DEVELOPING THE SOFTWARE PROJECT AND RESOLVING DIFFERENT TYPES OF RISKS ASSOCIATED

[1]Dr. J. Venkatesh, [2]Mr. D. Sathish Kumar, [3]Mrs. S. Thenmozhi and [4]Dr. P. Balasubramanie

[1]Associate Professor, School of Management Studies, Anna University : Chennai 600025, Regional Centre, Coimbatore, Jothipuram Post, Coimbatore - 641 047.Tamil Nadu, INDIA. Email: venkijv@gmail.com
[2]Full Time Ph.D Research Scholar, School of Management Studies, Anna University : Chennai 600025, Regional Centre, Coimbatore, Jothipuram Post, Coimbatore - 641 047.Tamil Nadu, INDIA. Email: sathish.mailme@yahoo.co.in
[3]Assistant Professor, Department of Computer Applications, Gnanamani College of Technology, AK Sumuthiram, Pachal - 637018, Namakkal District. Tamilnadu, India. Email: thenmozhi.s1983@gmail.com
[4]Professor, Department of Computer Science & Engineering, Kongu Engineering College, Perundurai, Erode - 638 052. Tamilnadu, India. Email: pbalu_20032001@yahoo.co.in

ABSTRACT: Manufacturing industries initiated software as vital role, for an accurate product development and manufacturing, instead of manual operation the system is fully automated and it gives an error less product, to develop software according to the customer need and satisfaction, softwares are developed with the customer specifications, in this article it describes the various risk in the software project development and the solutions to solve the risks. The purpose of the study is to focus on the different types of risks associated with development of a software project, to find out the best way to minimize those risks associated with development of software project and to analyze the impact of the risk factors on the project. A literature review in prominent academic journals in developing the software and resolving the risk associated and it has been conducted for the period 1997-2008. The objective is to collect, organize and synthesize existing knowledge relating to software development and the risk associated in developing the software. The study has described the different types of risks with development of software, the impact of the risk process on the project and finding the best way to minimize those risks with the calculations of percentage analysis and cross tabulation process. The value of this paper is to define the software development and the risk associated to analyze how research in this area has evolved during the period 1997-2008 and to recognize some lines of advance research.

KEYWORDS- Software projects, Different types of risks, minimize the risks, percentage analysis and cross tabulation

1. Introduction

Risk Management is a method that is used to reduce or eliminate risk before it can harm the productivity of a software project. With only 28% of software projects finishing on time and on budget, risk and the management of risk play an important role in software development. There are two ways that software engineers can handle risk. A reactive software engineer corrects a problem as it occurs, while a proactive software engineer starts thinking about possible risks in a project before they occur.

There are also specific risks associated with team members, customers, tools, technology, time estimation, and team size. Many of these risks can be minimized by the development methodology used for the project. There are many different tools that can be used to analyze the risk apparent in a project and that can help choose the best way to minimize or eliminate that risk.

Table 1- Several Types of Risk That Can Occur During a Software Development Project

RISK TYPE	DESCRIPTION
Generic risks	Generic threats across all projects. For example, requirements change, loss of team members, loss of funding
Product-specific risks	High level risks associated with the type of product being developed. For example: availability of testing resources
Project risks	affect project schedule or resources
Product risks	affect quality or performance of software
Business risks	affect the viability of the software

1.1 STATEMENT OF THE PROBLEM:

Recently it is witnessed that, the IT employees are facing more risk in developing the projects. Starting from the business planning level throughout the software development cycle, the management and employees are facing more risks in the software development projects. Due to those risks, the productivity and the quality of the projects are so badly affected. The business faces more losses. Therefore in order to reduce those risks and to improve the quality and the productivity of the projects an in depth investigation on the risk management is conducted in this study.

1.2 OBJECTIVES:

- To focus on the different types of risks associated with development of a software project
- To find out the best way to minimize those risks
- To analyze the impact of the risk factors on the project

1.3 SCOPE:

- This article provides details about the various risk causing factors & it's co factors.
- This article helps the manager to minimize the risk to certain extend.
- With the help of this study the organization will get awareness about their faults & will rectify their faults. So that the organization progress in all aspects.

1.4 LIMITATIONS:

- The article has been done in the middle level company. So that the findings may not be applicable to low level & top level of companies.
- Through this article the risk management of middle level company has been studied & suggestions are provided. So this may not be applicable to other similar concerns.
- The sample size is limited to 150.

2. REVIEW OF LITERATURE:

Klein (1999) find different types of risks will affect budget, user satisfactions, and system performance. Other studies indicate that 15 to 35% of all software projects are cancelled outright, and the remaining projects suffer from schedule slippage, cost overruns, or failure to meet their project goals.

Boehm and DeMarco (1997) mentioned that "our culture has evolved such that owning up to risks is often confused with defeatism". In many organizations, the tendency to 'shoot the messenger' often discourages people from bringing imminent problems to the attention of management. This attitude is the result of a misunderstanding of risk management.

Boehm (1991) identified 10 software risk items to be addressed by software development projects:

- Personnel shortfalls
- Unrealistic schedules and budgets
- Developing the wrong functions and properties
- Developing the wrong user interface
- Gold plating (adding more functionality/features than is necessary)
- Continuing stream of requirements changes
- Shortfalls in externally furnished components
- Shortfalls in externally performed tasks
- Real-time performance shortfalls
- Straining computer-science capabilities.

Jones (1998) further presented three key software risk factors and concerns of both executives and software managers.

- Risks associated with inaccurate estimating and schedule planning
- Risks associated with incorrect and optimistic status reporting
- Risks associated with external pressures, which damage software projects.

Boehm (1991) and Johnson et al.(2001). More extensive lists can be found in Addison and **Vallabh (2002), Barki et al. (1993), and Schmidt et al. (2001).** These lists are usually compiled from surveys of the experiences of stakeholders such as project managers who have been involved in software projects.

Boehm, 1988; de Camprieu et al., 2007; Keil et al., 2002; Mursu et al., 2003; Schmidt et al., 2001. This raises the prospect that risk assessment based on published checklists may be biased and/or limited in scope.

Generic options for responding to project risks by **DeMarco and Lister, 2003; Frame, 2003; Kerzner, 2003; Schwalbe, 2007**. Within these high-level options, specific responses can be formulated according to the circumstances of the project, the threat, the cost of the response and the resources required for the response.

MIS Quarterly released a special issue dedicated to the topic of offshore outsourcing in June **2008**. The field of risk management has been investigated by scientists from **1988 to 2000 (cf. Dibbern et al. 2004, pp. 34-38, and 54).**

3. RESEARCH METHODOLOGY:

Research is common parlance refers to a "search for knowledge". Once can also define research as a scientific and systematic search for pertinent information on a specific topic. In fact

Research is an "art of scientific investigation".Redmon and Mory define research as a "**systematic effort to gain knowledge".**

Research Design:
- o The type of research design used in this study is descriptive type of study.

Sample Size:
- o The sample size of the study is 150 employees in middle level companies.

Measurements & Scaling techniques:
- o The measurement scale that has been employed for this study is **Interval scale.**
- o The scaling technique that has been employed for the study is the Itemized Rating scale**.**

Data Collection Method:
- o In this study primary data have been used to collect the data for analysis and it has been collected through questionnaire method.

3.1 STATISTICAL TOOLS:

Percentage analysis
- o Percentage analysis is the total number of respondents divided by the total sample size and multiplied by 100.

Cross tabulation:
- o Statistical analysis provides techniques for studying the relationships among and between nominal variables. This is named as cross tabulation or cross tabs, cross classification and contingency table analysis.

4. PERCENTAGE ANALYSIS:

Table 2- Nature of Software Project the Respondent Works

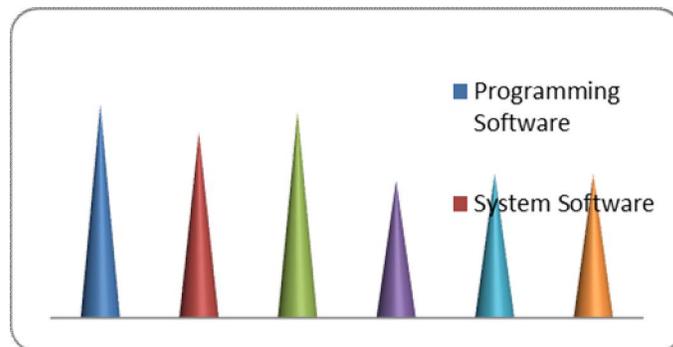| Nature Of Software Project	Frequency	Percentage
Programming Software	31	20.7%
System Software	27	18%
Application Software	30	20%
Inventory Management Software	20	13.3%
Utility Software	21	14%
Data Backup & Recovery Software	21	14%
Total	150	100

Source: Secondary Data

Chart 1-Nature of Software Project the Respondent Works

Inference:

From the above table & chart it's inferred that 20.7% of the employees are employed in developing programming software, 18% of the employees are in system software development, 20% of the employees are in application software development, 13.3% of the employees are employed with developing inventory management software, for utility software development 14% of the employees are appointed & remaining 14% employees are in data backup & recovery software development.

Table 3-Adaption of Risk Analysis in Business Planning Level

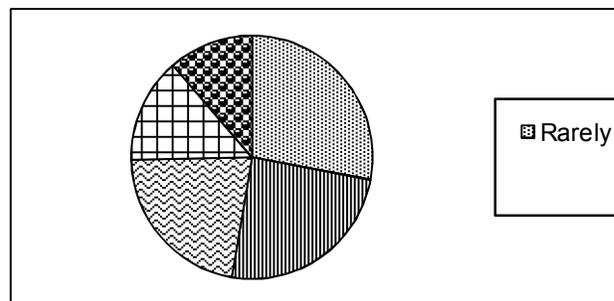| Risk Analysis On Business Planning Level	Frequency	Percentage
Rarely	42	28%
Occasionally	37	24.7%
Sometimes	33	22%
Frequently	21	14%
Very Frequently	17	11.3%
Total	150	100

Source: Secondary Data

Chart 2-Adaption of Risk Analysis in Business Planning Level

Table 4-Adaption of Risk analysis on Engineering Level

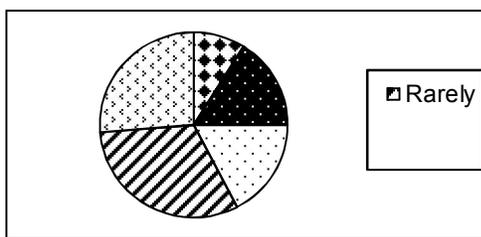| Risk Analysis on Engineering Level	Frequency	Percentage
Rarely	14	9.3%
Occasionally	25	16.7%
Sometimes	27	18%
Frequently	49	32.7%
Very Frequently	41	27.3%
Total	150	100

Source: Secondary Data

Chart 3-Adaption of Risk Analysis on Engineering Level
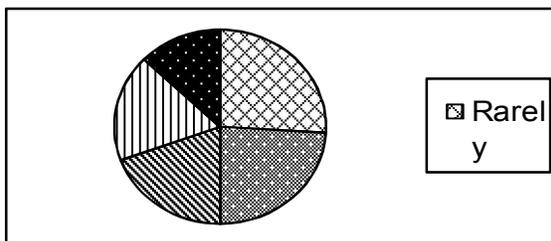
Inference:

From the above table & chart we can understand that at engineering level of software development process, the risk analysis is done frequently. Majority of the respondents say 32.7% accepts this statement.

Table 5-Adaption of Risk Management in Implementation Phase

Risk Analysis In Implementation Phase	Frequency	Percentage
Rarely	39	26%
Occasionally	36	24%
Sometimes	29	19.3%
Frequently	27	18%
Very Frequently	19	12.7%
Total	150	100

Source: Secondary Data

Chart 5-Adaption of Risk Management in Implementation Phase

Inference:

From the above table & chart, the inference that we have drawn is that the risk analysis in implementation phase is done rarely. 26% of the employees are accepting this statement.

TABLE 7-Minimization of Risk in Developing the Software

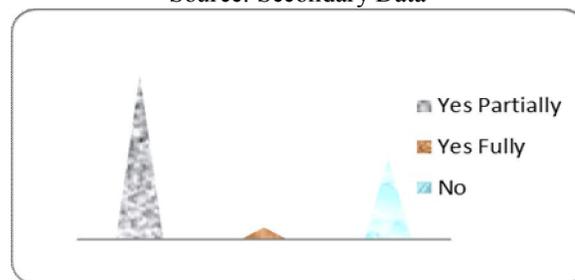| Minimization of Risk	Frequency	Percentage
Yes Partially	96	64%
Yes Fully	**7**	4.7%
No	47	31.3%
Total	150	100

Source: Secondary Data

Chart 6-Minimization of Risk in Developing the Software

Inference:

From the above table & chart the inference for the minimization of risk is that majority of the respondents say that the risk can be minimized to a certain extend. 64% of the respondents are accepting this statement.

5. CROSS TABULATION

Nature Of Software Project	Level Of Risk		
	High	Medium	Low
Programming Software	11	14*	6
System Software	11*	9	7
Application Software	12	14*	4
Inventory Software	5	6	9*
Utility Software	4	11*	6
Data Backup & Recovery Software	7	9*	5

Source: Secondary Data * Level of Risk Faced
*Highest Value

Inference:

From the above cross tabulation table, we can conclude that the development of System Software alone faces the highest risks whereas development of the Programming Software, Application Software, Utility Software and Data Backup & Recovery Software faces medium level of risk. Inventory software development project faces low level of risk.

TABLE 9-Nature of Software Project

Risk Factor	Nature Of Software Project	Level Of Risk			
		High	Medium	Low	NA
Software Requirement Risk	Programming Software	10*	7	8	6
	System Software	3	8	6	10*
	Application Software	7	6	9*	8
	Inventory Management Software	3	5	6*	6*
	Utility Software	6	3	7*	5
	Data Backup & Recovery Software	5	3	9*	4
Software Cost Risk	Programming Software	7	7	4	13*
Software Cost Risk Software Scheduling Risk	System Software	5	8	9*	5
	Application Software	7	7	9*	7
	Inventory Management Software	8*	5	5	2
	Utility Software	5	3	9*	4
	Data Backup & Recovery Software	7*	7*	3	4
	Programming Software	8	9*	7	7
Software Scheduling Risk Software Quality Risk	System Software	1*	6	5	5
	Application Software	9*	6	8	7
	Inventory Management Software	5	4	8*	3
	Utility Software	4	7*	4	6
	Data Backup & Recovery Software	8	6	5	2
	Programming Software	7	6	8	10*
Software Quality Risk Software Business Risk	System Software	5	7	4	11*
	Application Software	8	5	11*	6
	Inventory Management Software	3	5	10*	2
	Utility Software	8*	5	5	3
	Data Backup & Recovery Software	1	6	5	9*
	Programming Software	12*	7	4	8
Software Business Risk	System Software	7	7	9*	4
	Application Software	9	7	11*	3
	Inventory Management Software	6	8*	2	4
	Utility Software	6	7*	4	4
	Data Backup & Recovery Software	8*	6	7	0

Inference:

From the above cross tabulation table the following inferences are drawn:

- Software Requirement Risk brings high risk in developing programming software.
- Development of Inventory Management Software & Data Backup and Recovery Software is highly affected by software cost risk.
- Software Scheduling Risk is the highest risk causing factor for developing System Software project & Application Software.
- Developing a Utility Software is mainly retarded by the failure of quality factors.
- Failure s in managing the business risk is the main reason for the retardation of both the Programming Software & the Data Backup and Recovery Software.

6. FINDINGS

6.1 PERCENTAGE ANALYSIS:

20.7% of the employees are employed in developing programming software, 18% of the employees are in system software development, 20% of the employees are in application software development, 13.3% of the employees are employed with developing inventory management software, for utility software development 14% of the employees are appointed & remaining 14% employees are in data backup & recovery software development.

43.3% of the respondents say that medium level of risk is associated with their software development.

72.7% or the respondents are interested in following certain standards in risk management process.

Majority of the employees are interested in following risk management program in the software development project.

Majority of the respondents say 28% are initiating the management to follow risk management program in the business planning level of the software development cycle.

Majority of the respondents say 32.7% needs risk management program to be conducted frequently in the engineering level of the software development cycle.

Majority of the respondents say 26% are expressing to conduct risk management program rarely in the implementation phase of the software development life cycle.

6.2 CROSS TABULATION:

A) From the cross tabulation analysis between Nature of Software & the level of risk the following results have been drawn.

i) The development of System Software alone faces the highest risks whereas development of the Programming Software, Application

Software, Utility Software and Data Backup & Recovery Software faces medium level of risk. Inventory software development project faces low level of risk.

B) From the cross tabulation analysis between nature or the software & the risk factors, the following inferences have been drawn.

i) Software Requirement Risk brings high risk in developing programming software.

ii) Development of Inventory Management Software & Data Backup and Recovery Software is highly affected by software cost risk.

iii) Software Scheduling Risk is the highest risk causing factor for developing System Software project & Application Software.

iv) Developing a Utility Software is mainly retarded by the failure of quality factors.

v) Failure s in managing the business risk is the main reason for the retardation of both the Programming Software & the Data Backup and Recovery Software.

7. CONCLUSION

This article describes risks in software development. Every software development has the risk because software development is based on knowledge and new technologies, and the chances for success of a software development project are closely connected with successful risk addressing. As a result of that, we have closely investigated risks and risk impact areas in software development projects. With this paper, we propose a key element of modern software development practices to be software risk management. In order to achieve efficient risk management, we have proposed three risk management strategies suitable for different software development projects according to the amount of risk impact. It also have proposed a risk-based approach to development planning and risk management as an attempt to address and retire the highest impact risks as early as possible in the development process. The risk-based approach to software development should enable early risk addressing and conclusion when the expenses connected with risk materialization and project failure are small and insignificant. Strategies proposed in this work should enhance risk management on software development projects and increase project chances for success. From the global business perspective, the success of many enterprises

is becoming increasingly dependent on the success or failure of the software they build. It is not important if software is intended to be sold, internally used or to drive business transactions, the future of many enterprises is connected with the software they develop. Thus, risk management is not only a crucial development practice, but also a vital business practice.

Corresponding Author:
Full Time Ph.D Research Scholar, School of Management Studies, Anna University: Chennai 600025, Regional Centre, Coimbatore, Jothipuram Post, Coimbatore - 641 047.Tamil Nadu, INDIA.
Email: sathish.mailme@yahoo.co.in

REFERENCES

[1] Barki, H., Rivard, S., Talbot, J., 2001. An integrative contingency model of software project risk management. Journal of Management Information Systems 17 (4), 37–69.

[2] Bernstein, P.L., 1996. Against the Gods: The Remarkable Story of Risk. John Wiley & Sons, New York.

[3] Boehm, B.W., 1988. A spiral model of software development and enhancement. IEEE Computer 21 (5), 61–72.

[4] Boehm, B.W., 1989. Software Risk Management. Tutorial. IEEE Computer Society, Washington.

[5] Boehm, B.W., 1991. Software risk management: principles and practices. IEEE Software 8 (1), 32–41.

[6] Boehm, B.W., Ross, R., 1989. Theory-W software project management: principles and examples. IEEE Transactions on Software Engineering 15 (7), 902–916.

[7] Boehm, B., Turner, R., 2003. Using risk to balance agile and plan-driven methods. Computer 36 (6), 57–66.

[8] Boehm, B., Turner, R., 2004. Balancing Agility and Discipline: A Guide for the Perplexed. Addison-Wesley, Boston.

[9] Bourne, L.,2007. Avoiding the successful failure. In: PMI Global Congress, Asia Pacific, Hong Kong, 29–31 January.

[10] Brooks Jr., F.P., 1975. The Mythical Man Month: Essays on Software Engineering. Addison-Wesley, Reading.

[11] Charette, R.N., 1989. Software Engineering Risk Analysis and Management. McGraw-Hill, New York.

[12] Charette, R.N., 1996. The mechanics of managing IT risk. Journal of Information Technology 11 (4), 373–378.

[13] Charette, R.N., 2005. Why software fails? IEEE Spectrum 42 (9), 42–49.

[14] Cule, P., Schmidt, R., Lyytinen, K., Keil, M., 2000. Strategies for heading off project failure. Information Systems Management 17 (2), 65–73.

[15] Davis, G.B., 1982. Strategies for information requirements determination. IBM Systems Journal 21, 4–30.

[16] Camprieu, R., Desbiens, J., Feixue, Y., 2007. 'Cultural' differences in project risk perception: an empirical comparison of China and Canada. International Journal of Project Management 25 (7), 683–693.

[17] Rubenstein, D., 2007. Standish Group Report: There's Less Development Chaos Today. Software Development Times, 1 March, <http://www.sdtimes.com/ article/story-20070301-0.