Risk Management for the Development Of Software Project

Engr. Sobia Hashmi[1] MS (SE) Student,

Engr. Majid Kaleem[2] (Supervisor), and Engr. Ali Ahmed[3] (Supervisor)

Computer & Software Engineering Department

Bahria University, Karachi Campus

Karachi, Pakistan

sobia.hashmi454@gmail.com[1], majid.kaleem@bimcs.edu.pk[2], aiahmed@bimcs.edu.pk[3]

Abstract: Number of risks occurs for the development of successful software project. Sometimes it disturbs the schedule of whole project that result in the increase of cost. Therefore, the avoidance and mitigation strategies are used to overcome the effect of risk in the project. If software team use proactive approach then avoidance is the optimize solution but not all the risk can be avoided then it have to mitigate, and when risk become a reality, contingency plan has developed. The intention of this paper is to highlight number of risks, their arising issue and how to resolve them.

Keywords: Risk management, avoidance, mitigatation

1. INTRODUCTION

This is the fact majority of software development organization comprehend risk in a way that result in the increase of project development inconstancy and inadequacy. In project management, Kwak and lbbs analyze risk management as the least applied research area [1]. Moreover, risk management has not received enough consideration and does not appear to be broadly acknowledged within the software engineering, Adams and Pinto [5]. Avoidance of software risk management is fundamentally the organizational indolence and their resistant nature to change. [1]

Barry W. Boehm in his book "Software Risk Management", identified risk management in software projects as a separate area of research. That is studied particularly in the start of 90's. By the work of Boem on software risk management, an institution has been developed with the name "Software Engineering Institute (SEI)" in the middle of 90's.

The goal of this research is to accomplish an excellent and complete understanding of risk management.

2. LITERATURE REVIEW

2.1. Risk Identification

Various methods for risk identification have been used by managers of software project. According to Kulik and Weber, Four of them are going to be discussed [1].

The first one is *Formal Approach* for the identification of the various risks. According to this approach, complete, in-depth and comprehensive assessment of each risk accomplished [2]. The second approach is *Periodic Approach* includes the use of repetitive procedures for the specification and identification of the risks. The third approach is *Informal Approach*, that includes discussion with the system stakeholders involved in the project. The last one is *Adhoc Approach*, which gives an evaluation and mitigation of risks, when it appears initially in the project.

An international research on software development risk management by the Research Corporation KLCI, carried out in 2001, announced that the most common risk identification approach is the informal, used by 37% of the stakeholders.

2.2. Risk Classification

The software project risks consist of correlated classifications and every classification shall be defined individually. Moreover, berki review the research related to software development risk and classify them into five major categories [1]. McFarlan classify risk into three categories: project size, technology experience and project structure. He also suggested that project stakeholder should develop a software risk persona for every software project [3]. Sumner proposed nine risks that are unique in ERP projects through structured interviews, and compared the differences of software risks arise between ERP and MIS projects [1]. Han and Huang research consist of impact and the probability of each risk and calculate Risk Exposure (R.E) by the product of probability and consequence (RE = C*P). Where 'C' is the cost to the project should the risk occur, and 'P' is the probability of occurrence for a risk [4].

2.3. Performance Risk

According to Nidumolu [1], risk identified in the final development stages of a project is called Residual Performance Risk. It is divided into two parts Controllable Risk and Unforeseeable Risk, i.e.

Residual Performance Risk = Residual Controllable Risk + Unforeseeable Risk

Residual Controllable Risk can be controlled, and limited by using some techniques these risks exist continuously even the software has been deployed. While un-foreseeable risk neither identified nor controlled. Simply no idea about such risks when they arise.

3. METHADOLOGY

Research process is initialize by using descriptive method that consist of fact finding enquiries and surveys of different kinds including comparative and correlation methods. Data collection in research is done by observation and mailing questionnaire to practitioners (include project managers and programmers) of different software houses to evaluate the number of risks arise currently in the development of software project and then categorized them into 'very high', 'high', 'medium', 'low' & 'very low' according to the responses come from practitioners.

Questionnaire consist of various risks, i.e. funding and scheduling risks, requirements risks, risk associated staff, scope, relationship management, Sponsorship / ownership, staff turnover, development and environment risk, technology risk, product size risk, technical risk, management risks, reliability and other resource risks. All of these risks are discussed ahead.

4. AVOIDING AND MITIGATION STRATEGIES OF RISK

0.1. Funding and Scheduling Risks

When project is under funding for later years, because of artificial deadlines therefore project is under development, in fact the deadlines has reached. That results in deviation from budget.

1) *Avoidance:* Funding and scheduling risk can be avoided, when planning the project in phases or develop the plan as the phases progress, otherwise budget the project one phase at a time or approved the entire project in the beginning.

2) *Mitigation:* Funding and scheduling risk can be mitigate by re-evaluating the project cost-benefit analyses (CBA), trying to change the management process to approved more funding, otherwise explain flaws to the customer and their need for additional funding to continue the project.

0.2. Requirement Risks

This is due to the lack of understanding user needs, when the requirements are not freeze, or projects requirement are handled poorly. This situation left a bad catastrophic impact on customer.

1) *Avoidance*: Requirement Risks can be avoided by assign experienced project manager having skills to change management and controlling the project, establish a committee to change requirement and fulfill customer expectations or require management approval of changes.

2) *Mitigation:* This can be mitigate by meeting with customer and management to review the change management plan and adjust if necessary or revisit requirement, approved changes, and switch from initial or baseline requirements.

0.3. Risk Associated Staff

It includes lack of people with the right skills, lack of required knowledge in the project, or infrequent meetings of project team.

1) *Avoidance*: Risk Associated Staff can be avoided if we consider skill requirement before associating staff for particular project or review skills of already assigned staff and give them training, if necessary or do not commit for the project, if staff do not have required skills.

2) *Mitigation:* Mitigate this risk by replacing team members and train them, if there is enough time remaining to complete the project, or work with the existing team to determine reasons that how to overcome the shortage (that is shortage of skills and knowledge).

0.4. Scope

This is the basic element for starting any project. Therefore it is necessary for the software team to define goal and objectives before planning the project. The problem arises, when objectives are not clear or not understood.

1) *Avoidance*: To avoid risks arise due to scope factor conduct a project feasibility study (PFS) to determine its expected benefit to the business. If the scope is not clear then this is necessary to reject the project even in the selection phase. Otherwise define alternatives in the feasibility report, or quit the project and look for another. Because this is wrong decision to start the project whose scope is not clear.

2) *Mitigation*: Implement an alternative that was defined in risk planning report at the beginning or consider each phase of the project with top management. Otherwise implement any alternative approach from the risk plan or stop the project, if management doesn't have any other choice. At the end scope of the project should be clear and it fulfills user expectation completely.

0.5. Relationship Management

Risks that have been identified with managing project stake-holder relationships. It includes lack of cooperation from user, failure to meet user expectations, or higher expectations of users from developers and

management (that project will obviously fulfill their expectations).

1) *Avoidance:* Risk come into this category can be avoided when end user trying to participate in the team then feedback achieved is on the expectations of end user. Other way is to develop an approach to get feedback during the project that will tell current progress of project as well as relationship of management with each other, otherwise arrange requirements gathering sessions and involve right stakeholders in the session or define clearly scope and schedule to all stakeholders that project should be completed according to scope, in the schedule and within the budget.

2) *Mitigation:* These risks can be mitigate by including key stakeholders as soon as you consider a lack of user expectations (requirement do not fill the expectations and needs of user), meet with customers directly or face-to-face to actually review user expectations otherwise revisit to reevaluate scope and customer requirements.

0.6. Sponsorship / Ownership

Sponsorship/ownership risks include lack of corporate leadership, factors like management's loss of interest in the project, an unstable corporate environment, failure to make decisions at critical situations, some sort of confliction between user departments, and may be possible unethical behavior arise between communications.

1) *Avoidance:* These risks can be avoided when sponsor should set the project team's goals and vision, also encourage and support the project team, and remove roadblocks, if any. This is also necessary to communicate about the project status frequently to all stakeholders, otherwise identify weak links and resolve issues due to which the project performance become hidden.

2) *Mitigation:* Impact of such risk may vary in the development of project therefore to mitigate such risks first stop the project and meet with the management or sponsor to make a go/no go decision (either continue the project or no more). Otherwise work with the customer directly to understand the reasons for lack of commitment and bring management back on board using alternative communication methods.

0.7. Staff turnover

It comes when staff left the organization due to any reason. This left a critical impact on project cost, schedule and so on. Therefore, High turnover is a negative risk that may arise any uncertain condition.

1) *Avoidance:* There are different ways to avoid staff turnover. For example; to determine causes for turnover meet with current staff on the first hand (turnover because of any reason either low pay, poor working conditions, or some other reason), try to mitigate those causes before starting the project that are under our control. Once the project starts, assume turnover will occur and develop to ensure continuity when people leave. In order to avoid this situation, you have to organize project team in such a way that the information is broadcast about each development phase over each team member. Therefore, as the project phases are going to complete documents must be completed gradually. Moreover, assign a backup staff for any critical situation, if occur. Even a checklist of all possible negative or positive risk that may arise.

2) *Mitigation:* When there is no another option to avoid the staff turnover then mitigation strategy has been followed, information is documented, backup is available, and knowledge has been dispersed across the team. Those individuals who are leaving are asked to stop all work and spend their last weeks in "knowledge transfer mode" to the backup staff.

0.8. Development and Environment Risk

Overload of work to the developer and there is no peace of mind.

1) *Avoidance:* To provide peace of mind to the developer assign tasks in module and give a fix time slot to complete. Then the project will also complete in timely manner and chances of risks will be reduced.

2) *Mitigation:* If still there is no progress shown by developer then change the development strategy or process model for project.

0.9. Product size risk

This is associated with the overall size of the product. When size of the product is exceeded or reduced from estimated size or estimated size of the product in LOC or FP, therefore this is necessary to keep track of risk from beginning. If still the product size vary then try to overcome it much as possible. This usually occurs when size of variable selected randomly.

0.10. Technical Risk

Technical risks include problems with languages, project functionality, platforms methods, standards or processes. These risks arise, when so many useless constraints, foreign keys, candidate keys and primary keys

are used in the tables, lack of experience in the team, poorly defined parameters or dependencies on organization outside the direct control of the project team.

5. RESULTS AND DISCUSSION

By the analyses of risk assessment, among all the fourteen risks that have been discussed above, 8.2% are very high, 21.8% high, 29.6% medium, 35.7% low and 4.6% are very low. Percentage is calculated by using pie chart. This is shown in figure 5.1.
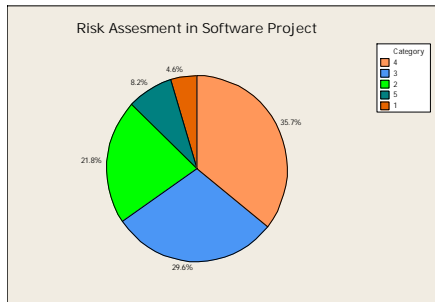

Figure 5.1

Analysis of variance (ANOVA) is a method for testing differences among means by analyzing variance. Here One-way anova is applied over fourteen groups of data. That result in $P=0.00$, therefore null hypotheses is rejected because P-value is less than 0.05

Normality has been calculated by Normal Probability plot, here most of the points are interconnected to the residual line.

The results from anova test indicate that technology risk is significantly lower than the other risks. Moreover, the corresponding p-value is being less than 0.05.
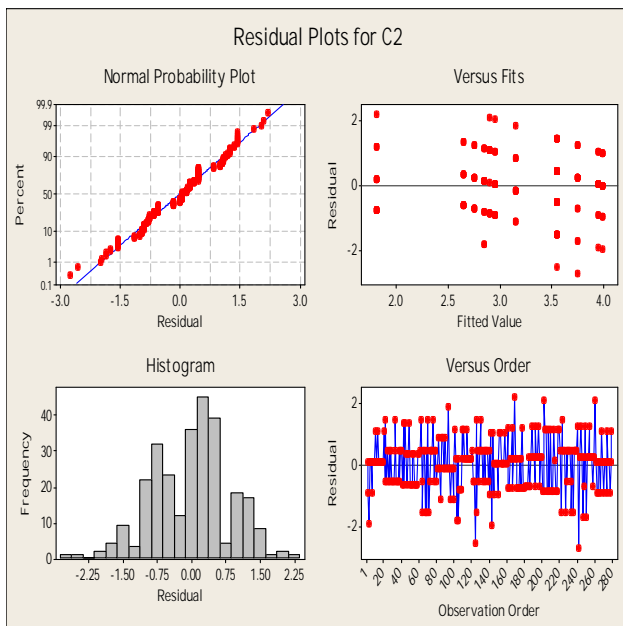

Figure 5.2

6. CONCLUSION

In this paper number of risks discussed with their avoidance and mitigation strategies. To overcome project failures, risks arise during the development of any software, this literature examined to identify those risk factors that destroy the development process of any project.

Most of the risks arise when there is a substantial difference between practical and theoretical prospective of both. Such differences are tried to be minimized in this paper.

7. REFERENCES

[1] Lazaros Sarigiannidis, Prodromos D. Chatzoglou, "Software Development Project Risk Management: A New Conceptual Framework" Journal of Software Engineering and Applications , 2011, 4, 293-305 doi:10.4236/jsea.2011.45032 Published Online May 2011

[2] Mira Kajko-Mattsson and et.al, "Mira Kajko-Mattsson", IAEND International Journal of Computer Science, Advance online publication: 20 Nov, 2008.

[3] S. J. Huang and W. M. Han, "Exploring the Relationship between Software Project Duration and Risk Exposure: A Cluster Analysis," Journal of Information and Management, Vol. 45, No. 3, 2008, pp. 175-182.

[4] W. M. Han and S. J. Huang, "An Empirical Analysis of Risk Components and Performance on Software Projects," The Journal of Systems and Software, Vol. 80, No.1, 2007.

[5] K. M. Adams and C. A. Pinto, "Software Development Project Risk Management: A Literature Review," Pro- ceedings of the 26th National Conference, Organizational Transformation: Opportunities and Challenges, American Society for Engineering Management, Rolla, October 2005, pp. 635-641.