An Incremental Risk Management Framework for Realizing Project Efficiency using Version Control

N Rama Rao
Associate Professor
Department of Computer Science & Engineering
Swarna Bharathi Institute of Science & Technology
Khammam

K Chandra Sekharaiah, Ph.D
Professor
School of Information Technology
Jawaharlal Nehru Technological University
Hyderabad

ABSTRACT

In software projects, often, risk handling capacities have a determinant power w.r.t. ensuring project efficiency. They can make or mar project efficiency. In this paper, an incremental risk management framework is presented. A unique sample risk management template with example is proposed to capture the modification. The revision to the existing risk management framework also captures inputs and outputs at every stage of risk management process. Together with these two, dashboard or graphical user interface (GUI) or web user interface (UI) is designed for ease-of-use and to build a work habit where risk management team members, developers, leads and managers must be able to state software or hardware risks at every stage of risk management process. The new framework maintains a repository to manage a central risk log which is at the center of spiral model; i.e. after every crucial step recording in repository takes place. The data and previous documentation from repository serve as input for the next step. As risk factors w.r.t. software configuration management is affected by the modification, a case study is presented w.r.t. how including version information in risk templates can be remedial for the software(s/w) risks involved.

General Terms

Risk Management, Source Code Management and Version Control.

Keywords

Software Risk Management (RM); Risk Template; Risk Repository; Dashboard; Roles & Responsibilities; Software Configuration Management (SCM); Version Control; and Incremental Risk Management Framework.

1. INTRODUCTION

Project risk management includes the processes of conducting risk management activities for a project. These processes interact with each other and with the processes in the other knowledge areas. Each process involves effort from one or more persons based on the needs of the project. Each process occurs at least once in every project and occurs in one or more of the project phases, if the project is divided into phases. To be successful, an organization should be committed to address risk management proactively and consistently throughout the project. Risk exists the moment a project is conceived. When the RM occur in project development life cycle? [7].The best time to start RM is at the time of concept commit phase.

There is presently a proven risk that software or hardware may be shipped without set of version information, which could lead to product failure and recall. The lack of embedded version information in files makes it hard for end users to accurately identify the running version of the software and thus, it would prove much harder to identify vulnerabilities in the software [6]. Given that projects never run just exactly as optimal plan. Systematic RM helps to lower project costs, higher revenues, reduced project threats or vulnerabilities or failures, and creates new opportunities for new projects.

The remainder of this paper is organized as follows. Section 2 presents risk management & plan. Section 3 presents risk identification, categories of risks & risk commit. Section 4 presents risk recording, repository & communication are as part of an incremental risk management framework for realizing project efficiency; Section 5 presents risk assessment & risk analysis. Section 6 presents risk response planning. Section 7 presents prevention, mitigation and execution commit. Section 8 presents software configuration management & deployment. Section 9 presents monitoring, tracking & reporting. Sections 10 presents control, measurement and learn, and finally section 11 presents the conclusion and future enhancement.

2. Risk Management & Plan (RMP)

The main objective of RM is project realization on-time and improved customer satisfaction. Proper RM includes maximizing the probability and consequences of positive events and minimizing the events adverse to the project objectives. RMP defines risks, roles & responsibilities, process, thresholds, reporting, tracking, controlling and learning [1-6], [10]. Managing risks in a project is imperative for its success. We need to have a process (or processes) in place for risk management to be effective. The six steps for a project manager for risk management are: (1) identifying (2) classifying (3) analyzing and prioritizing risks (4) taking action to reduce the exposure and developing mitigation plans (5) implementing contingency plans if the risk occurs (6) reviewing risks with higher-level management periodically during the project lifecycle to assure that they are properly addressed and not reproduced (7) Opportunity management identifies opportunities that, if taken advantage of, will improve the success of the program [1], [2], [3], [4]. The risk management is concerned with the dynamic allocation, utilization, and direction of resources (both human and technical), with time. Following paragraphs explain the roles & responsibilities.

Roles and Responsibilities:

Employees have legal responsibilities to co-operate with their employer’s efforts to improve performance of the product and contribute to the baseline. Psychological issues among the project team members are also important [1], [2]. The effects of organizational culture on risk management and the guidelines suggested cover the significance of culture during risk management process of software development project [1]. It is essential to assign clear responsibilities on who will lead on what action, and by when? The person who raises a risk do not necessarily also becomes the owner. Risks are best handled when they are predicted and are assigned with ownership.

Program Manager (PM)

The Program Manager has ultimate responsibility for ensuring that the risk management process is followed and integrated into the overall planning as outlined by the program plan. The PM participates in the identification, analysis of risks, keeps senior management informed of high exposure risks and opportunities. The PM is directly responsible for preparing and conducting risk identification meetings, managing the risk data for the program, and including risk summary information in program status reports.

Risk Management Team

The project's risk management team is responsible for identifying, analyzing, and mitigating project risks and opportunities. The risk owner is responsible for closely monitoring the risk and providing the RM team with updates.

Stake Holders

For effective RM, management team above the program manager addresses the following aspects: provide direct guidance and applaud successes concerning mitigation plans and actions that are effective; support identification of 'Best Practices'. Communication among the project manager, team members, project board and stakeholders is crucial in handling risks. In many failed projects, a team member knew of the fatal risk, but the project manager did not. Risks that have an expected value over a certain threshold may require special management attention. PM reviews high risks on regular basis. The process also includes authorization process, resources, budget and accountability. Risk is measured based on the response risk questionnaire [8]. Teams or Team Members are located across the globe. 24*7 is advantage of virtual teams. Schedule risks include decision making delay due to different time zones, waiting for approval, new tasks, decisions, network delay or failures (Virtual Private Networks) and communication gap. Sometimes directors or managers located in different time zones or countries do not approve until they personally come and verify the project. This leads to delay and collaboration problems. Solutions include video conferencing, chat tools: internal or external, Voice over IP (VoIP), team viewer, log me-in and web-ex, etc. Subject matter experts (SMEs)/project associates/ development team responsible for technical risks such as architecture, design, platform, software and security risks.

3. Risk Identification, Categories of risks & concept commit

Risk identification steps were briefed in [9]. The primary risks are the risks that affect the project positively or negatively. We need to identify the causes and the effects of risks. False positive risk refers to identifying a vulnerability that is not present; false negative risk refers to failing to identify the presence of vulnerability. When a change/enhancement/threat/ vulnerability/error/bug /issue mishandled, it leads to a risk. Risk check list is explained in [9].Risk categories include known vs. unknown, project, technical, business, and predictable, unpredictable risks [5], [9]. Other risks are software requirement, software cost, software scheduling, software quality, and software business risks briefed in [4]. Some other risk categories are budget, scheduling, operational, software development risks, risks associated with customer characteristics discussed in [1], [4], [9]. The process, platform and business risks were briefed in [4]. Detection methods such as interviews, brain-storming, desk research, case studies, surveys, SMEs and study project documentation are explained in [1].

The next step after risk identification is risk commit or concept commit. During the concept commit phase a specific risk or opportunity is defined and evaluated (i.e., solution overview). It involves extensive discovery on assigning roles for RM team, evaluating risks/issues, technologically feasible or not, roadmap, plan, resource management, milestones and approvals. The next step after risk identification is risk recording.

4. Risk Recording, Repository and Communication

This paper proposes a repository to manage a central risk log for scoring risks and their interpretation. Score based on probability and impact, highest-to-lowest sorting, thresholds: To decide which risks will and will not be acted upon. Regular update is needed on reporting risks, risk rating matrix along with current status, mitigation plan, next check-point date, and tracking information. The dashboard presents a meaningful summary of the vulnerabilities found, prioritize & explain vulnerabilities and provide possible remediation suggestions.

As shown in fig 1, the risk management process is cyclic nature of spiral Model which is a risk driven process model, starts from RMP & identification of a risk and it may result in identification of another new risk i.e. after every crucial step recording in repository takes place using risk template shown in Table 1, Table 2, Table 3 and Table 4. The data from repository will be taken as input for the next step. Findings are recorded and implemented. We need to develop software for dashboard as it provides GUI or Web UI that provides an easy interface for input/output. When a step is complete we must make sure that the results of the phase are documented in a clear fashion and are available to all persons involved in the project. This will include development overview, test overview which includes test cases, test summary reports, bug reports, and documentation. The document should be circulated to verify the truth of the information it contains. This circulation may well go outside of the immediate group of persons responsible for its content. The main aim is that all key parts which in total 'define' the project are unambiguous. This may not always be the case and such differences should be documented. All those responsible for the document should be noted. There should be a review of the data collected within the document to make sure it is relevant for the status of the project. When all this is done, it is ready for circulation. The sequence of steps in risk recording is: document milestones, deliverables, threats, vulnerabilities, opportunities, implementation plans and responses, etc., and then check, review and circulate.

5. Risk Assessment & Analysis

Risk analysis steps are: to characterize and risk assessment, assessing threats/vulnerabilities/risks, assign probability/ impact, timeframe, rank the risks, compute the total risk, qualitative effect analysis, semi-quantitative effect analysis,

Table 1. Risk Description

Risk ID # or Title	Sort of risk	Recorded Date	Stakeholder	Description
Failover	Project	13-07-2013	Client	The production environment does not exactly replicate the deployment environment so unforeseen issues may arise such as performance issues load sharing between high availability (HA) nodes.
Automation Reqs.	Testing	13-07-2013	Quality & Analysis (QA) Team	Test team has requested developers to provide API to simulate or to automate testing
Missing Version Info.	Software	15-07-2013	Employer	When software files are checked in, developers have to put in version field. Automation is required.
Poor Skill Set	Human Resource (HR)	13-07-2013	Employer	Lack of people(attrition) and poor skills in team
Change in Customer Requirements	Quality	15-07-2013	Client and Employer	Inability to meet customer expectations

Table 2. Risk Sorting Matrix

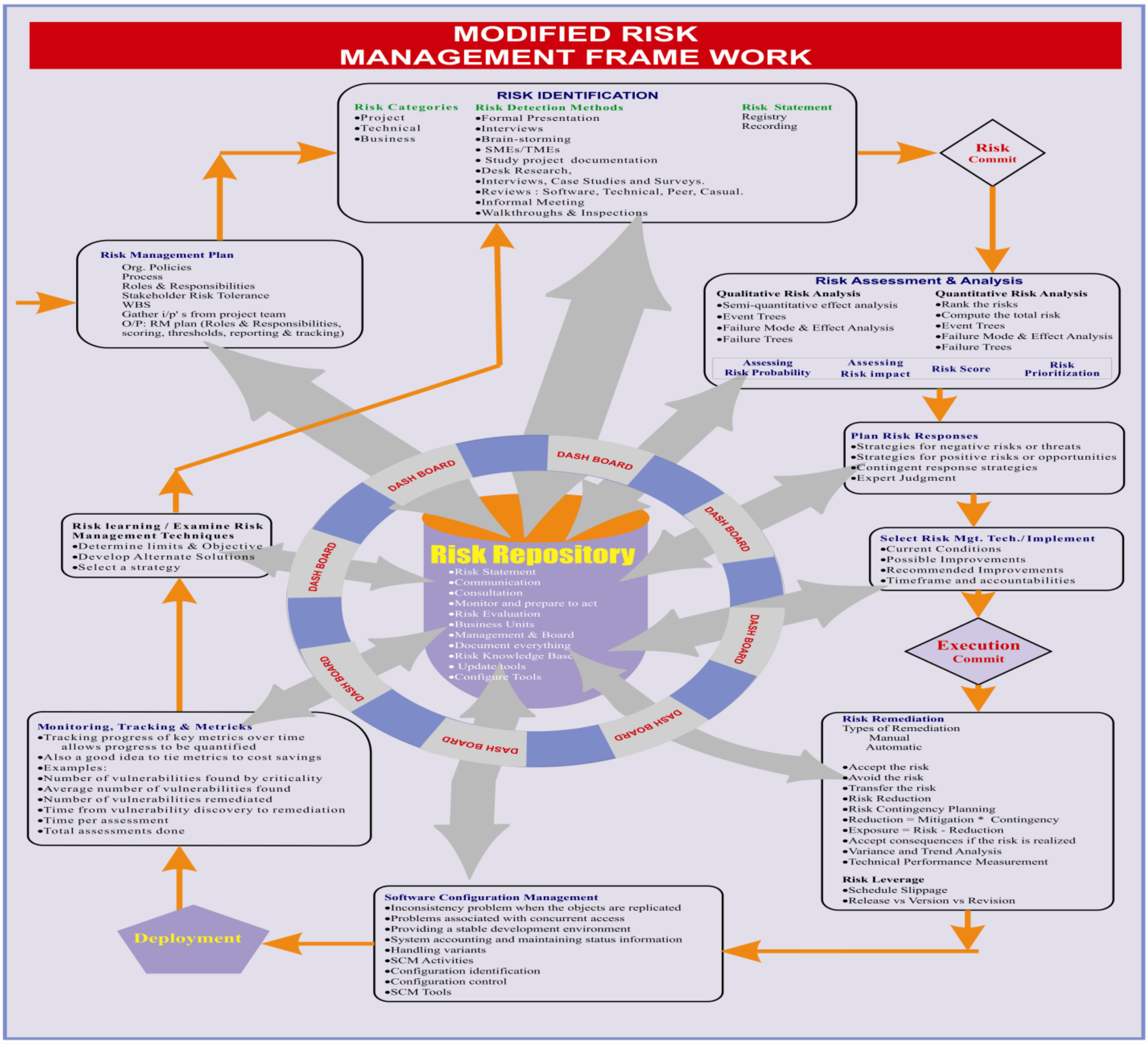
Risk ID # or Title	OS (Operating System)	Impact Region	Probability	Impact Score	Risk Score
Failover	Linux	External	2	3	6
Automation Reqs.	Windows/Linux	Schedule	3	5	15
Missing Version Info.	Linux	Project	5	10	50
Poor skill set	Windows/Linux	Schedule	4	8	32
Change in Customer Requirements	OS/Linux	Quality	3	5	15

Table 3. Risk Action Plan

Risk ID # or Title	Severity	Priority	Symptoms	Risk Status	Risk Reduction Tech.	Action Plan					
						Mitigate	Ignore	Accept	Prevention	Transfer	Exploit
Failover	MAJ	P2	Deployment Failure	Open	Create exact setup as Client	Perform alpha & Beta Testing	--	--	--	--	--
Automation API	MIN	P3	More time required to complete the testing	Open	Resolve longer test time	Commit additional test resources	--	--	--	Development team will provide API	--
Missing Version Info.	CRI	P1	Difficulty in identifying right source file.	Active	Need to be Addressed	--	--	--	It would be preferable to have those fields automatically updated since the release engineering system knows which values should go into those fields at check-in time. Hence it eliminates human error related to updating version fields & improve productivity by eliminating manual step.	--	Need to be documented any objections and consequences before approval.
Poor Skill Set	MAJ	P2	More defects	Observe	Provide training.	Recruit the team with required expertise, co-operation across teams,etc.	--	--	Knowledge Transfer to be done.	--	--
Change in customer requirements	MAJ	P2	--	Open	Accept	--	--	Accept & Implement	Issues need to be analyzed and to be implemented.	--	--

Table 4. Risk Control & Learn

Risk ID # or Title	Tracking	Control	Learn	Person Responsible	Comments / Remarks
Failover	Continuous monitoring of implementation.	Need to address the issues which arise during implementation	Meet with the Dev. and QA team to review the key risks and get a go /no-go decision to proceed with planning.	Project Manager (PM)	Please ensure accurate information regarding services which will run.
Automation API	Follow up	--	QA to be trained.	Dev. Team	Testing team may be trained to perform this itself.
Missing Version Info.	Need to document any subsequent actions taken.	Need to revise risk action plan.	Meet with the Dev. and QA team to review the key risks to insert version information in various files.	PM	Added Person X & Team as contributors to implement this due to their earlier significant contributions.
Poor Skill Set	Defect count reduced or not	--	Review	HR Team	Employer has to ensure good organization culture & Motivate people to retain the people.
Change in customer Requirements	Follow CR Process	Take corrective action when events occur.	Identify new risks resulting from mitigation	PM	Need to establish communications.



MODIFIED RISK MANAGEMENT FRAME WORK

RISK IDENTIFICATION

Risk Categories
•Project
•Technical
•Business

Risk Detection Methods
•Formal Presentation
•Interviews
•Brain-storming
• SMEs/TMEs
• Study project documentation
•Desk Research,
•Interviews, Case Studies and Surveys.
•Reviews : Software, Technical, Peer, Casual.
•Informal Meeting
•Walkthroughs & Inspections

Risk Statement
Registry
Recording

Risk Management Plan
Org. Policies
Process
Roles & Responsibilities
Stakeholder Risk Tolerance
WBS
Gather i/p' s from project team
O/P: RM plan (Roles & Responsibilities, scoring, thresholds, reporting & tracking)

Risk Assessment & Analysis

Qualitative Risk Analysis
•Semi-quantitative effect analysis
•Event Trees
•Failure Mode & Effect Analysis
•Failure Trees

Quantitative Risk Analysis
•Rank the risks
•Compute the total risk
•Event Trees
•Failure Mode & Effect Analysis
•Failure Trees

Assessing Risk Probability | Assessing Risk impact | Risk Score | Risk Prioritization

Plan Risk Responses
•Strategies for negative risks or threats
•Strategies for positive risks or opportunities
•Contingent response strategies
•Expert Judgment

Select Risk Mgt. Tech./Implement
•Current Conditions
•Possible Improvements
•Recommended Improvements
•Timeframe and accountabilities

Risk learning / Examine Risk Management Techniques
•Determine limits & Objective
•Develop Alternate Solutions
•Select a strategy

Risk Repository
•Risk Statement
•Communication
•Consultation
•Monitor and prepare to act
•Risk Evaluation
•Business Units
•Management & Board
•Document everything
•Risk Knowledge Base
• Update tools
•Configure Tools

Monitoring, Tracking & Metricks
•Tracking progress of key metrics over time allows progress to be quantified
•Also a good idea to tie metrics to cost savings
•Examples:
•Number of vulnerabilities found by criticality
•Average number of vulnerabilities found
•Number of vulnerabilities remediated
•Time from vulnerability discovery to remediation
•Time per assessment
•Total assessments done

Risk Remediation
Types of Remediation
Manual
Automatic

•Accept the risk
•Avoid the risk
•Transfer the risk
•Risk Reduction
•Risk Contingency Planning
•Reduction = Mitigation * Contingency
•Exposure = Risk - Reduction
•Accept consequences if the risk is realized
•Variance and Trend Analysis
•Technical Performance Measurement

Risk Leverage
•Schedule Slippage
•Release vs Version vs Revision

Software Configuration Management
•Inconsistency problem when the objects are replicated
•Problems associated with concurrent access
•Providing a stable development environment
•System accounting and maintaining status information
•Handling variants
•SCM Activities
•Configuration identification
•Configuration control
•SCM Tools

Fig 1: An Incremental Risk Management Framework

event trees, quantitative effect analysis, failure mode & effect analysis, failure trees, severity and prioritization were explained in [3], [5], [9], [10]. As shown in the risk template Table 1, Table 2, Table 3 and Table 4 with risk id as common field in every table and Fig 1 defines the risk management process. The column in Table 1 labeled "Risk ID" (Risk Identification Number) acts as the primary key in all four tables and the other columns include:

Risk ID : DDMMYYYY (e.g., 10032013 - March 10, 2013) like CR (Change Request) ID

Sort of Risk: General or Technical, Quality or Performance; Extl. = External; Orgl. = Organizational; PM = Project Management;

Stakeholder: Could be Customer, Employer, Development Team or QA Team

Impact Region: Quality, Cost, Schedule & Scope

Probability: 1/3/5/7/9 1 = LOW & 9 = HIGH

Impact Score: 1/3/5... 1 = LOW & 5 = HIGH

Risk Score: Impact x Probability

Priority: P1/P2/P3/P4/P5 P5 = Low & P1 = High.

Risk Severity: NOR - Normal, CRI – Critical, MAJ – Major, ENH – Enhancement, etc.

Symbols: Represents the risk events

Risk Status: Open, Resolved, Active & Closed.

Action Plan: Describe how to deal with the risk.

Mitigation - Reduce the impact of the risk event on the project

Accept - Continuous monitoring of the risk, re-assess that risk and consider assigning risk triggers and risk metrics.

Transfer – Using third party software or Assign to third party.

Prevention - Make the risk less likely to occur at all.

Ignorance – Ignore the risk

Risk Reduction Technique – Technique used to reduce the risk.

Contingency - Alternate course of action should the risk event occur

Control: Executing risk action plans

Risk needs to be quantified in two dimensions viz. impact of the risk and the probability of the risk. A risk assessment questionnaire is available in [9]. A risk probability/impact matrix is available in [5]. We arrive at a risk rating by combining the impact and the probability of occurrence. Ex: - Suppose you determine that the impact of a certain key employee leaving your organization will cost you Rs. 200000. You also determine that the probability of this risk occurring is 40 percent. The risk rating in this case would be 80000. The risk with the highest rating will have the highest priority and the others will follow in decreasing order of ratings. The impact of each risk driver on the risk component is divided into one of four impact categories-negligible, marginal, critical, or catastrophic [9].

6. Plan Risk Responses & Schedule

This step involves response strategy, generate responses, elaborate responses & select responses. It also involves exploring all the possible ways to reduce the impact of threats (or exploit opportunities) and planning actions to eliminate the risks (or enhance the opportunities). Action plans should be appropriate, cost effective, and realistic and be able to track the risks throughout the project.

7. Prevention/Mitigation and Execution Commit

As shown in the risk template action plan or Table 3, risk remediation consists of mitigate, ignore, accept, prevention, transfer and exploit the risk. The execute commit consists of development overview, test overview, development testing plan which includes interoperability testing, stress, performance, unit/acceptance test plan, regression testing accepted and completed. It also involves making a software product without defects/must-fix-bugs, or with systematic documentation, support, and training, etc.

8. SCM & Deployment

File versioning is a major risk under SCM. Risk template which is detailed in Table 1, Table 2, Table 3 and Table 4 includes risks/issues like version field and corresponding risk information. When software files are checked in, developers have to put in version field. As time progresses the existing versions will be superseded by newer software which fixes new defects or offers improved features, performance or security. It is also necessary to have an acceptable patching or upgrade strategy in place. Automation is required for version update. Regular updating or patch should be allowed.

9. Monitor, Track and Report or Reassessment

Risk management must make a log of risks solved for future reference and new members, track risks with associated tasks and make sure they don't reappear. This may be of great help on the organizational level. Risk tracking also includes monitoring the probability, impact, exposure, and other measures of risk for changes that could alter priority or risk plans and ultimately the availability of the service.

10. Control, Measure & Learn

Risk control includes the process of executing risk action plans and their associated status reporting. It includes measures taken to prevent, detect, minimize, or eliminate risk to protect the integrity, confidentiality, and availability of information. Risk control also includes initiating change control requests when changes in risk status or risk plans could impact the availability of the service. Measure the effectiveness of the planned action and controlling the risk impact by understanding risk triggers & timely implementation of planned actions. Usually, each individual have different opinions & ways to deal with risks. Some go for avoidance and others go with risk taking. So, while working for a project, the approach to risk should be consistent to meet project objectives & this need to be documented in a risk management plan. Tracking progress of key metrics over time allows progress to be quantified and also a good idea to tie metrics to cost savings. Learning in other words means to communicate and document the risks in risk repository for all personnel to understand the project's risks, mitigation alternatives as well as risk data and to make effective choices within the constraints of the project [3].

11. Conclusions

In this paper a sample risk template is proposed. A revision of the risk management framework [1], [2], [4], [6] is proposed by introducing the concept of risk repository. It captures inputs and outputs at every stage of risk management process.

Together with these two, dashboard is designed to build a work habit where technical lead & developers must be able to state their top risks at every stage of risk management process.

Risk recording is at the center of spiral model; i.e. after every crucial step recording in repository takes place. The data from repository will be taken as input for the next step. File versioning is a major risk under SCM. When software files are checked in, developers have to put in version field. Automation of version updates in tune with the file updates opens new work arena for future research. The revised risk management can provide enormous advantages to an organization by cutting down on costs. It also ensures proper delivery as per schedule and leads to on time project realization.

12. REFERENCES

[1] Charu Verma and Saad Ali Amin. "Significance of Healthy Organizational Culture for Superior Risk Management during Software Development", IEEE Conf. on Developments in E-systems Engineering (DESE 2010), 6-8 Sep. 2010. London, UK. pp.182 – 189.

[2] Edzreena Edza Odzaly, Des Greer and Paul Sage, "Software Risk Management Barriers: an Empirical Study", Third International Symposium on Empirical Software Engineering and Measurement (ESEM 2009), China. 15-16 Oct. 2009, FL U.S.A. pp. 418 – 421.

[3] Engr. Shehnila Zardari, "Software Risk Management", International Conference on Information Management and Engineering, 3-5 Apr. 2009. Malaysia. pp. 375 – 379.

[4] Hooman Hoodat and Hassan Rashidi, "Classification and Analysis of Risks in Software Engineering", World Academy of Science Engineering and Technology 2009, pp. 446-452.

[5] IAN Sommerville. "Software Engineering: Risk Management", 7th Edition. Print.

[6] Martin Boldt, Bengt Carlsson and Roy Martinsson. "Software Vulnerability Assessment Version Extraction and Verification", International Conference on Software Engineering Advances(ICSEA 2007), 25-31 Aug. 2007. France, pp.59-65.

[7] M. M. Sharma, Prakriti Trivedi, Anil K. Dubey, Akanksha Toshniwal and Himanshu Swarnkar. "Pioneering an Automated Risk Removal Tools in Software Engineering", International Conference on Information Systems and Computer Networks (ISCON), 9-10 March 2013, India. pp: 104-107.

[8] Puspamitra Mishra, Nabarun Roy, Balamukund S, Rajni M S, Shilpi Jain and Jude Fernandez. "A Risk Framework for New Country Compliance for Global Software Companies", Seventh International Conference on Global Software Engineering Workshops 27-30 Aug. 2012, Brazil. pp: 1-5.

[9] Rogger S Pressman. "Software Engineering: Risk Management", 6th Edition. Print.

[10] Zhang Jun-guang and Xu Zhen-chao. "Method Study of Software Project Risk Management", International Conference on Computer Application and System Modeling (ICCASM 2010), China. 22-24 Oct. 2010. V8-9 - V8-12.