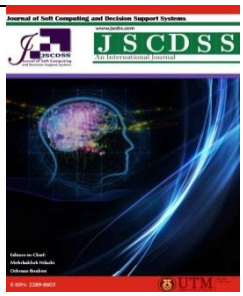
An International Journal
JSCDSS
http://www.jscdss.com
Vol.3 No.5 October 2016: 31-43
Article history:
Accepted 13 July 2016
Published online 13 July 2016

Journal of Soft Computing and Decision Support Systems



E-ISSN: 2289-8603

A Proposed Risk Assessment Model for Decision Making in Software Management

Bokolo Anthony Jnr. [a,*], Noraini Che Pa [a], Mustafa Salah Khalefa [a], Hamid Ali Abed Alasad [b], Hamzah Zmezm [c]
[a] Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400, Serdang, Selangor, Malaysia.
[b] Computers Sciences Department, University of Basra, Basra, Iraq.
[c] Department of Computer and Communication System Engineering, Universiti Putra Malaysia, 43400, Serdang, Malaysia.

*** Corresponding author email address**: bkanjr@gmail.com

Abstract

Software organization faces operational, technical and strategic risk. Hence, risk assessment is an important part of the decision-making process of software activities. Software management process has gained relevant during the last years, however there is still growing need of developing an innovative models that can support software practitioners in making decision to assess operational, technical and strategic risk. Existing risk assessment models adequately provide valuable insights to software practitioners to identify and measure the magnitude of risks associated in software activities, but they do not provide decision making support to software practitioners in assessing operational, technical and strategic risk. Thus, the aim of this paper is to propose a risk assessment model to support decision making of software practitioners when they assess risk that occurs in software management process. The developed model also provides software practitioners with the required risk assessment process and components, when they assess risk in their organisation. Semi–structured interview was used to collect data using two case studies involving a panel of software experts and software practitioners. Data was collected based on risk assessment practices in their respective software organisations. The case study was analysed using descriptive and narrative analyses. Results from the case studies shows that the current practice of assessing risk in software organisations is not effective due to inadequate decision making support to software practitioners when they measure and quantify identified operational, technical and strategic risk.

Keywords: Risk, Risk Assessment, Decision Making, Software Organisations, Software Management

1. Introduction

Risk is a combination of the probability of an event occurring, and the impact or consequence associated with that event. Risk management aims to manage and control risk effectively. If a risk is not identified it cannot be solved and trying to solve all risks is impossible. Thus, there is need for a risk assessment model to assess and solve risks in software management process (Abbinaya and Senthil, 2015; Feng, 2016). Risk assessment is the process of identifying and analysing the probability and impact of risk. Risk assessment in software based organisations is a summary of information and analyses used to evaluate the components of risk. Thus risk assessment is a systematic process of measuring and quantifying risk. Risk assessment assist in the selection of optimal or the most cost effective strategies for measuring risk, using a transparent decision making process (Sadiq et al., 2010; Omar, 2014). A decision can be defined as the act of reaching a conclusion. Good decision aids software management process to be effective. Software organisation involves the knowledge, techniques, and tools necessary to manage the development of software services. Software organisations adopt rules and regulations to guides practitioners in accomplishing their aims and objectives. Software organisations policy ensures that all of the project activities follow a certain predefined process. Thus software team members can guard against poor decision making through effective risk assessment strategies. The increasing complexity and dynamics of software process have plagued software practitioners with operational, technical and strategic risk. Therefore, risk assessment will assist software organisations to improve their performance of software projects (Xiaofei et al., 2014).

The assessment of risk is generally implemented by measuring the risk magnitude which may be assessed by considering two parameters: risk likelihood and risk effect. However, there are other risk processes that need to be considered in assessing risk in software organisations. In software organisations operational, technical and strategic risk can lead to failure of software based project and these risks should be considered in the assessment process (Ionita and Patriciu, 2014; Yao et al., 2016). Risk measurement is therefore introduced to structure and evaluate these risk and

integrate them into the decision making process when assessing the risk (Nasirzadeh et al., 2013; Muhammad and Adeel, 2014; Josua et al., 2015; Bokolo et al., 2015a). On the other hand in assessing risk software organizational knowledgebase is useful for supporting decisions making in assessing operational, technical and strategic risks. The assessment of risk can be done by software practitioners in making crucial decision based on the identified risk.

An effective decision making in software organisations can support software practitioners and software manager to assign tasks to minimize and assess risks. However existing model/frameworks lacks the capability to support software practitioners in making decisions on how to measure and assess risk. Therefore the objective of this work is to present a risk assessment model that can support software practitioners in making decisions when they measure identified risks in software process. The structure of this paper is organized as follows: Section 2 is literature review, Section 3 is the methodology, Section 4 is the case study findings, Section 5 is the proposed model, Section 6 is discussion and finally, Section 7 is the conclusion of the research paper.

2. Literature review

Today's software management process is usually distributed and often complex, thus it is crucial to achieve quality. Although software practitioners are concerned with schedule and cost issues, their major focus is on how make technical decisions. Software management process involves several system that interacts together to carry out software based objectives and goals.

However management of these software systems includes tasks required to control, measure and assess software risk that occurs. These management tasks are nowadays performed by software practitioners who are responsible for management of these risks (Nepomuceno and Fontana, 2013; Adeel et al., 2014; Bokolo et al., 2015a). Software risk management includes risk identification, risk measurement and evaluation, risk reduction or elimination, risk reporting and risk transfer (Bajo et al., 2012).

Software based project success depends on the software practitioner's ability to manage and assess identified risks. Therefore risk assessment provides a medium for software practitioners to measure identified risk. Risk assessment starts with software team training. The software team subsequently meets to discuss how to assess the identified risk assessment and to gain an understanding of risk management practices and risk assessment activities (Fayssal et al., 2015; Chi-An and Yu-Lun, 2015; Josua et al., 2015; Wang and You, 2015; Mukesh et al., 2015; Marko and Florian, 2015; Yao et al., 2016).

Fig. 1 shows the risk management process which involves two primary steps each with three subsidiary steps. The first primary step of risk management is risk assessment and it involves risk identification, risk analysis and risk prioritization (risk evaluation and risk mitigation). The other step of risk management is risk control and it involves risk management planning, risk resolution, and risk monitoring (Davide et al., 2012).

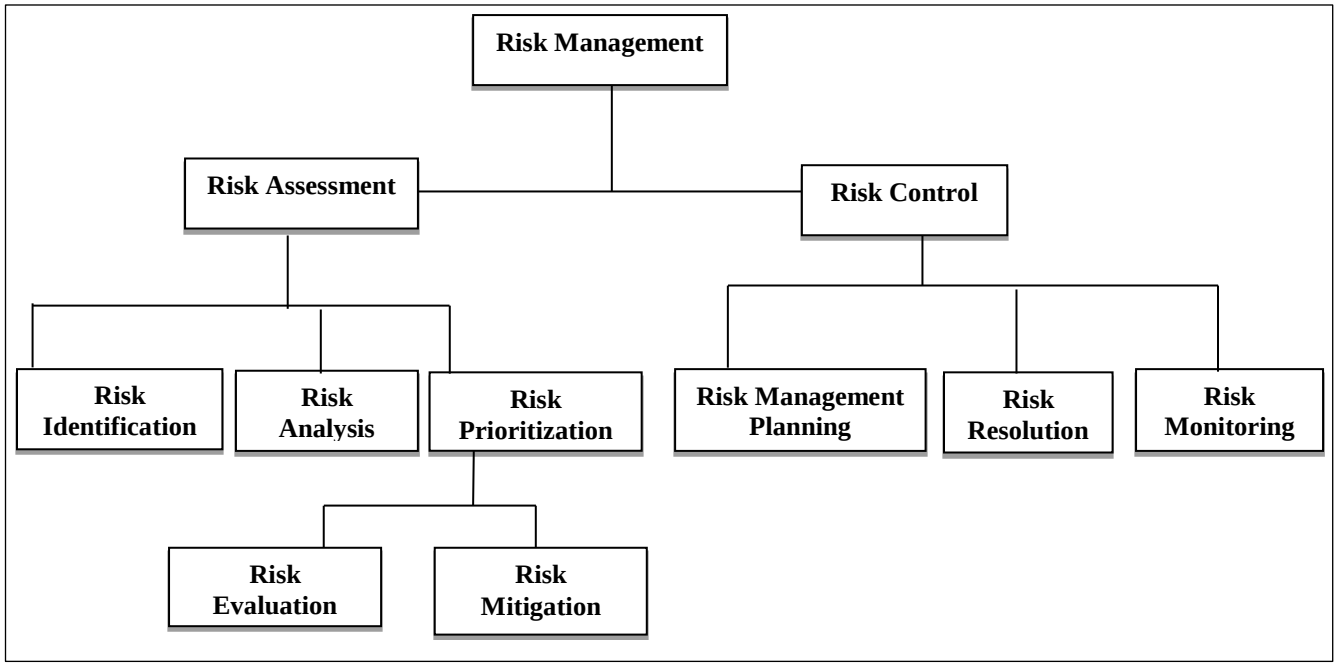


Fig. 1. Risk management process (Davide et al., 2012).

Risk identification produces a list of all recognized software risk factors which are likely to compromise software project success. Risk identification technique may include examination of decision drivers, assumption analysis and risk checklist. Risk analysis assesses the loss impact and loss probability for each risk. Risk analysis can

An International Journal JSCDSS E-ISSN: 2289-8603

be carried out using techniques such as performance models, network analysis, software cost models etc. Risk prioritization generates a ranked ordering of the identified risk. Risk evaluation involves highlighting the identified risk according to defined classes of risk and their risk level. By choosing an appropriate, operative risk aggregation procedure (Davide et al., 2012; Chi-An and Yu-Lun, 2015).

Risk Mitigation is the sub phase in risk assessment where by the risk is treated, avoided or transferred within software project. This is the stage in risk management involving our research because it's the phase were decision are made by software practitioner involved in managing the software (Davide et al., 2012). The other step in risk management is risk control and it involves risk resolution, risk management planning and risk monitoring. Risk control mainly aims to deal with the risks inherent in software project and thereby exercise better control over the software project and increase its chances of success. The main steps in the risk control stage comprises of risk resolution, risk management and risk monitoring. Risk management planning helps prepare software practitioners to address each risk item (Davide et al., 2012). Risk resolution produces a state in which the risk are eliminated or treated. Lastly risk monitoring involves tracking the controlled risk progress toward mitigating the risk and taking corrective action where appropriate. Risk monitoring technique includes risk event log analysis and milestone tracking.

However this research paper is mainly based on risk assessment only. Therefore this research is aimed at developing a risk assessment model with a detailed description of the model main components and model process which aims to support software practitioners in making decisions on assessing identified risks.

2.1 Existing risk assessment framework/models

This section reviews existing risk assessment model and frameworks. This will help in identifying the research gap existing in this research domain. Sadiq et al. (2010) designed a fuzzy multi benchmark decision-making for risk assessment to tackle the limitation of human experts who assesses risk solely based rely on their experience and judgement to estimate the risk. Fig. 2 shows the risk assessment model designed by Sadiq et al. (2010). The model quantifies existence risks in a in a repository to measure the risk. The model is applicable to identify and evaluate the magnitude of risks using quantitative approach.

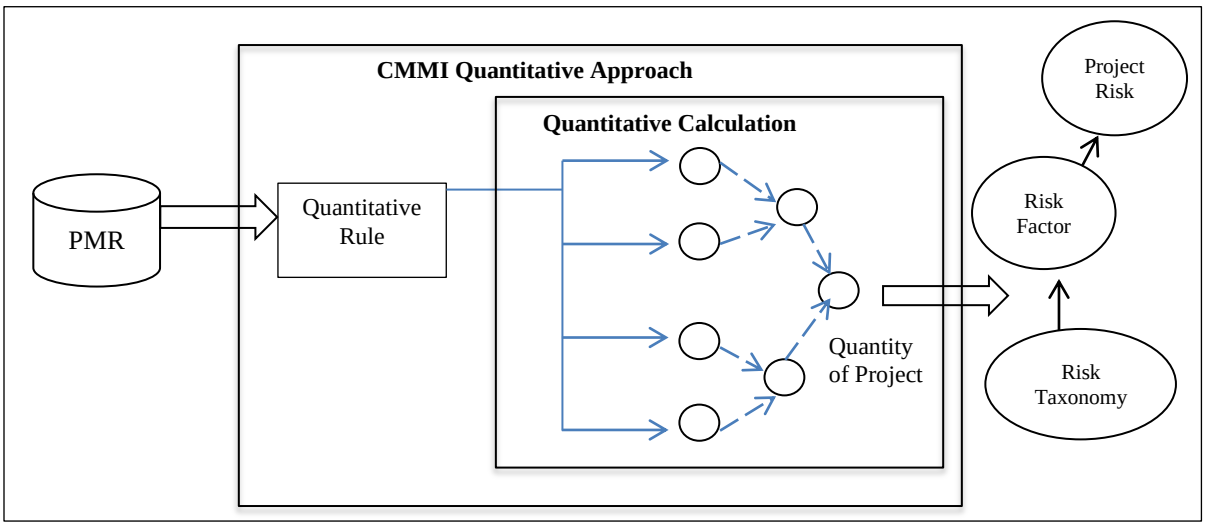


Fig. 2. Risk assessment model using quantitative approach adopted from (Sadiq et al., 2010).

Choetkiertikul and Sunetnanta (2010) proposed a risk assessment model using qualitative approach. The model offers a balanced basis for quantifying and checking risks and providing specific decision-making guide.

The model is shown in Fig. 3. The model assesses risk using quantitative approach thereby enabling an instinctive collection of data to evaluate the risk in software projects.

An International Journal JSCDSS E-ISSN: 2289-8603

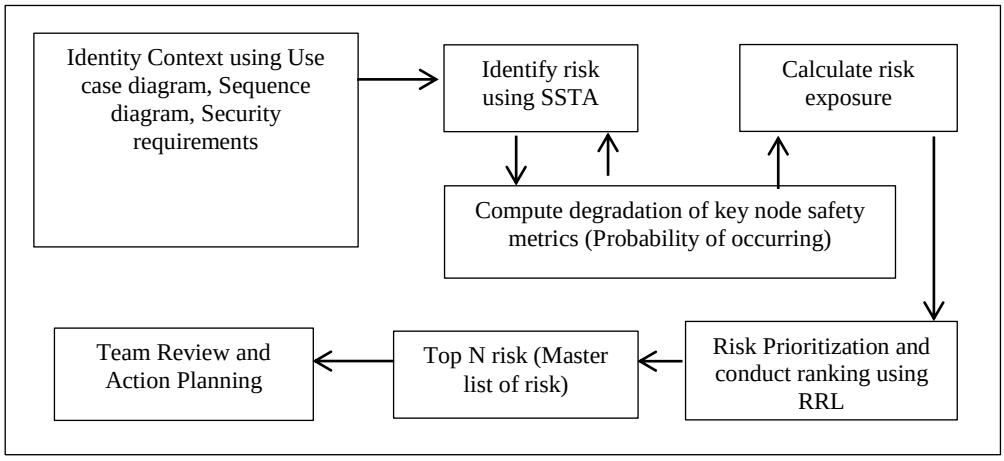


Fig. 3. Risk assessment model designed by (Choetkiertikul and Sunetnanta, 2010).

Previous models and frameworks proposed by other researchers includes the works of academicians such as Xiaofei et al. (2014) who designed a novel model for software risk assessment. The model comprises of software asset recognition, weakness analysis, consequence property confirmation and risk calculation. The model recognizes software assets using a "1 to 9" measures method using Analytic Hierarchy Process (AHP) techniques to assess the risk. The model computes the risk degree and the risk value are calculated using weighted average method and exponential method and respectively.

An automated risk assessment framework was proposed by Morakot et al. (2014) using a practical method. The framework helps in risk forecasting risk based on previous risk case studies. The framework comprises of the learning phase and the deployment phase. Abbinaya and Senthil (2015) developed a decision table using neural networks for effective software risk assessment. The developed approach uses decision table to compare existing risk and based on the compares method, analyse and produce an accurate risk assessment result. The neural network assists and trains the decision table using risk assessment training algorithm which is used to compare old risk data that has been assed and new risk that are yet to be assessed.

Chandan and Dilip (2015) suggested a probabilistic and estimation model to assess software risk that occurs in software based projects. The model uses a probabilistic software risk assessment model is proposed using Bayesian Belief Network (BBN) that emphases on the top software risk pointers for risk assessment in software based projects. Bokolo et al. (2015b) proposed a risk assessment model to provide collaborative support among software practitioners when they assess risk in their organisation. The model provides collaboration and communication among software practitioners. The model utilizes knowledge codification and software agent technology.

Omar (2014) proposed an operational risk assessment profile modelling for software quality calculation. The model is based on previously measured software quality assessment. The proposed model assist software practitioners in making quality decisions in assessing software risk related with software removal activities and defects detection. Muhammad and Adeel (2014) designed a practical V-Model Methodology, using Unified Modelling Language (UML) to model software risk assessment vsualization to help software practitioners to access risk in their organisation. The designed model provides solution to reduce risks using UML and visualize the software development processes in detail.

Nasirzadeh et al. (2013) presented a fuzzy group decision making approach to assessing risk with a decision making approach that will aid software practitioners in selecting the best alternative to assess identified risk. The components are risk avoidance, risk transfer, risk assessment and risk acceptance. Moeinzadeh and Hajfathaliha (2009) contributed by proposing a model combining fuzzy decision making method to risk assessment to make decisions that efficiently align software processes and decisions to utilize risk opportunities while concurrently minimizing risk. The model consists of two parts: risk identification, risk measurement and risk evaluation, where risk identification is the basis of risk evaluation. Shikha and Selvarani (2012) developed a proficient method of risk assessment using intelligent agents. The researchers ventured into how intelligent agents can be used in risk assessment by using enabling learning agents, which integrates decisions of the agent and the risks, but the current agent is capable of only identifying risk.

Manalif et al. (2013) presented an effort contingency model for risk assessment which helps to identify and estimation the value of identified software risk. The proposed model also integrates the effort estimation and risk assessment strategies because these activities are integral parts of the initial software project planning phase. The model risk assessment precision is based on effort estimates which also depend on the nature and level of the risks that are essential in software based projects.

Moorthy et al. (2013) presented a risk assessment model to reduce usability risk in software products. The model comprises of risk identification, risk prioritization, risk classification and risk analysis. The researcher mentioned that if usability risks can be identified, the overall chances of reducing risk of failure and producing usable software

An International Journal JSCDSS E-ISSN: 2289-8603

product could be increased. Moeinzadeh and Hajfathaliha (2009) only address decision making in risk assessment to make decisions that assist practitioners in minimizing risk. The researchers did not provide a risk knowledge source.

The motivation for conducting this researcher is based on the fact that the reviewed works only measure the risk probability and risk impact using different techniques and approaches. However the existing models and frameworks do not provide a decision making support to assist software practitioners when they assess operational, technical and strategic risk that occurs in software management process. Thus there is need for a model that will provide a knowledge base support of past risk that has been measured previously. Also the existing models/frameworks do not consider the risk assessment components but only considers the risk assessment process. Our proposed model will consider the risk assessment process and components.

2.2 Risk assessment practices in software organizations

Software practitioners are faced with risk in their industrial process. Thus there is need to assess the activities of software process that can result to risk. Risk assessment provides software practitioners with an accurate evaluation of risks (Sendi et al., 2012). The magnitude of risk can be assessed by considering two parameters: risk likelihood and risk severity. Risk magnitude and risk impact is therefore introduced and integrated into the decision making process of risk assessment (IIonita and Patriciu, 2014).

Risk assessment starts with the establishment of a risk assessment group that involves a range of experts with different background/discipline and necessary experience regarding software management. The risk assessment group undertakes the review of risk data and information, and resolution of risk criteria. The software members in risk assessment group are required to review all information related to the risks under consideration (Fayssal et al., 2015). Technical and operational risks are inherent in software usage in software organisations whose performance requirements exceed the capabilities of current software systems. If not assessed and solved early these risks can have negative effects on software project's cost, schedule and effectiveness. Technical and operational risks measure the probability and severity of adverse effects that are inherent in the projects and associated with its intended functions and performance requirements.

Technical and operational risk involves risk associated with software process quality, precision, accuracy, and performance over time of the developed software. In other words, technical and operational risk involves the risk associated within projects that meets intended functions and performance (Mukesh et al., 2015). Most software management processes are automated by software today. Thus, software's ever-increasing presence is following the trend of sub specialization. Compiler and language developers, real-time embedded systems experts, management information systems, user interface management systems, etc. All are examples of increasing software system. The increase in the influence and usage of software systems essentially accompanies an increase in operational and technical risk. Risk assessment in software management also involves the upper and technical management. Upper management: views risk almost exclusively in terms of profitability, schedule, and quality. Risk is also viewed in terms of the organization as a whole and the effect on multiple projects or a product line.

Software practitioners are concerned with cost, quality and schedules. Software practitioners in software management process involve Technical staffs (software engineers, hardware engineers) who are professionals concerns primarily with technical details of components, subassemblies, and products for one or more software projects. This inability to produce quality software on schedule and within a budget may be initiated by changing requirements, new technology, inexperienced personnel, inadequate management, or poorly developed cost or schedule. However, software practitioners and software management fail to communicate on these central issues (Marko and Florian, 2015). Making a decision is a multi-step process. Therefore, Laudon and Laudon (2012) described four different stages in decision making: intelligence, design, choice, and implementation. The process are adopted when software practitioners making decisions in software management process.

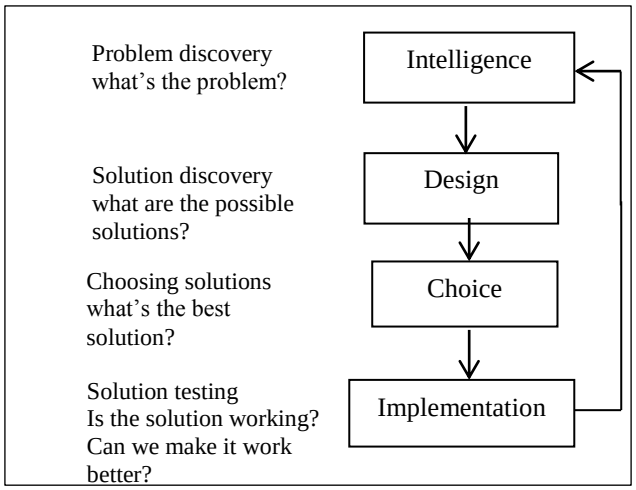


Fig. 4. Decision making process (Laudon and Laudon, 2012).

Fig. 4 shows the decision making process that comprises of four main phase. Intelligence consists of identifying, discovering and understanding the risk occurring in the software process, why the risk exists, where, and what effects it is having on software management process. Design involves identifying and exploring various solutions to the risk. Choice consists of choosing among solution alternatives and implementation involves deploying the chosen alternative solution and continuing to monitor how well the solution is working. What happens if the solution applied or chosen by software practitioners does not work, Fig. 4 shows that software practitioners can return to an earlier stage in the decision-making process and repeat it if necessary. Decisions are classified as structured, semi structured, and unstructured in software management. Fig.

An International Journal JSCDSS E-ISSN: 2289-8603

5 illustrates the decision making pyramid showing the software management members and software practitioners involved in the decision making process.

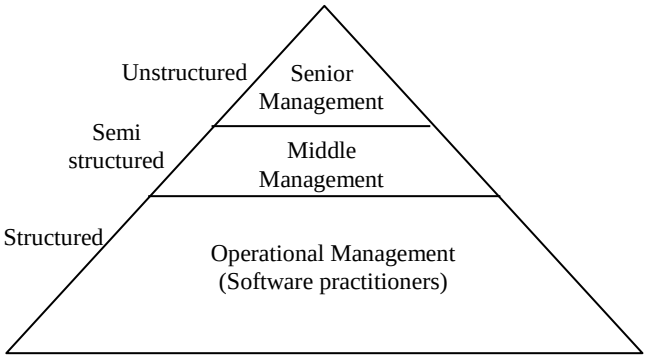


Fig. 5. Decision making pyramid (Laudon and Laudon, 2012).

In unstructured decision making, the decision makers provide judgment, evaluation, and insight to solve the risk. Each of these decisions is novel, important and non-routine. Software practitioners face many unstructured decision situations, such as establishing their firm's goals or deciding new approaches to assess risk. However, the answer would also require software practitioners to use their own best judgment and inquire from other software experts/managers for their opinions.

In semi structured decision software managers suggest routines that can be applied to asses identified risk. Middle management faces more structured decision scenarios but their decisions may include unstructured components.

Whereas structured decision involves software practitioners that assess risk in software management process. Their responsibilities range from making decisions, to writing reports, to attending meetings, and assessing operational and technical risk.

3. Methodology

This research adopted a qualitative research. The research is based on 4 different phases as follows:

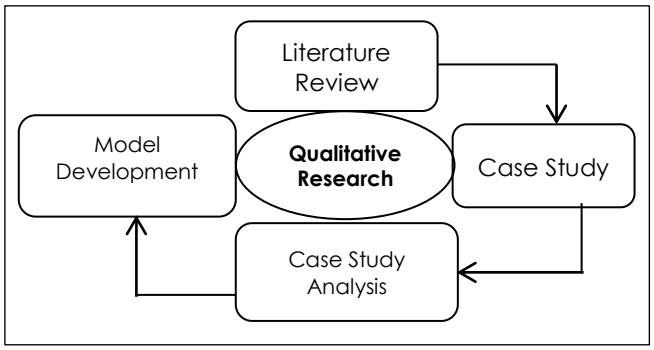


Fig. 6. Research methodology.

Fig. 6 shows the research methodology process implemented in developing the proposed risk assessment decision support model.

Phase 1 is the literature review which encompasses the reviewing of journals, conference proceeding, books literatures on risk assessment practices and process. This phase is important, since it lays the foundation for the research background and model development.

Phase 2 is the case study selection which comprises of data collection using case study by interview. This phase is carried out to confirm the risk assessment components and process derived from the literature. The case study was carried out using open ended interview. The case study used purposive sampling where the informants are selected based on their experienced and in depth knowledge of risk assessment in their organisation.

Phase 3 involves the case study analysis of the interview transcript which was analysed using a descriptive and narrative analysis based on risk assessment process and risk assessment components, because these are the modules of the proposed model.

Phase 4 involves the model development which involves the development of the proposed model. The model is developed based on the risk assessment process and risk assessment components derived from the literature and confirmed form the case study session.

3.1 Case study section

Case study was used in this research. A case study is a research strategy; it's a choice of object to be studied. It is a research design and it examines a phenomenon in its natural setting, employing multiple methods of data collection to gather information from one or a few entities (people, groups, or organizations). This section focused on descriptive justification of each selected organization based on the organization people, activity, technology, quality management and risk assessment process carried out by the organization. The instrument of the case study is a semi-structured interview. Two case studies were selected in this research. A total of 6 software practitioners/experts were interviewed in each case study to answer the interview questions based on how their organization assess and solve risk.

According to Yin (2004) the minimum number of informants form a case study is 3 and the maximum is unlimited. Thus data was collected from 6 informants via open ended interview. The analysed data is used to confirm the risk assessment process and components mention in the literature review.

Case Study A is a private organization practicing risk assessment by applying risk assessment methods. Currently they responsible to plan develop and implement ICT software activities and software projects in accordance with its mission, that is, "To develop and enhance research and development of enterprise systems through the provision of comprehensive information technology/software system.

Case Study B is a software/IT based organization practicing risk prevention and control by applying risk maintenance to their software and hardware process. Currently the department is responsible for managing and controlling all software/IT infrastructures in their

An International Journal JSCDSS E-ISSN: 2289-8603

organizations' at large. The findings from the literature review and the case study was important as a basis for the risk assessment model development.

Table 1 shows the interview questions used to collect data on risk assessment practices in the 2 case studies.

Table 1
Data collection instrument.

Components &Process	Questions
People	Who performs risk assessment? • Team or single person? • External (review) or internal? • IT People involved or other department? • Who are the best people in your opinion? What criteria are needed to select people for risk assessment?
Activities	What are the activities involved in risk assessment?
Approaches	How is the risk assessment carried out? • Quantitative or qualitative or others? • What are the methods used? • What techniques are used to perform risk assessment?
Technology	What technology is used for risk assessment? • Does the technology (hardware, software and network communication) used meet your expectations? • Name of technology used / how long has the technology been used / any problems using the technology?
Procedure	What are the procedures used for risk assessment?
Practice	In your opinion what are the possible practices involved in assessing risk?

4. Case study findings

Case Study A is a private organisation based in Malaysian. The ICT division in this organisation implement risk assessment/management practices on their IT/software infrastructures. They have been in the software related business for the past 16 years. The organisations staffs consist of highly trained, experienced and knowledgeable professionals and employ less than 100 practitioners.

4.1 Findings of case study A

People involved in risk assessment; Risk assessment is performed by Internal IT personnel in teams (organization). The criteria used to select people responsible for performing risk assessment in their organization are skill, experience and education, thus people (human) is a component in assessing risk in software organisations.

Risk assessment activities; The activities involved in risk assessment, the informants choose risk rating, screening examination of risk drivers, assumption risk analysis, review solution, benchmarking, cost benefit analysis, benchmark to state, mission knowledge mapping & standard risk management plan and added that the risk assessment should also be an activity for the assessment of risk. These activities are used for risk identification and risk measurement by the practitioners in their organisation.

Technology used for risk assessment; the software organization uses some technology such as Malaysian Risk Assessment Methodology Application (MIRAM) for risk assessment and for network communication (VPN, firewall, IDS, WAF and Antivirus) are used. However the informants mentioned that the current technology has been used for 10 years, but the organization has been experiencing license renewal subscription cost yearly for the security software that carters for risk assessment. Thus the hardware and software are components for assessing risk in their organisation.

Risk assessment approaches and procedures; The informants selected record keeping, proper documentation, structured workshop, feedback form, maintenance procedures, check list, training, cause-and-effect analysis, root cause analysis as procedures for risk evaluation in their organisation. The informants added that team discussion is also a procedure used for risk communication among the project team members in correlation to the identified risk that is being assessed.

Current risk assessment practices (process); Lastly the informants agreed that the risk assessment process should include estimation (evaluation) and monitoring (Communication) of the probability and magnitude of risk (Evaluation), calculation of potential risk using quantitative data (Measurement), provision of suggestions via database for monitoring activities, supporting collaborative decision-making process among risk assessment practitioners and they suggested that the model may also have risk prediction, to forecast the risk effect for organizations continuity and to improve/educate software practitioners.

4.2 Findings of case study B

Case Study B is also a Malaysian based organization that practice risk assessment in their ICT department. This organization has 103 fixed staff including professional and semi-skilled ones and it is working on design and software/IT infrastructures maintenance and installation services.

People involved in risk assessment; risk assessment is performed by internal and external IT (ICT members (expert and organisational staffs) personnel in teams. The criteria used to select people (Human) responsible for

An International Journal JSCDSS E-ISSN: 2289-8603

performing risk assessment in their organization are skill, experience and education.

Risk assessment activities; for the activities involved in risk assessment the informants mentioned informal/formal meeting and preventive maintenance as activities used for risk identification and risk measurement by the practitioners in their organisation.

Technology used for risk assessment; the organization uses some technology such as windows 2003 server windows, 2008 server, and switches/router. Thus the computer hardware and computer software are components for assessing risk in their organisation.

Risk Assessment approaches and procedures; the informants selected Maintenance procedures, proper documentation, checklist as approaches and procedures used for risk evaluation and structures workshop and formal meeting as procedure and activity used for risk communication. The risk communication is used to discuss about the identified risk that is to be assed in the organisation.

Current risk assessment practices (process); the informants added that risk assessment process should include monitoring (communication) of the identified risk. Also the informants added that risk decision should be one of the processes in carrying out risk assessment by measuring and evaluating the risk.

5. Proposed model

It is essential to develop a risk assessment model to support software practitioners in making decisions on how to measure risks in software management process. It is imperative to design an approach that utilizes data from knowledge base of past assessed risk impact and probability. However, the risk impact and probability are dependent on many components such as human factors, workplace factors, material factors and equipment factors etc., which are difficult to quantify and adequately measure in a traditional way. However, these components should be taken into consideration in the risk assessment process so that a reliable result can be obtained, in order to assess the risks associated with in software management process.

Fig. 7 shows the proposed model aimed to assist software practitioners in assessing operational and technical risk in their organisations pertaining to software usage and software project development. The model considers the components that lead to operational and technical risk. The components are seen in Fig. 7 as the hardware, software, human and the organisations or management. The model supports software practitioners in assessing risk; as a decision support tool based on the risk data from the knowledge-base.

The knowledge base contains past experience on how experts assess and control and quantify operational, technical risk. The proposed model is developed base on the findings from the review of existing risk assessment components (Human, Hardware, Software and Organisation) and confirmation from the case studies findings. The developed model is also similar to previous work of Noraini et al. (2015a). Whereas as the risk assessment process in the model (Risk identification, Risk Measurement, Risk Evaluation and Risk Communication) is derived from review of literatures in Section 2.1 of this research paper and from the findings from the case studies.

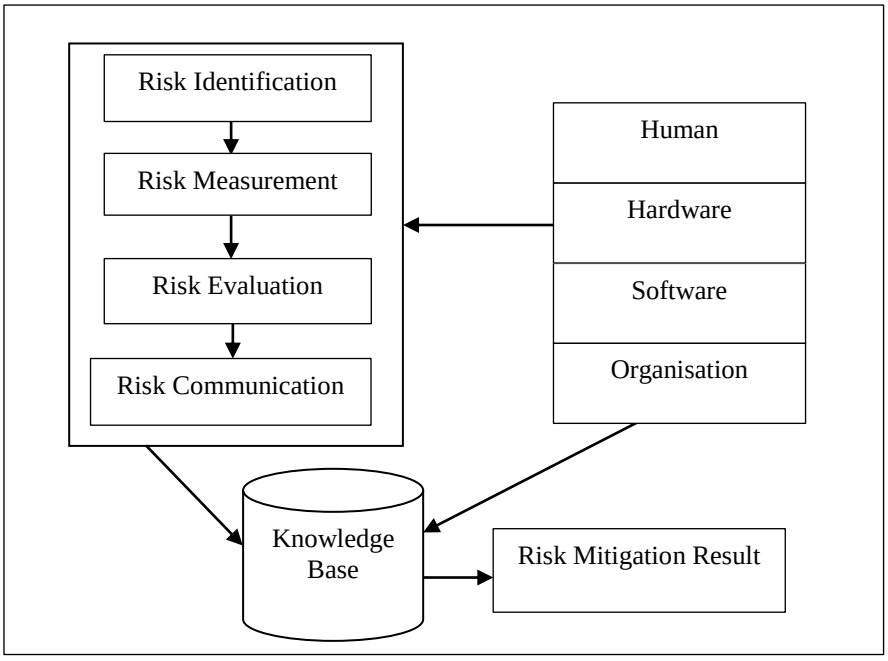


Fig. 7. Proposed risk assessment model in software management.

Fig. 7 shows the proposed risk assessment decision support model in software organisations. The model process involves the identification, measurement, evaluation, communication and mitigation of risk through

An International Journal JSCDSS E-ISSN: 2289-8603

the four risk components; human, organizational, hardware and software. The model aims to assist software practitioners in risk assessment process in software organisations based on the questions; (1) what can go wrong? (2) What is the likelihood that it will go wrong? (3) What are the consequences? These questions assist software practitioners in carrying out risk assessment in software organisations. To answer the three questions relating to in the risk assessment process, software practitioners need assistance from knowledge. The knowledgebase comprises of risk data that related to the four major sources of risk in software based organisations which are risk from hardware, software (includes software used in the development of software) organizational and human factors.

The proposed model assist software practitioners to identify what might go wrong and plan on what they might do about it. Software practitioners have to assess the risks that may affect a project, monitor these risks, and take action if problems arise. Risks may affect the project, the software that is being developed, or the organization. Thus the proposed model is important to assist in measuring the risk based on the risk impact and risk probability.

5.1 Risk assessment model process

Table 2 shows the risk assessment model process in software management process.

5.2 Risk assessment model components

Table 3 shows the components for risk assessment in software management process.

5.3 Risk likelihood and effect measurement

Risk effect and likelihood measurement is the systematic process to understand the nature of risk (by finding, recognizing and describing the risks) and to deduce the level of risk (by assigning values to impact and their probability) (Bokolo et al., 2015a; Chandan and Dilip, 2015). Risk effect and likelihood measurement provides the basis for making decisions about risk assessment as seen in Fig. 8, Table 4 and Table 5 in this research paper. During the risk measurement phase, software practitioners have to consider each identified risk and make a judgment about the effect and likelihood of that risk.

Software practitioners normally rely on their judgment and experience of previous projects and the problems that arose in them. It is not possible to make precise, numeric assessment of the effect and likelihood of each identified risk (Adeel et al., 2014; Bokolo et al., 2015b; Chi-An and Yu-Lun, 2015). Once the risks have been measured and ranked, software practitioners can make decisions on which of these risks are most significant. Risk measurement depend on a combination of the impact of the risk arising and the likelihood of the risk. Also the effect and likelihood of the risk can be saved in the database and can be reused to guide other software practitioners when they assess risk in future.

Table 4 and Table 5 shows the likelihood and effect scoring guideline for assessing risk based on the measurement of the risk to be assessed. The measurement is to be used by software practitioners in measuring identified risk. The application of this can be seen in Fig. 8 in this research paper. Where the identified risk is measured and assessed based on the risk impact and risk probability.

Table 2

Risk assessment model process.

Process	Description
Risk Identification	Produces lists of the projects-specific risk items likely to compromise a project success. A typical risk identification technique includes examination of decision drivers, assumption analysis, and checklist (Sadiq et al., 2012; Noraini and Bokolo, 2015a, Davide et al., 2012; Bokolo et al., 2015a, Nasirzadeh et al., 2013; Adeel et al., 2014; Ionita and Patriciu, 2014; Omar, 2014; Noraini et al., 2015c; Marko and Florian, 2015; Wang and You, 2015; Josua and Jaka, 2015; Bokolo et al., 2015b).
Risk Measurement	Computes the risk likelihood and risk effect for each identified risk item based on the magnitude of the risk (Davide et al., 2012; Bokolo et al., 2015a; Nasirzadeh et al., 2013; Ionita and Patriciu, 2014; Adeel et al., 2014; Noraini et al., 2015c; Josua and jaka, 2015).
Risk Evaluation	Produces a ranked ordering of the identified and measured risk items using techniques such as risk exposure analysis, risk reduction leverage. Involve cost-benefit analysis) (Davide et al., 2012; Adeel et al., 2014; Xiaofei et al., 2014; Chi-An and Yu-Lun, 2015; Josua and jaka, 2015).
Risk Communication	Includes status reporting on the execution of risk tracking and tracing in terms of probability, impact and risk metrics (Bokolo et al., 2015a; Chandan and Dilip, 2015; Adeel et al., 2014; Adeel et al., 2014; Chandan and Dilip, 2015).
Risk Mitigation Result	Involves the decision making and treatment of the risk that has been measured and evaluated. The assessment result is used to treat the operational and technical risk (Sadiq et al., 2012; Omar, 2014; Noraini and Bokolo, 2015a, Davide et al., 2012; Bokolo et al., 2015a, Nasirzadeh et al., 2013; Noraini et al., 2015c).
Knowledge Base	The knowledgebase stores risk assessment rules, standards and technical guidelines that provides support of past accessed risk that is utilized by software practitioners to make decisions on how to assess new risks (Sadiq et al., 2012; Adeel et al., 2014; Omar, 2014; Noraini and Bokolo, 2015a; Bokolo et al., 2015a, Nasirzadeh et al., 2013; Noraini et al., 2015c; Marko and Florian, 2015; Wang and You, 2015).

An International Journal JSCDSS E-ISSN: 2289-8603

Table 3

Risk assessment model components.

Components	Description
Human	Since software management process is an intellectual, labor intensive activity, the role of humans must be carefully understood to properly assess risk. The quality of decision making is based on the experience and background of the decision maker(s). That is, one can have all the relevant data and still make the wrong decision because of a lack of knowledge of how to apply the information (Sadiq et al., 2010; Davide et al., 2012; Choetkiertikul and Sunetnanta, 2010; Nepomuceno and Fontana, 2013). Thus the human (Software practitioners) are a determinate in making decisions related to risk assessment.
Hardware and Software	Software can also hinder the performance of hardware if its functions and interaction with hardware are not well understood. Thus, a change of software component can affect the performance of other software components and hardware components (Xiaofei et al., 2014; Muhammad and Adeel, 2014; Ionita and Patriciu, 2014; Bokolo et al., 2015a; Chi-An and Yu-Lun, 2015). Thus hardware and software are determinant components to be considered by software practitioners when they assess risk in their organisations.
Organization	The organizational level at which risk is identified is often different from the level at which decisions can be made and implemented. Thus, these different levels must interact effectively. To improve interaction within the organization, a common language must exist in addition to common definitions, common understanding, and a feedback mechanism for all decision makers (Shikha and Selvarani, 2012. Manalif et al., 2013; Adeel et al., 2014; Noraini and Bokolo, 2015a; Fayssal et al., 2015; Chi-An and Yu-Lun, 2015). Therefore the organizational management is a major determinant in decision making process when assessing risk in software management process. The organizational management sets all the policies and standards to be adhere to when assessing risk.

Table 4

Likelihood scoring guideline for risk assessment (Bokolo et al., 2015a).

Value	Likelihood/Probability Level
9-10	Very likely to occur
8-7	Will probably to occur
6-5	Equal chance of occurring or not
4-3	Probably not occur
1-2	Very unlikely

Table 5

Effect scoring guideline for risk assessment (Bokolo et al., 2015a).

Value	Effect/Impact Level
9-10	Very Significant
8-7	Significant
6-5	Significant and insignificant is equal
4-3	Insignificant
1-2	Very insignificant

Fig. 8 shows the risk effect and impact measurement. From the graph the impact is the same as the risk effect and the probability is the same as the risk likelihood. This measurement is used to quantify the risk as utilized by researchers such as Sadiq et al. (2010); Davide et al. (2012); Choetkiertikul and Sunetnanta, (2010) who used similar techniques in measuring risk in their proposed model.

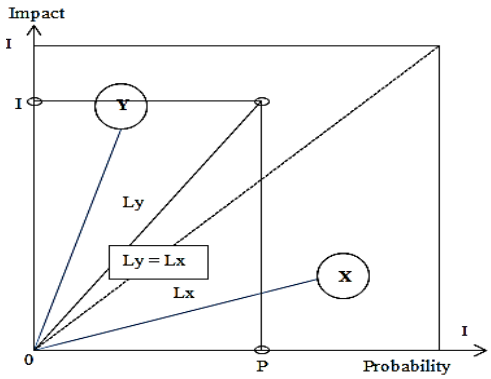


Fig. 8. Risk effect and impact measurement (Bokolo et al., 2015a).

6. Discussion

Information has become an essential resource for managing modern organizations. This is so because today's software environment is volatile, dynamic, turbulent and necessitates the burgeoning demand for accurate, relevant, complete, timely and economical information needed to drive the decision-making process in order to accentuate software organizational abilities to manage opportunities and risks. The effectiveness of software management process is dependent on the quality of decisions that informs its software operation. If decisions are right, it translates in positive software outcomes, but where software activities are executed in conditions of poor decisions resulting from insufficient or inaccurate information, such software process could be ruined. This is why decision making is a major determinant of software management's success or failure. Decision making is the process by which software practitioners choose specific course of action in response to threats and opportunities. Good decision result in courses of actions that help software practitioners to be effective, the opposite is its reverse (Omar, 2014).

Risk assessment is referred to as the critical process to aid software organisations achieving the new business changes, future investment in information and information system. In software organisations, the management perspective is included in the assessment and mitigation of risks. In general, risk assessment in software management process refers to an essential process aimed at supporting software enterprise achieving the new business changes, future investment in information and information system, an increasing IT threats and an increasing dependence on delivering information in system (Lainhart, 2010; Noraini et al., 2015b; Nancy, 2016).

Risk assessment in software organisations provides new approaches and methods for software managers to carry out both their traditional and newer roles, enabling them to monitor, plan, and forecast with more precision and speed than ever before and to respond more rapidly to the changing business environment. Risk assessment in

An International Journal JSCDSS E-ISSN: 2289-8603

software organisations have been most helpful to software managers by providing support for their roles in disseminating information, providing liaisons between organizational levels, and allocating resources. Risk assessment is an important part of planning in any software project because it allows software practitioners to predict potential risk that will affect the project and take steps to mitigate those problems. Therefore including risk assessment practice in project plan is an effective way to keep software project from being derailed by surprises or emergencies. Potential risks should be identified, and the relative probability and impact of each risk should be estimated.

Decisions can be structured, semi structured, or unstructured, with structured decisions clustering at the operational level of the organization and unstructured decisions at the strategic level. Decision making can be performed by individuals or groups and includes software practitioners as well as operational, middle, and senior managers (Noraini and Bokolo, 2015b). There are four stages in decision making: intelligence, design, choice, and implementation. Existing models/frameworks developed to support decision making in assessing risk do not always produce desired output, thus software practitioners faces difficult in making decisions on how to assess operational and technical risk that affects software performance. Therefore there is a need to assessment both operational and technical risk in software management process.

7. Conclusion

The concept of risk became popular during the 1920s. Since then, it has been successfully used in theories of decision making in computer science and decision science. Risk is defined as an uncertain event that, if it occurs, will have a negative effect (KarimiAzari et al., 2011; Ionita and Patriciu, 2014). Risk management is a series of steps that helps software teams to understand and manage uncertainty. Risk assessment is an important part of decision making process in software management. Risk assessment is the critical procedure of risk management. Despite many scholars, academicians, researchers and practitioners recognizing the risk assessment models in projects, insufficient attention has been paid by researchers to develop a risk assessment decision support model.

This research paper adopts a qualitative research that utilized case study to investigate how risk is being assessed in software based organisation. The case study was carried out through open ended interview questions using purposive sampling in 2 Malaysian based software/IT organisations with a total of 6 informants. The case study was analysed using descriptive and narrative analysed based on the risk assessment process and risk assessment components. Findings from the case study shows the identified process and components (technology, the people in the organisation that are involved in risk assessment, the activities, procedures, approaches) used in assessing risk.

Findings from the case study also highlights on type of people that makes decision on how to assess the risk in software organisations and lastly the current practices used by software practitioners in assessing risk in their organisation. The findings from the case study are in line with the earlier findings from the literatures (finding from the existing works as seen in Section 2.1 of this research paper). These findings are used in the development of the proposed risk assessment model for supporting decision making in software management process.

The advancement of the developed model with exiting risk mitigation models/framework supports software practitioner in making decisions on how to measure and assess risk in software organisation based on the knowledgebase. This is lacking in existing models were the risk are only measured alone, also existing works do not consider the risk assessment component but only concentrates on the risk assessment process. Thus the proposed risk assessment model can assist software practitioners to measure and assess operational and technical risks which affect software management process.

The model can successfully identify, quantify measure, evaluate, and provide assessment results for risk assessment. The model provides support for software practitioners in making decisions based on the risk components and risk assessment process. The risk components (human, hardware, software and organisation) and risk process (identification, measurement, evaluation and communication) are stored in the knowledgebase and are utilized by other software practitioners in assessing risk in their software management process. The risk assessment process and components are retrieved form the knowledgebase and are utilized as decision making support by software practitioners.

The practical implication of this research is that this research is based on a qualitative study; hence there is need to carry out a quantitative research study in line to the research to validate the model process and model component derived and confirmed from the case study section. Also there is need to utilize the developed model to implement a risk assessment system to support software practitioners in making decision in software management process.

The research limitation of this research is that this work is carried out in Malaysian based organisations. Thus the results from the case study cannot be generalized to other countries. This research is also based on qualitative study by case study. If a quantitative approach is adopted the results might differ. The practical limitation of this research is that our model depends on past experience of risk experts that is added to knowledge base. The information of assessed risk is based on the risk components and risk process added by human experts. Thus there is need to enhance the model to be more autonomic. This will enable the model to be intelligent to suggest risk assessment recommendations to software practitioners.

Future research direction will involve the integration of multi software agents and knowledge mapping techniques into the proposed model. The autonomous nature of software agents will support and recommend risk assessment suggestion to software practitioners in assessing

An International Journal JSCDSS E-ISSN: 2289-8603

operational, technical and strategic risk that occurs in their organisation. Since the present model knowledge is based on expert knowledge. The agents can support software practitioner in measuring the risk probability and risk impact automatically and saving the risk details in the knowledge base by mapping the risk information. Lastly there is need to conduct this study using a qualitative research study.

Acknowledgments

We would like to thank Universiti Putra Malaysia for giving us the opportunity to carry out this research. Much appreciation to the Editor-in-Chief and anonymous reviewers for their reviews and useful comments in improving this paper.

References

Abbinaya, S. Senthil, K. M. (2015). Software Effort and Risk Assessment Using Decision Table Trained by Neural Networks. Paper Presented at the IEEE ICCSP 2015 conference.

Bajo, J., Borrajo, M. L., Paz, J. F. D., Corchado, J. M. & Pellicer, M. A. (2012). A multi-agent system for web-based risk management in small and medium business. Expert Systems with Applications, 39(4), 6921–6931.

Bokolo, A. J. & Noraini, N. C. (2015). A Review on Tools of Risk Mitigation for Information Technology Management. Journal of Theoretical and Applied Information Technology, 11(1), 92-101.

Bokolo, A. J., Noraini, C. P., Teh, M. A., Rozi, N.H. N. & Yusmadi, Y. J. (2015a). Autonomic Computing Systems Utilizing Agents for Risk Mitigation of IT Governance. Jurnal Teknologi, 77 (18), 49-60.

Bokolo, A. J., Noraini, C. P., Rozi, N.H. N., & Yusmadi, Y. J. (2015b). A Risk Assessment Model for Collaborative Support in Software Management. Paper Presented at the 9th Malaysian Software Engineering Conference.

Chandan, K. & Dilip, K. Y. (2015). A Probabilistic Software Risk Assessment and Estimation Model for Software Projects. Paper Presented at the Eleventh International Multi-Conference on Information Processing-2015 (IMCIP-2015), Procedia Computer Science 54, 353 – 361.

Chi-An, C. & Yu-Lun, H.(2015). An Adjustable Risk Assessment Method for a Cloud System. Paper Presented at the IEEE International Conference on Software Quality, Reliability and Security - Companion.

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

Davide, A., Dulmin, R. & Mininno, V. (2012). Risk Assessment in ERP projects. Information Systems Journal, 37 (2), 183–199.

Fayssal, M. S., Richard G. S. & Zhaofeng, H. (2015). Reliability and Probabilistic Risk Assessment - How They Play Together. Paper Presented at the IEEE International conference.

Feng, J. (2016). Risk Assessment and Control for Accounting Information System based on Fuzzy Analytic Hierarchy Process. Paper Presented at the 2016 Eighth International Conference on Measuring Technology and Mechatronics Automation.

Ionita, M. & Patriciu, V. (2014). Achieving DDoS Resiliency in a Software Defined Network by Intelligent Risk Assessment Based on Neural Networks and Danger Theory. Paper Presented at the 15th IEEE International Symposium on Computational Intelligence and Informatics, 19–21 November, 2014, Budapest, Hungary.

Josua, J. P. S. & Jaka, S,. (2015). Risk Assessment Model of Application Development using Bayesian Network and Boehm's Software Risk Principles. Paper Presented at the International Conference on Information Technology Systems and Innovation (ICITSI) Bandung – Bali, November 16 – 19, 2015,

KarimiAzari, A., Mousavi, N., Mousavi, S. F. & Hosseini, S. (2011). Risk assessment model selection in construction industry. Expert Systems with Applications, 38(2). 9105–9111.

Lainhart, J. W. (2010). Why IT governance is a top management issue. The Journal of Corporate Accounting and Finance, 11(1), 33-40.

Laudon, K. C. & Laudon, K. P. (2012). Management Information System, Prentice Hall.

Manalif, E., Capretz, L. F. & Ho, D. (2013). Software Project Risk Assessment and Effort Contingency Model based on COCOMO Cost Factors. Journal of Computations & Modelling, 3(1), 113-132.

Marko, E. & Florian, T. (2015). Software Risk Assessment for Measuring Instruments in Legal Metrology. Paper Presented at the Proceedings of the Federated Conference on Computer Science and Information Systems 1113–1123.

Moeinzadeh, P. & Hajfathaliha, A. (2009). A Combined Fuzzy Decision Making Approach to Supply Chain Risk Assessment. World Academy of Science, Engineering and Technology, 60(2), 519-528.

Moorthy, J.T. S., Ibrahimn, S. B.& Mahrin, M. N. (2013). The Need for Usability Risk Assessment Model, SDIWC, 215-220.

Morakot, C., Hoa, K. D. & Thanwadee, T. S. (2014). A CMMI-based automated risk assessment framework. Paper Presented at the 21st Asia-Pacific Software Engineering Conference.

Muhammad, R. N. & Adeel, A. M. (2014). Using V-Model Methodology, UML Process-Based Risk Assessment of Software and Visualization. Paper Presented at the International Conference on Cloud Computing and Internet of Things (CCTOT 2014).

Mukesh, V. G., Shashank, M. S. & Santanu, K. R. (2015). Software Project Risk Assessment based on Cost Drivers and Neuro-Fuzzy Technique. Paper Presented at the International Conference on Computing, Communication and Automation (ICCCA2015).

Nancy J. L. (2016). An Innovative Goddard Space Flight Centre Methodology for using FMECA as a Risk Assessment and Communication Tool. USA. 1-9.

Nasirzadeh, F., Khanzadi, M. & Mianabadi, H. (2013). A Fuzzy Group Decision Making Approach to Construction Project Risk Management. International Journal of Industrial Engineering & Production Research, 1(1), 71-80.

Nepomuceno, V. S. & Fontana, M. E. (2013). Decision support system to project software management. Paper Presented at the 2013 IEEE International Conference on Systems, Man, and Cybernetics.

Noraini, N. C. & Bokolo, A. J. (2015a). A Model of Mitigating Risk for IT Organisations. Paper Presented at the 4th International Conference on Software Engineering and Computer Systems (ICSECS' 15).

Noraini, C. P. & Bokolo, A. J. (2015b). A Review on Decision Making of Risk Mitigation for Software Management. Journal

An International Journal JSCDSS E-ISSN: 2289-8603

of Theoretical and Applied Information Technology, 76(3). 333-341.

Noraini, C. P., Bokolo, A. J., Rozi, N. H. N. & Masrah, A. A. M. (2015a). A Review on Risk Mitigation of IT Governance. Information Technology Journal, 14 (1), 1-9.

Noraini, C. P., Bokolo, A. J., Rozi, N. H. N. & Masrah, A. A. M. (2015b). Risk Assessment of IT Governance: A Systematic Literature Review. Journal of Theoretical and Applied Information Technology, 71(2). 184-193.

Noraini, C. P., Bokolo, A. J., Rozi, N.H. N. & Yusmadi, Y. J. (2015c). Proposing a Model on Risk Mitigation In IT Governance. Paper Presented at the Proceedings of the 5th International Conference on Computing and Informatics, (ICOCI 2015).

Omar, A. (2014). Operational Profile Modeling as a Risk Assessment Tool for Software Quality Techniques. Paper Presented at the International Conference on Computational Science and Computational Intelligence.

Sadiq, M., Ahmad, M. W., Rahmani, K. I. & Jung, S. (2010). Software Risk Assessment and Evaluation Process (SRAEP) using Model Based Approach. Paper Presented at the 2010 International Conference on Networking and Information Technology.

Sendi, A. S., Shajari, M., Hassanabadi, M., Jabbarifar, M. & Dagenais, M. (2012). Fuzzy Multi-Criteria Decision-Making for Information Security Risk Assessment. The Open Cybernetics & Systemics Journal. 6(3) 26-37.

Shikha, P. & Selvarani, R. (2012). An Efficient Method of Risk Assessment using Intelligent Agents. Paper Presented at the Second International Conference on Advanced Computing & Communication Technologies.

Xiaofei, W., Xiaohong, L., Ruitao, F., Guangquan, X, Jing, H. & Zhiyong, F. (2014). OOPN-SRAM: A Novel Method for Software Risk Assessment. Paper Presented at the 19th International Conference on Engineering of Complex Computer Systems.

Wang, H. & You, L. (2015). Software Risk Assessment Method based on Fuzzy Neural Network. Paper Presented at the International Conference on Computer Science and Intelligent Communication (CSIC 2015).

Yao, Y., Cai, W. & Fang , N. (2016). Network & Information System Security Risk Assessment Technology. Paper Presented at the proceedings of 2016 13th International Bhurban Conference on Applied Sciences & Technology (IBCAST), Islamabad, Pakistan, 12th – 16th January, 2016.

Yin, K. R. (2004). Case Study Methods, Complementary Methods for Research in Education, Cosmos Corporation, American Education Research Association.

An International Journal
JSCDSS
E-ISSN: 2289-8603