 2.22 shows the risk matrix, which indicates five categories for likelihood, and another five categories for impact. With this categorisation, three prioritised plots are created for risks displayed in green, yellow, and red. The risks whose assessed priority falls in the green, yellow, and red zones are the low, moderate, and high risks, respectively. This matrix combines likelihood, with maximum impact, cost, timing, and performance and specifies the risk priority level.



Likelihood					
5	🟩 2	🟨 3	🟥 3	🟥 4	🟥 1
4	🟩 3	🟨 5	🟨 1	🟥 3	🟥 2
3	🟩 6	🟩 3	🟨 2	🟨 5	🟥 1
2	🟩 5	🟩 7	🟩 1	🟨 4	🟨 3
1	🟩 2	🟩 4	🟩 3	🟩 1	🟨 6
	1	2	3	4	5
	Consequence				

Figure 2.22: Risk Matrix for DOD 2015

The next step of DOD 2015 risk management is risk handling. The most important tasks in this step include determining the acceptable risks, developing plans and efforts for risk mitigation or avoidance. In risk handling, DOD has also recommended risk transfer in which risk responsibility may be delegated to another entity such as the government, contractor, or agent. For this step to proceed successfully, it is recommended to have a risk breakdown plan, which covers risk handling activities, using a time diagram. Thus, the progress of the risk mitigation plan will be clearer.

The last activity of the risk management process in the DOD 2015 standard is risk monitoring, which will show to what extent the risk handling plans have succeeded. Risk monitoring registers, keeps, and reports risk information (including risks that have or have not materialised) and the progress of the risk handling plan.

2.9.7 ISO/IEEE 16326-2009 Standard

System and software engineering lifecycle processes – project management

This standard recommends IEEE 16085 standard for risk management. IEEE 16085 standard was asserted again strongly endorsed in 2011 and will be an active standard up to 2016. This standard may be used for both systems and software.

2.9.8 ISO/IEC 27005-2011 standard

Standards of class ISO27K are adopted for information security risks. ISO 27005 standard encompasses a continuous process including a series of activities for risk management. These activities begin by establishing the context of risk management. The quantitative and qualitative assessment of risks and, finally, exercising control of the risk level and monitoring a spectrum of these activities. ISO 27005 is planned to include application of ISO 31000 2009. Nevertheless, to date, this standard has not specified, recommended, or even mentioned any specific risk management technique.

2.9.9 Generic Risk Management Process

Larson and Gray (2014) considered four fundamental steps in an iterative loop for the risk management process. As can be seen in Figure 2.23 shows that this model focuses on project risk management. The components of the generic model are described as follows:

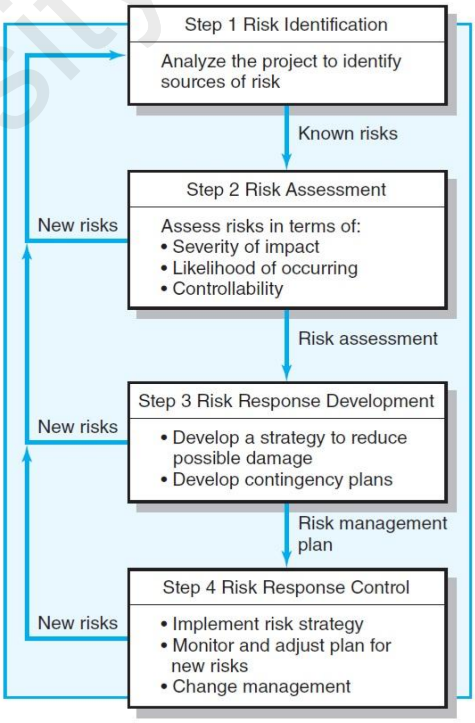

Step 1 Risk Identification
Analyze the project to identify sources of risk
Known risks
Step 2 Risk Assessment
Assess risks in terms of:
• Severity of impact
• Likelihood of occurring
• Controllability
Risk assessment
Step 3 Risk Response Development
• Develop a strategy to reduce possible damage
• Develop contingency plans
Risk management plan
Step 4 Risk Response Control
• Implement risk strategy
• Monitor and adjust plan for new risks
• Change management
New risks
New risks
New risks

Figure 2.23: Four-step Generic Risk Management Process (Larson & Gray, 2014)

2.9.9.1 First step: Risk Identification

In this step, a list of all potential risks to the project is drafted. For each identified risk, probable consequences are determined according to previous experiences, reports, or views of experts including managers and professionals. In this model, it is recommended to use the opinions of knowledgeable, experienced project managers. Among the key success factors in this step is encouraging people who are involved in the risk identification activity to apply critical thinking and consultation.

2.9.9.2 Second step: Risk Assessment

Determining the probability of the occurrence of a risk and its impact is considered to be the core of risk assessment.

This model incorporates the scenario analysis technique as the most common and the most practical technique. Each of the components of likelihood and impact are categorized into five grades in a risk assessment matrix. The risks comprise three zones - green (minor), yellow (moderate), and red (major). As the most important risk assessment activity, the risks rank may be determined by specifying impact and likelihood using risk severity matrix (Figure 2.24). The simplicity in identifying a risk may be added as a component to the FMEA severity matrix where the simplicity in identifying a risk is determined by using a five-point scale. Thus, risk value (the product of impact, probability, and detection) is used for ranking the risks. The use of this three-dimensional matrix includes a range from 1 to 125. In this context, a risk that has the lowest probability and the lowest impact can be easily identified - having a score of 1. A risk that has the highest impact and highest occurrence probability and is very difficult to identify will have a score of 125. Statistical techniques may also be employed for assessing project risks. Decision trees may be used for risk assessment. If a correlation can be established between past and current projects, S-curves and cash-flows will be usable. PERT simulation can also be a useful risk assessment tool if time- and resource-related data are available, with a proper statistical distribution.

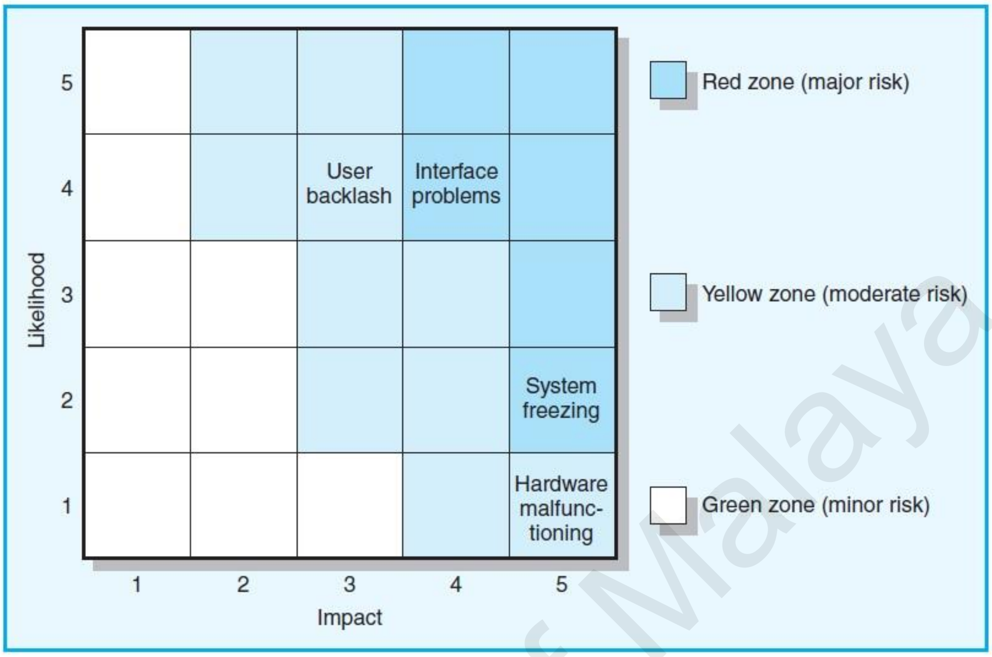


Figure 2.24: Risk Severity Matrix Proposed in the Generic Risk Management Model (Larson & Gray, 2014)

2.9.9.3 Third step: Risk Response Development

Decisions should be made as to what sort of response will be suitable for each identified risk. The proper response plan for risks can range from mitigation to retaining, and includes avoiding, sharing, and transferring, etc.

(a) ***Risk mitigation***

The two major risk mitigation strategies include the lessening of the consequences of the occurrence of a risk, if the risk materializes, and reduced probability of its occurrence.

(b) ***Risk avoidance***

One of the ways to face risk and threat is to prevent them from happening. In doing so, certain changes are made to the project plan. A wide spectrum of these changes range from revising the technology adopted to changing subcontractors or obtaining various permits.

(c) ***Risk transferring***

This policy or decision, i.e., delegating risk responsibility to another party, is also a solution to a risk. Passing risk responsibility does not mean risk elimination or decreasing the occurrence probability or consequences of a risk. It just means that another person or party will assume responsibility for dealing with the risk. Warranties, guarantees, utilizing foreign investment, sponsors, and financial supporters are instances of risk transfer.

(d) ***Risk retaining***

Risk acceptance or risk retaining is typically done for those risks for which proactive policies have not been adopted. Natural disasters and incidents caused by global crises are examples of factors for which the respective risks could not be countered using a pre-defined plan in a design project. What could be done is merely to devolve responsibility when they occur. Risk acceptance may also apply to risks with negligible consequences or whose combination of consequences and likelihood is very negligible. However, clear definitions and rules should be in place to handle such situations appropriately.

2.10 Using Experts in Risk Identification Step of the Model

Better identification of risks using experts to review documentation and study key project processes is beneficial to project success. This refers to dealing with the top-level class of risks and it is recommended to combine this method with low level methods such as brainstorming suggested in the DOD (2015) standard. Larson and Gray (2014) also explicitly proposed the use of experts' experiences by involving them directly in the risk identification step.

2.11 Review of Related Researches

A three-stage systematic literature review approach was used to review past related researches (Kitchenham & Charters, 2007) - planning, conducting, and reporting the review, as shown in Figure 2.25. Table 2.8 shows the list of bibliographic databases used to review the relevant literature for review.

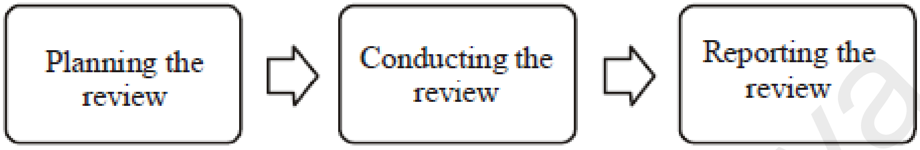


Figure 2.25: The Three-Stage Systematic Literature Review (Kitchenham & Charters, 2007)

Table 2.8: Bibliographic databases used

No.	Title	No.	Title
1.	Science and Technology of Advanced Materials	2.	Academic Info
3.	ScienceDirect	4.	Academic Search Engine
5.	Scientillion	6.	Arnetminer
7.	Semantic Scholar	8.	arXiv
9.	Social web	10.	Mendeley and Zotero
11.	The Collection of Computer Science Bibliographies	12.	BASE: Bielefeld Academic Search Engine
13.	Google Scholar	14.	Conclusion
15.	Inspec	16.	Copac
17.	iSeek Education	18.	CORE
19.	Association for Computing Machinery Digital Library	20.	dblp computer science bibliography
21.	Microsoft Academic Research	22.	Directory of Open Access Journals
23.	NERC Data Catalogue Service	24.	Dryad, , and FigShare,

According to Creswell (2012) and Cooper (2014), it is very important to use only valid resources for reviewing the academic literature. To review the published articles on research related to software project risk management, appropriate bibliographic databases were used. The search was conducted using keywords pertaining to software risk, which retrieved 4,300 papers on the topic from the IEEE, EBSCOhost, ResearchGate, Proquest, and Google Scholar databases. The search was further narrowed to retrieve only peer-reviewed papers, and this resulted in 385 papers retrieved. Table 2.9 lists some of the retrieved articles published from 2009 to 2016.

Table 2.9: Summarised list of risk-related scholarly articles (Chronologically Sorted)

Table 2.9 (Continued)					
No	**Year**	**Author**	**Title of Article**	**Focus**	**Resource Title**
1.	2016	Shiri, Teja & Ganesan	Advance Tools and Techniques for Software Risk Management	Software development faces many risks and challenges. This underscores the importance of risk management in the software development process. Software tools could be a lightweight approach to manage the risks in a better way. This paper also introduces some CASE and IDEs which are important for increasing the efficiency of risk management procedures.	International Journal of Advanced Research in Computer and Communication Engineering

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
2.	2016	Aruna	Impact of Team Skills in Software Quality - A Study on Twin Cities Small and Medium Software Development Units	This paper focuses on the skills of the software development teams. The author believes that the quality of software product is closely related to the expertise and skills of the development team members. The results of studies conducted on both small-sized and medium-sized software companies, concluded that the quality of the software developed is not dependent on the size, type and ownership of the software company, but on the development team members.	Splint International Journal of Professionals
3.	2016	Samantra, et al.	Interpretive structural modelling of critical risk factors in software engineering project	To identify the risk factors that can affect the success of software projects; it is suggested that risk factors be classified into four groups: autonomous, dependent, linkage and independent. A comprehensive structural model is developed to address the important risks factors	Benchmarking: An International Journal
4.	2016	Parthasara& Sharma	Impact of customization over software quality in ERP projects: an empirical study	The impact of major risks of customisation over ERP is investigated in this research. Using development of a framework for ERP customisation, the impact of customisation on ERP quality is rejected and the influence of customisation of source code and database over ERP quality is emphasised.	Software Quality Journal

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
5.	2015	Shukla & Husain	Study of Software Risk Analysis Models on Distributed Systems	Risk management is a vital part of the software development process. Risk assessment is the basic for risk activities. Determining each risk is a qualitative approach, while a quantitative model could be a better solution to manage the said risks. Quantitative methods are highlighted in this paper.	International Journal of Research and Development in Applied Science and Engineering
6.	2015	Roy & Dasgupta	A Study on Software Risk Management Strategies and Mapping with SDLC	Despite the many studies on software risk management, a large number of software projects still end up in failure. Weak risk assessment procedures is the main cause, although risk analysis is done on the whole project. This study identifies the key risk factors in different phases of the SDLC to achieve a better risk assessment.	2nd International Doctoral Symposium on Applied Computation and Security Systems
7.	2015	Quadri, Komal & Khalil	A Comprehensive Study on Risk Analysis and Risk Management in IT Industry	In view of the importance of risk management to the IT industry, this research focuses on the use of various tools and techniques for software risk analysis and management, and compares the results based on different aspects.	International Journal of Computer and Communication System Engineering

Table 2.9 (Continued)					
No	**Year**	**Author**	**Title of Article**	**Focus**	**Resource Title**
8.	2015	Kumar & Yadav	A Probabilistic Software Risk Assessment and Estimation Model for Software Projects	It is important to identify the risk factors that affect all the software development phases. This research basically tries to find a relationship between the risk factors and outcomes of the software projects	Eleventh International Multi-Conference on Information Processing
9.	2015	Elzamly & Hussin	Modelling and Evaluating Software Project Risks with Quantitative Analysis Techniques in Planning Software Development	This research focuses on the risk factors that are pertinent to implementing a better risk management process. It presents 10 risk factors for the planning phase, as well as another 30 critical risk factors. It concludes that successful software development can only be achieved by having better risk management.	Journal of Computing and Information Technology
10.	2015	Fulkerso, Thompson& Thompson	Team Member Perceptions of Software Team Leader Communication Influencing Motivation for Achievement of Project Goals	The results of a comprehensive multiple case studies using open-ended questions and NVivo 10 analysis tools are presented. The paper also presents the characteristics and skills leaders must possess in the software risk management teams.	Journal of Psychological Issues in Organizational Culture

Table 2.9 (Continued)					
No	**Year**	**Author**	**Title of Article**	**Focus**	**Resource Title**
11.	2015	Shrivastava & Rathod	Categorization of risk factors for distributed agile projects	This paper discusses the development of 45 risk factors for distributed agile development. The results show that the existence of several risks because between DSD and the agile approach.	Information and Software Technology
12.	2015	Choo & Goh	Pragmatic adaptation of the ISO 31000:2009 enterprise risk management framework in a high-tech organization using Six Sigma	This case study highlights the importance of feedback from stakeholders during the design phase of a framework for enterprise risk management. The integration of tools and resources is also emphasised.	International Journal of Accounting & Information Management

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
13.	2015	Ahonen et al.	Reported project management effort, project size, and contract type	The efforts data of 117 software projects are gathered and analyzed. Some statistical tests are conducted and the correlations show some shortfalls in the data reported. The issues are related to inaccuracy of reports are addressed and discussed. There is a need for the development of a new mechanisms for understanding the suppliers internal dynamics.	Journal of Systems and Software
14.	2015	Yang & Tamir	Offshore software project management: mapping project success factors	The factors related to a project and the offshore partners are investigated. The project-related factors are emphasised in the planning phase of software development. In the execution phase of software development, the outsourcer-related factors is emphasised.	International Journal of Project Organisation and Management
15.	2015	Jaafar, Janjua & Lai	(). Software Effective Risk Management: An Evaluation of Risk Management Process Models and Standards	This paper discusses the effective risk management techniques that have been used to evaluate the risk management models during the past 13 years. It also addresses the advantages and shortcomings of these techniques. The researchers believe that none of the current risk management models is perfect.	Information Science and Applications

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
16.	2015	Jowah	Project Management Tools and Techniques for Effective Project Execution	This paper investigates the role project management tools and techniques play in ensuring the success of projects. It provides recommendations for minimizing failures in projects through awareness of these tools.	Journal of Business and Economics
17.	2014	Wu et al.	OOPN-SRAM: A Novel Method for Software Risk Assessment	This paper proposes a risk assessment approach based on an object-oriented Petri Net which includes four phases. Using this approach leads to more effective and accurate software risk assessment	International Journal of Information and Computer Science
18.	2014	Talet & Talet	Incorporation of Knowledge Management with Risk Management and its Impact on IS/IT Projects	This paper highlights the importance of knowledge management for improving the risk management process. It presents the conceptual framework of a knowledge-based risk management (KBRM) process	International Proceedings of Economics Development and Research
19.	2014	Choo	Defining Problems Fast and Slow: The U-shaped Effect of Problem Definition Time on Project Duration	The research is based on the analysis of the data of 1,558 Six Sigma projects. The results show that the time of project completion could be saved by spending more time on planning during the early phases of project development.	Production Operations Management

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
20.	2014	Shahzad	Identification of Risk Factors in Large Scale Software Projects	Software risk of the software development incremental model are discussed in this research. The paper also describes the development of an avoidance and mitigation risk identification strategy.	International Journal of Knowledge Society Research
21.	2014	Mouratidis & Weippl,	An empirical study on the implementation and evaluation of a goal-driven software development risk management model	The research is focused on GSRM development to facilitate software risk management in view of three important constraints: requirements, change management, and user satisfaction.	Information and Software Technology
22.	2014	Serra & Kunc	Benefits Realisation Management and its influence on project success and on the execution of business strategies	The research shows that the integration of Benefits Realization Management and the corporate governance processes is very important for setting appropriate project success criteria.	International Journal of Project Management

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
23.	2014	Koolman	Top-10 risks in real-client software engineering class projects	This research reports on the weekly student's effort at the University of Southern California in risk identification and risk mitigation. The risks are encountered during the development of component-based software projects. This paper presents ten top risks based on various criteria.	IEEE 27th Conference on Software Engineering Education and Training (CSEE&T)
24.	2014	Lai	A WBS-Based Plan Changeability Measurement Model for Reducing Software Project Change Risk	This paper describes the development of a Changeability Measurement Model, and suggests that the risks in software development could be reduced by using the proposed WBS-based plan.	Lecture Notes on Software Engineering
25.	2014	Boehm	Software Project Risk and Opportunity Management	This paper reviews the recent models for risk management, with emphasis on opportunity management, and the techniques and tools used.	Software Project Management in a Changing World

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
26.	2014	Alsoghay& Djemame	Resource failures risk assessment modelling in distributed environments	This paper proposes a mathematical model for failure of resources risk prediction. Historical data is used to develop the model. The results of the evaluation of the model show that the resources risk of failure was correctly predicted.	Journal of Systems and Software
27.	2013	Ray et al.	A decision analysis approach to financial risk management in strategic outsourcing contracts	This research presents a comprehensive framework for managing the financial risks in outsourcing. The authors believe that managerial decisions about financial risks must only be taken within the organisations. Such frameworks can be useful for handling the risks properly	EURO Journal on Decision Processes
28.	2012	Salmeron& Lopez	Forecasting Risk Impact on ERP Maintenance with Augmented Fuzzy Cognitive Maps	ERP systems are needed for the new era. A practical ERP system is only useful if it has proper system of maintenance. The fuzzy approach is recommended to maintain ERP more accurately.	IEEE Transactions on Software Engineering

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
29.	2012	Ayodhya & Ramaiah	An Efficient Method of Risk Assessment using Intelligent Agents	Risk assessment is considered the most important task in software risk management process. This paper presents a new technique for performing risk assessment using intelligent agents.	2012 Second International Conference on Advanced Computing & Communication Technologies
30.	2012	Kruchten, Nord & Ozkaya	Technical debt: from metaphor to theory and practice	The technical debt phenomena in software development is categorised for many instances, and is discussed from various aspects.	IEEE Software
31.	2012	Taksande & Seaman	A Balancing Act: What software practitioners have to say about technical debt	To clarify the wide ranges of technical debt, 35 practitioners are interviewed. The paper discusses the data and information about the debt, and classification of the potential causes and effects.	IEEE Software

Table 2.9 (Continued)					
No	**Year**	**Author**	**Title of Article**	**Focus**	**Resource Title**
32.	2012	Diomidis	Don't Install Software by Hand	The complexity of interfaces and configurations are investigated in this research. The paper introduces control guidelines for using the IT configuration tools, and for simplifying the issues.	Software, IEEE
33.	2011	Thakurta	A Mixed Mode Analysis of the Impact of Requirement Volatility on Software Project Success	Requirements volatility is an important activity in software development. Requirements analysis is a crucial risk factor to consider when performing a risk management process. This research presents a two-phase study that includes interviews and surveys for managing requirements volatility	Journal of International Technology and Information Management
34.	2011	Lazzerini & Mkrtchyan	Analyzing Risk Impact Factors Using Extended Fuzzy Cognitive Maps	This paper describes the development of a group decision-making framework for risk handling. It also discusses risk handling in Software Project Management (SPM) using E-FCMs.	IEEE Systems Journal

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
35.	2011	Bernardi, Campos & Merseguer	Timing-Failure Risk Assessment of UML Design Using Time Petri Net Bound Techniques	This paper proposes a comprehensive risk assessment method for assessing the timing failure. The performance of the model was evaluated using a case study in a real-time embedded environment.	IEEE Transactions on Industrial Informatics
36.	2011	Tak Wah & Leung	A Risk Management Methodology for Project Risk Dependencies	This paper discusses risk dependency as well as the development of a management methodology. Three case studies were conducted to evaluate the efficiency of the model for systematically managing the risk dependencies.	IEEE Transactions on Software Engineering,
37.	2011	Rivard, James & Cameron	Software Project Risk Drivers as Project Manager Stressors and Coping Resources	This paper focuses on the emotional components of software project management. The role of the software project risk drivers is defined.	44th Hawaii International Conference on System Sciences

Table 2.9 (Continued)					
No	**Year**	**Author**	**Title of Article**	**Focus**	**Resource Title**
38.	2011	Avdoshin& Pesotskay	Software risk management	This paper reviews and compares the automated software tools, and describes a new risk analysis approach for information technology (IT) projects. The paper also presents recommendations for the software risk management process.	7th Central and Eastern European Software Engineering Conference (CEE-SECR)
39.	2011	Betz, Hickl & Oberweis	Risk Management in Global Software Development Process Planning	This paper presents the development of a new risk management model for planning global software development process. The model includes simulation and evaluation for process improvements.	37th EUROMICRO Conference on Software Engineering and Advanced Applications
40.	2011	Dongarra et al.	The International Exascale Software Project roadmap	This paper discusses the lack of planning and lack of determination in identifying the special risks, and the special aspects and requirements of extra-scale software projects.	International Journal of High Performance Computing Applications

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
41.	2011	Liu et al.	Relationships among interpersonal conflict, requirements uncertainty, and software project performance	The top 1,600 companies are investigated using surveys. The findings show that requirements diversity is the core of requirements instability, and has a negative effect on the performance of projects. Some recommendations are provided to overcome the problem.	International Journal of Project Management
42.	2010	Bardhan, Kauffman & Naranpana	IT project portfolio optimization: A risk management approach to software development governance	This research focuses on developing a prioritisation approach to integrate real option analysis with a project portfolio optimization model. The model helps the senior managers in developing the IT governance policies, and making strategic IT decisions.	IBM Journal of Research and Development

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
43.	2010	Dash& Dash	Risk Assessment Techniques for Software Development	This paper discusses the approach the Spiral model used for handling software risks. Some estimation models for software project risks are also discussed. The results show that using a project management tool will adversely affect the success of a project.	European Journal Of Scientific Research
44.	2010	Abdullah et al.	Risk Analysis of Various Phases of Software Development Models	This paper introduces the risks in each phase of software development, and the mitigation measures for those risks.	European Journal Of Scientific Research
45.	2010	Leitch	ISO 31000:2009-The New International Standard on Risk Management	The various aspects of ISO 31000:2009 are explained. The paper also present the advantages and shortcomings of the standard.	Risk Analysis

Table 2.9 (Continued)					
No	**Year**	**Author**	**Title of Article**	**Focus**	**Resource Title**
46.	2010	Benaroch, & Appari	Financial Pricing of Software Development Risk Factors	This paper introduces a comprehensive risk model. It also introduces the formulation of the risk-pricing information using different risk factors. Risk management is the key in handling project risks.	IEEE Software
47.	2010	Ahonen & Savolaine	Software engineering projects may fail before they are started: Post-mortem analysis of five cancelled projects	This paper investigates the main reasons and causes for the cancellation of software projects. Five software engineering projects were analysed through a post-mortem analysis method for this purpose.	Journal of Systems and Software
48.	2010	Chen	Fuzzy AHP-based method for project risk assessment	This paper describes the development of an improved Fuzzy AHP methodology for project risk assessment. The model facilitates decision-making by risk managers.	Seventh International Conference on Fuzzy Systems and Knowledge Discovery
49.	2010	De Bakker, Boonstra & Wortmann	Does risk management contribute to IT project success? A meta-analysis of empirical evidence	This paper investigates the relationship between project success and effective risk management approach. The results show that the perception of project success and risks by stakeholder as well as the behaviour of stakeholder in the risk management process are the two key issues.	International Journal of Project Management

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
50.	2010	Eveleens& Verhoef	The rise and fall of the Chaos report figures.	The results from 12,187 forecasts of 1,741 projects reveal the political biases of IT forecasts.	IEEE Software
51.	2010	Hong-bo, Hai-yang & Yan-ling	Research and application on risk assessment quantitative method based on fuzzy AHP	The paper describes the development of a ship integrated navigation system using an improved fuzzy AHP system. The results show the effectiveness of the model by simplification of the complexities.	5th International Conference on Computer Science & Education
52.	2010	Song & Dong	Risk evaluation in urban information system based on hierarchy fuzzy method	The risks in urban information systems are analysed. The identified risks are assessed using AHP and Fuzzy Logic methods.	2nd International Conference on Computer Engineering and Technology
53.	2009	Liu, Wang & Xiao	The Role of Software Process Simulation Modeling in Software Risk Management: a Systematic Review	Software projects are still vulnerable to various software risks. The Software Process Simulation Modelling (SPSM) still needs to be more focused on risk management. Also, most of the studies are related to risk planning and risk analysis, and most SPSM risk models are not applicable to real-life software projects.	3rd International Symposium on Empirical Software Engineering and Measurement

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
54.	2009	Zardari	Software Risk Management	This paper discusses the need for risk management in software development, especially the dynamic nature of software projects.	International Conference on Information Management and Engineering
55.	2009	Persson et al.	Managing Risks in Distributed Software Projects: An Integrative Framework	An integrated framework together with web-based tool are used to manage the risks for geographically distributed software.	IEEE Transactions on Engineering Management
56.	2009	Peng et al.	Empirical Evaluation of Classifiers for Software Risk Management	This paper presents a performance metric for software defect prediction. The metrics are used to evaluate the quality of classifiers. The top three classifiers are ranked by K-nearest-neighbor algorithm.	International Journal Of Information Technology & Decision Making
57.	2009	Benaroch& Goldstein	An Integrative Economic Optimization Approach to Systems Development Risk Management	This paper describes the development of an integrative economic optimization approach with a micro level technical view. The results obtained when using the model on some projects show that the model is suitable for mitigating the impact of software risks.	IIEEE Transactions on Software Engineering

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
58.	2009	Dongarra et al.	The International Exascale Software Project: a Call To Cooperative Action By the Global High-Performance Community	The wide range of conceptual and technical problems in the new era of open source software are explained. It discusses the need for cooperation and coordination among the software developers worldwide.	International Journal of High Performance Computing Applications
59.	2009	Islam et al.	Offshore-outsourced software development risk management model	This paper focuses on the development of the popular Offshore-outsourced software. An appropriate risk management model is introduced and the objective of the projects and the related risk factors are specified. The results show the effectiveness of applying risk management at the early phases of software development.	12th International Conference on Computers and Information Technology

Table 2.9 (Continued)

No	Year	Author	Title of Article	Focus	Resource Title
60.	2009	Kutsch & Hall	The rational choice of not applying project risk management in information technology projects	This research investigates IT projects using a survey and interview. The results show that one third of IT projects failed because there is no formal risk management process (FRMP). The main reason is the improper cost justification for implementing the FRMP.	Project Management Journal

Cortellessa et al. (2005) proposed a methodology for estimating performance-based risk factors by defining performance-based risk analysis and emphasising the crucial role of performance in system security. The methodology explains UML diagrams to find the occurrence probability of performance-related problems. This methodology estimates failure intensity by using functional failure analysis and combining it with the estimation done in the previous phase. The results enable the professionals to have different scenarios of system risks and software components risks in order to improve the software design.

Gorla (2012) identified the consequences of risk factors related to information systems outsourcing. He collected data from 150 companies, and found that outsourcing-related decisions are strongly affected by risks and previous system outsourcings (Gorla, Lau, & Mei, 2010).

Abdullah et al. (2010) used Rapid Application Development (RAD) technology to identify the different risks in the software development phases. They analysed these risks and suggested mitigation measures. The type of projects studied include real-time, mobile, service, distributed, Web, and database projects. They studied 100 projects and found that these risks are affected by four main parameters: i) cost, ii) scheduling, iii) reputation of the organization, and iv) credit and dependency on the risks of other phases. The results show that cost and scheduling risks are the two most important parameters in Web-based and database projects. Scheduling is the highest-risk parameter for all types of projects, except for service projects. Also, for most of the projects, organisational credit is of average importance and sensitivity.

Yaqoub and Ammar (2002) proposed a methodology for risk assessment at the software architecture level, and focuses on reliability-based risk. The architecture level refers to the first phase of the software development lifecycle. The methodology uses Dynamic Complexity and Dynamic Coupling metrics, which consider the components and the connectors as the architectural elements and define the complexity factors for them. Their proposed method for reliability risk analysis involves six steps: i) The system architecture is modeled using the Architecture Description Language (ADL), ii) Complexity analysis is done by simulation, iii) Severity analysis is done by simulation and failure mode, and effect analysis, iv) Risk heuristic factors involve the components and connectors, v) Components Dependency Graphs (CDG) are created for risk evaluation, and vi) Risk analysis and evaluation is done using graph traversal algorithms.

2.11.1 Distribution of Articles (1973-2016)

Table 2.10 shows the number of articles searched on risk-related topics. The first column lists the terms used in the search. The second column shows the number of articles with the term found in the title of the article, and the third column shows number of articles with the term found in the abstract and keywords section of the article, respectively. The last column shows the number of articles found with the term appearing in the research method, hypothesis, and research question (paper content) sections.

Figures 2.26 shows these distributions in a bar chart. It is obvious that has been a tremendous increase in the number of articles on risk management in recent years.

Table 2.10: Distribution of articles based on risk-related terminology

Term	No. of articles with terms in the Title	No. of articles with terms in the Abstract – Key word sections	No. of articles with terms in the Paper Content
Risk method	0	0	2
Risk correlation	1	2	2
Risk analyser/ risk analyst/ Risk personnel	1	0	4
Risk expert	0	8	4
Risk driver	1	5	6
risk dependencies	١	4	8
Fuzzy AHP*	2	4	8
Risk occurrence	0	12	13
Risk manager	0	1	17
Risk event	0	5	20
Risk estimation	5	7	20
AHP*	2	16	20
Risk avoidance	١	1	21
Risk model	6	9	22
Risk response	0	3	24
Risk impact	3	29	31
Risk monitoring	94	5	46
Risk exposure	1	9	57
Risk control	0	7	62
Risk evaluation	8	8	63
Software project risk	15	22	67
Fuzzy	19	36	78
Risk mitigation	6	14	86
Risk factor	9	60	92
Risk identification	4	15	127
Project risk	٢٠	26	133
Information Technology and risk	12	10	162
Risk analysis	18	16	178
Risk assessment	28	36	180
Software risk/ hazard	49	18	226
Risk management	72	80	268

Note: * Analytical Hierarchical Process

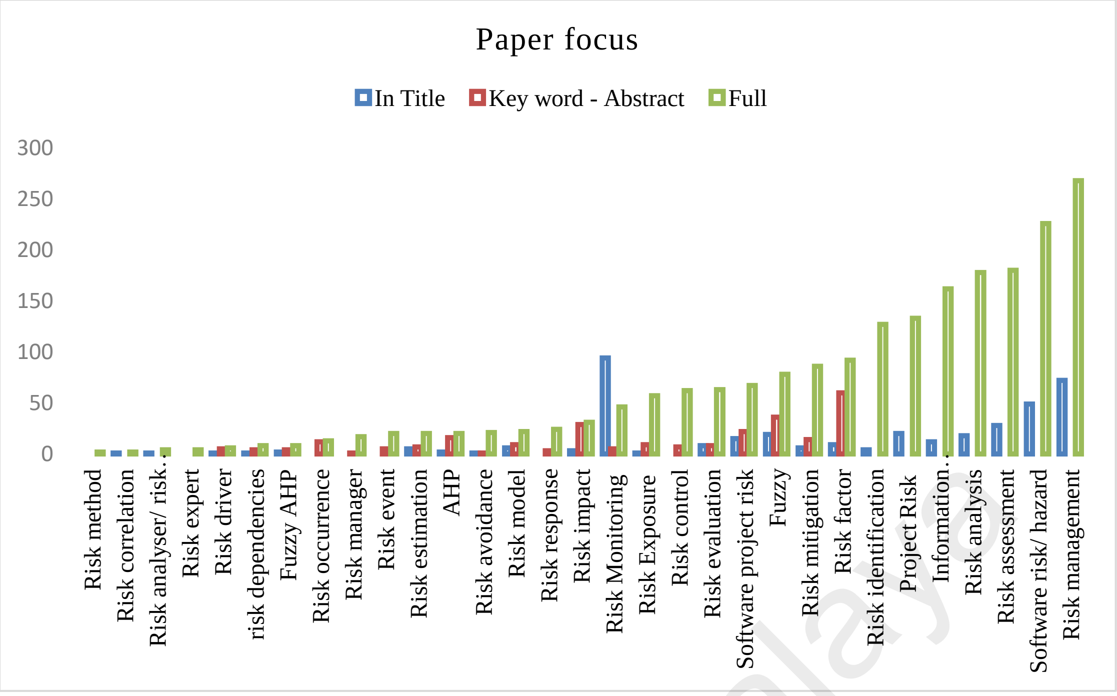


Figure 2.26: Distribution of Articles Containing Risk-Related Terminology

Figure 2.27 shows the distribution of articles (in percentage) based on the year of publication. The number of risk-related papers published has increased remarkably in 1999, 2002, 2004, 2008, 2010, and 2015. This shows that software project risk management has been widely implemented, and this will have positive impact on software development projects, as reported in the Standish Group's Chaos report.

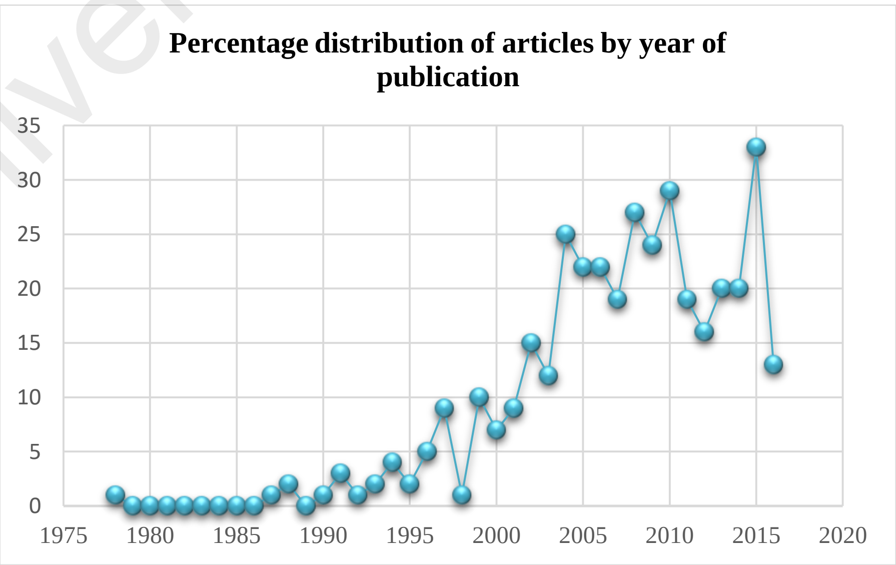


Figure 2.27: Percentage Distribution of Articles by Year Of Publication

Further analysis of the risk-related terms that appeared in the title of the articles shows that the terms risk monitoring (94 articles), risk management (72 articles), software risk / hazard (49 articles), and risk assessment (28 articles) reflect the top four topics, respectively, in the articles reviewed. It also shows that risk monitoring and risk management are the two main topics that have gained serious attention in recent years. Figure 2.28 shows the top four highest number (in descending order) of articles with the respective terms – software risk/hazard (124 articles), risk management (97 articles), risk factor (60 articles), and risk assessment (40 articles), found in the abstract and keywords section. Again, this analysis shows that risk management is also ranked second among the research topics of the articles reviewed by the researchers. Also, research on fuzzy risk analysis is gaining popularity with this term found in the abstract and keyword sections of 36 articles. Distribution of risk-related terminology found in the abstract and keywords sections of articles are shown in Figure 2.29.

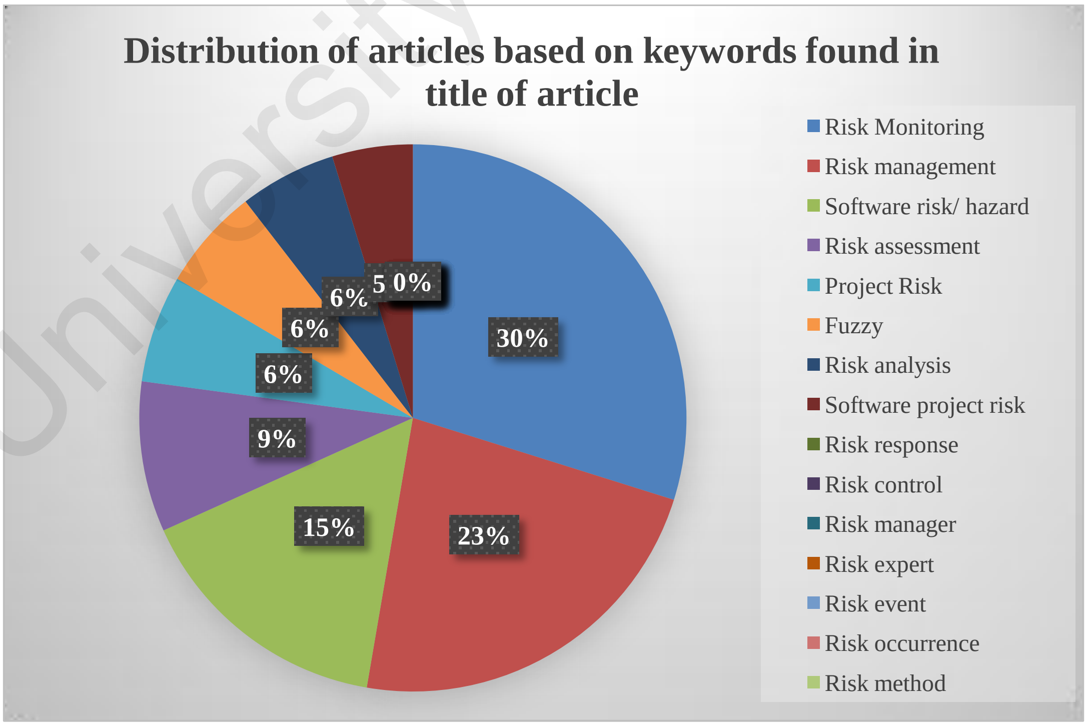


Figure 2.28: Distribution of Articles Based on Keywords Found in Title of Article

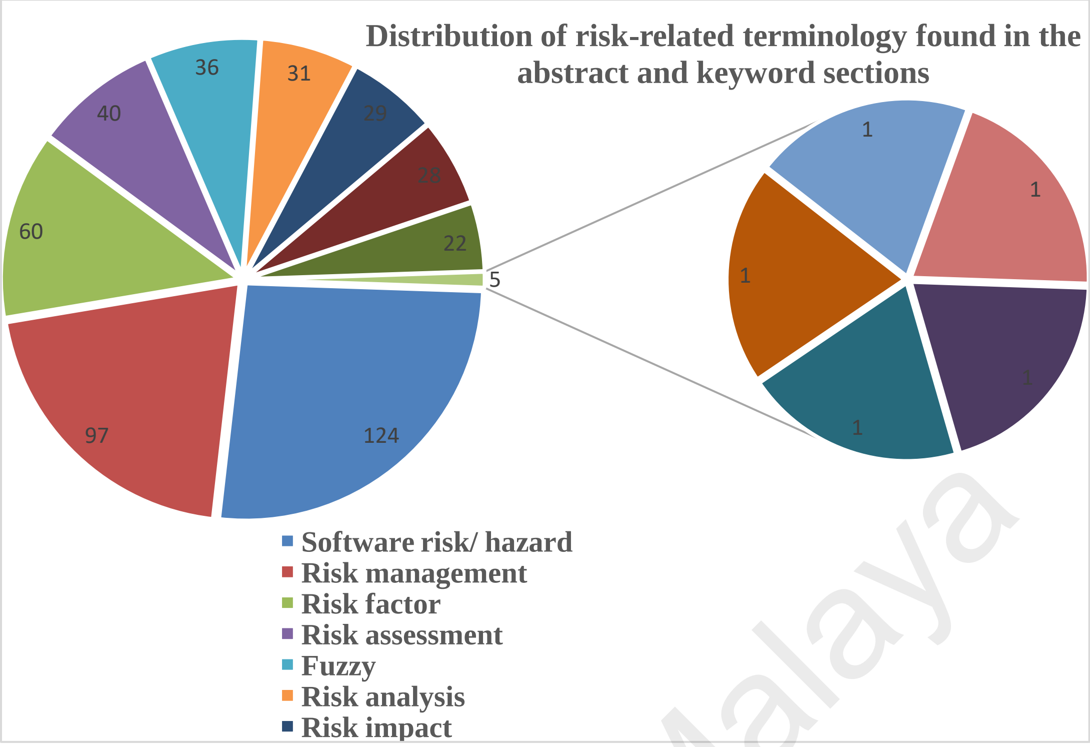


Figure 2.29: Distribution of Risk-Related Terminology Found in the Abstract and Keywords Sections of Articles

Figure 2.30 shows the focus of the content (full text) of all the papers reviewed. The highest frequencies were covered by risk management, software risk/hazard, risk assessment, risk analysis, Information Technology and Project Risk and risk identification, each with 268, 226, 180, 178, 162, 133, and 127 papers, respectively. The lowest frequencies were recorded by risk method, risk correlation, risk analyser/risk analyst/risk personnel and risk expert with 2, 2, 4, and 4 papers, each, respectively. Risk method was used only twice, indicating the little interest on the methodological aspect of risk.

The role of experts in risk management was only mentioned in six papers, which mainly focused on ranking the risks within a given project using fuzzy techniques. In fact, there has very few references to the nature and roles of risk analysts, even neglecting to investigate how subject matter experts (SMEs) can contribute to the success of the risk assessment process.

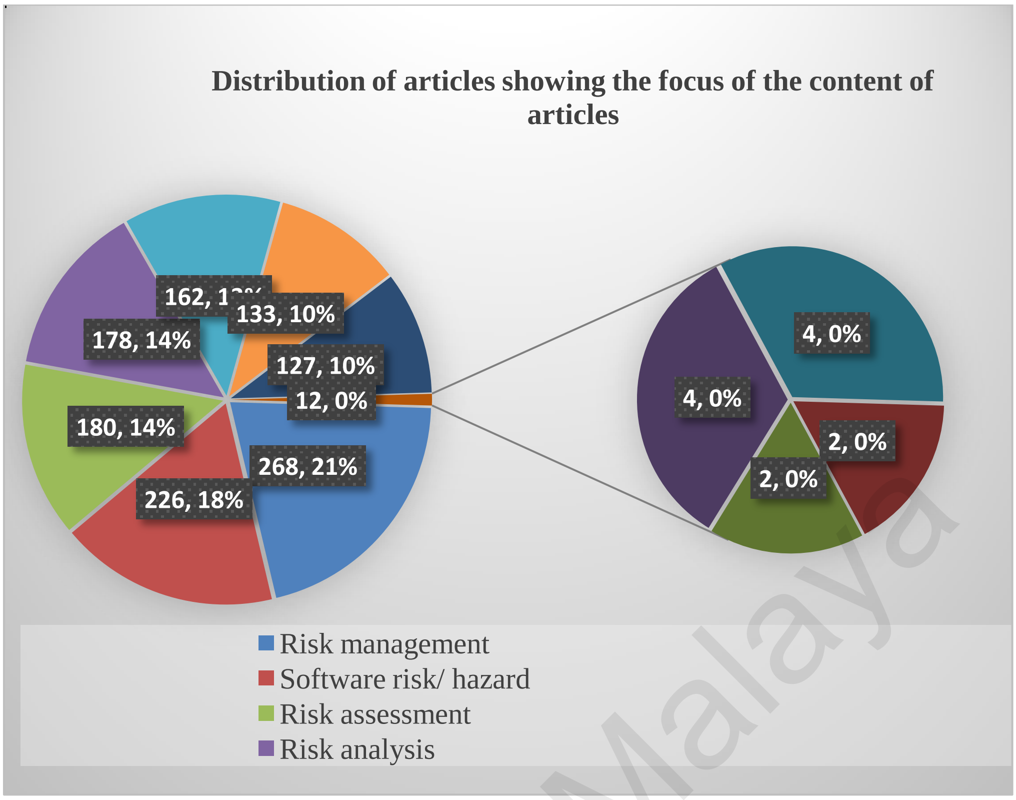


Figure 2.30: Distribution of Articles Showing the Focus of the Content of Articles (Full Text)

2.11.2 Challenges and Weaknesses of Current Risk Management Models

Boehm and DeMarco (1997), Lazzerini and Mkrtchyan (2011) and several researchers reviewed the relevant references and resources and reported some of the shortcomings and other pertinent issues to consider in any proposed risk management model, as follows:

i) Need for better collaboration: Reifer (2006) reported that project risk management has not been particularly emphasised in the workspace. Nevertheless, it is crucial to provide a virtual environment for the risk teams to collaborate (Gorla, Somers, & Wong, 2010). Besides hosting any relevant portals, it is essential to have an effective mechanism for collaboration within virtual environments (Ostvold & Jorgensen, 2005).

ii) Lack of integrated tools: Most of the existing models have overlooked the risks that occur following the completion of a project. This problem should be resolved by implementing integrated methods. Presently, some of the ad hoc methods and portals are operated by the same managers (Rabbi & Mannan, 2008).

iii) Busy managers: Risk management is handled by a project manager who is usually very busy attending to other tasks, and as a result, project risks are not properly managed. Risk managers rely on their personal experiences, as compared to experts, who have gained a far broader experience in similar projects (Verner, Sampson, & Cerpa, 2008).

iv) Project transition**:** Risk managers are substituted from one project to the next, thus adversely affecting the risk management process. Nevertheless, the independence of risk management linked with multiple projects can help the transfer of useful risk-related experiences. Such transfer can be facilitated through the use of appropriate online tools (Wanderley, Menezes Jr, Gusmãoa, & Lima, 2015).

v) Lack of measurement metrics: To evaluate the efficiency of project cost estimation in software development, suitable metrics and tools should be developed (Ammar, Nikzadeh, & Dugan, 2001).

vi) Incorrect risk factors: It is difficult to differentiate between a risk and a risk factor and this is the most common problem in software risk management models. The complexity of current approaches to clearly differentiate risk and risk factor makes their usage difficult, causes inaccurate analysis, and can even add new risks to the current risks of the project (Lazzerini & Mkrtchyan, 2011).

vii) Improper transform: One of the problems in software risk management is that information gathered for risk estimation is often qualitative in nature. It is very difficult for the risk analysts to convert the qualitative terms such as low, medium, and high into corresponding quantitative values.

viii) Unfamiliar with real-world necessities: Most of the risk management models neglect the real-world project necessities such as human resource requirement, and the efforts needed (Dedolph, 2003; Zowghi & Nurmuliani, 2002).

ix) Immatured process: Project Management Institute (PMI) reported that risk management is not mature yet because it is the least practised discipline compared with other aspects of project management practices on software-related issues (Charette, 2015; Rivard, St-James, & Cameron 2011).

x) Risk models are too general: The current software risk models are too generalised for use in real-world projects (Pandian, 2006, Sauer, Gemino, & Reich, 2007).

2.12 Summary

Risk management is an important process in software project management. A large number of completed software projects have resulted in schedule and budget over-runs, far exceeding the planned milestones. Researches have shown that incorrect estimation, project cost and schedule, incomplete and incorrect requirements specifications, and adopting the wrong methodologies are among the serious risks that can adversely affect a software development project.

Hence, it is crucial to conduct risk identification and to identify the various types of software project risks correctly in order to reduce the rate of project failures. In addition, the ability to prioritise the risks according to severity level and probability of occurrence of risks will further reduce the rate of software project failures. Thus, researches on these aspects in relation to the current project needs, development technology and that operating environment are crucial to ensure high rate of software project success.

CHAPTER 3: RESEARCH METHOD

This chapter describes the method used to carry out the research, and also explains the research activities and the research design. Two hypotheses were formulated and tested in this research. The last section of the chapter presents the internal and external validity of the research.

3.1 Research Methodology

The research is aimed at proposing an enhanced risk identification and risk assessment model for software risk management. The model is designed based on two important activities of risk management – risk identification and risk assessment. The performance of the proposed model was evaluated using data collected from 40 medium-sized commercial projects (20 projects each from Department P and Q, respectively) of a software company (See Appendix A). Figure 3.1 shows the sequence of activities involved in this research.

The first activity is the comprehensive review of related publications on risk management. Based on the information gathered from the literature review, the shortcomings in the existing risk management models were identified. Hence, an enhanced software risk management model is proposed to address these shortcomings. The Rapid Application Development (RAD) methodology is used for designing and developing a tool to facilitate the features and functions of the proposed model. The tool is used to record, analyse and generate reports using risk data collected during the risk identification and risk assessment processes. The following sections explain each of the research activities.

3.1.1 First Activity: Conduct Literature Review

Risk management is one of the most important activities in ensuring the success of software development projects (PMBOK, 5th ed.: PMI, 2013). From the literature review, it is found that poor risk management is one of the main causes of software project failure. A total of 340 research articles pertaining to software project risk management, and risk models proposed by researchers to overcome some of the software risk management problems, were reviewed. These articles include 220 academic journal articles, 98 papers in conference proceedings, and 10 standards on software risk management, published between 2009 and 2016. The literature review is presented in Chapter 2.

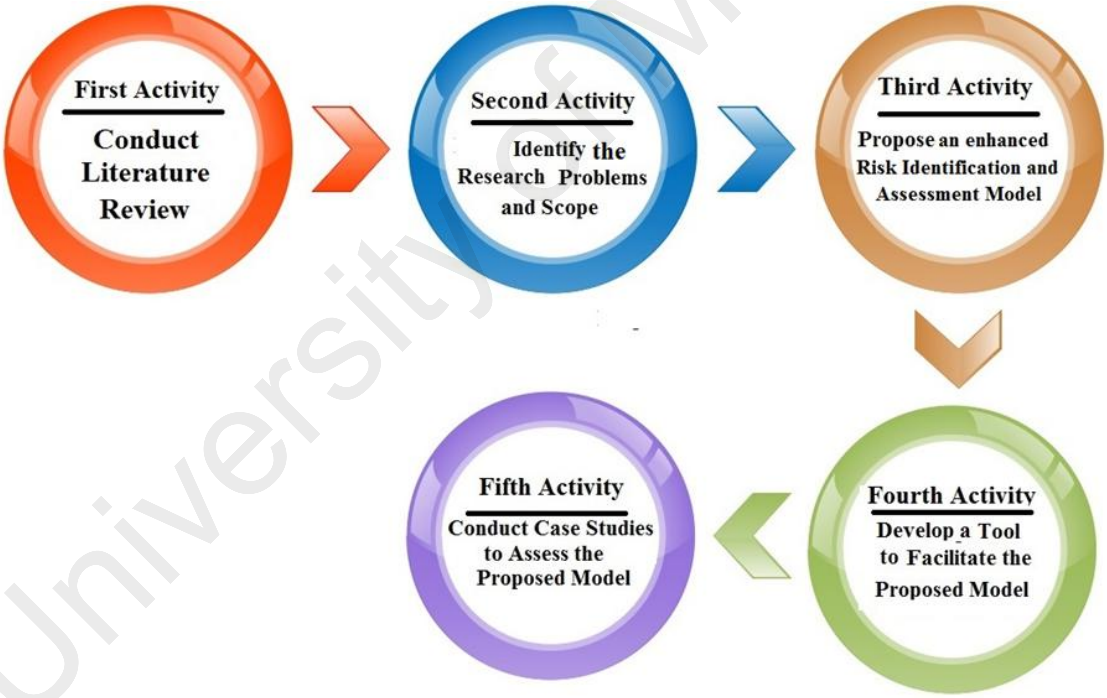

First Activity
Conduct Literature Review
Second Activity
Identify the Research Problems and Scope
Third Activity
Propose an enhanced Risk Identification and Assessment Model
Fourth Activity
Develop a Tool to Facilitate the Proposed Model
Fifth Activity
Conduct Case Studies to Assess the Proposed Model

Figure 3.1: Research Activities

3.1.2 Second Activity: Identify the Research Problems and Scope

The current risk identification methods are not effective because of the lack of a comprehensive risk classification system. The use of external experts are recommended in many of the studies, but there is no clear structure or framework to apply. Moreover, there is no recommendation for combining the experiences of internal risk analysts who are familiar with the specific needs of a company, and the rich experience of external risk analysts who are familiar with similar projects in many previous cases they have handled. The line of communication between the internal and external risk analysts in verifying the initial identified risks are also not addressed. In risk assessment, the current methods of getting the risk analysts' opinions is through surveys, which means they have to reveal confidential information of the company, and the projects with others. Moreover, getting the opinion of experts through the survey is a one-time activity, and is usually not repeated because of the security and privacy concerns of a company, while risk management is a continuous process. The proposed model is easily applicable, and external experts involved in the process have access to the projects documents, which have been analysed by internal analysts, as well as the results of their assessment. The existing risk identification and risk assessment models do not recommend for combining the opinions of internal and external risk experts. The framework and structure to create the right environment is recommended, but has yet to be implemented.

The research is confined to the following scope: i) covers risk identification and risk assessment, the two important phases of risk management; risk response is not covered in this research, but implementing an effective risk assessment will improve the risk response process; ii) covers risk management in commercial software systems; iii) only 20 risks are considered in each development phase, and 20 common risks are considered for each project; iv) each risk management team in the case study comprises five risk analysts; v) three experienced risk experts are engaged from an external software company to verify the risks identified and assessed by the two internal risk teams (RT3 and RT4). Using these three external experts could be a constraint and delimitation of this research, but the number of experts could be changed in a separate study to investigate whether the research outcome could be different with the involvement of more than three external risk experts.

3.1.3 Third Activity: Propose an Enhanced Risk Identification and Risk Assessment Model

To resolve the problems identified, an Enhanced Risk Identification and Assessment Model (E-RIAM) was proposed. E-RIAM incorporates five main enhancements: i) identifies a maximum of 20 risks (major and moderate) for each software development phase; ii) identifies a maximum of 20 common risks (major and moderate) for the entire project; iii) prepares a list of potential (major and moderate) software risks of each development phase and a list of common (major and moderate) risks that are pertinent to the entire project; iv) provides a risk database to store the potential, most serious, and common risks in the four software development phases (requirements analysis, design, programming and testing, and implementation and release), which can be used as a reference by the risk analysts during the risk identification process; and v) introduces an experienced risk team, known as Dynamic Verifier Core (DVC) team, to verify the list of risks that had been identified and assessed by the internal risk analysts. The DVC team also assesses the software project risks by considering their impact and likelihood of occurrence. The potential risks to a software project are ranked based on their Risk Value (RV), which is calculated using the formula: Likelihood * Impact. Chapter 4 explains E-RIAM in greater detail.

3.1.4 Fourth Activity: Develop a Tool to Facilitate the Features of the Proposed Model

To facilitate the risk identification and risk assessment processes, a support tool, Res-DVC, was developed using RAD development techniques, ASP programming language, and SQL-Server as the database management system. This tool records the different classes of risks according to the software development phases, and the outcomes of risk identification and risk assessment processes. Seven main databases are created to store the project data and various risk data in the system. These databases include: i) Project Details Database – to store the details of each project; ii) Risk Database – to store the potential risks to the four phases of software development (requirements analysis, design, programming and testing, and implementation); iii) Risk Identification Database – to store risks that have been identified by the risk teams during the risk identification process; iv) Risk Assessment Database – to store the values of the likelihood of occurrence (L) of each risk, using the scale ranging from 1 to 5, the impact (I) (consequence) of each risk if it materialises using the scale ranging from 1 to 5, the risk value (RV) of each risk which is calculated using the formula L×I, and the classification of each RV into the three categories – major, moderate, and minor risk – during the risk assessment process; v) Communication Log Database – to store the communication between RC and risk analysts; vi) Risk Team and DVC Team Database – to store the details of risk analyst and risk experts of the risk teams and DVC teams; vii) Verification Database – saves the verification process details of DVC teams. The tool generates various reports which include: i) lists of potential risks and the possible causes and effects of each risk; ii) values of the likelihood of occurrence (L), the risk impact (I), and risk value (RV) of each risk; and iii) risks severity category – major, moderate, or minor risk. Reports are generated for all the four phases of software development.

3.1.5 Fifth Activity: Conduct Case Studies to Assess the Proposed Model

Case studies were conducted to evaluate E-RIAM. Case studies investigate phenomena in real-world settings, for example, new technologies, communication in global software development, project risk and failure factors (Salo, 2004). Case study is a dominant research method within software engineering (Runeson et.al, 2009).

The case study method was chosen because: i) a risk coordinator (RC) was given explanation on how to conduct the case study systematically; ii) all the risk teams involved in the case studies are guided by the RC; iii) the case studies are based on past software projects that contain the relevant risk data needed (i.e., list of risks that had materialised, the risk values of all the risks that had materialised, and the classification of the risks into major, moderate or minor risk for each project) for comparison; and iv) the results of the case studies could be compared with the actual risk identification and risk assessment outcomes provided by the company (i.e., the data mentioned in iii). In this research, commercial software projects were used. The risk teams involved were required to:

i. identify 20 potential major and moderate risks for each software development phase and assess these 20 risks to determine the RV;
ii. identify 20 potential common major and moderate risks for the entire project and assess these 20 risks to determine the RV.

In software risk management, software companies must address the major and moderate risks, as they will impose serious impact if these risks are materialised. As stated in Chapter 1, this research aims to determine if identifying 20 potential major and moderate risks is sufficient for each of the software development phases, as well as 20 common major and moderate risks for the entire project. As mentioned in Chapter 1, risk response and control are not considered in this research.

Twenty-five software companies were invited to participate in the case study. However, only one software company agreed to participate, and was willing to provide information on 40 past medium-sized projects together with the related risks information.

As one of the objectives of this research is to determine whether the enhanced risk model (E-RIAM) can improve the risk identification and risk assessment processes, a balanced block design was used to form the control and treatment groups in the two case studies.

The 40 medium-sized projects of the software company consisted of 20 projects each from Department P and Department Q, respectively. Four risk management teams - risk teams RT1, RT2, RT3, and RT4 – were formed. Each risk team consists of five risk analysts with between 10 to 20 years of experience in software risk management. Another two risk teams, known as Dynamic Verifier Core (DVC-I and DVC-II) comprising three external risk experts, all with more than 20 years of experience in software risk management, were engaged to verify the risks that had been identified and assessed by risk teams RT3 and RT4. Table 3.1 shows the complete balanced block design of the two case studies. The complete balanced block design depicted in this section shows controlling of the bias in the research. The number of case studies, number of projects divided in two departments and similar size of projects, together with the number of risk teams and analysts indicate a structure balance in the research design to eliminate the various types of bias in this research, explained in detail in chapter 6.

Table 3.1. Balanced block design of the two case studies

Case Study	Department	No. of Medium-Sized Projects	Risk Team ID (Control Group)	Risk Team ID (Treatment Group)
1	P	20	RT1	DVC1 (RT3+DVC-I)
2	Q	20	RT2	DVC2 (RT4+DVC-II)

Keys:

RT1: Risk Team comprising five risk analysts from Department P.

RT3: Risk Team comprising five risk analysts from Department P.

RT2: Risk Team comprising five risk analysts from Department Q.

RT4: Risk Team comprising five risk analysts from Department Q.

DVC-I: Risk Team comprising three risk analysts from external company.

DVC-II: Risk Team comprising three risk analysts from external company.

RT1 and RT2 are the control groups (i.e., no treatment was assigned to these two teams). DVC-I and DVC-II are the treatment teams which will verify (i.e., treatment) the risks identified and assessed by RT3 and RT4, respectively. In the case study, teams RT1 and DVC1; and teams RT2 and DVC2 worked in pairs and in parallel. Hence, in the research, E-RIAM is the entity, and the attribute of the entity is the efficiency of E-RIAM, which is also the dependent variable. The team structure, without and with the intervention of the DVC team, i.e., (RT1, RT2) and (DVC1, DVC2), respectively, are the independent variables.

Teams RT1 and DVC1 (i.e., RT3 + DVC-I) were assigned to identify and assess 20 medium-sized projects of Department P. Similarly, teams RT2 and DVC2 (i.e., RT4 + DVC-II) were assigned to assess the risks of 20 medium-sized projects of Department Q. Details of the risk management teams, the case studies and their results are presented in Chapter 5. The research design of the two case studies is shown in Figure 3.2. All the risk teams, RT1, RT2, RT3, and RT4, identified and assessed the software risks using the ISO 31000/31010:2009 Standards and an Excel spreadsheet tool. Team DVC-I and Team DVC-II verified the risks identified and assessed by RT3, and RT4, respectively, using the proposed support tool, Res-DVC.

Each risk team (RT1, RT2, RT3, and RT4) identified and assessed 20 risks of the four development phases – requirements analysis, design, programming and testing, and implementation and release phases. Besides identifying the 20 risks for each development phase, another 20 common risks for the entire project (EnP) that could impact on each of the 20 software projects, were also identified and assessed, respectively, by all the four risk teams.

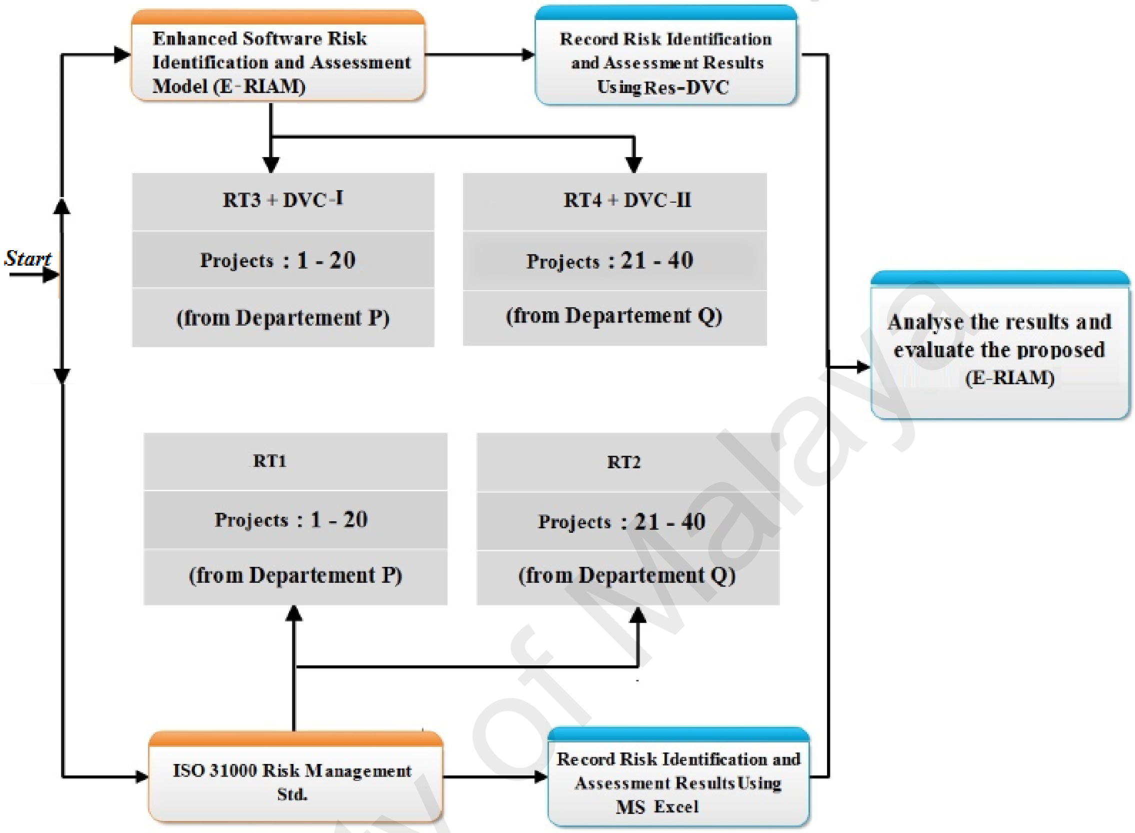

Enhanced Software Risk Identification and Assessment Model (E-RIAM)
Record Risk Identification and Assessment Results Using Res-DVC
RT3 + DVC-I
Projects : 1 - 20
(from Departement P)
RT4 + DVC-II
Projects : 21 - 40
(from Departement Q)
Start
Analyse the results and evaluate the proposed (E-RIAM)
RT1
Projects : 1 - 20
(from Departement P)
RT2
Projects : 21 - 40
(from Departement Q)
ISO 31000 Risk Management Std.
Record Risk Identification and Assessment Results Using MS Excel

Figure 3.2: Research Design of the Case Study

Team RT1 and Team RT2 identified and assessed software risks using the steps that are commonly used in the general risk management model based on the ISO 31000/31010: 2009 Standard (Purdy, 2010). The five risk analysts in teams RT1 and RT2 are from Department P and Department Q, respectively, of the company. These analysts were not aware of their involvement in the two case studies. This is aimed at preventing the risk analysts from attempting to identify more software risks, which might be irrelevant risks, in order to show that they are highly competent, if they were to know that they are involved in the research, and that they are being assessed on their risk identification and assessment capabilities. Team DVC1 and Team DVC2 are used to represent the team structure comprising RT3 and DVC-I, and RT4 and DVC-II, respectively. Risk teams

RT3 and RT4 consist of five risk analysts each who possess between 10 to 20 years of experience in risk management, respectively, similar to the five risk analysts each of risk teams RT1 and RT2. On the other hand, teams DVC-I and DVC-II consist of three risk experts each who were engaged from an external software company to verify the risks identified and assessed by RT3 and RT4, respectively. They possess more than 20 years of experience in software risk management. The use of experienced external experts was aimed at preventing bias in the case studies, which would happen if the risk experts who verified the risk identified and assessed by risk teams RT3 and RT4 were from the same department as RT3 and RT4, respectively. The five risk analysts in each risk teams RT3 and RT4 are from Departments P and Q, respectively. Again, risk teams RT3, RT4, DVC-I and DVC-II were not aware of their involvement in this research in order to ensure validity and reliability of the research outcome.

The data collected from the two case studies include: i) the 20 risks identified at each development phase, and 20 common risks of each software project by the four risk teams; ii) the likelihood of risk occurrence and the level of impact of each risk assessed by the four risk teams as well as the risk values verified by the two DVC teams. The data were compiled, analysed and compared with the actual list of risks that had materilaised, and their respective risk values for the 20 software projects from Departments P and Q, respectively. Two statistical hypotheses were formulated and appropriate statistical tests were used to prove these hypotheses (Chapter 5).

3.2 Metrics used to Determine Achievement of Research Objectives

As mentioned in Chapter 1, there are three main research objectives: i) to determine whether the identification of a maximum of 20 risks is sufficient in each development phase of a software project; ii) to determine whether the identification of a maximum of 20 common risks is sufficient for the entire software project; iii) to propose an

Enhanced Risk Identification and Assessment Model (E-RIAM). To achieve this research objective (iii), statistical method and measurement are used, as described in Chapter 5; iv) to evaluate whether E-RIAM can improve accuracy of the risk identification and risk assessment processes. To achieve objective (iv), two hypotheses are formulated and explained below.

Hypothesis 1

H0: The efficiency of E-RIAM in risk identification is the same as the efficiency of the generic risk identification process model.

H1: The efficiency of E-RIAM in risk identification is higher than the efficiency of generic risk identification process model.

Hypothesis 2

H0: The efficiency of E-RIAM in risk assessment is the same as the efficiency of the generic risk assessment process model.

H1: The efficiency of E-RIAM in risk assessment is higher than the efficiency of the generic risk assessment process model.

3.2.1 Test of Hypotheses

To test Hypothesis 1, the efficiency of the risk identification process is first calculated using Formula 1, and then tested using appropriate statistical test depending on the distribution of data of the datasets of Risk Identification Efficiency (RIE) (Chapter 5).

Risk Identification Efficiency (RIE) = No. of risks that had identified and materialised / Actual total no. of risks that had materialised (data from the company) x 100% (1)

To test Hypothesis 2, the risk value (RV) of each risk is first calculated by Formula 2:

Risk Value (RV) = Likelihood (Probability) of occurrence (L) x Impact (I)   (2)

The likelihood of occurrence and impact (consequence) of each risk are assessed using a scale ranging from 1 to 5, where 1 represents low likelihood of occurrence and low impact if the risk were to materialise; 5 represents high likelihood of occurrence and high impact if the risk were to materialise. Hence, the lowest RV is 1 (i.e., 1 x 1) and the highest RV is 25 (i.e., 5 x 5). Each of the RV is then classified into one of three categories – major, moderate or minor risk, based on the suggestion of Larson, and Gray (2014). Table 3.1 shows the classification of the RV. The red zone, yellow zone, and green zone indicate major, moderate, and minor risks, respectively. It is obvious that there are four sets of RVs, which fall in two different zones - 4 and 6 fall in both the green and yellow zones; 10 and 15 fall in both the yellow and red zones. In these conflicting situations, the classification of the risk will be determined based on the impact value. For example, if the impact is very high (i.e., scale of 5), the RV will fall in the yellow zone, otherwise, it will fall in the green zone (i.e., likelihood is scale of 5). Similarly, after the RVs have been calculated, they are tested using appropriate statistical test, as explained in Chapter 5.

Table 3.1: The risk value matrix (Larson & Gray, 2014)

Likelihood	**Risk Value = Likelihood x Impact**				
5 (Very likely)	**5**	**10**	**15**	**20**	**25**
4 (likely)	**4**	**8**	**12**	**16**	**20**
3 (Possible)	**3**	**6**	**9**	**12**	**15**
2 (Unlikely)	**2**	**4**	**6**	**8**	**10**
1 (Very Unlikely)	**1**	**2**	**3**	**4**	**5**
Impact	**1 (Very Low)**	**2 (Low)**	**3 (Medium)**	**4 (High)**	**5 (Very High)**

Key: Green zone: Minor, Yellow zone: Moderate, Red Zone: Major

3.3 Validity of Research

Every academic study should be subjected to a validity assessment to ensure validity of the research (Creswell, 2012). In this research, four important validity measures were employed.

3.3.1 Statistical Validity

Statistical validity is indicative that the dependent variable (efficiency of E-RIAM) is reliable as the number of risks identified and assessed are correctly and systematically recorded either using MS Excel or using Res-DVC, respectively. It is also indicative that correct formulas have been used to calculate the risk identification efficiency (RIE) and the risk values (RVs). Conflicting RVs are resolved consistently based on the impact value of each risk. The statistical tests used to prove the hypotheses were selected based on the advice of a statistician. The underlying assumptions of the statistical test – test on the data distribution normality – were tested to decide whether the parametric or non-parametric tests should be used to prove the hypotheses. The only statistical error that cannot be avoided in this research is the rejection of the null hypothesis incorrectly, in the statistical decision, α, which is set at 0.05.

3.3.2 Construct Validity

Construct validity concerns the extent to which a test is measuring what it claims, or purports, to be measuring (Cronbach & Meehl, 1955). This research aims to determine whether E-RIAM can improve the efficiency of the risk identification and risk assessment processes. Hence, assigning the same set of projects are assigned to two independent risk teams with similar level of risk management experience – one risk team without intervention of the DVC team (control group), and another risk team with intervention of the DVC team (treatment group, i.e., risk analysts are engaged from external company with similar number of years of work experience in risk management). Also, the risk identification and risk assessment results are compared with the actual total number of risks that had materialised and the actual risk values determined by the software company is a logical and right approach to adopt to achieve the objective (Cho, 2006). The metrics used, i.e., the number of risks that had materialised, and the RVs of the risks that are used to measure improvement of the risk identification and risk assessment processes, are very closely related to the efficiency of the processes. Hence, these measurements are the most appropriate for comparing and evaluating the two processes. Moreover, all the risk analysts involved (risk analysts of the four risk teams and the two DVC teams) did not know that they were selected or engaged to participate in the study (i.e., a blind study). They also did not know the actual total number of risks in each project. The risk identification and assessment processes were conducted simultaneously, and this will strengthen the construct validity of this research.

3.3.3 External Validity

External validity is indicative of whether the results of a study can be successfully generalised to other projects, studies, time, places, and participants (Creswell, 2012). To test the hypotheses and conduct the case studies, 40 similar medium-sized commercial projects were randomly selected from two departments of a software company. The same number of projects, i.e., 20 projects each, were selected for both case studies. All selected projects are assigned for each model E-RIAM, or ISO 31000/31010:2009. In each case study, 20 projects were considered for risk identification and risk assessment using both models by the risk control team and the risk treatment team. The risks of all projects were identified and assessed using both models (E-RIAM and ISO 31000/31010:2009), as well.

3.3.4 Internal validity

Internal validity concerns the extent to which a causal conclusion based on a study is warranted. It concerns the independent variable, and not some extraneous variables that can change the dependent variable (Graziano & Raulin, 2014).

Although the selected projects for the case studies were developed completely by different software development teams, they were all classified as medium-sized commercial projects, and were developed on homogenous platforms.

In addition, similar documents and candidate risks were presented to all teams. The project sizes were also similar as all projects were classified as medium-scale projects. Moreover, all the projects were developed in one company and under the same organizational environment, and thus, many risks share similar potential. The control team members for both case studies were selected from among the company’s employees who possess similar capabilities, and have similar set of skills and experience. Candidate risks were also provided to both groups working with the proposed model as well as ISO 31000/31010:2009.

3.4 Summary

This chapter describes the research method used in this thesis. The research method includes a review of the literature pertaining to risk management in software projects. An enhanced risk management model (E-RIAM) was proposed to address the main weaknesses of the existing models, and a support tool, Res-DVC, was designed to facilitate the risk identification and assessment processes. Forty medium-sized projects developed in two departments of a software company were used in two case studies to evaluate the efficiency of the E-RIAM model and compared it with the ISO 31000/31010:2009 standard. In assessing the enhanced risk identification and risk assessment model, a number of risks identified in each phase, and for the entire project and the risk values (RVs) were used as the main metrics to test the two hypotheses formulated for this research. Appropriate statistical tests were used to test the hypotheses. Finally, the research validity was confirmed using four validity measures.

CHAPTER 4: THE PROPOSED RISK IDENTIFICATION AND ASSESSMENT MODEL

This chapter presents the proposed Risk Identification and Assessment Model (E-RIAM) for software risk management. The model incorporates four new features: (i) classification of 20 specific risks for each software development phase, and 20 common software project risks; (ii) an enhanced risk management team structure, which also describes the responsibilities of the risk team members; (iii) an additional risk team known as Dynamic Verifier Core (DVC) to help verify the software risks; and (iv) enhancement to the existing risk identification and assessment process. These features are described in detail in the following sections.

4.1 Classification of Specific and Common Software Project Risks

Based on the literature review, failure to identify the potential risks that have high probability of occurrence and impact is one of the causes of software project failure (Nguyen, 2014; Rekha & Parvathi, 2015; Olteanu & Gheorghe, 2016). Hence, in this research, a list of specific risks that have high probability of occurrence and high impact were compiled based on information obtained from the literature review. These risks are classified according to the four software development phases – requirements, design, programming and testing, and implementation and release phases – and stored in a Risk Database (explained in section 4.5). Another list of common software project risks are also classified and stored in the Risk Database.

During the risk identification phase, the risk analysts can select the top 20 specific risks for each development phase and the top 20 common risks for the entire project from the Risk Database. Similarly, the DVC team can use this Risk Database to verify the risks identified by the risk team. This risks database not only helps in expediting and facilitating the risk identification process, but also in selecting the most likely-to-occur and high impact risks in the software project.

In this research, the decision for selecting 20 specific risks for each development phase and 20 common risks for the entire project, was made based on information gathered from the literature review (Ferguson (2004), Sonchan (2014), Samantra et al. (2016), and Elzamly, Hussin, & Salleh, 2016). Based on these studies, 20 risks are sufficient for each development phase and for the entire project, respectively, so that efforts can be prioritised to manage the major risks and appropriate risk response measures and risk control can be implemented as the potential cause(s) and effect(s) of each risk are also identified in this research. Appendix B shows the sample list of specific risks together with the causes and effects of each risk, for each development phase, and the common risks for the entire project.

4.2 Risk Management Team Structure and Responsibilities

A software risk management team, also known as the risk team in this research, consists of risk analysts who are able to identify the various types of software project risks, assess the risks, propose contingency plans and initiate control measures to respond to, and control those risks if they materialised. In addition, they are also familiar with different software development platforms, and the use of various enabling technologies and development environments. Although the number of risk analysts in each risk team depends on the size of a software project, each risk team usually consists of three to seven risk analysts, based on information gathered from the literature review (Kondabagil, 2007; Fulkerson et al., 2015).

In this research, each risk team consists of five risk analysts under the supervision of a risk coordinator (RC), also known as the supervisor. All the risk analysts involved in this research have 10 to 20 years of work experience in risk management. They are able to undertake risk management tasks in all the four general phases of risk management – risk identification, risk assessment, risk response development, and risk response control. On the other hand, the RC plays a major role in managing the risk teams. He/she selects the risk analysts to form the risk teams, assigns tasks to each risk analyst, monitors and manages all issues pertaining to risk management, and resolves conflicts (if any) among the risk analysts. Besides, the RC also documents the risk data and the results of the risk identification and risk assessment processes, and prepares the risk reports throughout the risk management process.

The coordinator has specialist knowledge of the risk management process, and should possess the following attributes (Brown, 2014; Kremljak & Kafol, 2014) : i) wide experience in managing risks of software projects in recent years, ii) ability in team formation and management, and good communication skills, iii) proficient in using word processors and spreadsheets software, and iv) good data analysis skills to generate informative and analytical reports.

4.3 Additional Risk Team: The Dynamic Verifier Core (DVC) Team

In the proposed risk identification and risk assessment process model (E-RIAM), an additional risk team, known as Dynamic Verifier Core (DVC) is introduced at the risk identification and risk assessment phases. The DVC team consists of three external risk experts - risk experts who are not from the risk teams of the company – who have more than 20 years of experience in software risk management. This DVC team will serve as a "risk-keeper" to verify the risks identified by the internal risk teams of the company.

In this research, during the risk identification phase, the internal risk teams (i.e., RT3 and RT4) will first identify up to 20 risks that they consider most likely-to-occur and having the most severe impact if they were to materialise. These risk teams refer to the risk documents of similar projects in the past in order to identify up to 20 risks. The

DVC team (i.e., DVC-I and DVC-II) verifies these risks and makes changes - eliminates existing risk(s), adds new risk(s), and/or modifies existing risk(s) - based on its rich experience in risk management, and eventually identifies the top 20 high probability of occurrence and high impact risks. The list of risks is then distributed to the risk teams for assessment. During the assessment, the risk teams (i.e., RT3 and RT4) make their assessment based on two parameters: probability of occurrence and the severity of impact, and calculates the risk value for each risk, as described in Chapter 3. The risk value (RV) is calculated using the following equation:

Risk Value (RV) = Probability of risk occurrence x Severity of risk impact (1)

The top 20 risks are prioritised based on the RV.

Following the assessment, the DVC team will verify the values of the two parameters: probability of occurrence and severity of each risk, provided by the risk teams. The DVC team (i.e., DVC-I and DVC-II) will modify the rating of these two parameters for each risk should they find that the ratings assigned are not reasonable. Furthermore, the DVC team conducts a brainstorming session, i.e., an additional parameter, by having face-to-face meeting on the new RVs based on their judgment, to reflect the intensity of the negative impact/consequence of each risk on the software project. Many researchers have highlighted that the risk experts' views/decisions in risk analysis should be considered objectively, and not subjectively (Lauritzen & Parry, 2012; Jose & Winkler, 2009; Lin & Huang, 2012). Hence, the decisions on the ratings of the two parameters for each risk are made objectively based on team consensus, and in a face-to-face risk assessment setting.

The risk value for each risk is thus calculated based on the values of the two parameters: i) Likelihood of risk occurrence, and ii) Impact of risk, as follows:

Risk Value (RV) = Likelihood of risk occurrence x Impact of risk (2)

However, they suggested that the ranking of RV must be determined based on consensus. Thus, after obtaining the RV, the DVC team should meet to discuss and finalise the ranking of the risks. The decisions by the experts are accurate, backed up by their experiences in handling similar projects in the past.

4.4 The Proposed Risk Identification and Assessment Model (E-RIAM)

The proposed risk identification and risk assessment process model (E-RIAM) is aimed at avoiding and/or mitigating some software risks which could threaten the success of software projects. The model incorporates some enhancements to the existing risk identification and risk assessment processes (see Figure 4.1).

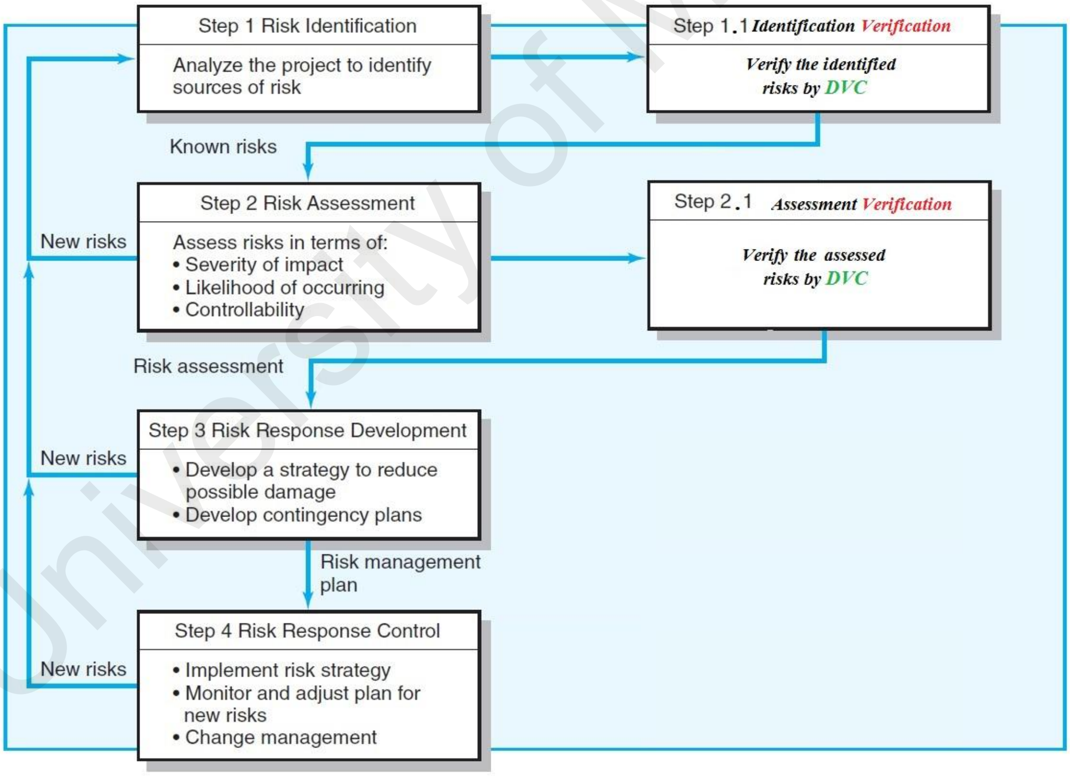
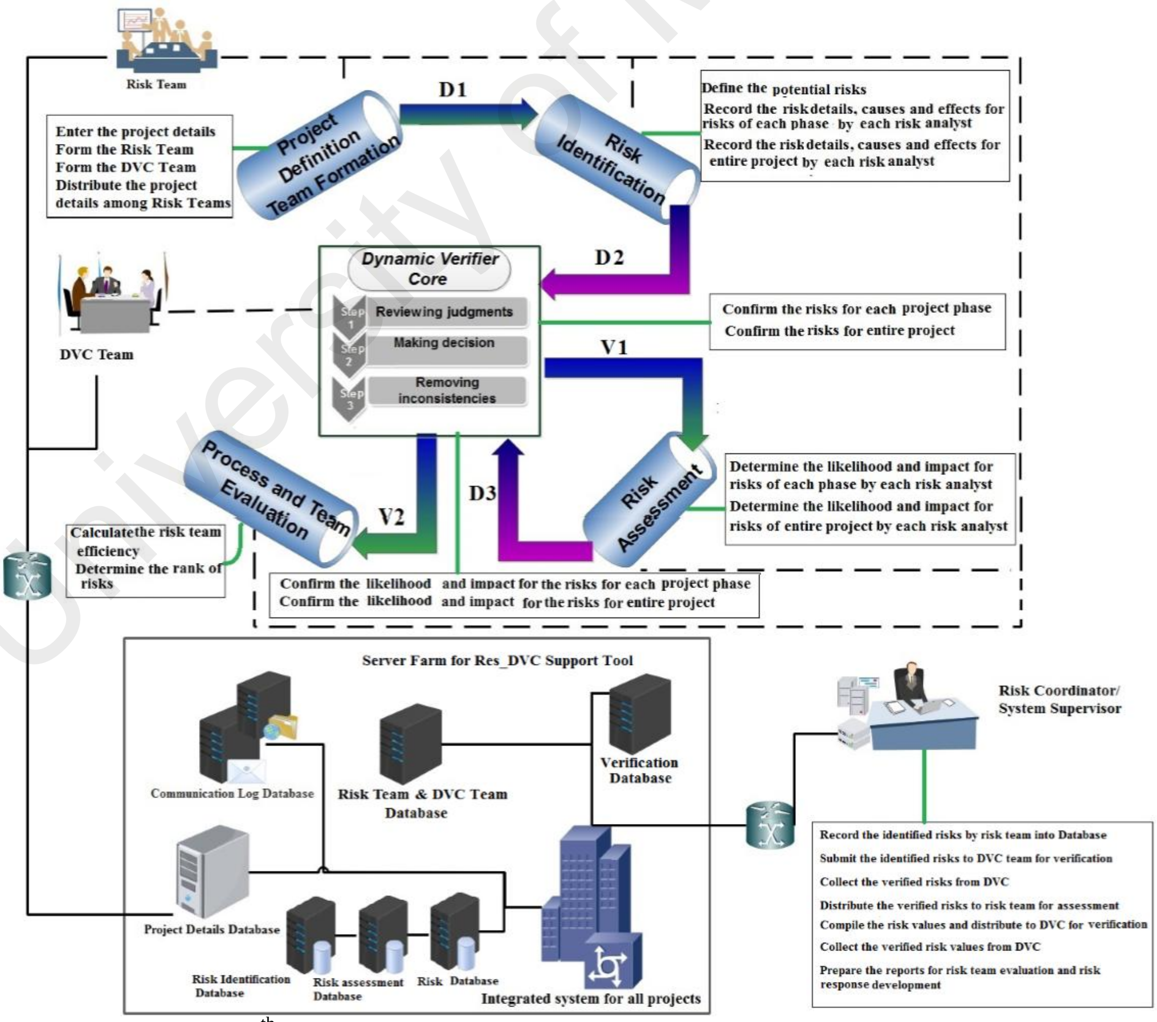

Step 1 Risk Identification
Analyze the project to identify sources of risk
Step 1.1 Identification Verification
Verify the identified risks by DVC
Known risks
Step 2 Risk Assessment
Assess risks in terms of:
• Severity of impact
• Likelihood of occurring
• Controllability
Step 2.1 Assessment Verification
Verify the assessed risks by DVC
New risks
Risk assessment
Step 3 Risk Response Development
• Develop a strategy to reduce possible damage
• Develop contingency plans
New risks
Risk management plan
Step 4 Risk Response Control
• Implement risk strategy
• Monitor and adjust plan for new risks
• Change management
New risks

Figure 4.1: Enhancements of E-RIAM compared to Larson and Gray (2014)

Figure 4.1 shows two verification steps (step 1.1 and step 2.1) that are added to the generic model of risk management as enhancements of E-RIAM. These two verification steps are depicted in detail together with the four main phases of the enhanced risk identification and risk assessment model, E-RIAM in Figure 4.2. The following section explains the four phases, the role of the Risk Coordinator (RC), and the Res-DVC support tool.

i) Phase 1: Project Definition and Formation of Risk Teams

In this phase, the risk coordinator (RC) of a large software company that has more than 100 employees (Slyngstad et. al 2008) is responsible for: the selection of risk analysts; formation of the risk team; defining the software project; assignment of software project and tasks to each of the risk team member; appointing external risk experts from other companies; and formation of the Dynamic Verifier Core (DVC) team. Besides, the RC also determines schedules of the risk management process.


Risk Team
Enter the project details
Form the Risk Team
Form the DVC Team
Distribute the project details among Risk Teams
Project Definition Team Formation
D1
Risk Identification
Define the potential risks
Record the risk details, causes and effects for risks of each phase by each risk analyst
Record the risk details, causes and effects for entire project by each risk analyst
DVC Team
Dynamic Verifier Core
Step 1
Reviewing judgments
Step 2
Making decision
Step 3
Removing inconsistencies
D2
Confirm the risks for each project phase
Confirm the risks for entire project
V1
Process and Team Evaluation
V2
D3
Risk Assessment
Determine the likelihood and impact for risks of each phase by each risk analyst
Determine the likelihood and impact for risks of entire project by each risk analyst
Calculatethe risk team efficiency
Determine the rank of risks
Confirm the likelihood and impact for the risks for each project phase
Confirm the likelihood and impact for the risks for entire project
Server Farm for Res_DVC Support Tool
Communication Log Database
Risk Team & DVC Team Database
Verification Database
Project Details Database
Risk Identification Database
Risk assessment Database
Risk Database
Integrated system for all projects
Risk Coordinator/ System Supervisor
Record the identified risks by risk team into Database
Submit the identified risks to DVC team for verification
Collect the verified risks from DVC
Distribute the verified risks to risk team for assessment
Compile the risk values and distribute to DVC for verification
Collect the verified risk values from DVC
Prepare the reports for risk team evaluation and risk response development

Keys: D_i : Dispatched i^{th}
V_j : Verified j^{th}

Figure 4.2: The E-RIAM Model

ii) Phase 2: Risk Identification

In this phase, the different types of risks that could occur and could have negative impact on a software project are identified by the risk teams. This identification process is iterated so that software project risks that are encouraged or materialised at any of the development phases can be identified as early as possible. The potential risks together with their cause(s) and effect(s) are first identified by each risk analyst and then discussed by all the risk team members.

Each risk analyst first selects up to 20 risks based on his/her past experience of similar projects, from the risk template of each phase. In this research, up to 20 risks are determined for a specific phase. Hence, there are altogether 80 risks for the four development phases (i.e., 20 risks x 4 phases = Maximum 80 risks). Also, another 20 common risks are determined for the entire project by the risk team. All the five risk analysts will determine up to 20 risks that are most likely-to-occur in each phase, and another 20 common risks that are most likely-to-occur in the entire project. All the risks (up to 80 specific risks and up to 20 common risks), are submitted by the risk team to the RC to be recorded into a database. The RC then submits these risks to the DVC team for verification.

During the verification process, the DVC team reviews the list of risks. The DVC team modifies the list of risks should they find that: some of the risks identified by the risk team have low probability of occurrence; the risk team had overlooked some other more severe and high probability of occurrence risks; or the risks are not correctly ranked. The DVC team confirms all the risks and returns them to the RC. The RC distributes the updated list of risks to the risk team for risk assessment, i.e., the third phase of risk management.

iii) Phase 3: Risk Assessment

In this phase, the risk team assesses the likelihood of risk occurrence and the impact of the risks, as determined by the DVC team. The assessments on the likelihood of risk occurrence and its impact (i.e., the severity of the consequences if a risk were to materialise), is made using a rating scale that ranges from 1 to 5, as explained below.

a) Likelihood of Risk Occurrence

Risk is an uncertain event, which may or may not materialise. A few standards have been introduced to guide the risk analysts in determining the likelihood of a risk occurrence. In E-RIAM, the likelihood of risk occurrence is rated using a scale ranges from 1 to 5, based on the ISO 31000 :2009 Standard.

In DOD 2009 Standards, likelihood of risk occurrence that is rated 5, indicates that the risk occurrence is more or less certain, although not completely certain. For instance, a software project that is far behind planned schedule, and is ending soon without provision for further extension, has a high likelihood (certain) of occurrence. If a risk will most probably occur, its likelihood of occurrence is rated 4. For example, having information about the history of similar risks or based on other evidences, a risk analyst can rate the risk 4. Rating 3 is assigned for risks that are equally likely to occur or not occur. No information regarding the number of times a risk has occurred may be available; however, there are previous records of its occurrence, in which case, it is rated 3. Rating of 2 indicates that the risks are not likely to occur. They have occurred in less than one-fifth of similar past projects. Lastly, a rating of 1 is assigned to risks where the probability of occurrence to less than five percent. Risk analysts’ experience in other projects or evidence that indicates little likelihood of threat from this kind of risks, can be rated 1. Table 4.1 shows a summary of the ratings on a scale rating from 1 to 5.

Table 4.1: Ratings for likelihood of risk occurrence

Rating	Characteristics
1	Risks with an occurrence probability of less than 5%.
2	Risks that are not likely to occur.
3	Risks that are equally likely to occur or not occur.
4	Risk that will most probably occur.
5	Risk occurrence is more or less certain (though not completely certain).

b) Risk Impact

Risk impact or consequence indicates the intensity of the adverse ramification of a risk in case it materialises. The impact of any risk occurrence can range from negligible to catastrophic. Immediate remedial measures must be taken to counter any catastrophic impact to a software project. E-RIAM adopts the five impact levels (Rank from 1 to 5) of the DOD 2006 standards, and ranking takes into consideration three important aspects of a software project – cost, schedule, and technology, as shown in Table 4.2.

In Table 4.2, Risk impact 1 (Rank 1) indicates ramifications that are tolerable, do not require preventive or management intervention, and can be handled as the project progresses, in case they occur. Risk impact 2 (Rank 2) indicates mild consequences that are negligible or might require only minimal management intervention. Risk impact 3 (Rank 3) indicates average ramifications that need to be monitored and managed. Risk impact 4 (Rank 4) indicates serious consequences that require constant preventive mitigation, and monitoring measures to pre-empt a catastrophic impact. Risk impact 5 (Rank 5) indicates very serious consequences that might lead to project failure or have very serious ramifications on the cost, personnel, security issues, and project delivery schedule.

Table 4.2: Impact levels and their relation to cost, schedule, and technology

Rank	Technology performance*	Cost**	Schedule***
1	No or negligible effect regarding the technology used.	Cost does not exceed the estimated cost in project design or additional cost is negligible.	No or negligible effect on the schedule.
2	Small impairment of performance i.e., exerts little effect on the project plan or it is tolerable.	Less than 1% of the cost, in addition to what was allocated in the budget.	Deviation from schedule but meet the main milestones.
3	Performance-related defects that slightly influence the main objectives of the project.	1% to 5% extra cost compared with what was allocated in the budget.	Schedule variations - schedule milestones may be changed or the approved milestones may be met by making little efforts.
4	Serious performance defects that could endanger project success or that are difficult to resolve.	5% to 10% extra cost compared with what was allocated in the budget.	Substantial variation in project schedule; critical path activities might not be completed on schedule.
5	Serious problems that prevent key performance parameters from being met.	More than 10% extra cost compared with what was allocated in the budget.	Serious problems that prevent the main milestones of the project from being met.

* The performance of the projects may be defined depending on the project type.

** This rating may be changed as the project managers deem necessary.

*** The milestones of each project are determined by the project managers.

c) Calculation of Risk Value

The risk value of each risk is calculated by multiplying the likelihood of occurrence, and the risk impact. It is rated using a scale ranging from 1 to 5. Hence, the most severe risk will have the value 25 (i.e., 5 x 5= 25), where 5 indicates the most severe impact and also most likelihood of risk occurrence, respectively; while the least severe risk will have the value 1 (i.e., 1 x 1 = 1), where 1 indicates the least severe impact, and also the least likelihood of risk occurrence, respectively (Larson & Gray, 2014).

This research is aimed at determining whether the enhancements made to the processes in the risk identification and risk assessment phases, and the involvement of a

DVC team, could help in identifying the top 20 most likely-to-occur and most severe risks for each software development phase, and for the entire software project, respectively. If these risks can be identified correctly, assessed accurately, avoided and/or mitigated, then a software project will have a better chance of success, and thus, avoiding/reducing the risk of project failure. Hence, the most likely-to-occur and severe impact risks are the most important risks to manage. The value of each risk that had been identified by the risk team is calculated using equation 1, as mentioned above:

$$\text{Risk Value} = \text{Likelihood of Occurrence (L)} \times \text{Severity of Impact (I)} \qquad (1)$$

The RC compiles the risk assessment values and distributes the list of values to the DVC team for verification. The DVC team will determine whether both the likelihood of occurrence and the impact of each risk have been assessed correctly, based on their experience, and on past records on risk assessment of similar software projects. As mentioned above, the DVC team can make a determination of each risk as to the severity of its impact on the project success. Hence, the risk value for each risk agreed the DVC team is calculated using Formula 1, as mentioned above. The results of the top 20 specific risks of each development phase and the top 20 common risks for the entire project are then reported to the RC, who then distributes them to the risk team for risk response development and risk control measures.

iv) Phase 4: Process and Team Evaluation

In this phase, the total numbers of risks that materialised in each development phase, and for the entire project, respectively are recorded. A risk evaluation report is produced and distributed to all the parties concerned. The results are compared to evaluate the efficiency of each risk team. In addition, the time taken by each risk team to identify and assess the risks for each development phase and for the entire project are recorded, and it can be used to calculate the productivity of each risk team, respectively.

Figure 4.3 shows the workflow of the proposed E-RIAM and the tasks performed by the risk personnel (i.e., the risk analysts and RC) in each step, and the outcomes/reports produced for the four risk phases. The figure shows that in comparison to the ISO 31000 and other standards the proposed model, E-RIAM, provides a clearer definition of the tasks and expected outcomes of the risk management team personnel.

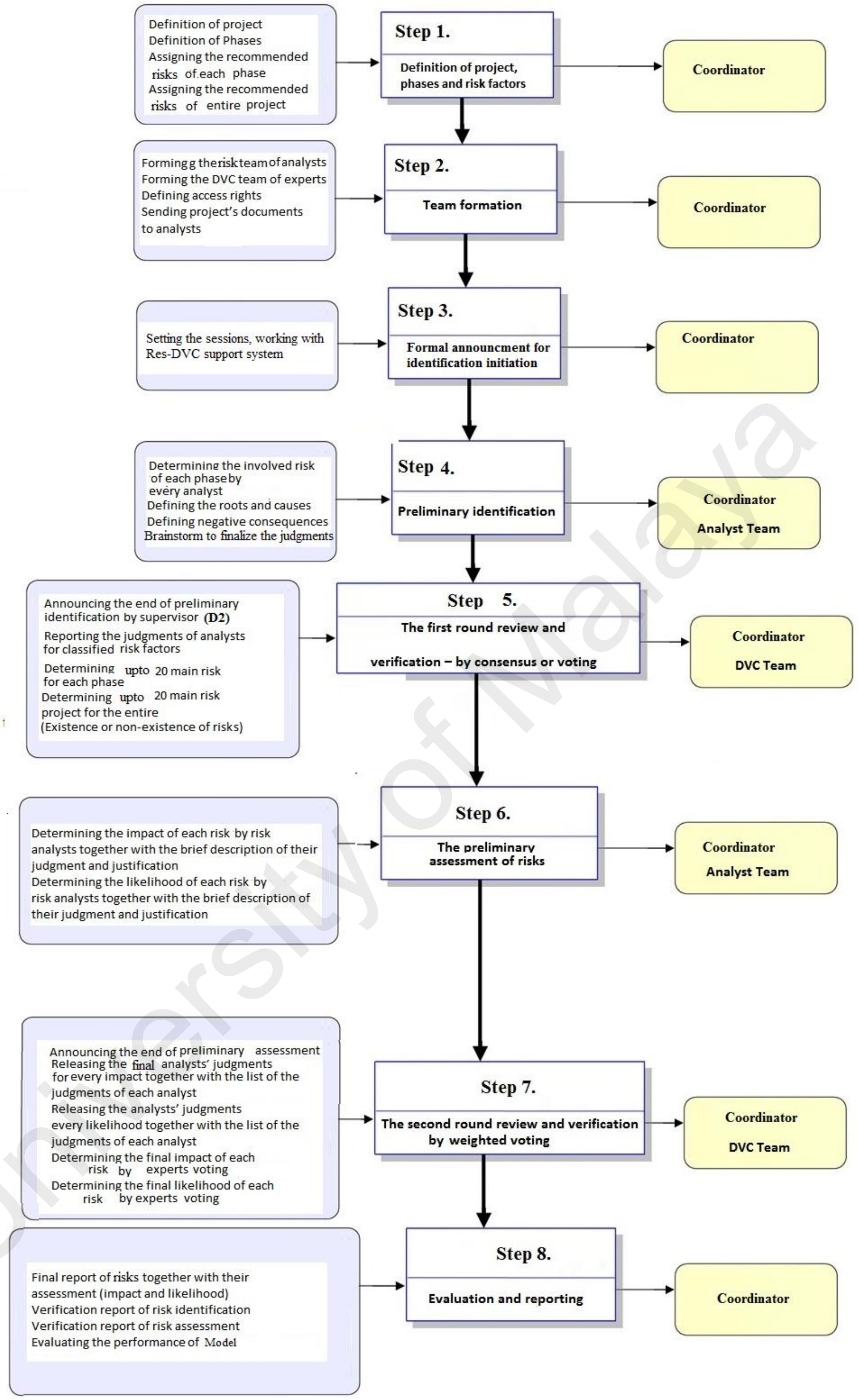

Definition of project
Definition of Phases
Assigning the recommended risks of each phase
Assigning the recommended risks of entire project
Step 1.
Definition of project, phases and risk factors
Coordinator
Formingg theriskteamofanalysts
Forming the DVC team of experts
Defining access rights
Sending project's documents to analysts
Step 2.
Team formation
Coordinator
Setting the sessions, working with Res-DVC support system
Step 3.
Formal announcment for identification initiation
Coordinator
Determining the involved risk of each phase by every analyst
Defining the roots and causes
Defining negative consequences
Brainstorm to finalize the judgments
Step 4.
Preliminary identification
Coordinator
Analyst Team
Announcing the end of preliminary identification by supervisor (D2)
Reporting the judgments of analysts for classified risk factors
Determining upto 20 main risk for each phase
Determining upto 20 main risk project for the entire
(Existence or non-existence of risks)
Step 5.
The first round review and verification – by consensus or voting
Coordinator
DVC Team
Determining the impact of each risk by risk analysts together with the brief description of their judgment and justification
Determining the likelihood of each risk by risk analysts together with the brief description of their judgment and justification
Step 6.
The preliminary assessment of risks
Coordinator
Analyst Team
Announcing the end of preliminary assessment
Releasing the final analysts' judgments for every impact together with the list of the judgments of each analyst
Releasing the analysts' judgments every likelihood together with the list of the judgments of each analyst
Determining the final impact of each risk by experts voting
Determining the final likelihood of each risk by experts voting
Step 7.
The second round review and verification by weighted voting
Coordinator
DVC Team
Final report of risks together with their assessment (impact and likelihood)
Verification report of risk identification
Verification report of risk assessment
Evaluating the performance of Model
Step 8.
Evaluation and reporting
Coordinator

Figure 4.3: Work Flow and Tasks Performed by Risk Personnel and Outcomes in Each Risk Phase

4.5 Support Tool and Databases

A support tool was developed to facilitate the risk identification and risk assessment processes in E-RIAM. This tool contains seven databases, known collectively as server farm (See Figure 4.2).

The information in some of the databases may be modified, except the information in the risk-related databases, which must remain intact. The details of each project together with the details of the various phases are stored in the project details database. The database of risk analysts and experts (risk team and DVC team database) contains their detailed personal information and their access rights to the system functionalities they can use depending on their roles. There are three databases related to risks: i) risk database – contains the risks pertinent to each development phase as well as those pertinent to the entire project; ii) risk identification database – contains details of the risks identified at each development phase of the project as well as the index of the respective risks; iii) risk assessment database – contains details of the risk assessed at each development phase of the project as well as the index of the respective risks, and the likelihood of risk occurrence and risk impact values. The verification database – contains information on the modifications made to the risk identification and risk assessment processes. After the risk analysts have identified and assessed the risks, the risk experts make the necessary modifications and information on the modifications will be kept in the respective database. All communication between the RC and analysts (experts) will be recorded in the communication log database.

4.6 The Unique Features of E-RIAM

Some researchers have strongly suggested involved external (independent) experts in risk management, but this is not mentioned in the ISO 31000 and other standards. As a result, interaction among risk teams in the risk identification and risk assessment phases is not mentioned. In E-RIAM, the DVC experts interact with each other during the brainstorming session, consensus making, or voting stages, but this is not mentioned in ISO 31000 and other standards. Furthermore, ISO 31000 does not suggest the use of risk experts' judgments and the involvement of DVC modifier experts. Another feature of the proposed model is combining the judgments of risk analysts, which will be useful for impact, probability of occurrence, and RV. This feature is not present in the ISO 31000 standard. Finally, the ISO 31000 standard does not provide for an evaluation of the model efficiency. However, this feature is incorporated in the fourth phase of the E-RIAM model. The use of an online support tool in the proposed model is unique. Other assessment models also provide offline or online tools to facilitate the assessment process, but they do not support the model completely.

4.7 Development of a Risk Identification and Assessment Support Tool (Res-DVC)

A support tool, known as Res-DVC, was developed to facilitate the process flow in E-RIAM. Res-DVC was developed using RAD methodology and ASP software. Figure 4.4 shows a snapshot of Res-DVC. As this research focuses only on risk identification and risk assessment, Res-DVC was thus designed to provide the functionalities to meet the enhanced features proposed for E-RIAM. These functions include:

1) Create and define new projects.

   Collect and store in the database information pertaining to the project, which includes the start date, end date, and project definition.

2) Create risk team and DVC team, and formally register risk analysts/experts into the risk teams;

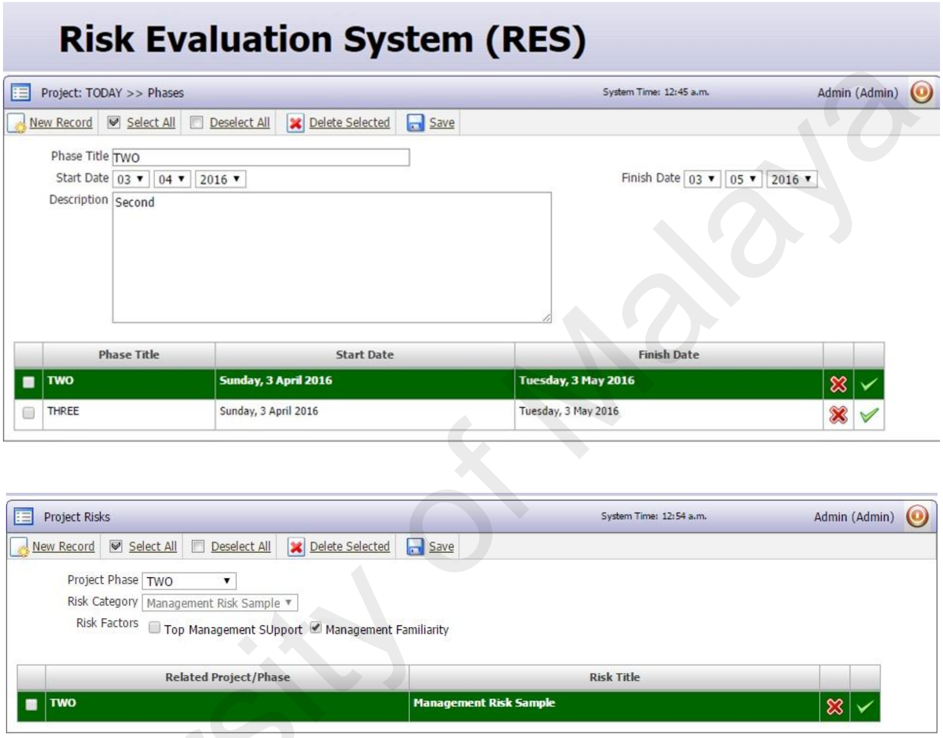
3) Assign project(s) to risk team and DVC team;

4) Store specific risks of each development phase (requirements, design, programming and testing, and implementation and release), and common risks of software projects;

5) Select specific risks of each development phase (requirements, design, programming and testing, and implementation and release), and common risks of a software project, by risk analysts/experts during the risk identification phase;

6) Record outcomes of the risk identification and risk assessment phases;

7) Produce reports of the risk identification phase.



Phase Title: TWO
Start Date: 03 ▾ 04 ▾ 2016 ▾
Finish Date: 03 ▾ 05 ▾ 2016 ▾
Description: Second

	Phase Title	Start Date	Finish Date		
☐	TWO	Sunday, 3 April 2016	Tuesday, 3 May 2016	✗	✓
☐	THREE	Sunday, 3 April 2016	Tuesday, 3 May 2016	✗	✓

Project Phase: TWO ▾
Risk Category: Management Risk Sample ▾
Risk Factors: ☐ Top Management SUpport ☑ Management Familiarity

	Related Project/Phase	Risk Title		
☐	TWO	Management Risk Sample	✗	✓

Figure 4.4: A Snapshot of Res-DVC Support Tool

4.8 Summary

The proposed E-RIAM comprises four main phases, namely, project definition and team formation, risk identification, risk assessment, and process and team evaluation. Four main enhancements were made to E-RIAM. These enhancements are aimed at overcoming or reducing some of the risk management problems revealed during the literature review (Chapter 2). In addition, a support tool (Res-DVC) was developed to facilitate the risk identification and risk assessment processes. This tool has seven databases to store details on the software projects, as well as pertinent risk-related data.

This tool makes it easy to produce risk identification and risk assessment reports. In addition, the productivity of risk analysts and risk experts can be calculated, and this will greatly help in the selection of risk analysts for future software projects. Two case studies were conducted to evaluate E-RIAM. Details of data collection and analysis for the evaluation of E-RIAM are presented in Chapter 5.

CHAPTER 5: CASE STUDIES, DATA COLLECTION AND ANALYSIS

In this research, two case studies were conducted to evaluate the efficiency of the E-RIAM model using risk data of 20 past software projects (medium-sized) from Department P and Department Q of a software development company (Company C), respectively. All these 40 commercial software projects were developed using software development lifecycle (SDLC) model and Rational Unified Process (RUP) technique implemented on .Net: ASP .Net with COM+ as application Server, IIS as Web Server, SQL Server and Oracle as Database Server with C# and ASP .Net programming languages.

In both the case studies, the two risk teams (RT1 from Department P, and RT2 from Department Q) identified and assessed the software risks following the common steps of the general risk management model based on ISO 31000/31010: 2009 Risk Management Standards. RT1 and RT2 used MS Excel to document the list of risks identified and assessed by them. On the other hand, RT3 (Department P) and RT4 (Department Q) identified the software risks based on the lists of risks classified according to the development phases (i.e., Requirements Analysis, Design, Programming and Testing, Implementation and release , and Common Risk for the Entire Project) that are stored in the Risk Database of Res-DVC tool (i.e. based on E-RIAM). However, in the event a new high probability of occurrence and high impact risk that either one or both of these two risk teams have identified is not found in the Risk Database, they can inform the Coordinator to include the risk together with its causes and effects into the Risk Database according to the classification of the respective software development phases. Both the RT3 and RT4 used the support tool, Res-DVC, to document the list of risks that they had identified and assessed throughout the risk identification and assessment processes. The two risk teams, DVC-I and DVC-

II, comprising three experienced risk analysts from the external company (Company D), also verified the risks identified by RT3 and RT4, respectively, based on the risks that are stored in the Risk Database. Similarly, they can request the Coordinator to include one or more new risks that they have identified during the risk identification process together with its respective causes and effects into the Risk Database. In this manner, the list of risks in the Risk Database will become more complete.

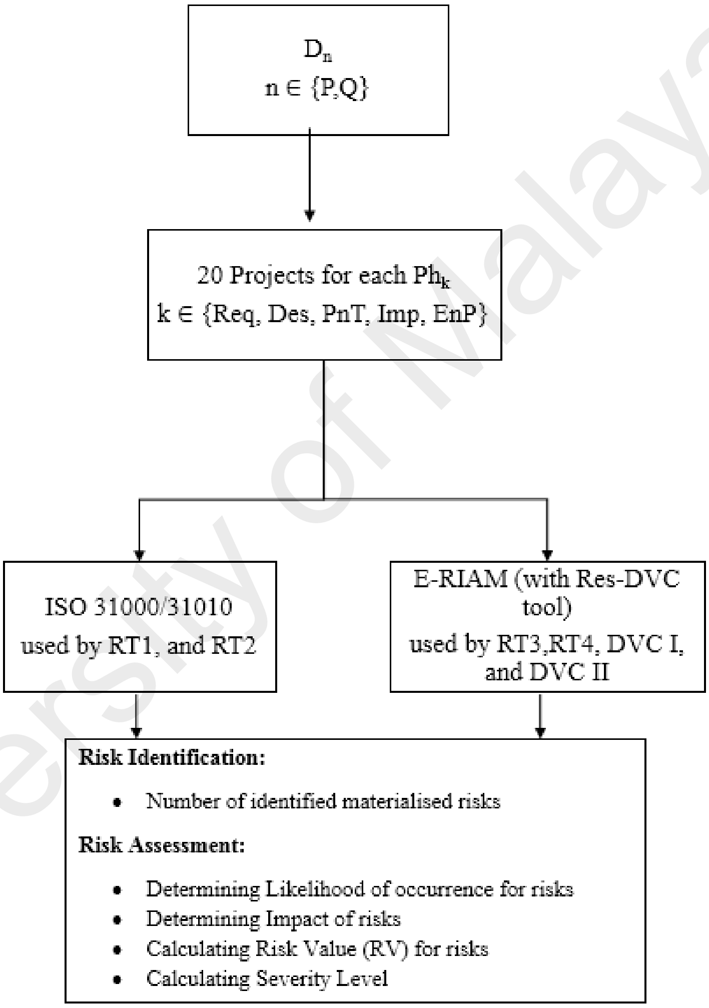
As the 20 software projects from Department P and Department Q, respectively, are historical commercial projects, the total number of risks that had materialised and the particulars of each risk, i.e., the risk ID, the development phase that the risk was detected, and the risk value calculated using Formula 2 (Chapter 3, Section 3.2.1: likelihood of occurrence of the risk x its impact), can be retrieved from the risk reports easily. Based on these risk data, the lists of risk that had materialised in the 40 software projects can be compared with the lists of risks that had been identified by RT1 and DVC1, RT2 and DVC2, to determine whether DVC1 and DVC2 were able to identify more risks that had materialised (i.e., more efficient) in the 20 software projects, respectively. Similarly, to determine whether DVC1 and DVC2 are more efficient in assessing the risks identified, the risk values (RVs) of the list of risks that had materialised in all the 40 software projects are compared with the RVs obtained by RT1 and DVC1 (Department P), RT2 and DVC2 (Department Q), respectively.

5.1 Case Studies

Both the identification and assessment on risks on the 20 software projects by the two Departments P and Q, respectively, were carried out simultaneously. All the risk teams from Department P (RT1 and RT3), Department Q (RT2 and RT4), and the two external risk teams, DVC-I and DVC-II, were given the following documents during the risk identification and assessment processes:

i) Design documents of the 20 projects (for each department), and

ii) Risk reports of a few past similar projects that record the type of risks that had materialised, the probability of risk occurrence, its impact and the risk value of each risk, are provided for referencing to improve the efficiency of the risk identification process.


D_n
$n \in \{P,Q\}$
20 Projects for each Ph_k
$k \in \{Req, Des, PnT, Imp, EnP\}$
ISO 31000/31010
used by RT1, and RT2
E-RIAM (with Res-DVC tool)
used by RT3,RT4, DVC I, and DVC II
Risk Identification:
Number of identified materialised risks
Risk Assessment:
Determining Likelihood of occurrence for risks
Determining Impact of risks
Calculating Risk Value (RV) for risks
Calculating Severity Level

Keys: Dn- Department number n, Where n= P, Q

Req: Requirements Analysis, Des: Design Phase, PnT: Programming and Testing,

Imp: Implementation and Release, EnP: Entire Project

Figure 5.1: Case Study: Research Design of The Two Independent Groups Research Design of The Case Studies

5.2 Selection of Risk Team Members

In Company C, there are 29 risk analysts with 23 of them having from 10 to 20 years of experience in software risk management. Hence, five risk analysts each time were assigned by the Coordinator to form the four risk teams – RT1, RT2, RT3 and RT4, respectively. Similarly, the risk analysts of the two risk teams, DVC-I and DVC-II, were selected from among the group of experienced risk analysts with more than 20 years of experience in software risk management from Company D (a sister company of Company C). The analysts of both companies did not have any information about the selected projects. In the two case studies, an equal number of risk analyst with the same level of risk management skills (i.e., having from 10 to 20 years of work experience) were chosen to identify and assess the risks in each development phase. The same approach was applied in the formation of the two experienced risk teams, DVC-I and DVC-II, as shown in Table 5.1. The assignment of an equal number of risk analysts of the same skill level for each case study will eliminate the confounding effect, which would have resulted if different number of risk analysts with different skill levels in software risk capabilities were involved.

Table 5.1: Number of risk analysts in each risk team

Case Study	Department	Risk Team ID (Control Group)	Risk Team ID (Treatment Group)	
			RT	DVC (Company D)
1	P	RT1: 5 RA	RT3: 5 RA	DVC-I: 3 RE
2	Q	RT2: 5 RA	RT4: 5 RA	DVC-II: 3 RE

Keys: RA – Risk Analysts
RE – Risk Experts

Note: RT1-RT4 – Less than 20 years of work experience in software risk management.
DVC-I and DVC-II – More than 20 years of work experience in software risk management.

5.3 Data Collection

In the two case studies, data were collected to prove the two hypotheses developed in Chapter 3. In case study 1, the five risk analysts of RT1 conducted a face-to-face meeting to identify the top 20 potentially high probability of occurrence and high impact risks, and record the outcomes using MS Excel. In the event the identification process could not be completed in one meeting, a second meeting or more face-to-face meetings will be conducted until all the risk analysts come to a comparison and agree that the top 20 potential risks have been identified. This identification process is repeated by RT1 to identify the top 20 potential software risks for other development phases as well as the top 20 common risks for the entire project (total sets of values), for all the 20 software projects. RT3 and RT4 also used the same procedures to identify the top 20 potential risks for the 20 software projects but based on the list of risks retrieved from the Risk Database of the Res-DVC tool. These two lists of identified risk were forwarded to the DVC-I and DVC-II teams by the Coordinator for verification. Both DVC-I and DVC-II teams used the same procedures to identify the potential risks in a face-to-face meeting but record the outcomes using the Res-DVC tool. The final (updated) lists of risks identified were then submitted to the Coordinator to forward to RT3 and RT4, respectively, for risk assessment.

During the risk assessment process, all the six risk teams (RT1, RT2, RT3, RT4, DVC-I and DVC-II) used the same procedures to assess the probability (likelihood) of occurrence (L) and the impact (I) of each risk identified, using a scale that ranges from 1 to 5. The risk value was also calculated (using the formula L x I), and recorded during the risk assessment process. The data collected during the risk identification and risk assessment processes include:

i) The total number of potential risks identified for each project, i.e., five sets of values per software project, giving a total of 100 sets of values for the total number of risks identified from 20 projects (5 sets of values for the total number of risks identified/project x 20 software projects = 100 sets of values);

ii) The risk ID for each risk identified in the four development phases as well as for the common risks for the entire software project;

iii) The probability (likelihood) of occurrence of each risk identified (L) for all the 20 software projects;

iv) The impact of each risk identified if it materializes (I) for all the 20 software projects; and

v) The risk value of each risk identified (calculated using the formula: L x I) for all the 20 software projects.

Besides the above data sets, the total number of risk that had materialised, and the details about each risk (i.e., the risk ID and its risk value) were also collected from Company C for proving of hypotheses. Table 5.2 shows a summary of the total number of sets of risk data, the total number of risks that had materialised (data obtained from Company C), and the total number of risks identified by the two control groups and treatment groups that materialised, collected from the two case studies for data analysis and proving the hypotheses.

Table 5.2: Data collected from the case studies for data analysis

Risks Data Collected	Case Study 1 (Department P: 20 software Projects: RT1↔DVC1)	Case Study 2 (Department Q: 20 software Projects: RT2↔DVC2)
Total number of sets of risk identified during the risk identification process by: i) Control Groups: RT1 and RT2	100	100
ii) Treatment Groups: DVC1 (RT3 + DVC-I) and DVC2 (RT4 + DVC-II)	100	100
Total number of risks that had materialised: (Data obtained from Company C)	1512	1605
Total number of risks identified by the two control groups that materialised: RT1 and RT2	1027	1005
Total number of risks identified by the two treatment groups that materialised: DVC1 and DVC 2	1386	1482

5.3.1 Administration of The Risk Identification and Risk Assessment Processes and Data Recording

During the two case studies, it is important to ensure that all related data are collected and recorded consistently. Also, data that are obtained based on the data collected (i.e., the risk values), are calculated correctly. In this regard, a data collection form was designed using MS Excel for the use by RT1 and RT2. The calculation of the risk value is done automatically using a built-in formula (L x I). A sample of the data collection form together with consent and commitment forms are included in Appendix C.

On the other hand, RT3, RT4, DVC-I and DVC-II used Res-DVC tool to record the risk data during the risk identification and risk assessment processes. The risk value of each risk is calculated automatically by the tool using the same formula (L x I). As Res-DVC is a new tool, a briefing session and demonstration was conducted by the researcher. Subsequently, the Coordinator held a similar briefing session and demonstration for the four risk teams (RT3, RT4, DVC-I and DVC-II) to ensure that they can use Res-DVC without any difficulty. This aims to avoid the four risk teams from the knowledge that they were involved in a research study.

Furthermore, each risk team is required to take not more than two weeks to identify and assess the risks for each project, respectively. A maximum of not more than two hours is also imposed on every face-to-face meeting to identify or assess the risks. A leader is selected in each risk team to ensure that the meeting time and the maximum duration imposed on each risk identification or assessment process are observed so that each software project can be completed according to the planned schedule. The total time to identify and assess the risks in each phase including the common risks for the entire project were recorded for future reference by other risk teams conducting risk identification and assessment on similar software projects. However, in these two case studies, although the time spent by all the four risk teams were recorded, these data are not used in data analysis due to the inconsistency in the number of risk analysts in each risk team (RT1, RT2, RT3 and RT4: 5 risk analysts; DVC-I and DVC-II: 3 risk analysts), the experience in software risk management (RT1, RT2, RT3 and RT4: between 10 and 20 years; DVC-I and DVC-II: more than 20 years), and the risk management practices in terms of the risk standards and guidelines used by the different companies (Company C uses ISO 31000/31010:2009 Risk Management Standards, Company D uses more than one standard).

5.4 Calculation of Risk Identification Efficiency

To test if the E-RIAM model is more efficient in risk identification than the risk identification process of the generic risk management model (Hypothesis 1), the efficiency of the risk identification process was first calculated using Formula 1 (Chapter 3) based on the data collected from the case studies:

Risk Identification Efficiency (RIE) = No. of risk identified by each risk team that materialised / Actual total No. of risks that had materialised (i.e., data provided by the software company) x 100% (1)

Table 5.3 shows the actual total number of risks that had materialised (obtained from Company C), the total number of risks identified by the five risk analysts of RT1, and DVC1 (i.e., risks identified by five risk analysts of RT3 and then verified by the three analysts of DVC-I) that materialised, for Projects 1 to 20 from Department P, respectively, together with their respective risk identification efficiency calculated using formula 1. Similarly, Table 5.4 shows the similar sets of risk data for case study 2, i.e., Projects 21 to 40, involving RT2 and DVC2 from Department Q.

Table 5.3: Data of the risk identification process (case study 1, department p, projects 1-20)

PRJ1	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	13	11	84.62	12	92.31
Des	12	9	75	11	91.67
PnT	14	6	42.86	14	**100**
Imp	15	11	73.33	13	86.67
EnP	15	13	86.67	14	93.33
PRJ2	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	13	11	84.62	11	84.62
Des	17	10	58.82	15	88.24
PnT	16	13	81.25	14	87.5
Imp	13	9	69.23	11	84.62
EnP	16	12	75	16	**100**
PRJ3	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	16	10	62.5	14	87.5
Des	14	7	50	12	85.71
PnT	16	11	68.75	15	93.75
Imp	15	10	66.67	15	**100**
EnP	14	11	78.57	10	71.43
PRJ4	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	16	11	68.75	14	87.5

Des	13	10	76.92	12	92.31
PnT	18	8	44.44	18	**100**
Imp	17	14	82.35	16	94.12
EnP	13	9	69.23	11	84.62
PRJ5	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	15	12	80	15	**100**
Des	17	11	64.71	16	94.12
PnT	14	11	78.57	13	92.86
Imp	12	8	66.67	10	83.33
EnP	14	10	71.43	13	92.86
PRJ6	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	12	7	58.33	11	91.67
Des	20	8	40	18	90
PnT	17	11	64.71	17	**100**
Imp	14	10	71.43	13	92.86
EnP	14	12	85.71	12	85.71
PRJ7	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	15	9	60	13	86.67
Des	15	9	60	14	93.33
PnT	17	10	58.82	15	88.24
Imp	15	11	73.33	13	86.67
EnP	15	12	80	13	86.67
PRJ8	Risk Identification				

Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	18	15	83.33	16	88.89
Des	19	14	73.68	15	78.95
PnT	15	11	73.33	14	93.33
Imp	14	11	78.57	13	92.86
EnP	17	12	70.59	15	88.24
PRJ9	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	17	12	70.59	17	**100**
Des	14	10	71.43	13	92.86
PnT	15	11	73.33	13	86.67
Imp	16	11	68.75	14	87.5
EnP	17	11	64.71	16	94.12
PRJ10	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	18	11	61.11	17	94.44
Des	17	12	70.59	16	94.12
PnT	13	9	69.23	13	**100**
Imp	13	6	46.15	12	92.31
EnP	16	11	68.75	14	87.5
PRJ11	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	17	13	76.47	16	94.12
Des	17	4	23.53	16	94.12
PnT	13	9	69.23	12	92.31
Imp	12	7	58.33	12	**100**

EnP	17	11	64.71	15	88.24
PRJ12	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	14	8	57.14	13	92.86
Des	14	9	64.29	14	**100**
PnT	16	8	50	15	93.75
Imp	15	10	66.67	13	86.67
EnP	12	8	66.67	11	91.67
PRJ13	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	15	10	66.67	13	86.67
Des	13	9	69.23	12	92.31
PnT	13	9	69.23	13	**100**
Imp	15	11	73.33	14	93.33
EnP	12	7	58.33	10	83.33
PRJ14	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	15	11	73.33	12	80
Des	15	12	80	15	**100**
PnT	17	12	70.59	16	94.12
Imp	15	9	60	13	86.67
EnP	19	13	68.42	18	94.74
PRJ15	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	19	11	57.89	16	84.21
Des	10	5	50	10	**100**

PnT	16	11	68.75	16	**100**
Imp	13	7	53.85	12	92.31
EnP	15	11	73.33	14	93.33
PRJ16	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	19	17	89.47	19	**100**
Des	16	10	62.5	14	87.5
PnT	18	13	72.22	17	94.44
Imp	17	13	76.47	15	88.24
EnP	14	5	35.71	12	85.71
PRJ17	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	17	11	64.71	16	94.12
Des	17	13	76.47	16	94.12
PnT	15	12	80	15	**100**
Imp	15	11	73.33	15	**100**
EnP	16	10	62.5	16	**100**
PRJ18	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	14	9	64.29	13	92.86
Des	15	11	73.33	14	93.33
PnT	13	10	76.92	12	92.31
Imp	15	9	60	14	93.33
EnP	14	9	64.29	12	85.71
PRJ19	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1

Req	18	14	77.78	16	88.89
Des	15	10	66.67	14	93.33
PnT	15	12	80	14	93.33
Imp	11	8	72.73	10	90.91
EnP	16	13	81.25	16	**100**
PRJ20	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT1	RIE_RT1	TNRM_DVC1	RIE_DVC1
Req	16	9	56.25	14	87.5
Des	18	13	72.22	17	94.44
PnT	9	6	66.67	7	77.78
Imp	13	12	92.31	12	92.31
EnP	16	8	50	13	81.25

Keys:
TNRM_Comp = Total number of risks materialised (obtained from Company C, Department P)
TNRM_RT1 = Total number of risks identified by Risk Team 1 (RT1) that materialised
TNRM_DVC1 = Total number of risks identified by by the DVC Team 1 (DVC1) that materialised
RIE_RT1 = Risk Identification Efficiency of Risk Team 1 (RT1)
RIE_DVC1 = Risk Identification Efficiency of DVC Team 1 (DVC1)
Req = Requirements Analysis Phase
Des = Design Phase
PnT = Programming and Testing Phase
Imp = Implementation and Release Phase
EnP = Entire Project

Table 5.4: Data of the risk identification process (case study 2, department q, projects 21-40)

PRJ21	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	17	11	64.71	16	94.12
Des	16	10	62.5	15	93.75
PnT	17	12	70.59	16	94.12
Imp	18	11	61.11	17	94.44

EnP	17	12	70.59	15	88.24
PRJ22	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	16	12	75	16	**100**
Des	13	7	53.85	11	84.62
PnT	16	9	56.25	14	87.5
Imp	19	10	52.63	18	94.74
EnP	16	11	68.75	14	87.5
PRJ23	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	16	7	43.75	15	93.75
Des	16	9	56.25	16	**100**
PnT	15	10	66.67	14	93.33
Imp	19	15	78.95	18	94.74
EnP	14	8	57.14	11	78.57
PRJ24	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	16	10	62.5	15	93.75
Des	17	11	64.71	15	88.24
PnT	15	9	60	13	86.67
Imp	14	9	64.29	13	92.86
EnP	16	9	56.25	15	93.75
PRJ25	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	17	12	70.59	16	94.12
Des	17	11	64.71	16	94.12

PnT	15	9	60	15	**100**
Imp	15	10	66.67	15	**100**
EnP	14	8	57.14	12	85.71
PRJ26	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	18	11	61.11	18	**100**
Des	16	5	31.25	14	87.5
PnT	16	12	75	15	93.75
Imp	20	12	60	19	95
EnP	18	10	55.56	18	**100**
PRJ27	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	15	11	73.33	13	86.67
Des	17	13	76.47	16	94.12
PnT	16	14	87.5	15	93.75
Imp	17	12	70.59	16	94.12
EnP	15	9	60	14	93.33
PRJ28	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	17	12	70.59	17	**100**
Des	17	11	64.71	15	88.24
PnT	17	11	64.71	13	76.47
Imp	13	8	61.54	11	84.62
EnP	14	6	42.86	13	92.86
PRJ29	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2

Req	18	10	55.56	14	77.78
Des	18	11	61.11	18	**100**
PnT	17	7	41.18	13	76.47
Imp	17	10	58.82	16	94.12
EnP	16	9	56.25	15	93.75
PRJ30	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	18	11	61.11	18	**100**
Des	15	12	80	11	73.33
PnT	17	13	76.47	17	**100**
Imp	13	8	61.54	12	92.31
EnP	17	11	64.71	16	94.12
PRJ31	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	19	9	47.37	18	94.74
Des	12	4	33.33	11	91.67
PnT	15	11	73.33	13	86.67
Imp	17	13	76.47	15	88.24
EnP	17	12	70.59	16	94.12
PRJ32	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	19	11	57.89	18	94.74
Des	11	8	72.73	10	90.91
PnT	15	10	66.67	15	**100**
Imp	15	5	33.33	12	80
EnP	17	12	70.59	17	**100**

PRJ33	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	15	9	60	14	93.33
Des	14	10	71.43	13	92.86
PnT	14	9	64.29	12	85.71
Imp	17	9	52.94	17	**100**
EnP	16	11	68.75	14	87.5
PRJ34	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	15	12	80	14	93.33
Des	17	8	47.06	16	94.12
PnT	15	8	53.33	14	93.33
Imp	15	10	66.67	13	86.67
EnP	17	11	64.71	17	**100**
PRJ35	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	16	13	81.25	15	93.75
Des	13	7	53.85	11	84.62
PnT	18	12	66.67	17	94.44
Imp	15	10	66.67	14	93.33
EnP	16	11	68.75	14	87.5
PRJ36	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	16	9	56.25	16	**100**
Des	13	7	53.85	12	92.31
PnT	18	10	55.56	17	94.44

Imp	14	9	64.29	11	78.57
EnP	18	11	61.11	16	88.89
PRJ37	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	15	10	66.67	14	93.33
Des	15	12	80	14	93.33
PnT	18	11	61.11	16	88.89
Imp	14	9	64.29	13	92.86
EnP	17	12	70.59	15	88.24
PRJ38	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	17	10	58.82	15	88.24
Des	17	11	64.71	15	88.24
PnT	18	8	44.44	18	**100**
Imp	16	7	43.75	15	93.75
EnP	18	14	77.78	17	94.44
PRJ39	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	17	13	76.47	16	94.12
Des	18	14	77.78	18	**100**
PnT	17	6	35.29	16	94.12
Imp	16	7	43.75	16	**100**
EnP	13	9	69.23	12	92.31
PRJ40	Risk Identification				
Risk Phase	TNRM_Comp	TNRM_RT2	RIE_RT2	TNRM_DVC2	RIE_DVC2
Req	14	6	42.86	13	92.86

Des	15	10	66.67	15	**100**
PnT	13	8	61.54	12	92.31
Imp	18	10	55.56	16	88.89
EnP	17	13	76.47	16	94.12

Keys:
TNRM_Comp = Total number of risks materialised (obtained from Company C, Department Q)
TNRM_RT2 = Total number of risks identified by Risk Team 2 (RT2) that materialised
TNRM_DVC2 = Total number of risks identified by by the DVC Team 2 (DVC2) that materialised
RIE_RT12 = Risk Identification Efficiency of Risk Team 2 (RT2)
RIE_DVC2 = Risk Identification Efficiency of DVC Team 2 (DVC2)
Req = Requirements Analysis Phase
Des = Design Phase
PnT = Programming and Testing Phase
Imp = Implementation and Release Phase
EnP = Entire Project

5.5 Test of Normality

Before choosing a statistical test to prove hypothesis 1, the Kolmogorov-Smirnov Tests of Normality was used to determine if the risk identification efficiency (RIE) of RT1 and DVC1; RT2 and DVC2, are normally distributed, respectively.

Tables 5.5 and 5.6 show the Kolmogorov-Smirnov normality test results for Case study 1 (Department P) and Case study 2 (Department Q), respectively.

Table 5.5: Kolmogorov-Smirnov test of normality for case study 1 (department p, projects 1-20)

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Statistic	Df	Sig.	Statistic	df	Sig.
Risk Identification Efficiency (RT1)	.106	100	.007	.953	100	.001
Risk Identification Efficiency (DVC1)	.129	100	.000	.934	100	.000

a. Lilliefors Significance Correction
Note: Kolmogorov-Smirnov test, if sig. value, p > 0.05, indicates normal distribution, otherwise not normal distribution.

Table 5.6: Kolmogorov-Smirnov tests of normality for case study 2 (department q, projects 21-40)

	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Statistic	Df	Sig.	Statistic	df	Sig.
Risk Identification Efficiency (RT2)	.088	100	.054	.968	100	.015
Risk Identification Efficiency (DVC2)	.180	100	.000	.895	100	.000

b. Lilliefors Significance Correction
Note: Kolmogorov-Smirnov test, if sig. value, p > 0.05, indicates normal distribution, otherwise not normal distribution.

The two sets of the Kolmogorov-Smirnov test results show that the risk identification efficiency (RIE) of RT1 and RT2 are normally distributed as their respective p values are p = 0.07, and p = 0.054, respectively (i.e. both p > 0.05). However, the RIE of DVC1 and DVC2 are both 0.000, respectively, indicating that the data are not normally distributed. Hence, the non-parametric test, Wilcoxon Signed Ranks Test is used to test hypothesis 1. In both case studies, the three assumptions underlying the Wilcoxon Signed Ranks Test are confirmed:

Assumption 1: Each pair of the RIE represents a random sample from a population and is independent of every other pair of RIE, in both case studies.

Assumption 2: The sample sizes of both case studies consist of more than 16 pairs of non-tied RIE.

Assumption 3: The difference in the RIE of both case studies are continuously distributed.

5.5.1 Test of Hypothesis 1

Tables 5.7 and 5.8 show four data sets for RIE of risks identified by the control teams and the treatment teams of the two departments' projects, respectively.

Table 5.7: Results of Wilcoxon Signed Ranks test (case study 1, department p)

Ranks

		N	Mean Rank	Sum of Ranks
Risk Identification Efficiency (DVC1) - Risk Identification Efficiency (RT1)	Negative Ranks	1[a]	7.00	7.00
	Positive Ranks	96[b]	49.44	4746.00
	Ties	3[c]		
	Total	100		

a. Risk Identification Efficiency (DVC1) < Risk Identification Efficiency (RT1)

b. Risk Identification Efficiency (DVC1) > Risk Identification Efficiency (RT1)

c. Risk Identification Efficiency (DVC1) = Risk Identification Efficiency (RT1)

Test Statistics[a]

	Risk Identification Efficiency (DVC1) - Risk Identification Efficiency (RT1)
Z	-8.528[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

Table 5.8: Results of Wilcoxon Signed Ranks test (case study 2, department q)

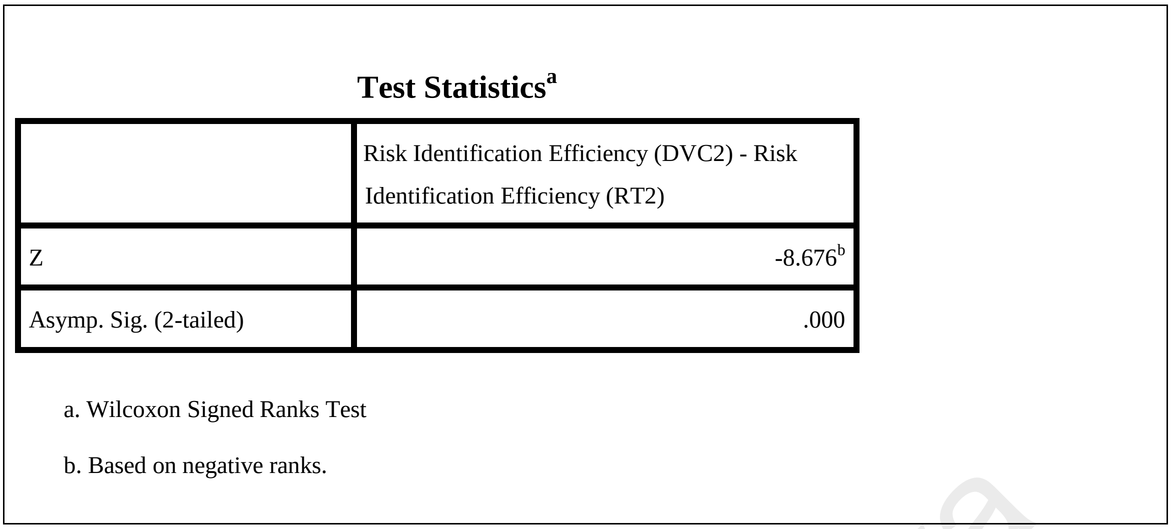
Ranks

		N	Mean Rank	Sum of Ranks
Risk Identification Efficiency (DVC2) - Risk Identification Efficiency (RT2)	Negative Ranks	1[a]	2.00	2.00
	Positive Ranks	99[b]	50.99	5048.00
	Ties	0[c]		
	Total	100		

a. Risk Identification Efficiency (DVC2) < Risk Identification Efficiency (RT2)

b. Risk Identification Efficiency (DVC2) > Risk Identification Efficiency (RT2)

c. Risk Identification Efficiency (DVC2) = Risk Identification Efficiency (RT2)



Test Statistics[a]

	Risk Identification Efficiency (DVC2) - Risk Identification Efficiency (RT2)
Z	-8.676[b]
Asymp. Sig. (2-tailed)	.000

a. Wilcoxon Signed Ranks Test

b. Based on negative ranks.

As shown in Table 5.7, the results of the Wilcoxon Signed Ranks Test show that in case study 1 (Department P), 96 sets of the Risk Identification Efficiency (DVC1) are greater than (>) Risk Identification Efficiency (RT1), four (4) sets of the Risk Identification Efficiency (DVC1) are less than (<) Risk Identification Efficiency (RT1), and only one (1) set of the Risk Identification Efficiency (DVC1) is equal to (=) the Risk Identification Efficiency (RT1) (this is a tie, implying that the RIE of DVC1 and RT1 is the same, i.e., the number of risk identified by DVC1 that materialised is the same as the number of risk identified by RT1 that materialised). It can be inferred that the proposed E-RIAM model that incorporates the risk verification process conducted by the three experienced risk analysts based on more than one risk management standards, can identify more risks that eventually materialise. Also, the test statistics of the Z value of the Wilcoxon Signed Ranks Test, at α= 0.025 (Asymp. Sig. (2-tailed) test), gives -8.528[b] (Based on negative ranks), indicates that DVC1 is more efficient than RT1 in identifying the software risks in the 20 software projects (Projects 1-20) of Department P.

Similarly, as shown in Table 5.8, the results of the Wilcoxon Signed Ranks Test show that in case study 2 (Department Q), 99 sets of the Risk Identification Efficiency (DVC2) are greater than (>) the Risk Identification Efficiency (RT2), only one (1) set of the Risk Identification Efficiency (DVC1) are less than (<) the Risk Identification Efficiency (RT1). There is no tie in case study 2 (as shown by 0^c in Table 5.8). Again, it can be inferred that the proposed E-RIAM model that incorporates the risk verification process conducted by three experienced risk analysts conducted based on more than one risk management standard can identify more risks that eventually materialise. Also, the test statistics of the Z value of the Wilcoxon Signed Ranks Test, at α= 0.025 (Asymp. Sig. (2-tailed) test), gives -8.676^b (based on negative ranks), indicates that DVC2 is more efficient than RT2 in identifying the software risks in the 20 software projects (Projects 21-40) of Department Q.

Based on the test results of the Wilcoxon Signed Ranks Test of the two case studies, it is justified to reject the null hypothesis (H0) and accept the Alternate hypothesis (H1), implying that the efficiency of the E-RIAM model in the risk identification process is higher than the efficiency of the generic risk management model.

5.5.2 Test of Hypothesis 2

Hypothesis 2 aims to determine whether the efficiency of the E-RIAM model in risk assessment is higher than of the generic risk assessment process model. In order to test this hypothesis, data on the risk assessment of each project were collected and analysed. These data include the assessment on the likelihood (probability) of occurrence of each risk (L) and its impact if it materialises (I). As explained in Chapter 3, these two attributes are measured using a scale that ranges from 1 to 5, where one (1) represents low likelihood of occurrence and low impact if the risk materialises, while five (5) represents high likelihood of occurrence and high impact if the risk materialised. The risk value (RV) which is obtained using Formula 2 (Chapter 3) was calculated, and the classification of the RV into the three different zones of major risk (red zone), moderate risk (yellow zone), and minor risk (green zone) was then determined, and used in the test of hypothesis 2, explained in the next section. However, based on the company policies, risks with less than 10 percent influence on budget and schedule assumed as Minor, between 10 to 30 percent deviation in cost or schedule are categorised as Moderate risks, and more than 30 percent are classified as Major risks.

5.5.2.1 Risk Assessment Data of the Two Case Studies

To test Hypothesis 2, data of the risk assessments of: i) the risks that had materialised (obtained from Company C); and ii) the risks identified by the RT1 and DVC1 that had materialised, for Projects 1 to 20 from Department P, respectively, together with the respective RVs which were calculated using formula 2, are compiled as shown in Appendix D. Table 5.9 shows a sample data of Project 1 of Case study 1. The naming convention used in the risk ID is explained below:

D_P_P1_A_1 denotes risk 1 of the requirements analysis phase of Project 1 from Department P, i.e. D_P refers to Department P, P1 refers to Project 1, A refers to requirements analysis phase, (D is the Design phase, P is the Programming and Testing phase, I refers to the Implementation and release, and Prj refers to the entire project. The value "0" denotes any risk that has not been identified by the risk team concerned. For example, in Table 5.9, for the risk ID: D_P_P1_A_4, RT1 failed to identify the risk that had materialised, but DVC1 was able to identify the risk, and calculated the risk value (RV) and the severity level (SL) correctly (i.e., these two sets of values matched with the RV and SL of the risk that had materialised). On the other hand, for the risk ID: D_P_P1_A_14, both the risk teams, RT1 and DVC1, failed to identify the risk that had materialised. Similarly, Table 5.10 shows similar sets of sample risk assessment data for Case Study 2, i.e., for Project 21 from Department Q. The complete sets of risk assessment data of Department Q are included in Appendix D.

Table 5.9: Sample data of the risk assessment process (case study 1, department p, project 1)

Risk ID	RV (C)	SL (C)	LoO (RT1)	Imp (RT1)	RV (RT1)	SL (RT1)	LoO (DVC1)	Imp (DVC1)	RV (DVC1)	SL (DVC1)
D_P_P1_A_1	10	3	3	5	15	3	4	3	12	2
D_P_P1_A_2	15	3	4	3	12	2	4	3	12	2
D_P_P1_A_3	20	3	2	5	10	3	4	3	12	2
D_P_P1_A_4	**25**	**3**	**0**	**0**	**0**	**0**	**5**	**5**	**25**	**3**
D_P_P1_A_5	20	3	3	5	15	3	4	4	16	3
D_P_P1_A_6	15	2	5	2	10	2	4	4	16	3
D_P_P1_A_8	20	3	3	2	6	1	4	4	16	3
D_P_P1_A_10	10	3	2	2	4	1	4	3	12	2
D_P_P1_A_11	15	3	1	1	1	1	5	4	20	3
D_P_P1_A_13	15	2	5	4	20	3	4	3	12	2
D_P_P1_A_14	**5**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
D_P_P1_A_16	20	3	3	2	6	1	3	3	9	2
D_P_P1_A_17	25	3	2	4	8	2	4	3	12	2
D_P_P1_D_3	20	3	1	2	2	1	5	4	20	3
D_P_P1_D_4	15	3	3	3	9	2	0	0	0	0
D_P_P1_D_5	20	3	5	3	15	2	5	4	20	3
D_P_P1_D_6	20	3	2	4	8	2	4	4	16	3
D_P_P1_D_7	25	3	4	4	16	3	5	5	25	3
D_P_P1_D_9	15	3	3	4	12	2	4	3	12	2
D_P_P1_D_10	25	3	1	2	2	1	5	5	25	3
D_P_P1_D_11	15	2	0	0	0	0	5	5	25	3
D_P_P1_D_12	15	3	4	2	8	3	4	4	16	3
D_P_P1_D_14	20	3	4	3	12	2	5	4	20	3
D_P_P1_D_15	25	3	0	0	0	0	5	5	25	3
D_P_P1_D_17	15	3	0	0	0	0	4	3	12	2
D_P_P1_P_2	15	2	4	5	20	3	4	3	12	2
D_P_P1_P_4	25	3	3	4	12	2	5	5	25	3
D_P_P1_P_5	20	3	1	1	1	1	4	2	8	3
D_P_P1_P_9	5	1	0	0	0	0	3	2	6	1
D_P_P1_P_10	20	3	3	4	12	2	5	4	20	3
D_P_P1_P_11	5	1	0	0	0	0	3	2	6	1
D_P_P1_P_13	20	3	0	0	0	0	3	3	9	2
D_P_P1_P_14	15	2	4	3	12	2	4	3	12	2
D_P_P1_P_15	10	2	0	0	0	0	5	2	10	2
D_P_P1_P_16	10	2	3	4	12	2	3	3	9	2
D_P_P1_P_17	20	3	0	0	0	0	4	4	16	3
D_P_P1_P_18	5	1	0	0	0	0	3	3	9	2
D_P_P1_P_19	5	1	0	0	0	0	3	2	6	1
D_P_P1_P_20	15	2	0	0	0	0	5	4	20	3
D_P_P1_I_2	25	3	4	3	12	2	5	3	15	2
D_P_P1_I_4	25	3	4	5	20	3	5	4	20	3
D_P_P1_I_5	5	1	3	4	12	2	3	2	6	1
D_P_P1_I_6	20	3	3	4	12	2	4	4	16	3
D_P_P1_I_8	25	3	2	2	4	1	0	0	0	0
D_P_P1_I_9	20	3	5	2	10	2	5	4	20	3
D_P_P1_I_11	5	1	0	0	0	0	3	2	6	1
D_P_P1_I_12	25	3	4	4	16	3	5	5	25	3
D_P_P1_I_13	10	2	0	0	0	0	5	2	10	2
D_P_P1_I_14	20	3	4	3	12	2	5	4	20	3
D_P_P1_I_15	5	1	0	0	0	0	4	2	8	3
D_P_P1_I_17	10	2	0	0	0	0	5	2	10	2

D_P_P1_I_18	10	2	4	3	12	2	4	3	12	2
D_P_P1_I_19	20	3	3	4	12	2	5	4	20	3
D_P_P1_I_20	10	2	1	1	1	1	0	0	0	0
D_P_P1_Prj_2	10	2	2	1	2	1	4	3	12	2
D_P_P1_Prj_3	15	2	3	3	9	2	4	3	12	2
D_P_P1_Prj_4	5	1	2	4	8	2	3	2	6	1
D_P_P1_Prj_5	15	2	2	4	8	2	5	3	15	2
D_P_P1_Prj_6	25	3	0	0	0	0	5	5	25	3
D_P_P1_Prj_7	25	3	3	4	12	2	5	5	25	3
D_P_P1_Prj_8	20	3	4	5	20	3	5	4	20	3
D_P_P1_Prj_9	20	3	5	5	25	3	5	4	20	3
D_P_P1_Prj_10	20	3	3	3	9	2	5	3	15	2
D_P_P1_Prj_12	10	2	3	1	3	1	0	0	0	0
D_P_P1_Prj_13	5	1	4	3	12	2	3	2	6	1
D_P_P1_Prj_15	25	3	1	1	1	1	5	5	25	3
D_P_P1_Prj_16	25	3	0	0	0	0	5	5	25	3
D_P_P1_Prj_18	15	3	4	3	12	2	4	4	16	3
D_P_P1_Prj_19	15	2	3	2	6	1	5	3	15	2

Keys:

D_P : Department P
A: Requirements analysis phase
P: Programming and testing phase
Prj: Entire project
RV: Risk Value
LoO: Likelihood of Occurrence of the risks

PN1: Project number (N1 = 1, .., 20)
D: Design phase
I: Implementation and release phase
n: Denotes the risk number (1 ≤ n ≤ 20)
SL: Severity Level (Major, Moderate, or Minor)
Imp: Impact of the risk

Table 5.10: Sample data of the risk assessment process (case study 2, department q, project 21)

Risk ID	RV (C)	SL (C)	LoO (RT2)	Imp (RT2)	RV (RT2)	SL (RT2)	LoO (DVC2)	Imp (DVC2)	RV (DVC2)	SL (DVC2)
D_Q_P1_A_1	20	3	2	1	2	1	4	4	16	3
D_Q_P1_A_4	20	3	4	3	12	2	5	3	15	2
D_Q_P1_A_5	10	3	0	0	0	0	5	2	10	2
D_Q_P1_A_6	25	3	0	0	0	0	3	3	9	2
D_Q_P1_A_7	20	3	0	0	0	0	5	4	20	3
D_Q_P1_A_8	20	3	3	2	6	1	4	4	16	3
D_Q_P1_A_9	20	3	5	3	15	2	5	4	20	3
D_Q_P1_A_10	25	3	0	0	0	0	5	2	10	2
D_Q_P1_A_11	25	3	2	1	2	1	0	0	0	0
D_Q_P1_A_12	25	3	3	1	3	1	4	2	8	3
D_Q_P1_A_13	25	3	5	2	10	2	5	3	15	2
D_Q_P1_A_15	10	3	4	1	4	1	5	2	10	2
D_Q_P1_A_16	10	3	0	0	0	0	5	3	15	2
D_Q_P1_A_17	20	3	4	3	12	2	5	4	20	3
D_Q_P1_A_18	20	3	1	1	1	1	4	4	16	3
D_Q_P1_A_19	20	3	0	0	0	0	4	3	12	2
D_Q_P1_A_20	20	3	3	3	9	2	4	3	12	2
D_Q_P1_D_1	25	3	0	0	0	0	5	3	15	2
D_Q_P1_D_2	20	3	4	4	16	3	4	4	16	3
D_Q_P1_D_3	5	1	0	0	0	0	3	2	6	1
D_Q_P1_D_4	20	3	0	0	0	0	5	4	20	3
D_Q_P1_D_6	5	1	0	0	0	0	3	2	6	1
D_Q_P1_D_7	25	3	5	3	15	2	5	5	25	3

D_Q_P1_D_8	15	2	3	3	9	2	4	3	12	2
D_Q_P1_D_9	5	1	1	5	5	2	4	2	8	3
D_Q_P1_D_10	20	3	4	3	12	2	5	4	20	3
D_Q_P1_D_11	10	2	4	1	4	1	3	2	6	1
D_Q_P1_D_12	25	3	4	3	12	2	5	4	20	3
D_Q_P1_D_14	25	3	0	0	0	0	3	3	9	2
D_Q_P1_D_15	25	3	4	4	16	3	5	4	20	3
D_Q_P1_D_17	25	3	0	0	0	0	0	0	0	0
D_Q_P1_D_18	20	3	1	5	5	2	5	3	15	2
D_Q_P1_D_20	25	3	3	3	9	2	5	5	25	3
D_Q_P1_P_1	20	3	0	0	0	0	1	5	5	2
D_Q_P1_P_2	15	2	5	3	15	2	5	3	15	2
D_Q_P1_P_3	5	1	1	5	5	2	3	2	6	1
D_Q_P1_P_4	15	2	4	4	16	3	4	3	12	2
D_Q_P1_P_5	25	3	3	3	9	2	5	5	25	3
D_Q_P1_P_6	5	1	5	5	25	3	3	2	6	1
D_Q_P1_P_8	25	3	5	5	25	3	5	5	25	3
D_Q_P1_P_9	20	3	4	3	12	2	5	4	20	3
D_Q_P1_P_10	25	3	4	3	12	2	4	3	12	2
D_Q_P1_P_11	20	3	0	0	0	0	5	4	20	3
D_Q_P1_P_12	20	3	2	1	2	1	5	4	20	3
D_Q_P1_P_13	15	2	4	2	8	3	4	4	16	3
D_Q_P1_P_15	25	3	0	0	0	0	0	0	0	0
D_Q_P1_P_16	15	3	3	3	9	2	5	3	15	2
D_Q_P1_P_18	10	3	2	1	2	1	3	2	6	1
D_Q_P1_P_19	10	2	0	0	0	0	4	3	12	2
D_Q_P1_P_20	10	3	0	0	0	0	4	3	12	2
D_Q_P1_I_1	10	3	3	1	3	1	5	2	10	2
D_Q_P1_I_2	5	1	0	0	0	0	5	2	10	2
D_Q_P1_I_3	20	3	4	3	12	2	5	3	15	2
D_Q_P1_I_4	5	1	4	3	12	2	3	2	6	1
D_Q_P1_I_5	5	1	0	0	0	0	4	2	8	3
D_Q_P1_I_6	25	3	5	5	25	3	5	5	25	3
D_Q_P1_I_7	20	3	0	0	0	0	0	0	0	0
D_Q_P1_I_9	25	3	4	4	16	3	5	5	25	3
D_Q_P1_I_10	15	2	0	0	0	0	5	3	15	2
D_Q_P1_I_11	20	3	5	2	10	2	5	4	20	3
D_Q_P1_I_12	5	1	5	2	10	2	4	2	8	3
D_Q_P1_I_13	10	3	0	0	0	0	5	2	10	2
D_Q_P1_I_14	10	2	4	1	4	1	5	2	10	2
D_Q_P1_I_15	10	2	0	0	0	0	5	2	10	2
D_Q_P1_I_16	5	1	4	4	16	3	5	2	10	2
D_Q_P1_I_18	25	3	4	3	12	2	5	5	25	3
D_Q_P1_I_19	10	2	0	0	0	0	5	2	10	2
D_Q_P1_I_20	15	2	4	4	16	3	4	4	16	3
D_Q_P1_Prj_1	15	3	2	1	2	1	4	4	16	3
D_Q_P1_Prj_3	25	3	3	3	9	2	5	4	20	3
D_Q_P1_Prj_4	25	3	3	2	6	1	5	3	15	2
D_Q_P1_Prj_5	20	3	1	1	1	1	0	0	0	0
D_Q_P1_Prj_6	20	3	5	2	10	2	4	3	12	2
D_Q_P1_Prj_8	5	1	0	0	0	0	3	2	6	1
D_Q_P1_Prj_9	25	3	5	5	25	3	5	5	25	3
D_Q_P1_Prj_10	15	2	0	0	0	0	4	4	16	3
D_Q_P1_Prj_11	10	3	4	2	8	3	4	2	8	3
D_Q_P1_Prj_12	15	2	0	0	0	0	5	3	15	2
D_Q_P1_Prj_13	10	2	5	2	10	2	5	2	10	2
D_Q_P1_Prj_14	20	3	4	3	12	2	5	4	20	3

D_Q_P1_Prj_15	25	3	0	0	0	0	5	4	20	3
D_Q_P1_Prj_16	5	1	1	5	5	2	1	5	5	2
D_Q_P1_Prj_17	20	3	3	1	3	1	4	4	16	3
D_Q_P1_Prj_18	20	3	5	2	10	2	5	4	20	3

Keys:

D_Q : Department Q
A: Requirements analysis phase
P: Programming and testing phase
Prj: Entire project
RV: Risk Value
LoO: Likelihood of Occurrence of the risks

PN1: Project number (N1 = 21, …, 40)
D: Design phase
I: Implementation and release phase
n: Denotes the risk number (1 ≤ n ≤ 20)
SL: Severity Level (Major, Moderate, or Minor)
Imp: Impact of the risk

5.5.2.2 Approaches to Test Hypothesis 2

To test Hypothesis 2, both the dependent and independent variables, and the total number of risks that had materialised must first be determined. The dependent variable is the severity level of a risk which is an ordinal scale (i.e., 0 – Not Identified, 1 – Minor Risk, 2 – Moderate Risk, and 3 – Major Risk), and is obtained from the two independent variables: i) Likelihood of occurrence of a risk (L), and ii) Impact of a risk if it materialises (I), which are both assessed using a scale ranging from 1 to 5, respectively (refer to Section 5.3 above). The total number of risks that had materialised in Case Study 1 (i.e., Department P: Projects 1-20, total number of risks: 1,512), and in Case Study 2 (i.e., Department Q: Projects 21-40, total number of risks: 1,605) are obtained from Company C (i.e., Risk Management Reports of the 40 projects as these are past projects). However, there are some risks that had materialised but the RT teams (i.e., RT1 and RT2) and DVC teams (i.e., DVC-I and DVC-II) had failed to identify them. These risks are denoted as “Not Identified”, and are included in the sample size. This is because it is assigned a value 0 (an ordinal scale), and can be considered one of the measurement scales for the dependent variable – severity level.

Table 5.11 – Table 5.14 show the total number of risks that had materialised for the three categories of severity level for Case Study 1 (RT1 and DVC1) and Case Study 2 (RT2 and DVC2), respectively. Based on Table 5.3 and Table 5.4, it is observed that

none of the risk teams (RT1, RT2, DVC1 and DVC2) had identified more risks than the total number of risks that had materialised in both the case studies (Total: 40 projects). On the other hand, DVC1 and DVC2 were able to identify 19 sets of the number of risks identified and which had materialised, correctly (i.e., the value 100, indicated in boldface in the RIE column of Table 5.3 and Table 5.4), with the total number of risks reported in the Risk Management Report of the 20 projects of Case Study 1 and Case Study 2, respectively.

Table 5.11: Total number of risks according to severity levels (RT1)

Assessed by RT1

	Observed N		Observed N
Not Identified	485		
Minor Risk	306	Minor Risk	306
Moderate Risk	451	Moderate Risk	451
Major Risk	270	Major Risk	270
Total	1512	Total	1027

Note: Not Identified refers to risks that had materialised (data obtained from the Risk Management Report of Company C) but were not identified by the risk team.

Table 5.12: Total number of risks according to severity levels (DVC1)

Assessed by DVC1

	Observed N		Observed N
Not Identified	126		
Minor Risk	129	Minor Risk	129
Moderate Risk	501	Moderate Risk	501
Major Risk	756	Major Risk	756
Total	1512	Total	1386

Note: Not Identified refers to risks that had materialised (data obtained from the Risk Management Report of Company C) but were not identified by the risk team.

Table 5.13: Total number of risks according to severity levels (RT2)

Assessed by RT2

	Observed N
Not Identified	600
Minor Risk	301
Moderate Risk	452
Major Risk	252
Total	1605

	Observed N
Minor Risk	301
Moderate Risk	452
Major Risk	252
Total	1005

Note: Not Identified refers to risks that had materialised (data obtained from the Risk Management Report of Company C) but were not identified by the risk team.

Table 5.14: Total number of risks according to severity levels (DVC2)

Assessed by DVC2

	Observed N
Not Identified	123
Minor Risk	156
Moderate Risk	625
Major Risk	701
Total	1605

	Observed N
Minor Risk	129
Moderate Risk	501
Major Risk	756
Total	1482

Note: Not Identified refers to risks that had materialised (data obtained from the Risk Management Report of Company C) but were not identified by the risk team.

To test Hypothesis 2 – i.e., to determine whether the efficiency of the E-RIAM model in risk assessment is higher than the generic risk assessment process model, the following approaches can be used, i.e., to compare and determine:

i) The total number of risks identified and had materialised in each category of severity level (SL) (i.e., major, moderate and minor), as assessed by RT1 and DVC1; and assessed by RT2 and DVC2, thus, matching the SL of the corresponding risks that had materialised in Case Study 1, and Case Study 2, respectively.

ii) The total number of risks that had materialised in both case studies (data obtained from the Risk Management Report of Company C), but not identified by the risk teams (RT1, DVC1, RT2, and DVC2).

5.5.2.3 Approach 1 to test Hypothesis 2

The total number of risks identified and which had materialised in each category of severity level (SL) (i.e., major, moderate and minor), assessed by RT1 and DVC1; and assessed by RT2 and DVC2, thus, matching with the SL of the corresponding risks that had materialised in Case Study 1, and Case Study 2, respectively, are determined based on case summaries of all the risks in the two case studies. Table 5.15 shows a sample of the case summaries of the first 20 risks generated using IBM SPSS version 22, for Case Study 1. In this table, case numbers 1, 3, 5 and 18 (in blue) of the risks assessed by RT1 (Total: 4 risks) matched with the SL (i.e., Major Risk) of the corresponding risks assessed by the company (based on the Risk Management Report of Case Study 1). Similarly, case numbers 4, 5, 7, 9, 14, 16, 17, 18 and 20 (in red) of the risks assessed by DVC1 (Total: 9 risks) match with the SL (i.e., Major Risk) of the corresponding risks assessed by the company. Only one (1) risk with "Moderate" SL assessed by RT1 (case number 6, in green) and DVC1 (case number 10, in purple), match with the corresponding risk assessed by the company, respectively. Appendices E and F show the case summaries of Case Study 1 (Total: 1,512 risks), and Case Study 2 (Total: 1,605 risks), respectively.

Table 5.15: Sample case summaries of the first 20 risks of case study 1 for comparison of severity levels

Case Summaries

Case Number	Severity Level of the Risks Assessed by the Company	Severity Level of the Risks Assessed by RT1	Severity Level of the Risks Assessed by DVC1
1	**Major Risk**	**Major Risk**	Moderate Risk
2	Major Risk	Moderate Risk	Moderate Risk
3	**Major Risk**	**Major Risk**	Moderate Risk
4	**Major Risk**	Not Identified	**Major Risk**
5	**Major Risk**	**Major Risk**	**Major Risk**
6	**Moderate Risk**	**Moderate Risk**	Major Risk
7	**Major Risk**	Minor Risk	**Major Risk**
8	Major Risk	Minor Risk	Moderate Risk
9	**Major Risk**	Minor Risk	**Major Risk**
10	**Moderate Risk**	Major Risk	**Moderate Risk**
11	Minor Risk	Not Identified	Not Identified
12	Major Risk	Minor Risk	Moderate Risk
13	Major Risk	Moderate Risk	Moderate Risk
14	**Major Risk**	Minor Risk	**Major Risk**
15	Major Risk	Moderate Risk	Not Identified
16	**Major Risk**	Moderate Risk	**Major Risk**
17	**Major Risk**	Moderate Risk	**Major Risk**
18	**Major Risk**	**Major Risk**	**Major Risk**
19	Major Risk	Moderate Risk	Moderate Risk
20	**Major Risk**	Minor Risk	**Major Risk**

Table 5.16 and Table 5.17 show a summary of the total number of risks that had materialised distributed according to the SL (major, moderate, or minor) of each risk based on the data obtained from Company C (i.e., Risk Management Report, row 1 of Tables 5.16 and 5.17, respectively), and the total number of risks identified by the two pairs of risk teams (RT1 and DVC1; RT2 and DVC2) that had materialised and matched with the SL of the corresponding risks of Case Study 1 and Case Study 2, respectively (rows 2 and 3 of Tables 5.16 and 5.17, respectively).

Table 5.16: Total no. of risks assessed by RT1 and DVC1 that match the SL of the corresponding risks of company c (case study 1)

Case Study 1: Department P, Projects 1-20 (Total: 1,512 risks that had materialised)				
Severity Level **No. of Risks that had Materialised**	**Major**	**Moderate**	**Minor**	**Total**
1. No. of Risk that had Materialised (Obtained from Company C)	926 (61.3%)	364 (24.0%)	222 (14.7%)	1,512 (100%)
2. No. of Risk Identified and which had Materialised (Assessed by RT1)	208 (22.5%)	110 (30.2%)	34 (15.3%)	352
3. No. of Risk Identified and which had Materialised (Assessed by DVC1)	642 (69.3%)	236 (64.8%)	102 (45.9%)	980

Table 5.17: Total no. of risks assessed by RT2 and DVC2 that match the SL of the corresponding risks of company c (case study 2)

Case Study 2: Department Q, Projects 21-40 (Total: 1,605 risks that had materialised)				
Severity Level **No. of Risks that had Materialised**	**Major**	**Moderate**	**Minor**	**Total**
1. No. of Risk that had Materialised (Obtained from Company C)	995 (62.0%)	396 (24.7%)	214 (13.3%)	1,605 (100%)
2. No. of Risk Identified and which had Materialised (Assessed by RT2)	178 (17.9%)	103 (26%)	15 (7%)	296
3. No. of Risk Identified and which had Materialised (Assessed by DVC2)	581 (58.4%)	290 (73.2%)	99 (46.3%)	970

Figure 5.2 and Figure 5.3 show the distribution of the risks according to SL in bar charts for Case Study 1 and Case Study 2, respectively. In Case Study 1, the total number of risks that had materialised and distributed according to SL are: major risks – 926 (61.3%), moderate risks – 364 (24.0%), and minor risks – 222 (14.7%), giving a

total of 1512 risks. Out of 1512 risks, RT1 was able to identify 208 major risks (22.5% = 208/926 x 100%), 110 moderate risks (30.2% = 110/364 x 100%), and 34 minor risks (15.3% = 34/222 x 100%), correctly, giving a total of 352 risks. Similarly, DVC1 was able to identify 642 major risks (69.3% = 642/926 x 100%), 236 moderate risks (64.8% = 236/364 x 100%), and 102 minor risks (45.9% = 102/222 x 100%), correctly, giving a total of 980 risks. Based on Table 5.16 and Figure 5.3, DVC1 was able to identify more risks distributed according to SL correctly than RT1 which had identified 434 major risks, 126 moderate risks, and 68 minor risks, giving a total of 628 (41.5% = 628/1512 x 100%) risks out of the total 1,512 risks that had materialised in Case Study 1.

Similarly, based on Table 5.17 and Figure 5.4, DVC2 was able to identify 581 (58.4%) major risks, 290 (73.2%) moderate risks, and 99 (46.3%) minor risks, correctly, giving a total of 970 risks. Overall, DVC2 was able to identify more risks distributed according to SL correctly than RT2 by 403 major risks, 187 moderate risks, and 84 minor risks, giving a total of 674 (42.0% = 674/1605 x 100%) risks out of the total 1,605 risks that had materialised in Case Study 2. It can be concluded that the two DVC teams (DVC1 and DVC2), who used E-RIAM in risk assessment show higher efficiency than the RT teams (RT1 and RT2) who assessed the risks based on the generic risk assessment process model, and the ISO 31010:2009 Risk Assessment Standards. Hence, the null hypothesis (H0) is rejected and the alternate hypothesis (H1) is accepted.

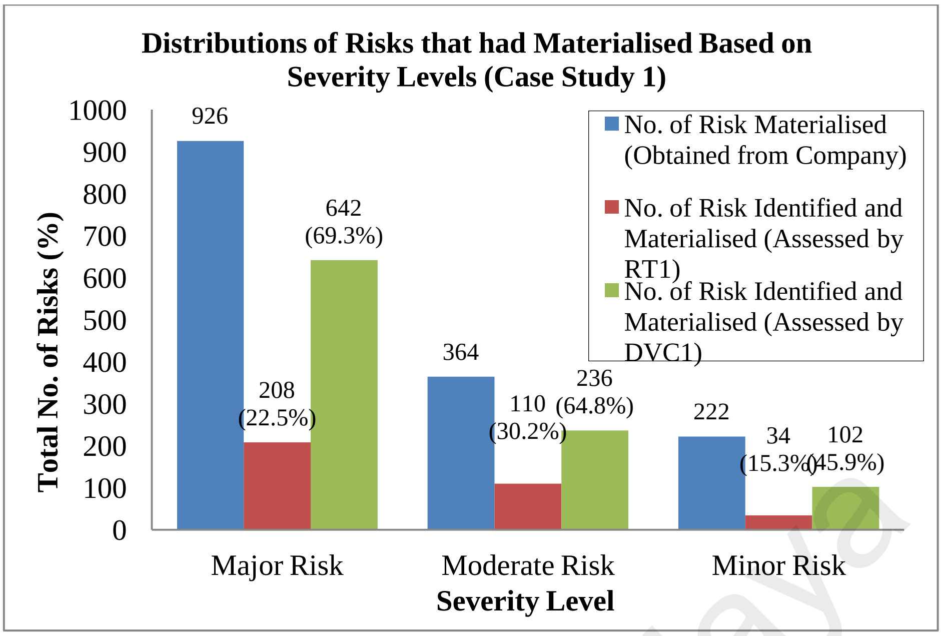


Figure 5.2: Distribution of Risks that had Materialised Based on Severity Levels Assessed by RT1 and DVC1 (Case Study 1)

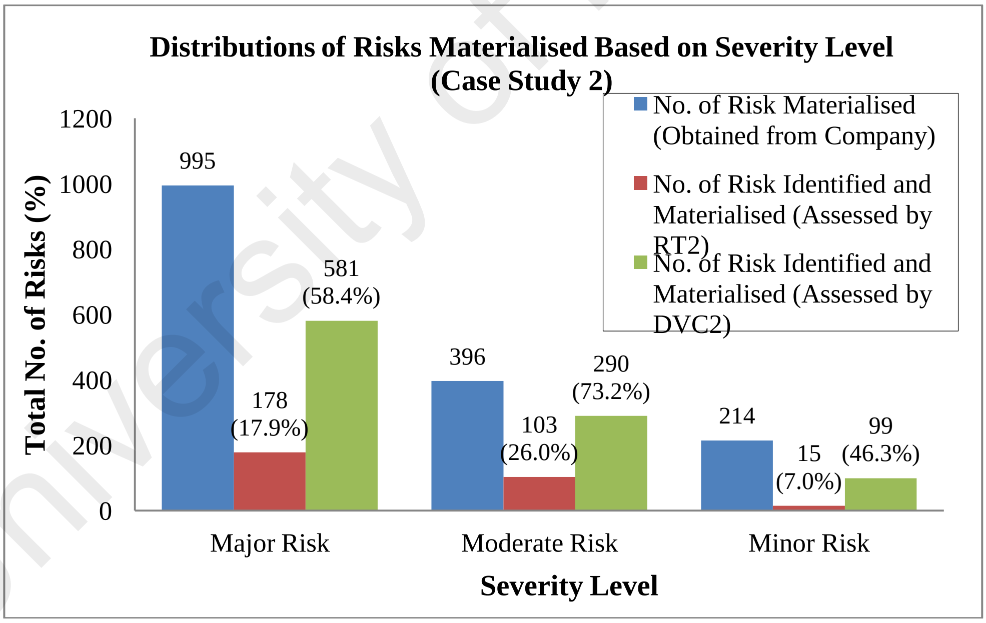


Figure 5.3: Distribution of Risks that had Materialised Based on Severity Levels Assessed by RT2 and DVC2 (Case Study 2)

5.5.2.4 Approach 2 to test Hypothesis 2

In this approach, the total number of risks that had materialised in both case studies (data obtained from the Risk Management Report of Company C), but were not identified by the risk teams (RT1, DVC1, RT2, and DVC2) are compared and analysed. Based on Table 5.18 and Figure 5.4, overall, the two RT teams were unable to identify more risks that had materialised than the two DVC teams, in the two case studies, respectively. The difference in the number of risks that were not identified between RT1 and DVC1, and between RT2 and DVC2 are 359 (22.4%), and 477 (29.7%), respectively. These figures imply that introducing an external experienced risk team to verify the risks identified by a less experienced risk team can contribute to significant improvement in the risk identification and assessment processes.

Table 5.18: Total number of risks that were not identified by the four risk teams

Case Study 1	
Risk Team	**Not Identified**
RT1	485 (32.1% = 485/1512 x100%)
DVC1	126 (8.3% = 126/1512 x100%)
Difference (RT1-DVC1)	359 (22.4% = 359/1605 x100%)

Case Study 2	
Risk Team	**Not Identified**
RT2	600 (37.4% = 600/1605 x100%)
DVC2	123 (7.7% = 123/1605 x100%)
Difference (RT2-DVC2)	477 (29.7% = 477/1605 x100%)

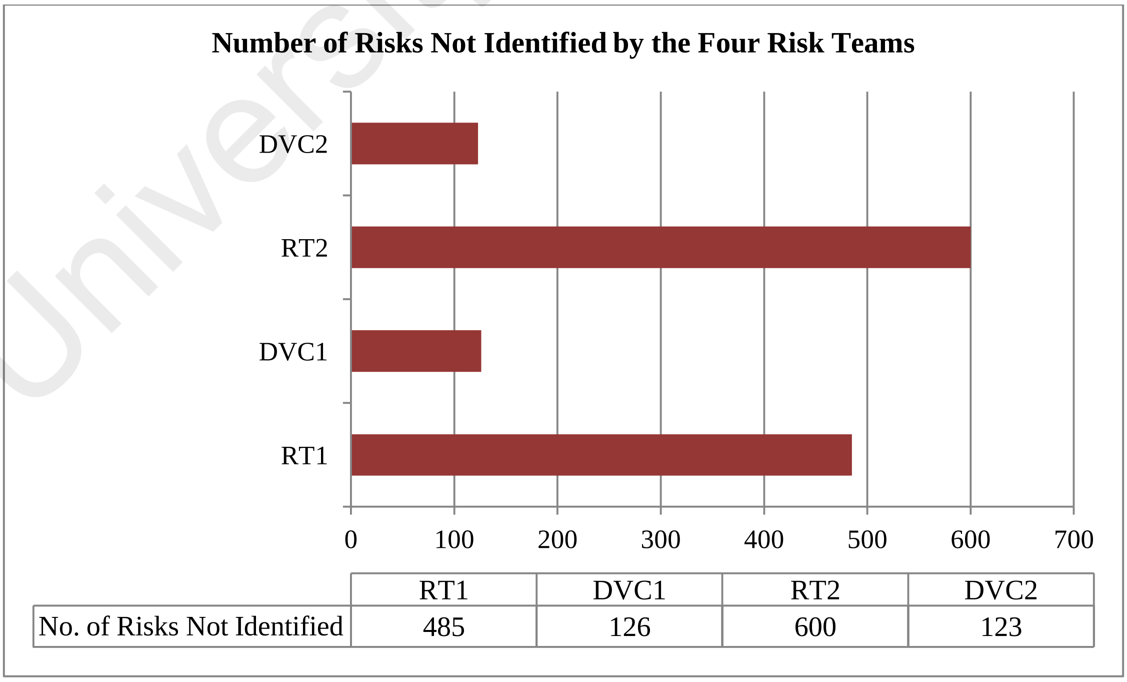


Figure 5.4: Total Number of Risks that were not Identified by the Four Risk Teams

5.6 Determining the Total Number of Potential Moderate and Major Risks in Each Development Phase

Besides proposing an enhanced risk identification and risk assessment process model (E-RIAM), this research also aims to determine:

i) whether the identification of a maximum of 20 potential (moderate and major) risks is sufficient in each development phase of a software project; and
ii) whether the identification of a maximum of 20 common (moderate and major) risks is sufficient for the entire software project.

To confirm these two propositions, cross-tabulations of the three variables – Severity Level (Moderate and Major risk categories only) for each risk assessed by the Company C, Development Phase, and Project ID were generated for Case Study 1 (Projects 1-20) and Case Study 2 (Projects 21-40). Table 5.19 shows a cross-tabulation for Case Study 1 (Projects 1-20). The cross-tabulation tables for the risks assessed by RT1 and DVC1 (Case Study 1), and Case Study 2 (i.e., risks assessed by Company, RT2 and DVC2) are included in Appendix E, and F, respectively.

Based on the results shown Table 5.19, and all the cross-tabulation tables in Appendices G and H (i.e., all data for Case Study 1 and Case Study 2), none of the projects has a total of 20 or more moderate and major risks, materialised in each of the development phase as well as the common risks for the entire project. Hence, it can be concluded that identifying a maximum of 20 potential (moderate and major) risks is sufficient for such purpose.

Table 5.19: Cross-tabulation for case study 1 (projects 1-20): severity level of the risks assessed by the company * development phase * project id

Severity Level of the Risks Assessed by the Company * Development Phase * Project ID Cross-tabulation

Project ID			Development Phase					Total
			Design Phase	Entire Project	Imp. Phase	P&T Phase	Req. Phase	
Project 1	Severity Level of the Risks Assessed by the Company	Moderate Risk	1	5	4	5	2	17
		Major Risk	11	8	8	5	10	42
	Total		12	13	12	10	12	59
Project 10	Severity Level of the Risks Assessed by the Company	Moderate Risk	5	4	6	1	5	21
		Major Risk	10	10	6	9	10	45
	Total		15	14	12	10	15	66
Project 11	Severity Level of the Risks Assessed by the Company	Moderate Risk	3	3	3	3	1	13
		Major Risk	10	13	7	9	14	53
	Total		13	16	10	12	15	66
Project 12	Severity Level of the Risks Assessed by the Company	Moderate Risk	3	2	4	2	2	13
		Major Risk	7	7	8	9	9	40
	Total		10	9	12	11	11	53
Project 13	Severity Level of the Risks Assessed by the Company	Moderate Risk	2	2	3	2	4	13
		Major Risk	8	7	10	8	8	41
	Total		10	9	13	10	12	54
Project 14	Severity Level of the Risks Assessed by the Company	Moderate Risk	5	3	3	2	3	16
		Major Risk	9	10	11	12	9	51
	Total		14	13	14	14	12	67
Project 15	Severity Level of the Risks Assessed by the Company	Moderate Risk	2	4	4	6	5	21
		Major Risk	6	8	7	8	10	39
	Total		8	12	11	14	15	60
Project 16	Severity Level of the Risks Assessed by the Company	Moderate Risk	5	4	3	4	5	21
		Major Risk	6	9	14	13	10	52
	Total		11	13	17	17	15	73
Project 17	Severity Level of the Risks	Moderate Risk	3	0	0	4	4	11

	Assessed by the Company	Major Risk	14	9	14	10	12	59
	Total		17	9	14	14	16	70
Project 18	Severity Level of the Risks Assessed by the Company	Moderate Risk	7	4	3	4	2	20
		Major Risk	7	8	11	7	8	41
	Total		14	12	14	11	10	61
Project 19	Severity Level of the Risks Assessed by the Company	Moderate Risk	5	5	2	5	4	21
		Major Risk	9	10	7	7	13	46
	Total		14	14	14	15	9	12
Project 2	Severity Level of the Risks Assessed by the Company	Moderate Risk	3	6	2	4	1	16
		Major Risk	7	9	10	10	10	46
	Total		10	15	12	14	11	62
Project 20	Severity Level of the Risks Assessed by the Company	Moderate Risk	3	4	5	6	3	21
		Major Risk	11	11	5	1	12	40
	Total		14	15	10	7	15	61
Project 3	Severity Level of the Risks Assessed by the Company	Moderate Risk	3	2	2	4	5	16
		Major Risk	9	11	8	8	10	46
	Total		12	13	10	12	15	62
Project 4	Severity Level of the Risks Assessed by the Company	Moderate Risk	2	2	7	6	4	21
		Major Risk	9	11	10	8	9	47
	Total		11	13	17	14	13	68
Project 5	Severity Level of the Risks Assessed by the Company	Moderate Risk	4	2	4	2	1	13
		Major Risk	11	11	8	11	14	55
	Total		15	13	12	13	15	68
Project 6	Severity Level of the Risks Assessed by the Company	Moderate Risk	4	4	2	6	1	17
		Major Risk	11	8	8	9	10	46
	Total		15	12	10	15	11	63
Project 7	Severity Level of the Risks Assessed by the Company	Moderate Risk	6	2	6	7	4	25
		Major Risk	7	10	8	7	8	40
	Total		13	12	14	14	12	65

Project 8	Severity Level of the Risks Assessed by the Company	Moderate Risk	8	3	4	6	5	26
		Major Risk	7	13	8	8	13	49
	Total		15	16	12	14	18	75
Project 9	Severity Level of the Risks Assessed by the Company	Moderate Risk	3	4	6	5	4	22
		Major Risk	7	12	8	8	13	48
	Total		10	16	14	13	17	70
Total	Severity Level of the Risks Assessed by the Company	Moderate Risk	77	65	73	84	65	364
		Major Risk	176	195	176	167	212	926
	Total		253	260	249	251	277	1290

Keys: Imp: Implementation and release P&T: Programming and Testing
Req: Requirements Analysis

5.7 Summary

This chapter presents the evaluation of E-RIAM using the data collected from two case studies on 40 software development projects from Company C. Statistical test was used to prove the Hypothesis 1 developed in Chapter 3. Before testing the hypothesis, the Kolmogorov-Smirnov test of normality, was employed to determine whether the data of the dependent variable (i.e., risk identification efficiency) are distributed normally, in both the case studies. Using Wilcoxon Signed Ranks Test, 96 sets and 99 sets of the Risk Identification Efficiency of DVC1 and DVC2, are greater than (>) the Risk Identification Efficiency of RT1 and RT2, in case study 1, and case study 2, respectively. Also, the test statistics of the Z values of the Wilcoxon Signed Ranks Test, at α= 0.025 (Asymp. Sig. (2-tailed) test), give -8.528[b] and -8.676[b] (both based on negative ranks), implying that both the DVC1 and DVC2 are more efficient than RT1 and RT2 in identifying the software risks in the 20 software projects of Department P and Department Q, respectively.

In this research, Hypothesis 2 was developed to determine whether the efficiency of E-RIAM in risk assessment is higher than the generic risk assessment process model. Two approaches were used to test this hypothesis. The first approach compares the severity level (i.e., major, moderate, and minor) of the risks identified, materialised and assessed by the two RT teams and DVC teams, respectively, that matched with the severity level (SL) of the corresponding risks (i.e., based on the data given by Company C). The total number of matching risks, distributed according to the three categories of severity levels are compiled and analysed. The outcomes show that DVC1 was able to identify 628 (41.5%) risks more than the total number of risks that RT1 could identify correctly, in Case Study 1. Similarly, DVC2 was able to identify 674 (42.0%) risks more than the total number of risks that RT2 could identify correctly, in Case Study 2.

Further analysis on the number of risks that had materialised but were not identified by the pair of RT1 and DVC1 teams, and by RT2 and DVC2 teams, respectively, show that RT1 and RT2 failed to identify 359 (22.4%), and 477 (29.7%) risks that DVC1 and DVC 2, were able to identify, respectively, and successfully. These two sets of findings imply that introducing an external experienced risk team to verify the risks identified by a less experienced risk team can contribute to significant improvements in the risk identification and assessment processes. Hence, it can be concluded that the null hypothesis (H0) of the two hypotheses developed in this research are rejected, and the two alternate hypotheses (H1) are accepted – implying that E-RIAM has higher efficiency in software risk identification and assessment processes.

Although a sample size of 40 medium-sized software projects were used in the two case studies, the findings show that using the enhanced risk identification and risk assessment processes (E-RIAM), and its standalone support tool, Res-DVC, the less experienced software risk analysts can improve efficiency in the risk identification process by identifying more risks that eventually materialised, and be able to assess the risks identified according to the three categories of severity level (major, moderate and minor), more accurately. Furthermore, the findings on the identification of a maximum of 20 potential (moderate and major) risks for each software development phase as well as for the common potential (moderate and major) risks for the entire project, will also be useful to the software risk analysts when conducting risk identification on similar medium-sized commercial software projects in future.

CHAPTER 6: CONCLUSION

This chapter discusses issues pertaining to the validity and reliability of the research. It explains how bias is minimized in different aspects of this study. It also discusses the problems encountered, and how they were resolved. The protection of the confidentiality of project data and other relevant information, and the protection of personal information of participants, the limitations and delimitations of this study, are also presented in this section. Finally, the main contributions of the research are highlighted. Proposals and suggestions for future research provide the conclusion to the thesis.

6.1 Validity of Research

This research was subjected to four validity assessments - external, internal, statistical, and construct validity, as described in detail in Chapter 3. We have made all efforts to satisfy the various criteria of the validity measures. For research reliability, risk analysts and experts participating in previous case studies played no role in identifying the risks of the selected projects or in the software development. Hence, they have neither any prejudgment nor interest in the risks per se. The teams were formed with an equal number of analysts in each team who have similar capabilities. The teams involved in the two case studies used the same tools and procedures, worked in similar environment, and had access to similar facilities, be it the DVC1 and DVC2 teams that worked with the E-RIAM model and its support tool (Res-DVC), or the RT1 teams that used the ISO 31000 standard and an Excel-based support tool. The statistical tests showed data non-normality, therefore, non-parametric methods were adopted. An academic statistician, acted as a consultant in this thesis, and interpreted and explained the use of suitable statistical tests based on the research goals, characteristics of tests, and their compatibility with the data sets and the hypotheses to be tested.

The number of risks identified using the two models in the two case studies is a very important variable that must be considered. The respective values were recorded for the 40 projects in the two case studies using both the ISO 31000 and E-RIAM models. This variable is used in the test of the first hypothesis of this study. The impact and the likelihood of occurrence of the materialised risks are also important variables, which are used in the test of the second hypothesis. They were recorded in all the four phases of the 40 projects by the teams that used the E-RIAM and ISO 31000 models, respectively. They were also used to test the hypothesis.

6.2 Bias in Research

The number of cases assigned to the control and treatment groups is balanced to prevent biases in the research. The control groups RT1 and RT2, as well as the treatment teams DVC1 and DVC2, were each allocated 20 projects for each case study. This may be considered a good balance. The 20 projects each for the first case study and second case study to departments P and Q, respectively, also indicate a good balance. For each project, the 20 risks are encountered in the four phases of software development - requirements analysis, design, programming and testing, and implementation and release. All the risks are also considered for each project. The use of the same number of risks for all projects and phases helps to minimize bias.

Table 6.1: Balanced block for research design

Measure	Case Study 1 (Department P)		Case Study 2 (Department Q)	
	ISO31000 used RT1	E-RIAM used DVC1	ISO31000 used by RT2	E-RIAM used by DVC2
Number of Projects	20	20	20	20
Total number of projects	40		40	

Number of potential risks for identification and assessment in Requirements Analysis phase	20	20
Number of potential risks for identification and assessment in Design phase	20	20
Number of potential risks for identification and assessment in Programming and Testing phase	20	20
Number of potential risks for identification and assessment in Implementation and release phase	20	20
Number of Potential risks for identification and assessment in Entire Project	20	20
Total potential risks	100	100

Table 6.1 shows a good balance in the number of risks considered for the control teams RT1 and RT2, as well as the treatment teams DVC1 and DVC2. Three types of bias pertinent to research validity was considered:

6.2.1 Response Bias

Response bias occurs when there are few participants involved in the research. To prevent this, suitably qualified people who have sufficient experience in risk analysis were invited to participate in the case studies.

6.2.2 Observer Bias

Observer bias occurs when the researcher or people involved in the project unduly influence the opinions, ideas, and decisions of the participants (risk analysts). To avoid this bias, the team members studied the documents separately before the brainstorming sessions, to arrive at the final decisions. In this way, their opinions are not biased and not influenced by anyone.

6.2.3 Selection Bias

Selection bias occurs when participants are influenced or guided into achieving pre-determined results or those based on wrong criteria. This bias is avoided in this research because the analysts were selected based on their qualification at random. Hence, they do not have any relationship with the researcher.

6.3 External Validity of Research

External validity, which is fully explained in Chapter 3 of this thesis, was applied to the methodology as well as the data collection and data analysis stages of the research. The selected projects were medium-sized, and the selected company is a big commercial software company with more than 100 employees. Two case studies were conducted using two separate groups of risk analysts. The results were similar and the hypotheses were confirmed in both case studies, and these indicate the validity of the research findings. A total of 40 projects were used in the case studies, and data collection took almost seven months. The top 20 risks were considered to give higher accuracy in the findings, compared with the majority of other studies, which had used only on the top 10 risks.

6.4 Ethical Issues of Research

The selected risk analysts in the case studies had collaborated voluntarily. The profiles of the analysts, the project managers who made available project information, and the manager of the company who provided the case studies, were not disclosed during the research. The analysts were assured that information on their identity will not be disclosed. Moreover, information as to the efficiency of analysts in risk identification or risk assessment was not and will not be provided to their managers, thus, their job performance will not be affected. Project information was provided to the external experts with the consent of the company's managers. The external experts were committed to keep this information confidential. Information and documents which were not necessary for the case study, such as the identity of people involved in system development, were not made known to the risk teams.

The tools used to collect and analyze data were installed on the servers of the two departments of the company. Unauthorized people have no access to the tools and data. The access rights of the participants in the study was controlled based on their role in the case study, thus preserving data and information confidentiality.

6.5 Problems Encountered During Research

The process of conducting the research went generally quite smoothly. Nonetheless, the following issues, which needed prompt attention, were resolved before the start of the case studies:

6.5.1 Finding Appropriate Companies and Projects for the Case Study

Some companies or project managers were reluctant to participate in the case studies because of fear of disclosure of confidential information pertaining to medium- and large-sized projects. To find a company that would collaborate, we sent emails to many software companies, and we also enlisted the assistance of our friends in the academia as well as those in the software industry. We earned the trust of companies by presenting a consent form, and giving our commitment to protection of their confidential information, and undertaking to return all information and data pertaining to the projects and case studies to the company without the identities of team members who participated. Assurance was also given that no unauthorized copy will be made of any information or documents.

6.5.2 Number of Projects

A sufficient number of datasets had to be prepared in order to conduct the statistical tests in order to give validity to the results. There should be more than 30 projects, and they should have some uniformity so that the results could be compared. Any difference in identity, shape, or construct could pose problems to the research validity. Aside from the company's manager, the project managers also have to be briefed and they have to agree to cooperate. This problem was sorted out by communicating with the company's manager and the company's IT senior manager and department and project managers, who were encouraged to collaborate. Face-to-face meetings also helped to fix the aforesaid problem along with the consent form, and letter of assurance on data confidentiality and privacy protection.

6.5.3 Number of Risk Analysts Participating in Case Studies

It was rather difficult to find 20 risk analysts and encourage them to spend several hours a day for over six months on the case studies. This problem was resolved with the treatment team members who were told that the use of the proposed model and its support tool can enhance their knowledge and capability. When they present the results to the control team, if would also make them to be aware of their own weaknesses and strengths, thus, enriching their experience. The support provided by the company's senior managers and departments to this study also rekindled the interest of team members to participate actively in the case study.

6.6 The Specific Contributions of This Research

This research has established a strong case for the use of external experts for identifying and assessing software project risks, improving risk management, and consequently, improving software project management. The contribution made by this research is clearly seen from the three advantages the E-RIAM model and its support tool have over the ISO 31000 standard: i) more accurate identification of additional number of important risks; ii) more accurate assessment of important risks; and iii) better composition of risk teams. In this section, the most significant contributions of the E-RIAM model and its support tool were compared with risk identification and assessment results of the ISO 31000. The advantages of E-RIAM are elaborated in the sections, below.

6.6.1 More Accurate Identification of Higher Number of Important Risks

Identifying a higher number of important risks that could have high impacts on software and can cause serious threats to the project in terms of time and cost over-run, or performance, is the most important contribution of this research. The failure to identify even one important risk is enough to seriously threaten or cause failure to the software project. Many reports have testified to this serious oversight.

6.6.2 More Accurate Assessment of Likelihood of Risk Occurrence and Impact

More accurate recognition of the likelihood of materialised risks and more accurate recognition of risk impacts, i.e. undesirable consequences in case of risk occurrence, are other important achievements of this study. Analysis of the data collected from the case studies shows that the E-RIAM model and its support tool can assess more accurately the likelihood of risk occurrence and its impact (in case of occurrence). This is an important contribution of this study. More accurate risk assessment contributes positively to improved risk management, especially in making more accurate allocation of budget, and in better formulation of risk response plans. Improved project management will eventually lead to improved software product in terms of cost, time, and performance.

6.6.3 Better Composition of Risk Teams

This research had emphasised on the formation and composition of risk teams with internal risk analysts, and the involvement of external risk experts. The structure of the risk teams, their role and interaction, decision correction cycle, and judgments are all communicated transparently and systematically, and their competency is assessed through analysis of the collected data. Another contribution of this study is that the internal risk analysts and external risk experts can gain their own unique experience in the use of the proposed model, at the same time. The advantage of using internal analysts is their familiarity with the software risks the company has encountered before, and also their familiarity with similar projects whose risks they have analyzed in the past. On the other hand, the use of experienced external analysts who have neither interest nor bias towards the company, nor concern about their job security, can offer impartial and unbiased views or assessment that can complement the decisions and opinions of risk analysts.

6.6.4 Use of Res-DVC Support Tool

The support tool, Res-DVC, incorporates all the features and workflow of the E-RIAM model. The various databases stores and updates project information from the different phases of each project. Furthermore, the tool also facilitates the process of risk identification, the entry of risk information into the system, and risk verification by the experts. By storing information on likelihood of occurrence and impact of each risk, as well as the comments, will facilitate the revision and verification of every risk. In addition, the second round of verification by risk experts will be more convenient and more accurate as the risk assessment information is already recorded in the database. Risk information can be stored and retrieved from the system and reports can be produced. This eliminates or reduces the operational errors, which would otherwise occur if the processes were to be done manually. Controlling the access of analysts and experts to the information on the projects, phases, and risks, reduces the probability of errors, and provides flexibility in simultaneous identification of the risk of several phases or several projects. In addition, as the tool is an integrated system, all functions are performed using the same software, thus, dispensing with the need for other software.

6.6.5 Registering Causes and Effects of Risks

The proposed model also records the reasons for risk occurrence and their potential consequences. Other models do not provide this useful feature, because there has not been any report on this from the literature review. In E-RIAM this information may be modified in the support tool and they may be updated when the risks eventually materialised. E-RIAM also allows us to have a comprehensive information bank on the project types and more information is added from one project to another. This will contribute to improvement to the risk identification and risk assessment processes. Also, the availability of information pertaining to risk causes and consequences would be useful for formulating appropriate risk response plan.

6.6.6 Meeting of Risk Analysts and Experts and Preventing Divergence of Opinion

ISO 31000 recommends having meetings for the exchange of ideas among risk analysts, such as through brainstorming among risk analysts held in a room or using online tools. The tool can be used to keep information and decisions made in all the meetings and also to forward the sub-team decisions to the DVC teams.

ISO 31000 also mentioned the use of different methods to combine analysts’ opinions; however, it does not recommend any specific mechanism for doing this. The E-RIAM recommends brainstorming to converge the ideas and recommendations of the risk analysts and the risk experts. Brainstorming is held after the individual investigations, and after the analysts and experts have made a decision, thus reducing any possible biases. Consensus or vote-taking is incorporated into the model and the session scheduling is controlled by a supervisor. Thus, there will be no dispute in the decision-making process, and in arriving at the final results of the risk identification or risk assessment processes.

6.6.7 Selecting Internal and External Risk Analysts

There is no specific recommendation in ISO 31000 regarding the selection of internal and external risk analysts. In the E-RIAM model, internal analysts are selected based on their qualifications and experience. It recommends analysts with more than 10 years of experience in conducting risk identification and risk assessment in similar projects over the past three years. In ISO 31000, the use of external risk analysts can only be considered, whereas the use of external risk analysts is a feature incorporated in the E-RIAM model, specifically as verifiers of risk analysts' comments in the risk identification and risk assessment processes. These external risk analysts must have more than 20 years of experience in risk identification and risk assessment of similar type of projects.

6.6.8 Roles and Responsibilities of Risk Analysts in the E-RIAM Model

In ISO 31000, the participants play various roles in the risk identification and risk assessment processes. The extended roles they play weaken management of their core tasks and this can cause responsibility overlap. In the E-RIAM, however, the roles have become more transparent and more confined. A system supervisor can be considered for the role of risk coordinator, aimed at improving performance. Finally, there is no direct communication between the internal and external risk analysts in ISO 31000. In E-RIAM, the verification of the initial findings of internal analysts and the allowed modification of their decisions by external experts, are essential features of the model.

Table 6.2 shows a brief comparison between the features of the E-RIAM model and the ISO 31000 standard.

Table 6.2: Comparison between the E-RIAM model and the ISO 31000 standard

No.	Characteristics	E-RIAM	ISO 31000
1	The relationship between internal and external risk analysts	The verification of the initial findings of internal analysts and the modification of their decisions by external experts are essential features of the model.	The relationship between internal and external risk analysts is not mentioned.
2	Use of automatic support tools for risk identification and risk assessment	The model support tool (Res-DVC) incorporates all the features of E-RIAM and facilitates collection of data related to risk identification and risk assessment.	Use of automatic tools is recommended; however, no specific tool is suggested in this standard.
3	The reason for risks occurrence and their potential consequences	The reason for risks occurrence and their potential consequences are mentioned in other studies and official reports for candidate risks, facilitating risk identification and assessment.	Registering and recording information on why risks occur and their consequences are recommended; however, there is no database for this purpose and on the whole, the issue is not addressed.
4	Eliminating incompatibilities	Consensus or vote-taking is considered and implemented in the model with the session schedule being controlled by a supervisor. Thus, there will be no dispute in final decision-making in the risk identification and risk assessment processes.	No specific mechanism is recommended for eliminating divergence in the decisions of the internal analysts and external experts.
5	Meetings of risk analysts and experts	Brainstorming among analysts held in a room or via online tools. The tool can be used to keep information and decision made in all the meetings, and to forward sub-team decisions to the DVC teams.	Has recommendations for exchange of ideas among risk analysts.
6	Selecting internal risk	Internal analysts are	There is no specific

	analysts	selected based on their qualifications and experience. Recommends analysts with more than ten years of experience in conducting risk identification and risk assessment for similar projects over the past three years.	recommendation regarding the selection of internal risk analysts.
7	Responsibilities and roles in the risk identification and risk assessment processes	The roles of the participants are more transparent and more confined. A system supervisor can be considered for the role of risk coordinator, aimed at improving performance.	Participants play various roles in the risk identification and risk assessment processes. This weakens management of their core tasks and can cause responsibility overlap.
8	Use of external risk analysts	The use of external risk analysts is incorporated, specifically as verifiers of risk analysts' comments in the identification and risk assessment processes. External risk analysts must have more than 20 years of experience in risk identification and risk assessment in similar types of projects.	Use of external risk analysts can only be considered.
9	Use of internal risk analysts	Internal analysts are selected based on their qualification and experience. Recommends analysts with more than 10 years of experience in risk identification and risk assessment in similar projects over the past three years.	There is no specific recommendation regarding the selection of internal risk analysts.

6.7 Future Research

Limitations, delimitations, and constraints of this study may change in future. These changes should lead to new knowledge or findings that will extend the scope of this study or bring about improvement in the risk management process. Better, more effective models can be developed by addressing all the identified threats and risks of this research. Future research, which can complement or extend the scope of this study from different aspects include:

i) Identifying and assessing the different candidate risks at other phases of a project, which are not covered in this study, and building a comprehensive list of risks;

ii) Identifying new risks or risks pertaining to non-commercial projects, and keep a list of risks together with the causes and consequences of each risk, in the database;

iii) Exploring the optimum number of risk team members for large projects and those with tight delivery deadline;

iv) Brainstorming among risk analysts and risk experts, is highly advocated in this model. Future study could consider giving more weight to the judgment of senior risk experts;

v) This study involved 40 projects - an appropriate number for an academic research. It is good to explore a higher number of projects conducted in different development environments and having different types of threats and risks. The findings will give a wider perspective of potential threats and risks in the risk management plans of various organisations;

vi) Use of other methods, such as experiments, to strengthen research validity to make the findings more acceptable and reliable;

vii) Explore whether having more than three external risk experts can result in a more efficient and accurate risk management process;

viii) In this study, risk was investigated from a negative perspective - having adverse or undesirable effects on software projects. However, some academic standards and references consider risks from two aspects - a threat and an opportunity. This means that certain risks that have adverse effects, can also have positive effects and create opportunities. More in-depth research should be conducted to understand this interesting perspective of risks;

ix) Conduct research on the risk of various software projects including those that have different management environment or those for the development of different software products. The efficiency of any model is best assessed on how it handles projects with different complexities pertaining to risk management and product outcomes;

x) More features should be incorporated into the support tool for use in future case studies, such as standardising the descriptions and information of the risks. This feature will enhance the capability of E-RIAM and its support tool for use in case studies with different complexities.

REFERENCES

Abdullah, T., Mateen, A., Sattar, A., & Mustafa, T. (2010). Risk Analysis of Various Phases of Software Development Models. European Journal Of Scientific Research, 40(3), 369-376.

Ahonen, J. J., & Savolainen, P. (2010). Software engineering projects may fail before they are started: Post-mortem analysis of five cancelled projects. Journal of Systems and Software, 83(11), 2175–2187. doi:10.1016/j.jss.2010.06.023

Ahonen, J. J., Savolainen, P., Merikoski, H., & Nevalainen, J. (2015). Reported project management effort, project size, and contract type. Journal of Systems and Software, 109, 205–213. doi:10.1016/j.jss.2015.08.008

Alsoghayer, R., & Djemame, K. (2014). Resource failures risk assessment modelling in distributed environments. Journal of Systems and Software, 88, 42–53. doi:10.1016/j.jss.2013.09.017

Ammar, H.H.; Nikzadeh, T.; Dugan, J.B.; , "Risk assessment of software-system specifications,", IEEE Transactions on Reliability, 50(2)171-183, Jun 2001doi: 10.1109/24.963125

ANSI/ASSE Z690.1–2011. (2011). Vocabulary for Risk Management. Washington, D.C.: American National Standards Institute.

ANSI/ASSE Z690.2–2011. (2011). Risk Management Principles and Guidelines. Washington, D.C.: American National Standards Institute.

ANSI/ASSE Z690.3–2011. (2011). Risk Assessment Techniques. Washington, D.C.: American National Standards Institute.

Antinyan, V., Staron, M., Meding, W., Osterstrom, P., Wikstrom, E., Wranker, J., … Hansson, J. (2014). Identifying risky areas of software code in Agile/Lean software development: An industrial experience report. 2014 Software Evolution Week - IEEE Conference on Software Maintenance, Reengineering, and Reverse Engineering (CSMR-WCRE). doi:10.1109/csmr-wcre.2014.6747165

Aruna, G. (2016). Impact of Team Skills in Software Quality - A Study on Twin Cities Small and Medium Software Development Units

Avdoshin, S. M., & Pesotskaya, E. Y. (2011). Software risk management. 2011 7th Central and Eastern European Software Engineering Conference (CEE-SECR). doi:10.1109/cee-secr.2011.6188471

Bardhan, I. R.; Kauffman, R. J.; Naranpanawe, S.; , "IT project portfolio optimization: A risk management approach to software development governance," IBM Journal of Research and Development ,54(2), 2:1-2:18, March-April 2010 doi: 10.1147/JRD.2009.2039824

Basile, C., Canavese, D., D'Annoville, J., Sutter, B. D., & Valenza, F. (2015). Automatic Discovery of Software Attacks via Backward Reasoning. 2015

IEEE/ACM 1st International Workshop on Software Protection. doi:10.1109/spro.2015.17

Beaver, J. M., & Schiavone, G. A. (2006). The effects of development team skill on software product quality. ACM SIGSOFT Software Engineering Notes, 31(3), 1. doi:10.1145/1127878.1127882

Beins, B. C. (2013). Experimenter Effects. The Encyclopedia of Cross-Cultural Psychology, 527–529. doi:10.1002/9781118339893.wbeccp214

Benaroch, M., & Appari, A. (2010). Financial Pricing of Software Development Risk Factors. IEEE Software, 27(5), 65–73. doi:10.1109/ms.2010.28

Benaroch, M., & Goldstein, J. (2009). An Integrative Economic Optimization Approach to Systems Development Risk Management. IIEEE Trans. Software Eng., 35(5), 638–653. doi:10.1109/tse.2009.25

Betz, S., Hickl, S., & Oberweis, A. (2011). Risk Management in Global Software Development Process Planning. 2011 37th EUROMICRO Conference on Software Engineering and Advanced Applications. doi:10.1109/seaa.2011.64

Boehm, B. (2014). Software Project Risk and Opportunity Management. Software Project Management in a Changing World, 107–121. doi:10.1007/978-3-642-55035-5_5

Boehm, B. W. (1988). "A spiral model of software development and enhancement," Computer , 21(5),61-72, May 1988. doi: 10.1109/2.59

Boehm, B., & Bhuta, J. (2008). Balancing Opportunities and Risks in Component-Based Software Development. IEEE Software, 25(6), 56–63. doi:10.1109/ms.2008.145

Boehm, B.W.; DeMarco, T. (1997). "Software risk management," Software, IEEE , 14(3), 17-19. doi: 10.1109/MS.1997.589225

Burke, R., & Barron, S. (Eds.). (2012). Project Management Leadership. doi:10.1002/9781119207986

Carlin, J., & Doyle, L. (2001). 4: Basic concepts of statistical reasoning: Hypothesis tests and the t-test. J Paediatr Child Health, 37(1), 72–77. doi:10.1046/j.1440-1754.2001.00634.x

Chang, C. P. (2015). Software Risk Modeling by Clustering Project Metrics. International Journal of Software Engineering and Knowledge Engineering, 25(06), 1053–1076. doi:10.1142/s0218194015500175

Charette, R. N. (2015). Enterprise Risk Management. The Next Wave of Technologies, 265–283. doi:10.1002/9781119199946.ch15

Chen, Y. (2010). Fuzzy AHP-based method for project risk assessment. 2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery. doi:10.1109/fskd.2010.5569128

Choetkiertikul, M., Dam, H. K., Tran, T., & Ghose, A. (2015). Characterization and Prediction of Issue-Related Risks in Software Projects. 2015 IEEE/ACM 12th Working Conference on Mining Software Repositories. doi:10.1109/msr.2015.33

Choo, A. S. (2014). Defining Problems Fast and Slow: The U-shaped Effect of Problem Definition Time on Project Duration. Prod Oper Manag, 23(8), 1462–1479. doi:10.1111/poms.12219

Choo, B. S. Y., & Goh, J. C. L. (2015). Pragmatic adaptation of the ISO 31000:2009 enterprise risk management framework in a high-tech organization using Six Sigma. Int J Acc & Info Management, 23(4), 364–382. doi:10.1108/ijaim-12-2014-0079

Cortellessa, V.; Goseva-Popstojanova, K.; Kalaivani Appukkutty; Guedem, A.R.; Hassan, A.; Elnaggar, R.; Abdelmoez, W.; Ammar, H. H.; (2005). "Model-based performance risk analysis", IEEE Transactions on Software Engineering, 31(1) 3-20, Jan. 2005

Creswell, J. W. (2012). Educational research: Planning, conducting, and evaluating quantitative and qualitative research. 4th Ed., Upper Saddle River, NJ: Pearson. 2012

Cronbach, L. J., & Meehl, P. E. (1955). Construct validity in psychological tests. Psychological Bulletin, 52(4), 281–302. doi:10.1037/h0040957

Daojin Fan. (2010). Analysis of critical success factors in IT project management. 2010 2nd International Conference on Industrial and Information Systems. doi:10.1109/indusis.2010.5565760

Dash, R., & Dash, R. (2010). Risk Assessment Techniques for Software Development. European Journal Of Scientific Research, 42(4), 615-622.

De Bakker, K., Boonstra, A., & Wortmann, H. (2010). Does risk management contribute to IT project success? A meta-analysis of empirical evidence. International Journal of Project Management, 28(5), 493–503. doi:10.1016/j.ijproman.2009.07.002

Dedolph, F. M. (2003). The neglected management activity: Software risk management. Bell Labs Technical Journal, 8(3), 91–95. doi:10.1002/bltj.10077

Department of Defense (2015). Risk, Issue, and Opportunity Management Guide for Defense Acquisition Programs.

Di Tullio, D., & Bahli, B. (2013). The impact of Software Process Maturity on Software Project Performance: The Contingent Role of Software Development Risk. Systèmes D'information & Management, 18(3), 85. doi:10.3917/sim.133.0085

Dongarra, J., Beckman, P., Aerts, P., Cappello, F., Lippert, T., Matsuoka, S., … Valero, M. (2009). The International Exascale Software Project: a Call To Cooperative Action By the Global High-Performance Community. International Journal of High Performance Computing Applications, 23(4), 309–322. doi:10.1177/1094342009347714

Elzamly, A., & Hussin, B. (2014). Managing Software Project Risks (Analysis Phase) with Proposed Fuzzy Regression Analysis Modelling Techniques with Fuzzy Concepts. CIT, 22(2), 131. doi:10.2498/cit.1002324

Elzamly, A., & Hussin, B. (2015). Classification and Identification of Risk Management Techniques for Mitigating Risks with Factor Analysis Technique in Software Risk Management. Review of Computer Engineering Research, 2(1), 22–38. doi:10.18488/journal.76/2015.2.1/76.1.22.38

Elzamly, A., & Hussin, B. (2015). Modelling and Evaluating Software Project Risks with Quantitative Analysis Techniques in Planning Software Development. CIT, 23(2), 123. doi:10.2498/cit.1002457

Elzamly, A., Hussin, B., & Salleh, N. M. (2016). Top Fifty Software Risk Factors and the Best Thirty Risk Management Techniques in Software Development Lifecycle for Successful Software Projects. International Journal of Hybrid Information Technology, 9(6), 11–32. doi:10.14257/ijhit.2016.9.6.02

Eveleens, J. L., & Verhoef, C. (2010). The rise and fall of the Chaos report figures. IEEE Software, 27(1), 30–36. doi:10.1109/ms.2009.154

Federal Aviation Administration. (2003). Advisory Circular 39–8. Washington, D.C.: Federal Aviation Administration.

Ferguson, R. (2004), "A Project Risk Metric", CrossTalk, The Journal of Defense Software Engineering, 17(4), 12-15, Apr

Fischer, R. (2004). Standardization to Account for Cross-Cultural Response Bias: A Classification of Score Adjustment Procedures and Review of Research in JCCP. Journal of Cross-Cultural Psychology, 35(3), 263–282. doi:10.1177/0022022104264122

Frey, T. (2014). IT project portfolio management – Evaluating, selecting, and staffing IT projects. Governance Arrangements for IT Project Portfolio Management, 35–85. doi:10.1007/978-3-658-05661-2_3

Fulkerson, R. E., Thompson, R. L., & Thompson, E. H. (2015). Team Member Perceptions of Software Team Leader Communication Influencing Motivation for Achievement of Project Goals. Journal of Psychological Issues in Organizational Culture, 6(3), 24–39. doi:10.1002/jpoc.21202

Gorla, N. (2012). Information Systems Service Quality, Zone of Tolerance, and User Satisfaction. Journal of Organizational and End User Computing, 24(2), 50–73. doi:10.4018/joeuc.2012040104

Gorla, N., Somers, T. M., & Wong, B. (2010). Organizational impact of system quality, information quality, and service quality. The Journal of Strategic Information Systems, 19(3), 207–228. doi:10.1016/j.jsis.2010.05.001

Graziano, A. M., & Raulin, M. L. (2014). Research methods – A process of inquiry. 8th ed. Essex: England: Pearson Education Limited.

Green, S. B., Salkind, N. J. and Akey, T. M. Using SPSS for Windows: Analysing and Understanding Data. (1997). Upper Saddle River, NJ, Prentice-Hall Inc.

Hoermann, S., Aust, M., Schermann, M., & Krcmar, H. (2012). Comparing Risks in Individual Software Development and Standard Software Implementation Projects: A Delphi Study. 2012 45th Hawaii International Conference on System Sciences. doi:10.1109/hicss.2012.168

Hong-bo, L., Hai-yang, Y., & Yan-ling, H. (2010). Research and application on risk assessment quantitative method based on fuzzy AHP. 2010 5th International Conference on Computer Science & Education. doi:10.1109/iccse.2010.5593805

Huang, Y.-P., & Lin, J.-W. (2012). Interactive remote computations for retaining wall design and risk assessment. Natural Hazards, 66(2), 985–993. doi:10.1007/s11069-012-0519-4

Husain, M., Shukla, S. (2015). Study of Software Risk Analysis Models on Distributed Systems. International Journal of Research and Development in Applied Science and Engineering (IJRDASE)

Hydari, H. (2015). The Rules of Project Risk Management: Implementation Guidelines for Major Projects. Project Management Journal, 46(4), e4–e4. doi:10.1002/pmj.21516

Islam, S., Houmb, S. H., Mendez-Fernandez, D., & Joarder, M. M. A. (2009). Offshore-outsourced software development risk management model. 2009 12th International Conference on Computers and Information Technology. doi:10.1109/iccit.2009.5407292

ISO Guide 73. (2009). Risk Management Terminology. Geneva, Switzerland: International Organization for Standardization (ISO).

ISO/IEC 16085:2006, Standard for Software Engineering - Software Life Cycle Processes - Risk Management. (n.d.). doi:10.1109/ieeestd.2006.288594

Jaafar, J., Janjua, U. I., & Lai, F. W. (2015). Software Effective Risk Management: An Evaluation of Risk Management Process Models and Standards. Information Science and Applications, 837–844. doi:10.1007/978-3-662-46578-3_99

Jensen, J. R. (2009). Computerized occurrence reporting system: Development, implementation, and impact. Perspectives in Healthcare Risk Management, 10(1), 10–16. doi:10.1002/jhrm.5600100105

Jian Song, & Zhaoyang Dong. (2010). Risk evaluation in urban information system based on hierarchy fuzzy method. 2010 2nd International Conference on Computer Engineering and Technology. doi:10.1109/iccet.2010.5486017

Jose, V. R. R., & Winkler, R. L. (2009). Evaluating Quantile Assessments. Operations Research, 57(5), 1287–1297. doi:10.1287/opre.1080.0665

Jowah, L. E. (2015). Project Management Tools and Techniques for Effective Project Execution. JBE, 6(10), 1762–1774. doi:10.15341/jbe(2155-7950)/10.06.2015/011

Judith C. and Kate M. (2007), 'Information Technology Workforce skills: Does Size matter?', information Systems Management, 24(4) 345-359.

Kondabagil, J. (2007). Risk Management in Electronic Banking. doi:10.1002/9781118390436

Koolmanojwong, S. (2014). Top-10 risks in real-client software engineering class projects. 2014 IEEE 27th Conference on Software Engineering Education and Training (CSEE&T). doi:10.1109/cseet.2014.6816805

Kruchten, P., R.L. Nord, and I. Ozkaya, (2012). Technical debt: from metaphor to theory and practice. IEEE Software, 2012. 29(6): p. 18-21

Kumar, C., & Yadav, D. K. (2015). A Probabilistic Software Risk Assessment and Estimation Model for Software Projects. Procedia Computer Science, 54, 353–361. doi:10.1016/j.procs.2015.06.041

Kutsch, E., & Hall, M. (2009). The rational choice of not applying project risk management in information technology projects. Project Management Journal, 40(3), 72–81. doi:10.1002/pmj.20112

Kwan, T. W., & Leung, H. K. N. (2011). A Risk Management Methodology for Project Risk Dependencies. IIEEE Trans. Software Eng., 37(5), 635–648. doi:10.1109/tse.2010.108

Lai, S. T. (2014). A WBS-Based Plan Changeability Measurement Model for Reducing Software Project Change Risk. Lecture Notes on Software Engineering, 94–99. doi:10.7763/lnse.2014.v2.102

Larson, E. W. & Gray, C. F. (2014). Project Management: The Managerial Process. McGraw-Hill International Edition.

Lazzerini, B., & Mkrtchyan, L. (2011). Analyzing Risk Impact Factors Using Extended Fuzzy Cognitive Maps. IEEE Systems Journal, 5(2), 288–297. doi:10.1109/jsyst.2011.2134730

Lee, K., Oh, S., & Yoo, J. K. (2013). Method-Free Permutation Predictor Hypothesis Tests in Sufficient Dimension Reduction. Communications for Statistical Applications and Methods, 20(4), 291–300. doi:10.5351/csam.2013.20.4.291

Leitch, M. (2010). ISO 31000:2009-The New International Standard on Risk Management. Risk Analysis, 30(6), 887–892. doi:10.1111/j.1539-6924.2010.01397.x

Lim, E., N. Taksande, and C. seaman. (2012). A Balancing Act: What software practitioners have to say about technical debt. IEEE Software, 2012. November/December: p. 22-27.

Lindholm, C. (2015). Involving user perspective in a software risk management process. J. Softw. Evol. and Proc., 27(12), 953–975. doi:10.1002/smr.1753

Liu, D., Wang, Q., & Xiao, J. (2009). The role of software process simulation modeling in software risk management: A systematic review. 2009 3rd International Symposium on Empirical Software Engineering and Measurement. doi:10.1109/esem.2009.5315982

Liu, J. Y. C., Chen, H. G., Chen, C. C., & Sheu, T. S. (2011). Relationships among interpersonal conflict, requirements uncertainty, and software project performance. International Journal of Project Management, 29(5), 547–556. doi:10.1016/j.ijproman.2010.04.007

Marchewka, M. (2015). English. JBE, 6(5), 996–1002. doi:10.15341/jbe(2155-7950)/05.06.2015/015

Molokken-Ostvold, K., & Jorgensen, M. (2005). A comparison of software project overruns - flexible versus sequential development models. IIEEE Trans. Software Eng., 31(9), 754–766. doi:10.1109/tse.2005.96

Nguyen, T. (2014). Software Project Management - Towards Failure Avoidance. Proceedings of the 9th International Conference on Software Engineering and Applications. doi:10.5220/0004992605600567

Olteanu, F. C., & gheorghe, C. (2016). Aspects regarding the qualitative analysis of risks due to the occurrence of low probability and very high impact events. Review of the Air Force Academy, 14(1), 133–140. doi:10.19062/1842-9238.2016.14.1.19

Parry, M., Dawid, A. P., & Lauritzen, S. (2012). Proper local scoring rules. The Annals of Statistics, 40(1), 561–592. doi:10.1214/12-aos971

Parthasarathy, S., & Sharma, S. (2016). Impact of customization over software quality in ERP projects: an empirical study. Software Quality Journal. doi:10.1007/s11219-016-9314-x

PENG, Y., KOU, G., WANG, G., WANG, H., & KO, F. S. (2009). Empirical Evaluation of Classifiers for Software Risk Management. International Journal Of Information Technology & Decision Making, 8(4), 749-767.

Persson, J.S.; Mathiassen, L.; Boeg, J.; Madsen, T.S.; Steinson, F.; , "Managing Risks in Distributed Software Projects: An Integrative Framework,", IEEE Transactions on Engineering Management, 56(3) 508-532, Aug. 2009 doi:10.1109/TEM.2009.2013827

PMI Southwest Missouri Chapter (2013). Project management body of knowledge (PMBOK) guide: 5th edition. Available at http://pt.slideshare.net/yaparicio/pmbok-5-thedition.

Poth, A., & Sunyaev, A. (2014). Effective Quality Management: Value- and Risk-Based Software Quality Management. IEEE Software, 31(6), 79–85. doi:10.1109/ms.2013.138

Pozzebon, R. C. B., Silva, L. A. L., Fontoura, L. M., & Campbell, J. A. (2014). Argumentation schemes for the reuse of argumentation information in collaborative risk management. Proceedings of the 2014 IEEE 15th International Conference on Information Reuse and Integration (IEEE IRI 2014). doi:10.1109/iri.2014.7051888

Project Management Institute. (2013). A guide to the Project Management Body of Knowledge (PMBOK guide) (5th ed.). Newton Square, PA: Project Management Institute.

Purdy, G. (2010). ISO 31000:2009-Setting a New Standard for Risk Management. Risk Analysis, 30(6), 881–886. doi:10.1111/j.1539-6924.2010.01442.x

Quadri, A.T., Komal,M, Zaineb Khalil, Z. (2015). A Comprehensive Study on Risk Analysis and Risk Management in IT Industry, International Journal of Computer and Communication System Engineering (IJCCSE), 2 (4), 2015, 561-568

Rabbi, M. F., & Mannan, K. O. B. (2008). A Review of Software Risk Management for Selection of Best Tools and Techniques. 2008 Ninth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing. doi:10.1109/snpd.2008.127

Ravindranath Pandian, C. (2006). Applied Software Risk Management. doi:10.1201/9780849305313

Ray, B. K., Tao, S., Olkhovets, A., & Subramanian, D. (2013). A decision analysis approach to financial risk management in strategic outsourcing contracts. EURO J Decis Process, 1(3-4), 187–203. doi:10.1007/s40070-013-0013-6

Reifer, D. (2002). Ten deadly risks in Internet and intranet software development. IEEE Software, 19(2), 12–14. doi:10.1109/52.991324

Reifer, D. J., & Boehm, B. (2007). Software Management. doi:10.1109/9780470049167

Rekha, J. H., & Parvathi, R. (2015). Survey on Software Project Risks and Big Data Analytics. Procedia Computer Science, 50, 295–300. doi:10.1016/j.procs.2015.04.045

Rivard, S., St-James, Y., & Cameron, A. (2011). Software Project Risk Drivers as Project Manager Stressors and Coping Resources. 2011 44th Hawaii International Conference on System Sciences. doi:10.1109/hicss.2011.381

Runeson, P, & H¨ost, M. (2009). Guidelines for conducting and reporting case study research in software engineering," Empirical Software Engineering, 14, 131–164, 2009.

Roy, B., Dasgupta, R., & Chaki, N. (2015). A Study on Software Risk Management Strategies and Mapping with SDLC. Advanced Computing and Systems for Security, 121–138. doi:10.1007/978-81-322-2653-6_9

Ruhe, G.; Saliu, M.O. (2005). "The art and science of software release planning," Software, IEEE , 22(6) 47- 53. doi: 10.1109/MS.2005.164\

S. Islam, H. Mouratidis, and E. R. Weippl. (2014). "An empirical study on the implementation and evaluation of a goal-driven software development risk management model," Information and Software Technology, 56(2), 117–133.

Salmeron, J. L., & Lopez, C. (2012). Forecasting Risk Impact on ERP Maintenance with Augmented Fuzzy Cognitive Maps. IIEEE Trans. Software Eng., 38(2), 439–452. doi:10.1109/tse.2011.8

Salo, O., & Abrahamsson, P. (2004). Empirical evaluation of agile software development: The controlled case study approach. In F. Bomarius and H. Iida, editors, *Proceedings of the 5th International Conference Product Focused Software Process Improvement*, Number 3009 in *LNCS*, Springer-Verlag, 2004, 408–423.

Samantra, C., Datta, S., Mahapatra, S.S. and Debata, B.R. (2016) 'Interpretive structural modelling of critical risk factors in software engineering project', Benchmarking: An International Journal, 23(1), 2–24. doi: 10.1108/bij-07-2013-0071.

Saravanamuthu, K., Brooke, C., & Gaffikin, M. (2013). The next phase in information management: using risk to integrate data and facilitate social learning about sustainability. International Journal of Business and Systems Research, 7(3), 266. doi:10.1504/ijbsr.2013.055320

Sauer, C., Gemino, A., & Reich, B. H. (2007). The impact of size and volatility on IT project performance. Commun. ACM, 50(11), 79–84. doi:10.1145/1297797.1297801

Serra, C. E. M. & Kunc, M. (2014). Benefits Realisation Management and its influence on project success and on the execution of business strategies. Also, no. International Journal of Project Management, Issuehttp://dx.doi.org/10.1016/j.ijproman.2014.03.011.

Shahzad, B. (2014). "Identification of Risk Factors in Large Scale Software Projects:," International Journal of Knowledge Society Research, 5(1) 1–11, 2014.

Shikha, & Selvarani, R. (2012). An Efficient Method of Risk Assessment Using Intelligent Agents. 2012 Second International Conference on Advanced Computing & Communication Technologies. doi:10.1109/acct.2012.19

Shrivastava, S. V., & Rathod, U. (2015). Categorization of risk factors for distributed agile projects. Information and Software Technology, 58, 373–387. doi:10.1016/j.infsof.2014.07.007

Slyngstad, O. P. N., Conradi, R., Babar, M. A., Clerc, V., & Vliet, H. van. (2008). Risks and Risk Management in Software Architecture Evolution: An Industrial Survey. 2008 15th Asia-Pacific Software Engineering Conference. doi:10.1109/apsec.2008.70

Smith, D., & Politowski, R. (2013). Managing Risk the ISO 31000 Way. doi:10.3403/9780580675126

Sonchan, P., & Ramingwong, S. (2014). Top twenty risks in software projects: A content analysis and Delphi study. 2014 11th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON). doi:10.1109/ecticon.2014.6839820

Spinellis, D. (2012) "Don't Install Software by Hand," in IEEE Software, 29(4) 86-87. doi: 10.1109/MS.2012.85

Standish Group, (2015). “CHAOS Manifesto 2015,” Standish Group International Inc.

Stephen N. L. (2013): Risk Management Terminology, Quality Engineering, 25:3, 292-297

Sudhaman, P., & Thangavel, C. (2015). Efficiency analysis of ERP projects—software quality perspective. International Journal of Project Management, 33(4), 961–970. doi:10.1016/j.ijproman.2014.10.011

Tak Wah Kwan; Leung, H.K.N. (2010). "A Risk Management Methodology for Project Risk Dependencies,", IEEE Transactions on Software Engineering, 37(5), 635-648, Sept.-Oct. 2011. doi: 10.1109/TSE.2010.108

Talet A. N., Talet M. Z. N.. (2014). Incorporation of Knowledge Management with Risk Management and Its Impact on IS/IT Projects. International Proceedings of Economics Development & Research 69(6) DOI: 10.7763/IPEDR

Taylor, H. (2006). Critical risks in outsourced IT projects: The intractable and the

Teja, S.S.K., Nithya Ganesan, N. (2016). Advance Tools and Techniques for Software Risk Management. International Journal of Advanced Research in Computer and Communication Engineering, 5(3), DOI 10.17148/IJARCCE.2016.53110

Thakurta, r. (2011). A Mixed Mode Analysis of the Impact of Requirement Volatility on Software Project Success. Journal of International Technology & Information Management;2011, 20 Issue ½.

Thomas, S., & Bhasi, M. M. (2011). A Structural Model for Software Project Risk Management. Vilakshan: The XIMB Journal Of Management, 7(3), 71-84.

Tiwana, A., & Keil, M. (2004). The one-minute risk assessment tool. Commun. ACM, 47(11), 73–77. doi:10.1145/1029496.1029497

Traniello, J. F. A., & Bakker, T. C. M. (2015). Minimizing observer bias in behavioral research: blinded methods reporting requirements for Behavioral Ecology and Sociobiology. Behav Ecol Sociobiol, 69(10), 1573–1574. doi:10.1007/s00265-015-2001-2

unforeseen. Communications of ACM, 49(11), 74-79.

Van den Berghe, A., Scandariato, R., Yskout, K., & Joosen, W. (2015). Design notations for secure software: a systematic literature review. Software & Systems Modeling. doi:10.1007/s10270-015-0486-9

Verner, J., Sampson, J., & Cerpa, N. (2008). What factors lead to software project failure? 2008 Second International Conference on Research Challenges in Information Science. doi:10.1109/rcis.2008.4632095

Vinnem, J.-E. (2013). Use of Risk Indicators for Major Hazard Risk. Offshore Risk Assessment 2, 791–839. doi:10.1007/978-1-4471-5213-2_22

Wanderley, M., Menezes, J., Gusmão, C., & Lima, F. (2015). Proposal of Risk Management Metrics for Multiple Project Software Development. Procedia Computer Science, 64, 1001–1009. doi:10.1016/j.procs.2015.08.619

Wu, X., Li, X., Feng, R., Xu, G., Hu, J., & Feng, Z. (2014). OOPN-SRAM: A Novel Method for Software Risk Assessment. 2014 19th International Conference on Engineering of Complex Computer Systems. doi:10.1109/iceccs.2014.28

Yacoub, S.M., Ammar, H.H. (2002). "A methodology for architecture-level reliability risk analysis,", IEEE Transactions on Software Engineering, 28(6) 529-547.

Yahav, I., Kenett, R. S., & Bai, X. (2014). Risk Based Testing of Open Source Software (OSS). 2014 IEEE 38th International Computer Software and Applications Conference Workshops. doi:10.1109/compsacw.2014.107

Yang, Y. H., & Tamir, G. (2015). Offshore software project management: mapping project success factors. International Journal of Project Organisation and Management, 7(2), 111. doi:10.1504/ijpom.2015.069613

Zardari, S. (2009). Software Risk Management. 2009 International Conference on Information Management and Engineering. doi:10.1109/icime.2009.138

Zowghi, D., & Nurmuliani, N. (2002). A study of the impact of requirements volatility on software project performance. Ninth Asia-Pacific Software Engineering Conference. doi:10.1109/apsec.2002.1182970

LIST OF PUBLICATIONS AND PAPERS PRESENTED

Khatavakhotan, A. S & Ow Siew Hock. 2016. A Generic Software Risk Tolerance Model (GSRTM): An Improvement in Software Risk Assessment Process. Proceedings of the International Conference on Computer and Applications (ICCA'2016). Dubai, September 14-15, 2016. (IEEE / ISI-Indexed)

Khatavakhotan, A. S & Ow Siew Hock. 2016. A New Risk Identification Model Based on Improbability of Potential Software Project Risks: Validated by Real Empirical Studies. Proceedings of the International Conference on Computer and Applications (ICCA'2016). Dubai, September 14-15, 2016. (IEEE / ISI-Indexed)

Khatavakhotan, A. S & Ow Siew Hock. 2015. DEVELOPMENT OF A SOFTWARE RISK MANAGEMENT MODEL USING UNIQUE FEATURES OF A PROPOSED AUDIT COMPONENT. Published by the Malaysian Journal of Computer Science (MJCS), 28(2), 110-131. (ISI-Indexed)

Khatavakhotan, A. S., Hashemitaba, N., & Siew Hock Ow. GISOS: A Model for Rectifying Complexities and Mitigating the Risks of Global Information System Development. Signal Processing and Information Technology. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. 62, 2012, 64-169. Springer- Verlag Berlin Heidelberg. (Chapter in Book)

Khatavakhotan, A. S., Hashemitaba, N., Ow, S.H. 2012. MRMM: A Mathematical Risk Management Model for Iterative IT Projects based on the Smart Database. International Journal of Information and Electronics Engineering, 2(2), 88-95. (Non-ISI/Non-SCOPUS)

Khatavakhotan, A. S., Hashemitaba, N., & Ow, S.H. 2012. A Novel Model for Software Risk Mitigation Plan to Improve the Fault Tolerance Process. International Journal of Information Technology & Computer Science. Information Integration and Computing Applications ,5, September/October 2012. 38-42. (Non-ISI/Non-SCOPUS)

Khatavakhotan, A. S & Ow Siew Hock. 2012. Dynamic Verifier Core: A Practical Solution to Mitigate the Risks of Software Risk Management Process. Software Engineering Journal, 2(5): 203-207. DOI: 10.5923/j.se.20120205.04 (Non-ISI/Non-SCOPUS)

Khatavakhotan, A.S.; Siew Hock Ow. An Innovative Model for Optimizing Software Risk Mitigation Plan: A Case Study. Modelling Symposium (AMS), 2012 Sixth Asia, Topic(s): Communication, Networking & Broadcasting ; Components, Circuits, Devices & Systems ; Computing & Processing (Hardware/Software) Digital Object Identifier: 10.1109/AMS.2012.55 Publication Year: 2012 , Page(s): 220 – 224

Khatavakhotan, Ahdieh Sadat; Ow, Siew Hock. Rethinking the Mitigation Phase in Software Risk Management Process: A Case Study . Computational Intelligence, Modelling and Simulation (CIMSiM), 2012 Fourth International Conference on Topic(s): Computing & Processing (Hardware/Software) Digital Object Identifier: 10.1109/CIMSim.2012.62, Publication Year: 2012 , Page(s): 381 – 386.

Khatavakhotan, A. S. & Ow Siew Hock. 2012. Managing the Risks of Software Risk Management Process: An Innovative Model using Dynamic Verifier Core. INTERCOMP 2012, September, Vienna, Austria.

Khatavakhotan, A. S. & Ow Siew Hock. 2012. Rethinking the Mitigation Phase in Software Risk Management Process: A Case Study. Postgraduate Research Excellence Symposium PGRES 2012, 25 Sep, Kuala Lumpur, Malaysia.

Khatavakhotan, A. S., Hashemitaba, N. & Siew Hock Ow. 2012. A Novel Model for Software Risk Mitigation Plan to Improve the Fault Tolerance Process. Proceedings of the International Conference on Information Integration and Computing Applications (ICIICA 2012), Singapore, August 14-15, 2012. (ISI-Indexed)

Khatavakhotan, A. S. & Ow Siew Hock. 2012. Improving IT Risk Management Process by an Embedded Dynamic Verifier Core; Towards Reducing IT Projects Failure. Proceedings of the 3rd International Conference on Intelligent Systems, Modelling and Simulation (ISMS2012), Sabah, February 8-10, 2012. (IEEE / ISI-Indexed)

Khatavakhotan, A. S., Hashemitaba, N., Ow Siew Hock. 2011. GISOS: A Model for Rectifying Complexities and Mitigating the Risks of Global Information System Development. Proceedings of Second International Conference on Recent Trends in Information Processing & Computing (IPC 2011), Kuala Lumpur, November 14-15 2011, pp. 23-27. (Non-ISI/Non-SCOPUS)

Khatavakhotan, A. S., Hashemitaba, N. & Ow Siew Hock. 2011. From Identification to Budget Allocation: A Novel IT Risk Management Model for Iterative Agile Projects. Proceedings of the 2011 3rd International Conference on Software Technology and Engineering, Kuala Lumpur, August 12-14 2011, pp. 509-513 (IEEE / ISI-Indexed)

Khatavakhotan, A. S., Hashemitaba, N., Ow Siew Hock. 2011. A Case Study: Using a Comprehensive IT Risk Management Model. Proceedings of the United Kingdom, Malaysia, Ireland Engineering Science Conference 2011, (UMIES 2011), Kuala Lumpur, July 12-14, 2011. (Non-ISI/Non-SCOPUS)