International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org, editorijaiem@gmail.com
Volume 2, Issue 7, July 2013 ISSN 2319 - 4847

Software Risk Management and Mitigation Model

[1]Narendra Kumar Rout, [2]Nirjharinee Parida, [3]Sushruta Mishra

[1,2&3]Gandhi Engineering College, BBSR

ABSTRACT

Software risk management is a software engineering practice with processes, methods, and tools for managing risks in a project. It provides a disciplined environment for proactive decision-making to assess continuously what can go wrong; determine what risks are important to deal with; and implement actions to deal with those risks. Risk management planning addresses the strategy for risk management, the risk management process, and the techniques, methods, and tools to be used to support the risk management process. This paper recognizes the increasing role of risk management in present software projects and aims at providing more support in this area. First we take a look how software risk enters into the enterprise then we show a data model of risk identification with the overview of risk assessment and mitigation process. We then show how we can apply insurance to Software. And finally we present a solution to managing software risks.

Keywords: Software project risk, Risk identification, Outsourcing, Risk mitigation, Risk insurance

1. INTRODUCTION

The software industry is one of the largest manufacturing industries in the world, with $350 billion in off-theshelf software sold each year and over $100 billion in customized code on top of that. Risk management is an investment; that is, there are costs associated with identifying risks, analyzing those risks, and establishing plans to mitigate those risks. Software risk management is a software engineering practice with processes, methods, and tools for managing risks in a project. The main objective of Risk Management is to identify potential problems before they occur so that risk handling activities can be planned and invoked as needed across the life of the product or project to mitigate adverse impacts on achieving objectives. It should begin at the earliest stages of project planning and continue throughout the total life cycle of the project. Software project risk management is an ethic in which the project team continually asseses what may negatively impact the project, determines the probability of such events occuring, and determines the impact should such events occur. It provides a disciplined environment for proactive decision-making to assess continuously what can go wrong; determine what risks are important to deal with; and implement actions to deal with those risks. Risk management planning addresses the strategy for risk management, the risk management process, and the techniques, methods, and tools to be used to support the risk management process.

Risk is a function of the likelihood of a given threat-source's exercising a particular potential vulnerability, and the resulting impact of that adverse event on the organization [1]. In IT systems, risk can be introduced from the internet, servers, networks, malicious insiders and even lapses in physical security. Risk is the possibility of loss. It is a function of both the probability of an adverse event occurring and its impact; the impact manifests itself in a combination of financial loss, time delay, and loss of performance. A risk is the precursor to a problem; the probability that, at any given point in the software life cycle, the predicted goals cannot be achieved within available resources. Risk cannot be eliminated from a software project, but it can be managed. Risk management is critical to the success of any software effort and is a strategic aspect of all software projects. Forces that contribute to loss or damage constitute elements of risk. Some influences are external to the enterprise and other influences are internal to the enterprise. These forces cannot be completely eliminated, and, hence, the enterprise has to take a calculated risk on its IT investment. IT risks are somewhat peculiar to each industry and/or firm. Risk can be classified into systematic and unsystematic risk [2]. Systematic risk refers to that portion of risk caused by external factors; this is common and may affect all firms. Virus, hacking, fire, natural disasters and power loss are sources of systematic risk. Their effect is felt by many of the companies that are placed in the same position. For example, a loophole in the Internet browser that is vulnerable for hacking affects all of the firms that use the browser. Whereas, unsystematic risk is the portion of total risk that is unique to the firm. The factors such as misuse of data, loss of data, application error, human interaction, inside attackand equipment malfunction can be cited for unsystematic risk. Unsystematic factors are largely independent of factors affecting the IT industry in general. Since these factors affect one firm, they must be examined for each firm. The proportion of systematic and unsystematic risk denotes degree of vulnerability of the firm to the external or internal factors. Systematic risk is also known as generic risk, and unsystematic risk is also known as specific risk. Even though systematic risk is common for all firms of similar nature, its effect is not the same across all firms. This may be due to differences inToday's application

has become the enterprise's ''new perimeter''. With better network-level security technology hardening the network perimeter, malicious attackers are now focusing their efforts to strike at the least defended points - the application. While hackers were once satisfied with defacing Web sites, unleashing denial-of-service attacks and trading illicit files through targeted networks, modern attackers are profit-driven. Financial and customer data have become valuable commodities and applications must be secure enough to protect them. Recent industry statistics confirm this trend. Data from Computer Emergency Response Team (CERT) reveals that the number of software vulnerabilities has risen dramatically and has eclipsed 7,000 new software vulnerability disclosures in the past year [3] – for example: personnel shortfalls, unrealistic schedules and budgets, developing the wrong software functions, developing the wrong user interfacing, gold plating, continuing stream of requirements changes, shortfalls in externally furnished components, shortfalls in externally performed tasks, real-time performance falls, straining computer science capabilities, etc. Meanwhile, Gartner and NIST report that 95% of all reported vulnerabilities are in software [4], 78% of threats target business information, and 75% of attacks target the application level [5]. Yet, even with these findings, most enterprises allocate less than 10% of their security spending to application security.

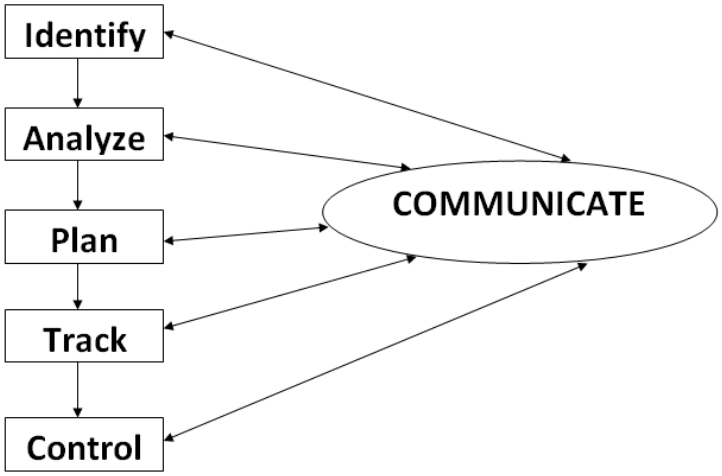

Fig. 1 Risk management process

2. SOFTWARE RISK MANAGEMENT PROCESS

There are several models available for risk management.

The model recommended in this section was developed bythe Software Engineering Institute (SEI) [7] and is shown in Figure 1.

Identify: Before risks can be managed its must be identified before adversely affecting the project. Establishing an environment that encourages people to raise concerns and issues and conducting quality reviews throughout all phases of a project are common techniques for identifying risks.

Analyze: Analysis is the conversion of risk data into risk decision-making information. It includes reviewing, prioritizing, and selecting the most critical risks to address. The Software Risk Evaluation (SRE) Team analyzes each identified risk in terms of its consequence on cost, schedule, performance, and product quality.

Plan: Planning turns risk information into decisions and actions for both the present and future. Planning involves developing actions to address individual risks, prioritizing risk actions and creating a Risk Management Plan. The key to risk action planning is to consider the future consequences of a decision made today.

Track: Tracking consists of monitoring the status of risks and the actions taken against risks to mitigate them.

Control: Risk control relies on project management processes to control risk action plans, correct for variations from plans, respond to triggering events, and improve riskmanagement processes. Risk control activities are documented in the Risk Management Plan.

Communicate: Communication happens throughout all the functions of risk management. Without effective communication, no risk management approach can be viable. It is an integral part of all the other risk management activities.

3. Risk Assessment

Risk assessment is the first process in the risk management methodology. It is based on three concepts: reviews, snapshots and reports that underpin the three layers of processing the risk-related information: identification, analysis and reporting. Reviews establish the framework for risk identification, snapshots pass the identified risks for further analysis

and reports communicate the results of risk assessment. The risk identification layer uses reviews to gather risk related information from a project. Reviews differ in terms of their scope, duration, participants and identification techniques. It is possible that two reviews overlap in time, however differing in their scope and/or participants. Risk related information collected during a review is represented as risk indication and identifies a particular risk, the involved project stakeholder, timestamp, the identification technique and possible comments. After the identification and analysis, the risk assessment report is generated. The report is a sort of "risk summary" of the present view at risks. It can then be used as input for risk mitigation related activities. It may also be taken as an input to the next risk review action. The output of the risk assesment process helps to identify appropriate controls for reducing or eliminating risk during the risk mitigation process [1]. The risk assessment methodology encompasses nine primary steps such as System Characterization, Threat Identification, Vulnerability Identification, Control Analysis, Likelihood Determination, Impact Analysis, Risk Determination, Control Recommendations, and Results Documentation.

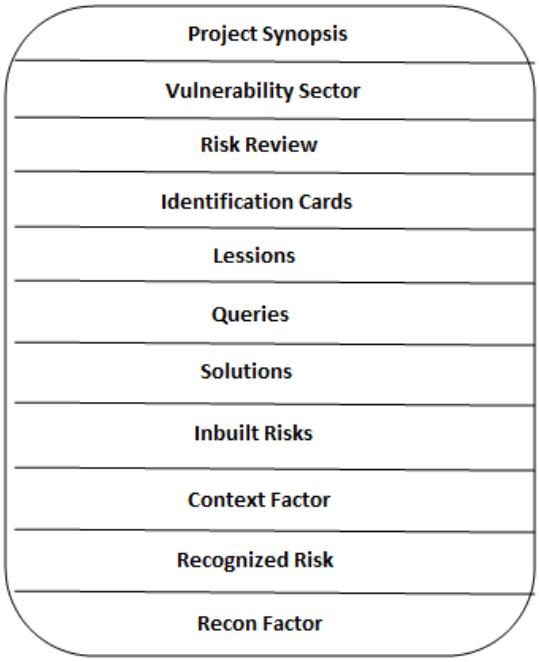

Fig. 2 Elements of risk management

4. Data Model of Risk Management

The model comprises the following elements:

Project synopsis: General project description (process, methodology, organization, size, initiation date).

Vulnerability sector: Area of a project that is exposed to a common type of risks (e.g. requirement specification, personnel management etc.)

Risk Review: This is the root object of the identification phase. Opening a new review starts risk identification activities whereas closing the review ends the risk information acquisition.

Identification cards: Checklists are used to collect information that helps to identify risks. A checklist includes its name, description, author's identification and its components.

Lessions: It is a component of a checklist. It can be hierarchically decomposed into more fine structuring elements as shown in Figure 3.

Queries: This is the lowest structuring level of a checklist (nevertheless it may include some sub-questions).

Solution: It represents the answer to a checklist question (it may be of different type: yes/no, range etc).

Inbuilt risk: Risk that is stored in the risk knowledge base. It may be selected by one or more answers to the questions.

Context factor: Risk factor providing the context for a risk stored in the risk knowledge base.

Recognised risk: Detailed risk description (from the risk knowledge base) in the context of a particular project. It is extracted from the knowledge base using the selection ofpredefined risks resulting from the answers to the checklist questions.

Recon factor: Context of the identified risk extracted from the risk knowledge base.

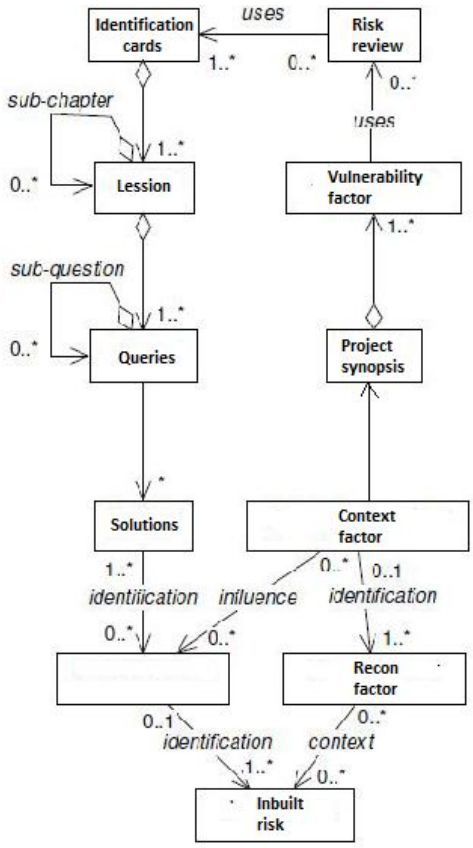

Fig. 3 Risk identification data model

5. Risk Mitigation

Risk mitigation, the second process of risk management, involves prioritizing, evaluating, and implementing the appropriate risk-reducing controls recommended from the risk assessment process. Because the elimination of all risk is usually impractical or close to impossible, it is the responsibility of senior management and functional and business managers to use the least-cost approach and implement the most appropriate controls to decrease mission risk to an acceptable level, with minimal adverse impact on the organization's resources and mission. Senior management, the mission owners, knowing the potential risks and recommended controls, may ask, "When and under what circumstances should I take action? When shall I implement these controls to mitigate the risk and protect our organization?" The risk mitigation chart in Figure 4 addresses these questions. Appropriate points for implementation of control actions are indicated in this figure by the word YES. This strategy is further articulated in the following rules of thumb, which provide guidance on actions to mitigate risks from intentional human threats:

When vulnerability (or flaw, weakness) exists -> implement assurance techniques to reduce the likelihood of a vulnerability's being exercised.
When a vulnerability can be exercised -> apply layered protections, architectural designs, and administrative controls to minimize the risk of or prevent this occurrence.
When the attacker's cost is less than the potential gain -> apply protections to decrease an attacker's motivation by increasing the attacker's cost (e.g., use of system controls such as limiting what a system user can access and do can significantly reduce an attacker's gain).
When loss is too great -> apply design principles, architectural designs, and technical and nontechnical protections to limit the extent of the attack, thereby reducing the potential for loss. The strategy outlined above, with the exception of the third list item ("When the attacker's cost is less than the potential gain"), also applies to the mitigation of risks arising from environmental or unintentional human threats (e.g., system or user errors). (Because there is no "attacker," no motivation or gain is involved.)

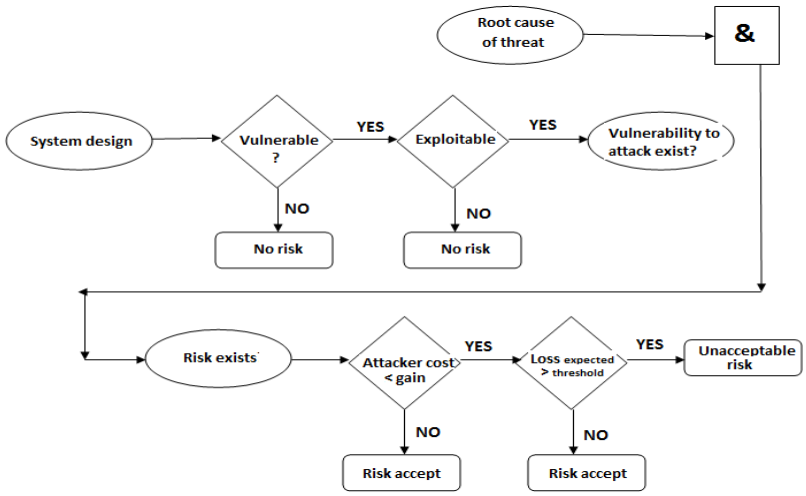

Fig. 4 Risk mitigation action points

6. CONCLUSION

The most important thing for a software project to do is to get focused on its critical success factors. For various reasons, including the influence of previous document-driven software management guidelines, projects get focused on activities which are not critical for their success. We can take some steps such as, ranking the project's most significant risk items, establishing a regular schedule for higher management reviews of the project's progress and so on to keep tracking on major risk factors. In this paper, we discuss about the risk management process with risk assessment and risk mitigation techniques, and also present software insurance concepts with a real life solution. We observed, still now software risk management reside in back seat but we should keep more focus on it. However, risk management is not a cookbook approach. To handle all of the complex people-oriented and technology-driven success factors involved in software projects, a great measure of human judgment is required.

REFERNECES

[1.] NIST Risk Management Guide for Information Systems Special Publication 800-30. July, 2002
[2.] Reilly, F.K.; K. Brown; Investment Analysis and Portfolio Management, Harcourt College Publishers, 2002
[3.] Microsoft Security Intelligence Report 2008 –Based on data from the DHS NVD & CERT)
[4.] Mark Curphey, "SoftwareSecurity Testing: Let's Get Back to Basics" October, 2004, SoftwareMAG.com
[5.] Theresa Lanowitz, "Now Is the Time for Security at the ApplicationLevel" 2005, Gartner
[6.] Mary Hayes Weier, "The Second Decade Of Offshore Outsourcing: Where We're Headed", Nov.2007, InformationWeek
[7.] Software Engineering Institute Web site: http://www.sei.cmu.edu/risk
[8.] Based in Burlington, Mass., Veracode is backed by .406 Ventures, Atlas Venture and Polaris Venture partners. Web site: http://www.veracode.com