Integration of Supportive Processes with Elementary Processes for Making Current Practices of Software Project Risk Management More Effective

Uzair Iqbal Janjua, Jafreezal Jaafar, Izzatdin B A Aziz
Department of Computer & Information Sciences
UNIVERSITI TEKNOLOGI PETRONAS
Perak, Malaysia
uzair.iqbal.janjua@gmail.com, jafreez@petronas.com.my, izzatdin@petronas.com.my

***Abstract*—In literature to increase the success rate of software project development, several risk management approaches have been presented by researchers and practitioners. Usually, these risk management approaches are in the form of process models, steps and professional standards. Despite of many proposed risk management approaches, still the success rate of software project development is very low. This directed the researchers of software engineering field to address the need of effective risk management approaches. To make risk management effective, many researchers and practitioners, on the basis of their theoretical knowledge and practical experiences have defined strategies and have given many recommendations. The objective of this study is to integrate effective techniques and supportive processes identified from literature with the elementary processes of software risk management to make it more effective.**

Keywords-; software; project;effective risk management

I. INTRODUCTION

Since last forty years, researchers and practitioners reported many failures of software project development [1-4]. In 2004, only loss of two failure software projects, cost near to $1000 million [3]. Where, in some reports researchers have emphasized on the failure rate of software project development, while as many other researchers conducted numerous studies to determine the factors that can increase the success rate of software project development. Risk management is one of the several factors that have been found to contribute to project success [2, 4-9]. Most of the researchers, explicitly considered 'Risk Management' as an important success factor [10], however a few other implicitly relate the risk management as a primary factor for the success of software project development. For instance in [11], author identified primary and secondary factors of project success. However, author also stated that the identified primary and secondary success factors cannot be qualified without the effective risk management.

Since 1996, importance of risk management has increased when practitioners realized that risk management is more than the cost and schedule estimation activity [11]. In addition, in later years, researchers also identified many other advantages of risk management. For example, it can help the development team to avoid adversities, rework, and overload [2, 12] and most importantly, it helps to reduce the failure rate of software projects [11, 13-16]. This led the researchers and practitioners to consider risk management as one of the significant aspect of sound project management.

Advantages, mentioned above, emphasized the authors of project management books to include a full chapter or a specific section on risk management. Meanwhile, this directs the practitioners in industry to include risk management plans in overall project plans. Now applying risk management in software development project is not a question of choice but a necessity [7]. However the high failure rate [17, 18] of software project development directs the researchers and practitioners to address the need of more effective risk management [19].

According to research of Microsoft Corporation by putting 5% work into effective risk management, organizations have 50% –75% opportunity to complete project before the deadline [20]. Despite of popularity and significance of effective risk management for the successful development of software projects, limited work has been done to integrate the effective recommendations available in literature with most common risk management processes to increase the effectiveness of risk management. Hence, objective of this study is, *"to integrate effective techniques and supportive processes identified from literature with the elementary processes of software risk management to make it more effective"*.

The paper is structured as follows: section II is about the brief review of literature and previous studies conducted by the authors of this study. Section III presents the integration of basic and a few effective risk management processes. Section IV describes the conclusion.

II. LITERATURE REVIEW

The philosophy of risk management was not well discussed in the software development community until the late 1980s [21]. However, the need to identify the risks of the software projects to be developed and the mitigation of them has though, always been esteemed.

To manage risk factors, researchers over the last three decades have proposed different software risk management process models and methods. In literature of software engineering a lot of software project risk management professional standards [22, 23], guidelines and process models [12, 14, 24-26] are available. However, still the success rate of software projects development is very low [17, 27, 28]. Survey result conducted in 2008 shows; out of 100, only 32 software projects succeeded, 24 projects were completely failed and 44 projects were seriously threatened by risks [18]. This indicates the researchers to address the need to

978-1-4799-7896-0/15/$31.00 ©2015 IEEE

make risk management more effective to increase the success rate of software project development.

Researchers and practitioners, on the basis of their theoretical knowledge and practical experiences have given many suggestions and advices for using different tools and methods to make the current practices of software risk management more effective. However, still there is no model, which can be called the de facto effective risk management model [29-31].

A. Preivous Studies Conducted by Authors

In order to make current practices of risk management more effective, the authors of this study have conducted the two different studies in different times [32, 33].

In the first study, authors conducted the systematic literature review of two online databases from the year 1996 to 2012 with the key stream 'software effective risk management'. The main objective of study [32] was to identify the different techniques proposed by different researchers to make current practices of risk management more effective. After following the selection criteria, 34 papers were selected for the secondary study on the mutual understanding of two authors of the study. During the analysis of selected papers authors have identified many effective techniques suggested by the different researchers. Sixty eight percent of the selected papers in [32], addressed the manipulation or the addition of some of the key steps or processes in current software project risk management practices in order to make the risk management more effective. In addition, evidences gathered for the need of effective risk management and its positive impact on software projects success is enough to advocate the worthiness of effective risk management. Results of the study also indicated that effective risk management in software development not only helps to manage the critical risks but also help in decision making and ensures the customers' satisfaction and overall improved financial performance of the organizations.

Figure 1. Minimum Required Processes for Effective Risk Management for Software Development [33]

Recently, the authors of this study conducted another study with objective [33] to examine different risk management process models and standards with a specific focus on effective risk management techniques and processes, with the intention of gaining a better understanding of their valuable features and limitations. During the study [32] authors identified the most elementary processes by evaluating different risk management models, methods and professional standards on the basis of commonality as depicted in Figure 1. Following the evaluation, it was also found that all risk management models and professional standards have their own limitations as illustrated in Table I. Hence, there is need to incorporate the suggestions addressed in the literature by different researchers to make the models more effective to increase the success rate of software project development. During the evaluation, 2 professional standards, 4 general risk management process model /steps and 5 specific-process oriented risk management models have been studied. Evaluations of professional standards and models have been presented in Table I. In Table I, 'Y" represent that in this particular article author have conferred specific effective strategy or process. "N' represents that author remain silent about effective strategy or process. "P" means in article effective strategy/ process was discussed partially. Moreover, in Table I 'RM' is used to represent 'Risk Management'.

TABLE I. : EVALUATION OF RISK MANAGEMENT MODEL AND PROFESSIONAL STANDARDS AGAINST EFFECTIVE PROCESSES [33]

		Risk Management Plan	**Risk Identification**	**Risk Assesment**	**Risk Controlling**	**Risk Monitoring**	**Project Identification**	**Historical Database**	**Stakeholder Identification**	**Lesson Learned**
RM Professional Standards	[22]	Y	Y	Y	Y	Y	N	P	Y	Y
	[23]	Y	Y	Y	Y	Y	N	N	Y	N
General RM Process Model	[14]	Y	Y	Y	Y	Y	N	N	N	N
	[24]	Y	Y	Y	Y	Y	Y	P	Y	Y
	[12]	Y	Y	Y	Y	Y	N	N	N	N
	[26]	Y	Y	Y	Y	Y	N	N	P	Y
Specific Process Oriented RM Model	[34]	Y	Y	Y	Y	P	N	Y	N	N
	[35]	P	Y	Y	Y	N	P	N	N	N
	[36]	Y	Y	Y	Y	Y	N	P	P	Y
	[37]	Y	Y	Y	Y	N	Y	N	Y	P
	[8]	Y	Y	Y	Y	Y	N	N	N	N

III. EFFECTIVE RISK MANAGEMENT PROCESS MODEL

The process model is a most common risk management approach found in the literature [2]. Process models specify orderly sequenced interrelated sub processes for managing risks of the software projects. Usually, they specify tools, techniques and the individual sub process, believed to be necessary for managing risk in software projects.

Required processes and database for effective risk management as shown in Figure 2 with their need for the effective risk management are discussed now.

A. Project Specification

Software project development is the process of developing software project on the base of documented project specification. Project specification is designed and documented on the basis of customer/client demands and requirement.

A software specification is a complete description of a software project to be developed with all functional and non-functional requirements. Before starting on any project, it is the primary responsibility of the project manager to recognize and document all the requirement of customer to design a complete project specification. Without a project specification, project manager cannot start risk management. However to make risk management proactive for its effectiveness, project managers must start the risk management even before actual starting of the project.

B. Project Identification Process

Project identification can be initial phase of the software project effective risk management process model. In strategic planning process, project identification is common and first process. However the objective and use of project identification process for the software project effective risk management is different. The main purpose of this process is to identify the nature of the project to be developed before starting the actual risk management. Nature of project can be identified on the basis of project specification through the following parameters.

- Size [30, 38]
- Complexity and criticality [30, 38]
- Sector [37]
- Application [37]

Once the nature of project is identified, it might be beneficial to project manager in many ways. First and the foremost benefit would help him/her to search most similar developed projects with different parameters from the past already developed projects database. Identification of similar project can be helpful to the project manager for the early identification of common crucial risk. Usually, it has been observed that the projects developed for same type of application under same sector have many common risks [37]. Furthermore, lessons learned from past projects assist the manager to avoid the less effective risk controlling techniques. In addition, project type, size and complexity define how much risk management must be effective.

Project identification process also helps to identify the need of a software project risk manager other than the project manager [32] for risk management activities. The software project risk manager is a role in software development that is responsible to assist the project manager in risk management of underdevelopment project [39].

C. Historical Database

Historical database is the repository of already developed projects. Historical database contains the all information of risk management [5] activities performed in the previously developed projects. After completing the project identification process, a project manager can easily identify the similar projects from the historical database. Similar project can help the project manager [31] to make the risk management proactive which is the basic requirement of the effective risk management [32]. Furthermore, a project manager and risk manager can identify the effective risk mitigation techniques for the similar risks of project to be developed. This will help them to decrease the efforts, time and cost for the development of new effective mitigation techniques.

In addition to benefits of historical database mentioned above, it can also help to predict the required resources for risk management activities.

D. Stakeholder Identification Process

Every project to be developed involves some stakeholders and has its own requirement. And every requirement has many potential risks related to involved stakeholders. As the complexity and size of the project increases, list of potential risks of project to be developed also expands. This increase the range of stakeholders that need to be considered during the risk management activities [37]. To make the risk management more effective, all responsible stakeholders for risk management activities must be involved in different phases of risk management. But, to involve the responsible stakeholders, first there is need to identify all stakeholders of project to be developed [37]. In addition, at this stage risk manager and project manager must assign the role and responsibility of involved stakeholder to avoid conflict in later stages. Historical database can also be helpful to identify relevant stakeholders and their role in different phases of effective risk management.

Main objective of stakeholder identification process is to identify all type of stakeholders need to be involved in risk management. It must also identify the stakeholder which may be directly affected by crucial risks but not necessarily to be involved for the risk management activities. Result of this process will be in form of list of stakeholders. List may have two main types of stakeholders, external stakeholders (customer, user etc.) and internal stakeholders (development team, analyst etc.). Normally, project manager and risk managers are interested to just find the expectations of external stakeholders from the project and their opinion in a few areas of risk management activities. However, second type of stakeholders is more crucial for the effectiveness of risk management because of their direct involvement in risk management activities and direct influence of project on them.

E. Communication Management Process

Formal, disciplined, structured and proper communication is essential for effective risk management of software project development [40-42]. Effective communication of the information about proactively identified risks can help the project manager and risk manager to develop a consensus between different already identified stakeholders. Therefore, it is mandatory for the project manager and risk manager to manage the communication about risk management activities between all affected and involved stakeholders effectively.

Purpose of this process is to identify the method to be used for the communication about the crucial risks. Project manager and risk manager can choose one or many communication methods within same project. A few of communication methods are: Meetings, Brainstorming sessions, emails, Questionnaires/surveys, voice chat, Newsletters and bulletins.

F. Risk Management Planning Process

Once, project type, appropriate communication method and stakeholder identified then next task of the project and risk manager is to design a risk management plan. A risk management plan is a document that a project manager prepares to predict risks of the project to be developed, estimate the impact of predicted risks, and define control strategies to predicted risks.

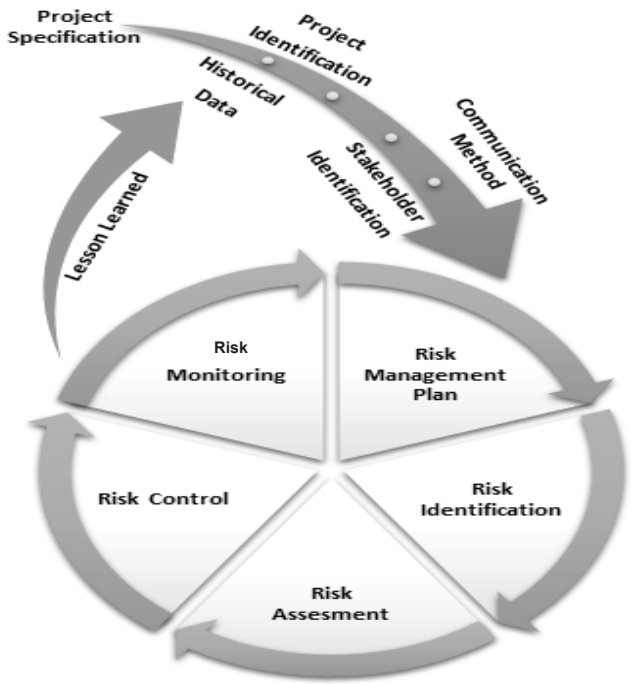

Figure 2. Software Project Effective Risk Management

Main objective of risk management planning process is to develop a comprehensive, documented risk management plan for each crucial risk on the base of available resources, funds, time and project specification. However, risk management planning process is not a onetime activity. It is a continuous and iterative process. Moreover, to make it more effective, risk management plan should be integrated with the overall project plan. Most of the models studied in [33] have this process with little differences in the functionality like in [12, 14, 22].

G. Risk Identification Process

Risk identification is the first and foremost phase of many conventional risk management guidelines. However, after following the literature of the general risk management and software project risk management, in this study it is considered as a second elementary process after risk management planning process.

Some degree of threats (risk) always exist in the projects and the purpose of risk identification process is to examine all key areas to trace the possible threats and their root cause [43] which can disrupt the success of the software project.

Risk Identification can be done by any one of the following methods; checklists of potential threats, Lessons learned files, work breakdown structure surveys, key process evaluation, meetings and brainstorming, Expert judgment, reviews of plans, processes, and work products [9, 44]. A checklist is probably the most frequent tools for risk identification. However, some researcher mentioned using of maximum method can make the identification of risk more effective [45].

H. Risk Assessment Process

After, the risk identification process, there is need to analyze quantitatively and qualitatively the identified and approved risks [9]. Indeed, this step helps the risk managers to shorten the list of identified risks; because, in practice, it is not possible to make a control strategy for each and every identified risk. In risk analysis/assessment processes all risks are analyzed carefully by the experts and redundant and tolerable risks are dropped from the list. Different techniques like performance model, modelling and simulation, cost model and decision analysis can be helpful for the risk analysis activities. Moreover involvement of experts for finding the risk exposure qualitatively will be beneficial.

I. Risk Controlling Process

Purpose of risk controlling process is to develop and apply the effective strategies to control the top prioritized risks. Although, it's top priority of all risk managers and project managers to mitigate risk completely. However in some scenarios it's not possible. In such scenarios the manger tries to decrease the severity of risk if it occurs. Other significant techniques used by the mangers are the risk acceptance, risk transfer and risk avoidance. In some risk management guidelines the process of development of controlling strategy is named as risk mitigation. However in general project risk management standards like 'PMBOK guide' and books like 'Project Management: A Systems Approach To Planning, Scheduling, and Controlling' [9], risk controlling process is known as "Plan Risk Response".

J. Risk Monitoring Process

After the initiating of risk controlling process, there is need to monitor systematically the prioritized risks against the risk mitigation or risk control strategies. While monitoring, expert can reconsider the risk identification and risk assessment process on the basis of project progress and actions. As a result new mitigation strategies can be introduced to mitigate risks where appropriate.

Where some authors believes, managers should monitor top ten risks [46] [13], there are also few who believe that for large projects there is need to track top twenty risks [47]. Different techniques can be used to monitor the risks like

"*milestone tracking, top-10-risk-item list that highlighted in weekly or monthly, milestone project review followed up appropriately with reassessment of the risk item*" [48]. In literature risk monitoring is also known as risk tracking.

K. Lesson learned

Experience is an excellent teacher in risk management [9]. To make current practices of risk management of software development more effective, keeping record of lesson learned while the development of a project can be significant for the future software project and for the project manages. In literature, this database have different names like in [43], "Lesson Learned" is named as "Risk Prediction". Whereas, in [49], it is stated as "Learn" process. Whatever the name of this database is, its main objective is to record all information about risks, which will become the objective source for upcoming projects [11]. Lesson learned database of current project can be the historical database for the future projects as shown in Figure 3.

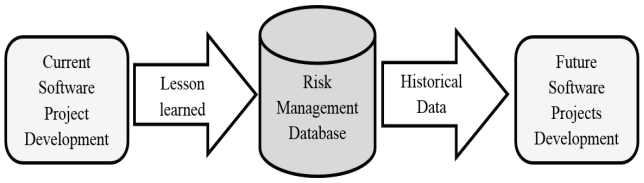

Figure 3. Relationship of lesson learned and Historical Database

L. Continuous Risk Management

Most of the renowned researchers of software risk management field agreed that risk management approaches cannot be effective until and unless it is continuously applied throughout the software project development lifecycle [2, 19, 30, 42, 50]. Continuous risk management is not only limited to the continuous risk identification. Along with the risk identification process other processes: risk assessment risk controlling and risk monitoring must be continually applied throughout the project to make the risk management more effective.

IV. CONCLUSION

Following the results of our previous different studies, in this study a few most effective risk management recommendations of different researchers are integrated well with elementary risk management processes in the form of database, and other process. Though, this is first step towards proposing the software project risk management process model, however, evidence of the need for effective risk management and its positive impact on software projects is enough to advocate the worthiness of effectiveness of the software project effective risk management model.

In future, the intention is to integrate other recommendations of researchers to make current practices risk management more effective. Moreover, in future, to find the significance of this model author of this study will evaluate it empirically.

REFERENCES

[1] A. M. Jenkins, J. D. Naumann, and J. C. Wetherbe, "Empirical investigation of systems development practices and results," *Information & Management,* vol. 7, pp. 73-82, 4// 1984.

[2] P. L. Bannerman, "Risk and risk management in software projects: A reassessment," *Journal of Systems and Software,* vol. 81, pp. 2118-2133, 2008.

[3] R. N. Charette, "Why software fails [software failure]," *Spectrum, IEEE,* vol. 42, pp. 42-49, 2005.

[4] G. J. Hidding and J. Nicholas, "Reducing I.T. Project Management Failures: A Research Proposal," in *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on*, 2009, pp. 1-10.

[5] L. L. Lobato, P. da Mota Silveira Neto, I. do Carmo Machado, E. S. de Alemida, and S. R. de Lemos Meira, "Risk management in software product lines: An industrial case study," in *Software and System Process (ICSSP), 2012 International Conference on*, 2012, pp. 180-189.

[6] S. Koolmanojwong and B. Boehm, "A look at software engineering risks in a team project course," in *Software Engineering Education and Training (CSEE&T), 2013 IEEE 26th Conference on*, 2013, pp. 21-30.

[7] H. Hashimi, A. Hafez, and M. Beraka, "A Novel View of Risk Management in Software Development Life Cycle," in *Proceedings of the 2012 12th International Symposium on Pervasive Systems, Algorithms and Networks*, 2012, pp. 128-134.

[8] A.-G. Tang and R.-l. Wang, "Software project risk assessment model based on fuzzy theory," in *Computer and Communication Technologies in Agriculture Engineering (CCTAE), 2010 International Conference On*, 2010, pp. 328-330.

[9] H. R. Kerzner, *Project Management: A Systems Approach To Planning, Scheduling, And Controlling* John Wiley & Sons, 2009.

[10] Y. Hu, X. Zhang, E. Ngai, R. Cai, and M. Liu, "Software project risk analysis using Bayesian networks with causality constraints," *Decision Support Systems,* vol. 56, pp. 439-449, 2012.

[11] H. R. Kerzner, *Project management: a systems approach to planning, scheduling, and controlling*: John Wiley & Sons, 2013.

[12] Linda H. Rosenberg, Al Gallo, Ted Hammer, and F. Parolek, "Continuing Risk Management at NASA," *CrossTalk, The Journal of Defense Software Engineering,* February, 2000.

[13] B. W. Boehm and R. Ross, "Theory-W software project management principles and examples," *Software Engineering, IEEE Transactions on,* vol. 15, pp. 902-916, 1989.

[14] B. W. Boehm, "Software risk management: principles and practices," *Software, IEEE,* vol. 8, pp. 32-41, 1991.

[15] R. N. Charette, *Software engineering risk analysis and management*: Intertext Publications, 1989.

[16] Y. Qu, M.-J. Yuan, and F. Liu, "The risk factor analysis for software project based on the interpretative structural modelling method," in *Machine Learning and Cybernetics (ICMLC), 2012 International Conference on*, 2012, pp. 1019-1024.

[17] J. Dhlamini, I. Nhamu, and A. Kaihepa, "Intelligent risk management tools for software development," in *Proceedings of the 2009 Annual Conference of the Southern African Computer Lecturers' Association*, 2009, pp. 33-40.

[18] G. J. Hidding and J. M. Nicholas, "Reducing IT Project Management Failures: Early Empirical Results," in *System Sciences (HICSS), 2014 47th Hawaii International Conference on*, 2014, pp. 4305-4314.

[19] Y. Tao, "A study of software development project risk management," in *Future Information Technology and Management Engineering, 2008. FITME'08. International Seminar on*, 2008, pp. 309-312.

[20] Y. Hu, J. Du, X. Zhang, X. Hao, E. Ngai, M. Fan, *et al.*, "An integrative framework for intelligent software project risk planning," *Decision Support Systems,* vol. 55, pp. 927-937, 2013.

[21] Y.-H. Kwak, "A brief history of Project Management," ed: Greenwood Publishing Group, 2005.

[22] "IEEE Standard for Software Life Cycle Processes - Risk Management," *IEEE Std 1540-2001,* pp. i-24, 2001.

[23] G. Purdy, "ISO 31000: 2009—setting a new standard for risk management," *Risk analysis,* vol. 30, pp. 881-886, 2010.

[24] J. Kontio, *Software engineering risk management: a method, improvement framework, and empirical evaluation*: Helsinki University of Technology, 2001.

[25] NASA, "NASA Procedures and Guidelines NPG 7120.5 B," in *Risk Management*, ed, 2002, p. 122.

[26] G. Holt, "Risk Management Fundamentals in Software Development," *Crosstalk: Journal of Defense Software Engineering,* August 2000.

[27] "The Standish Group. (2009).CHAOS Summary 2009: The 10 Laws of Chaos," 2009.

[28] C. López and J. L. Salmeron, "Risks Response Strategies for Supporting Practitioners Decision-Making in Software Projects," *Procedia Technology,* vol. 5, pp. 437-444, 2012.

[29] R. Stern and J. C. Arias, "Review of Risk Management Methods," *Business Intelligence Journal,* 2011.

[30] M. Kajko-Mattsson, J. Lundholm, and J. Norrby, "Industrial Opinion on the Effectiveness of Risk Management Methods," in *Computer Software and Applications Conference, 2009. COMPSAC'09. 33rd Annual IEEE International*, 2009, pp. 642-643.

[31] G. Holt, "Software Risk Management From a System Perspective," *Crosstalk: Journal of Defense Software Engineering,* vol. 18, 2005.

[32] U. Janjua, A. Oxley, and J. Jaffer, "Effective Risk Management of Software Projects (ERM): An Exploratory Literature Review of IEEE and Scopus Online Databases," in *Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013)*. vol. 285, T. Herawan, M. M. Deris, and J. Abawajy, Eds., ed: Springer Singapore, 2014, pp. 445-452.

[33] J. Jaafar, U. I. Janjua, and F. W. Lai, "Software Effective Risk Management: An Evaluation of Risk Management Process Models and Standards," in *Information Science and Applications*, ed: Springer, 2015, pp. 837-844.

[34] A. S. Khatavakhotan and S. H. Ow, "Rethinking the Mitigation Phase in Software Risk Management Process: A Case Study," in *Computational Intelligence, Modelling and Simulation (CIMSiM), 2012 Fourth International Conference on*, 2012, pp. 381-386.

[35] S. Islam, "Software development risk management model: a goal driven approach," in *Proceedings of the doctoral symposium for ESEC/FSE on Doctoral symposium*, 2009, pp. 5-8.

[36] P. K. Dey, J. Kinch, and S. O. Ogunlana, "Managing risk in software development projects: a case study," *Industrial Management & Data Systems,* vol. 107, pp. 284-303, 2007.

[37] D. Gotterbarn and S. Rogerson, "Responsible risk assessment with software development: creating the software development impact statement," *Communications of the Association for Information Systems,* vol. 15, p. 40, 2005.

[38] D. Gotterbarn and T. Clear, "Using SoDIS™ as a risk analysis process: a teaching perspective," in *Proceedings of the Sixth Australasian Conference on Computing Education-Volume 30*, 2004, pp. 83-90.

[39] U. I. Janjua, A. Oxley, and J. B. Jaafar, "Classification of software project risk managers: Established on roles and responsibilities," in *Computer and Information Sciences (ICCOINS), 2014 International Conference on*, 2014, pp. 1-6.

[40] J. Samad and N. Ikram, "Managing the risks: An evaluation of risk management processes," in *Multitopic Conference, 2006. INMIC'06. IEEE*, 2006, pp. 281-287.

[41] J. Nyfjord and M. Kajko-Mattsson, "Communicating Risk Information in Agile and Traditional Environments," in *Software Engineering and Advanced Applications, 2007. 33rd EUROMICRO Conference on*, 2007, pp. 401-408.

[42] S. M. Avdoshin and E. Y. Pesotskaya, "Software risk management," in *Software Engineering Conference in Russia (CEE-SECR), 2011 7th Central and Eastern European*, 2011, pp. 1-6.

[43] C. Schmidt, P. Dart, L. Johnston, L. Sterling, and P. Thorne, "Disincentives for communicating risk: a risk paradox," *Information and Software Technology,* vol. 41, pp. 403-411, 1999.

[44] P. Jalote, A. Palit, and P. Kurien, "The timeboxing process model for iterative software development," *Advances in Computers,* vol. 62, pp. 67-103, 2004.

[45] E. H. Conrow, "Risk Management for Systems of Systems," *Crosstalk: Journal of Defense Software Engineering,* vol. 18, February 2005.

[46] T. Addison and S. Vallabh, "Controlling software project risks: an empirical study of methods used by experienced project managers," in *Proceedings of the 2002 annual research conference of the South African institute of computer scientists and information technologists on Enablement through technology*, 2002, pp. 128-140.

[47] T. W. Kwan and H. K. Leung, "A risk management methodology for project risk dependencies," *Software Engineering, IEEE Transactions on,* vol. 37, pp. 635-648, 2011.

[48] J. Wan, D. Wan, and H. Zhang, "Case study on business risk management for software outsourcing service provider with ISM," *Technology and Investment,* vol. 1, p. 257, 2010.

[49] C.-g. Pan and Y.-w. Chen, "An optimization model of CMMI-based software project risk response planning," *International Journal of Applied Mathematics and Computer Sciences,* vol. 1, pp. 155-159, 2005.

[50] Z. Yali, Y. Huixiu, and J. Xiangmei, "Study of Project Risk Continuous Process Pattern," in *Affective Computing and Intelligent Interaction*, ed: Springer, 2012, pp. 507-513.