



**THE EFFECTS OF ORGANIZATIONAL CULTURE
ON RISK MANAGEMENT
DURING SOFTWARE DEVELOPMENT**

by

Charu Verma

**A thesis submitted in the partial fulfillment of
the requirements for the degree of**

MSc in Information Technology Management

Faculty of Informatics

June 2009

Declaration

I Charu Verma, confirm that this research carried out for assessment is my original work and is completed in my own words. Works of other authors in any form (e.g. statements, methodology, tools, ideas, figures, text) are correctly acknowledged and a complete list of the references has been included at the end.

Signed:

Date:

Acknowledgements

To begin with thanking God for the ability he has given me, I take this great opportunity to express my respect to my dissertation supervisor Dr. Saad Amin for letting me choose a topic of my choice. He always provided me with the necessary assistance and support whenever I needed.

I am pleased to thank my elder sister, who has always guided me in right direction and encouraged in tough times, making me feel better and stronger.

I would like to thank all the interviewees, my colleagues and friends all across the world, who had been of great help by answering the survey.

I would like to express my deep gratitude to my parents and my spouse without whose love, support and care, it would have been difficult for me to complete this work. Very warm thanks to my mother Chander Kanta Verma, for every little thing she did which made me good enough to be able to be, what I am today.

Abstract

The inevitable challenge of time, schedule, cost and scope in software projects has been a matter of concern even for powerful names like Microsoft and IBM. No matter how well equipped an establishment is, these projects come entrapped in risks. However, dealing proactively with these risks could truly be rewarding. There are several books, methodologies and tools which attempt to deal with Software Project Risk Management; nevertheless there is always another side to solving a problem, which the research is going to explore. The research is centered on the subject of studying whether it is the methodology or the work culture or both, which is responsible for the success or the failure of a software project. The subject of "Cultural Issues Impacting Risk Management" during software project development has been explored by very few authors with a narrow coverage. The main aim of this study is to obtain an understanding of key cultural barriers to effective Software Project Risk Management. The study begins with identifying the software project risks and risk behavior during various Software Development Life Cycles, followed by studying the means of culture emergence in an organization and how it influences the software development process. Moreover, the initial studies are researched further by means of interviews, surveys and case studies. The outcome of which is used to develop a support structure to aid organizations identify and assess Software Project Risks, by making it an integral part of the organizational conduct.

Contents

1. Introduction	12
1.1 Importance of software Projects	13
1.2 Software Project - A Risky Business	13
1.3 Organization Culture and Software Projects.	13
1.4 Aims, Objectives and Key Questions.	13
1.5 Organization of Research	14
 2. Literature Review – I	 15
2.1 The Current Climate.	16
2.2 Defining Risk	16
2.3 Categories of Software Project Risks.	17
2.4 What is SDLC	18
2.5 Risk Behavior in Various SDLCs	18
2.5.1 Waterfall Model	18
2.5.2 V-Shaped Model	20
2.5.3 Incremental Model	21
2.5.4 Spiral Model	21
2.6 Risk management	22
2.6.1 Risk Identification	23
2.6.2 Risk Analysis	24
2.6.3 Risk Planning	25
2.6.4 Risk Monitor and Control	25
2.7 Place of Risk Management in an Organization	26
 3. Literature Review - II	 27
3.1 The Current Climate	28
3.2 Defining Organization Culture	28
3.2.1 How Does the Organization Culture Emerge?	30
3.3 Essential Elements of Risk Culture	30
3.3.1 Consistent Terminology and Risk Language	31
3.3.2 Tolerances, Thresholds, and Triggers	31
3.3.3 Mandatory Processes	32
3.3.4 Risk Reporting and Controls	32

3.4 Importance of Organizational Culture – IBM Vs Microsoft	32
3.5 Building Risk Management Culture – How to Begin?	33
3.5.1 Evaluating Organization Culture	33
3.5.2 Measuring Risk Value	33
3.5.3 Understanding the Key Value Attributes	34
3.5.4 Organizational Suitability of Risk Valuation Techniques	35
3.5.5 Organizational Approach to Risk Management	38
3.6 Successful Techniques of Software Development	38
3.6.1 Agile Method – Encourages Team Empowerment	39
3.6.2 Lean Software Development	40
3.7 Human Factor in the Domain of Risk Management Culture	41
4. Research Methodology	42
4.1 Research Framework	43
4.2 Case Studies	45
4.2.1 Progress	45
4.2.2 Problems Encountered	45
4.2.3 Accomplishment	45
4.3 Desk Research	46
4.3.1 Progress	46
4.3.2 Problems Encountered	46
4.3.3 Accomplishment	46
4.4 Interviews	47
4.4.1 Progress	47
4.4.2 Problems Encountered	47
4.4.3 Accomplishment	47
4.5 Survey	48
4.5.1 Progress	48
4.5.2 Problems Encountered	48
4.5.3 The Success and Significance Outcome	48
4.6 Research Assessment	48
4.6.1 Types of Research Techniques	48
4.6.3 Overall Limitations of Study	49

5. Research Findings	50
5.1 Findings	51
5.2 Desk Research	51
5.2.1 Software Project Risks Barriers and Problems within an Organization	52
5.2.2 Software Project Management Risk Management components	53
5.2.3 Organization Culture Components which Impact Risk Management	54
5.3 Interview Findings	55
5.4 Case Studies Findings	56
5.5 Survey Findings	58
5.5.1 Main Findings of the Survey	58
5.6 Overall Findings of the Study	59
 6. Discussion	 60
6.1 The issues identified	61
6.1.1 Management Judgment about Project Outcome	62
6.1.2 Lack of Clear Vision and Management Support	63
6.1.3 Lack of Line Sponsorship	63
6.1.4 Insufficient User Involvement and Commitment	64
6.1.5 Poor Productivity Due to Lack of “Sense of Urgency”	64
6.1.6 Poor Communication	64
6.1.7 Human – Just Hired or Nurtured?	64
6.1.8 Loose Monitoring and Controlling Procedures	65
6.1.9 Reactive Approach	65
6.1.10 Lack of willingness to do a lessons learned	65
6.2 Summary	66
 7. Solution Development	 67
7.1 The Framework Foundation	68
7.2 Software Risk Management Culture Development – Mind Mapping	70
7.3 Solution Development	71
7.3.1 Risk Management Guidelines – Superior Culture	72
7.4 How a Good SDLC Culture Manages Risks	73
7.4.1 Project Planning and Requirements Gathering - Identify Risks	73
7.4.2 Project Design – Assess and Analyze Risks	75

7.4.3 Project Development and Testing -- Handle risks	76
7.4.4 Project Implementation – Risk Treatment for Future Reference	77
7.5 Evaluation	77
8. Recommendations and Conclusion	78
8.1 Review & Recommendations	79
8.2 Conclusion and Future Work	80
9. References	82
A Risk Management Survey	86
B Survey Questionnaire Report	91
C Interview Questionnaire	94

List of Figures

2.1	Interdependencies of Risks	16
2.2(a)	Waterfall Development Lifecycle	18
2.2(b)	Coping with the Change	19
2.2(c)	Waterfall Risk Profile	19
2.3	V-model	20
2.4	Incremental Model	21
2.5	Spiral Model	21
2.6	Approaches to Risk Management	22
2.7	Risk Identification approach Summarized	23
2.8	Software Risk Analysis Chart by Change Management Tool Book	24
2.9	Place of Risk Management in an Organization	26
3.1	Organization Culture	28
3.2	Organization Culture Vs Society Culture	29
3.3	Seven Determinants of Culture in an Organization	30
3.4	Risk Terminology Summarized.	31
3.5	EVM of an Over Scheduled Project	34
3.6	IBM Software Brands Summarized	35
3.7	Project Value Measurement Techniques	35
3.8	Agile Software Development Principles Summarized	39
3.9	Lean Software Development Principles Summarized, one method of Agile	40
4.1	Methodologies Adopted	43
5.1	Bubble Diagrams Presentation	51
5.2	Software Risks – Barriers and Problems	52
5.3	Software Risk Management – Components	53
5.4	Organization Culture Components Impacting Risk Management	54
5.5	Survey Findings	58
6.1	Software Project Failure - Main Issues Identified	62
6.2	Culture vs. Structure	63
6.3	Various Factors Brainstormed for Solution Development	66

7.1	Software Risk Management Dependence on the Organization Culture	68
7.2	Approach to Solution Development	69
7.3	Aspects Considered while Developing the Framework	70
7.4	Superior Cultural Aspects of Risk Management Process	72
7.5	Requirements Categories Summarized	73
7.6	Risk Categories Summarized	74
7.7	Analyzing Which Risks to Respond	75

List of Tables

2.1	Summary of Tools Implemented by Microsoft and IBM	25
4.1	Description of Research Methods	44
4.2	Aim behind Choosing Case Studies	45
4.3	Desk Research Rationale	46
4.4	Overview of Interviews	47
5.1	Presentation Methods to Illustrate Findings	51
5.2	Interview Findings	55
5.3	Case Studies Findings	57

Chapter 1

Introduction

“The first step in the risk management process is to acknowledge the reality of risk. Denial is a common tactic that substitutes deliberate ignorance for thoughtful planning”.

-- Charles Tremper (n.d.)

1.1 Importance of Software Projects

Software is analogous to oxygen for two reasons; it is ubiquitous and it is necessary for survival. It enables to get us cash from an ATM, make a phone call, and drive our sedan. A classic mobile phone now contains almost 2 million lines of software code; by 2010 it is expected to have 10 times as many. General Motors Corporation estimates that by then its each car will have 100 million lines of code. Charette (2005) acknowledges that “Over recent years there has been a clear shift from a production based to a knowledge-based economy, and in this economy, IT is key”.

With statistics showing most companies now concede IT (Information Technology) to be their primary necessity, at the same time they see it as a risky business. The focus of this project is to provide an understanding of the key barriers to effective software project risk management and how to overcome them by making it a part of the work culture. Finally, a framework will be developed in order to aid effective Risk Management Culture within organizations.

1.2 Software Project – A Risky Business

According to the Nonprofit Risk Management Center (2008), a risk is any uncertainty about a future event that threatens an organization’s ability to accomplish its mission statement. The particular risk concerned may either be predictable and foreseeable or unpredictable and unforeseeable. While in the case of software development business, it generally involves unpredictable risks, unpredictable to an extent that some medical software failures had even caused deaths. Standish Group (1995) studied software projects and found out that 16% were successful (fully functional, on-time, and on-budget), 53% were unsuccessful (reduced functionality, late, over-budget) and 31% were failed (cancelled). More recent data (2008) suggests an improvement of mainly: 35%, 46%, and 19% respectively.

1.3 Organization Culture and Software Projects

While computers have revolutionized business, software development is embedded with risks of feature creep, complexity of high-maintenance technology, incomplete understanding of user requirements, time and cost overruns, also poor project management. However, the most important force driving above mentioned risks is the people involved. Any project is undertaken for the people and by the people. Many times it is their inefficiencies during software development life cycle which prevent a business from realizing full benefits from its investments in technology assets. For that reason, in spite of following a fool proof Project Management Strategy, software projects fail for various reasons.

For the very reason of variety of surprises inherent in software project environment, every software project is unique, irrespective of employing similar development processes and tools. As a result, software development projects have a gloomy track-record of cost and schedule overruns along with quality and usability problems, also Jiang and Klein (1999) state that “different types of risks affect budget, user satisfactions, and system performance”. Therefore, it becomes the basic necessity of the project team to prepare itself and get ready to welcome these surprises rather than moaning over them. It becomes the responsibility of the management, followed by all the stake holders to create a healthy environment which encourages smooth operations during the tough times of development.

1.4 Aim, Objectives and Key Questions

Aim

To obtain an understanding of key cultural barriers to successful software development, and propose a framework, which enables project team to build a healthy organizational culture in which they are able to recognize the risks for their proper treatment.

Objectives

- ⇒ To understand how Software Project Risk Assessment is currently conducted on business and enterprise-wide level.
- ⇒ To understand good business, project planning and management practices.
- ⇒ To research tools and views by experts within the Software Project risk field.
- ⇒ To identify a Risk management Culture rather than Risk management process which can be used effectively and effortlessly by organizations in order to avoid larger deviation from set targets.

Key questions

What are the current limitations of software development risk management?

How inducing a healthy culture can encourage effective software development risk management?

1.5 Organization of Research

The entire study develops through seven chapters. For “Culture” being an abstract topic, several diagrams have been especially created in MS Visio, in order to elucidate an analysis or verify a view point. Brief outline of each chapter is as follows: **Literature Review-I** chapter focuses on developing an understanding of the key areas within Software development risks, Software Development Life Cycles’ (SDLC) treatment of risks and risk management in general. **Literature Review-II** chapter centers on developing an understanding of key areas within Software project risk culture notably organization culture, measuring the project value and the risk value. The research conducted for the literature review outlined a number of key issues. In the **Research Methodology** chapter, the initial findings are researched further by seeking the views and opinions of experts. The main avenues of research were case studies, interviews and a survey conducted on software risk management with respect to organization culture within small and big setups. The subsequent four chapters are as follows:

Findings

In this chapter, case studies and interview findings are tabulated. For desk research and survey findings, bubble diagrams are developed in order to bring main issues into light and identify similar and contradictory views, leading to the identification of good and poor organization culture practices, effecting software risk management.

Discussion - Issues Identified Within Software Risk Management Culture

As reported by Verma (2009), some of the main issues found to be limiting software risk management were: lack of cross level communication, reluctance to accept the risk situation, a strong blame culture and lack of individual involvement. The discussion is concluded by gaining an understanding of how software risk management is currently conducted with in organizations. This is formulated using views and opinions obtained through interviews, surveys and case studies of three best IT organizations.

Development

The findings of the discussion helped in developing a framework for managing software risks by means of identifying good cultural practices. A diagram has been sketched in order to understand the effects of work culture on the risk management process during software development. Also, a mind map has been developed in order to identify a list of good practices. The main areas within this framework are: Identify, Assess, Analyze and Implement. This may sound like any other risk management process, however with a touch of very strong pre-planning, vision and team involvement in a healthy environment. The psychology aspect of stakeholders and environment aspect of the organization are taken into account.

Recommendations & Conclusion

A synopsis of the main problem is provided first. This is then followed by recommendations made for various cultural issues involved in limiting software risk management.

Chapter 2

Literature Review - I

Software Risks, Risk Types and Risk Behavior in Various SDLCs

“What is necessary is never a risk.”

-- *Cardinal De Retz, Memoires, 1718*

2.1 The Current Climate

During the 1970s and 80s, software development was not at the heart of many businesses, rather it was merely a support function. Today, companies striving for a competitive edge are obligated to develop commercial software in order to survive the competition. Moreover, the nature of these businesses and indefinite culture in the organizations decide software projects' degree of success.

Despite this surge in software development projects; the state of software project management for business critical software projects is still abysmal. In general nearly 40 percent of Software projects fail before completion. This view has been expressed in an article by IT Cortex (2001). These facts question this high failure percentage of software projects.

However, this study submits the basic problem area that is insufficiently addressed in today's "state of art" era: The effective risk management culture during the project lifecycle. Before discussing the risk management culture, it is important to first obtain an understanding of various types of risks encountered by a software development team and how these risks influence the project, during various Software Development Life Cycles (SDLCs).

2.2 Defining Risk

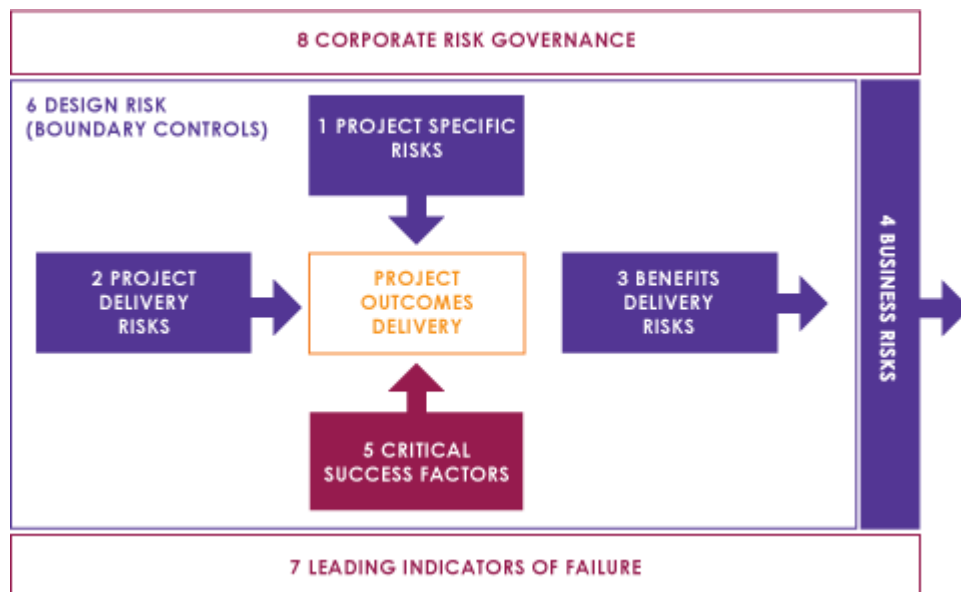


Figure 2.1: Interdependencies of Risks (Simms, 2007)

Etuzun (2008) states that, "risk is the possibility of bearing a loss. The loss could be anything from diminished quality of an end product to increased cost, missed deadlines, or project failure". He also proposes that risk and opportunity go hand in hand. Many development projects strive to enhance existing potential and accomplish something that has not been tried before. The opportunity for enhancement cannot be realized without taking risk. According to Scoy & Roger (1992), risk in itself is not bad; risk is essential to progress, and failure is often a key part of learning. But we must learn to balance the possible negative consequences of risk against the potential benefits of its associated opportunity. This idea is very well illustrated in the above figure, which shows that somewhere in the middle of risks, there exists some critical success factors which in reality, enable the successful completion of a project.

2.3 Categories of Software Project Risks:

Software Engineering Institute (1990) categorizes Software project risks as following:

2.3.1 Schedule Risks:

Project schedule slips when project tasks and schedule release risks are not addressed properly. Schedule risks primarily influence the project and finally the company economy and may lead to project failure. Schedules usually slip due to following reasons:

- ⇒ Resources like staff, systems, and skills of individuals not tracked properly.
- ⇒ Incorrect time estimation.
- ⇒ Failure to detect intricate functionalities and time required to build them.
- ⇒ Sudden projects scope extension.

2.3.2 Budget Risks:

These risks lead to increase in project budget, which may be caused by inaccurate budget estimation, cost overruns or project scope expansion.

2.3.3 Operational Risks:

These risks refer to losses caused by inappropriate process execution, system failure or some external forces. Causes of Operational risks could be following:

- ⇒ Failure to address priority conflicts.
- ⇒ Failure to resolve the responsibilities.
- ⇒ Insufficient or Inefficient resources.
- ⇒ Inappropriate subject training.
- ⇒ Poor or no planning for resources.
- ⇒ Lack of required communication in the team.

2.3.4 Technical risks:

Technical risks generally lead to failure of functionality and performance which may be a result of any of the following reasons.

- ⇒ Continuous changing requirements.
- ⇒ No advanced technology available or the existing technology is in initial stages.
- ⇒ Product is complex to implement.
- ⇒ Difficulties in project modules integration.

2.3.5 Programmatic Risks:

These are the external risks which are beyond the functional boundary. These external events could be one or all of the following:

- ⇒ Running out of fund.
- ⇒ Market development.
- ⇒ Changes in customer product policy and priority.
- ⇒ Changes in Government rule.

The study focuses on how to minimize the risks related to or caused by an organization's internal environment. Future is unpredictable, so are the external changes. However, dealing competently with the cultural (internal) aspects of an organization could lead to successful management of many software risks and result in comparatively successful project outcome. The next section studies risks' behavior during various stages of variety of SDLCs.

2.4 What is SDLC?

SDLC is an acronym for System Development Life Cycle which forms the basis for Software Project Management. SDLC is the process of developing information systems through investigation, analysis, design, implementation and maintenance. SDLC is also known as information systems development or application development. According to Folleth (2002), SDLC is a systems approach to problem solving and is made up of several phases, each comprised of multiple steps, which are as follows:

- ⇒ The software conception - identifies and defines a need for the new system.
- ⇒ A requirements analysis - analyzes the information needs of the end users.
- ⇒ The architectural design - creates a prototype for the design with the necessary requirements for the hardware, software, people and data resources.
- ⇒ Coding and debugging - develops and programs the final system.
- ⇒ System testing – assesses the system's actual performance in relation to anticipated functionality.

2.5 Risk Behavior in Various SDLCs

Almost all the software projects have associated risks depending upon their complexities. One of the key sources of risks is consistent changes in scope, which dazes the project team during the project's lifecycle. In its most common form, this risk is seen as changing user requirements. It is, however, not confined only to this area. There are various causes which could bring real risks to projects, for instance changes to the arrangement of the project team or in stakeholders, changes in the technology being used or changes to any external systems with which the new software must work. Dealing with these type of changes to a project is the key to reducing development risks and increasing the overall possibility of project success. In order to accomplish this, a deep understanding of various SDLCs is necessary, which will help in making the right decision while choosing a SDLC and dealing with risks.

2.5.1 Waterfall Model

The waterfall model is the mother of all the SDLCs. They all are further extensions of this model. The basic SDLC model involves series of steps from conception till completion. This model treats the project in series of steps – planning, requirements gathering, analysis, design, development and testing. In waterfall model, Requirements (gathering) completes before Analysis starts. Therefore, Design phase does not start until Analysis is complete and signed off. Figure 2.2(a) shows the typical lifecycle of a waterfall development process. Here, many times a Project Manager literally sees the project as a linear and predictable model as shown below. He/She may enjoy these assumptions, however later on might also struggle with risks which always come embedded in these assumptions.

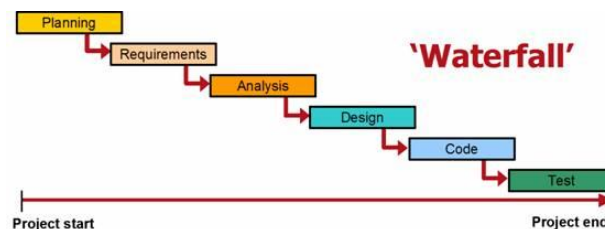


Figure 2.2(a): Waterfall Development Lifecycle (Murphy, 2000)

This model sometimes presents a false illusion of everything being alright. However, software development is naturally an unpredictable process. Change might inescapably pop up during the project lifecycle, and this is usually the result of feedback from users. As the figure 2.2(b) depicts, waterfall processes have a slow feedback cycle for the stage that really matters, which is the actual software development but not the signed off paperwork.

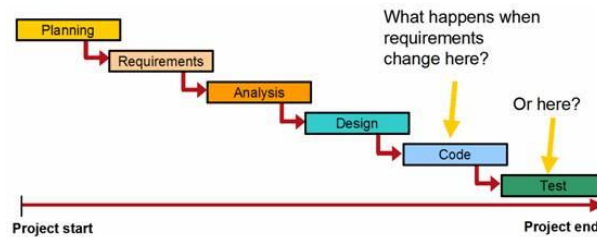


Figure 2.2(b): Coping with the Change (Ibid.)

As a result, changes might occur during the later stages of development or even during integration. Come to think of writing off a 100 man-hours project, after having already spent 80 man-hours of the budget for the small reason of inefficient understanding of user requirements. The risk behavior for waterfall processes, as shown in Figure 2.2(c) below, is inapt for a poorly defined software development.

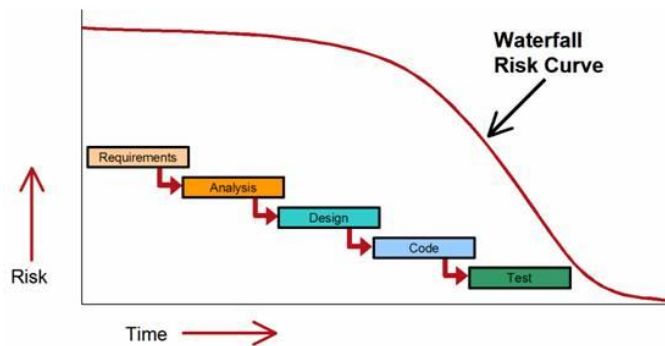


Figure 2.2(c): Waterfall Risk Profile (Ibid.)

The major problem with waterfall processes is; it relies on the assumption that progress of a software project can be predicted with reasonable accuracy from the outset and a reliable completion date can be derived from this. The very assumption of lifecycle being predictable ensures a high degree of risk within a project.

2.5.2 V-Shaped Model

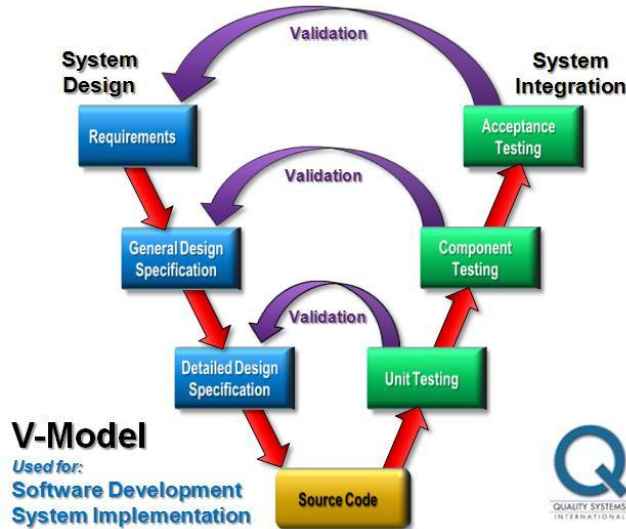


Figure 2.3: V-model (Lewallen, 2005)

V-Shaped life cycle is a series of pre-defined stages, much similar to waterfall model. Each phase has to be completed before the next phase begins. Testing is encouraged more in this model contrary to waterfall. The test plans are developed long before the development begins. The test plan focuses on matching up to the functionality specified in the requirements gathering phase.

The high-level design phase focuses on system architecture and design. An integration test plan is worked out in order to test the software system modules' capacity to work together. Whereas unit tests are designed in order to test the low-level components of the system. Coding takes place during the implementation stage, after which all the test plans are implemented and matched against their respective planned functionalities.

Therefore, this model involves adequate testing to take care of most of the functional risks, although it may run over-budget and over-schedule due to its very nature of planning for tight backup. However, it works well for medium sized projects in which requirements are complete and easily understood.

2.5.3 Incremental Model

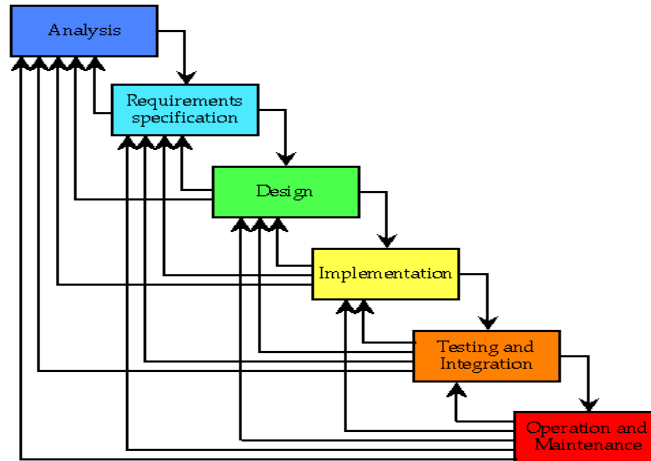


Figure 2.4: Incremental Model (Ibid)

An insightful approach to the waterfall model is incremental model, where numerous development cycles take place which makes it a “multi-waterfall” cycle. Cycles are divided up into smaller and more easily manageable iterations in which each iteration goes through the requirements, design, implementation and testing phases. A running version of software is completed at the end of first iteration and the final system is built on successive iterations build on the basis of first iteration.

It is more flexible with changing requirements; hence lesser chances of encountering scope change. However, challenges may occur concerned to system architecture because not all requirements are gathered at one go for the complete cycle.

2.5.4 Spiral Model

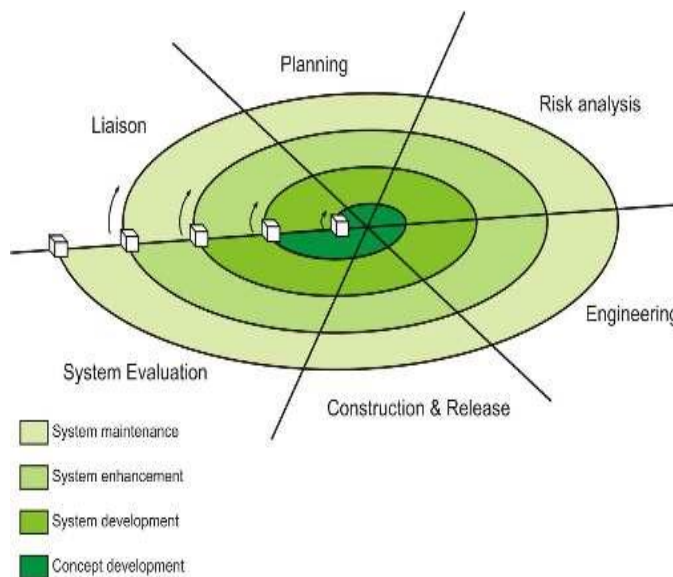


Figure 2.5: Spiral Model (Ibid)

Spiral model is similar to incremental model, as it places more emphasis on risk analysis. It has four stages, mainly Planning, Risk Analysis, Engineering and System Evaluation. A software project cyclically passes through these stages in iterations (called Spirals in this model). The foundation spiral, starting in the planning stage, followed by requirements gathering and risk assessment. Each succeeding spiral builds up on the foundation spiral.

Planning stage involves requirements gathering. In the next stage of risk analysis, risks are identified and responses are planned. A prototype is produced at the end of the risk analysis stage. Engineering stage involves software development, along with testing at the end. The evaluation stage enables the customer to evaluate the outcome of completed project till date before the project proceeds to the next spiral .In the spiral model, the angle symbolizes progress, and the radius symbolizes expenses incurred.

Therefore this model is apt for large and mission-critical projects; however it can prove to be a costly model to use. Seeing that the project success truly depends on risk management which makes it lesser suitable for smaller projects.

These are the four main SDLCs adopted by many software organizations. Moreover, there are various strategies and distinctive processes developed exclusively for Risk Management which will be discussed in the next chapter. Let us study “Risk Management” in general.

2.6 Risk Management

There are two perspectives of managing risk, firstly managing corporate risk which entails setting in place the necessary strategies and associated control measures and secondly, managing organizational risk which encourages the use of risk based procedures for making day to day business decisions. As for this study focuses more on the second perspective, thus an analysis of this area is provided. Mostly risk management approaches consist of five main areas or stages which are illustrated in the following figure. The only difference between the two is the way they implement it.

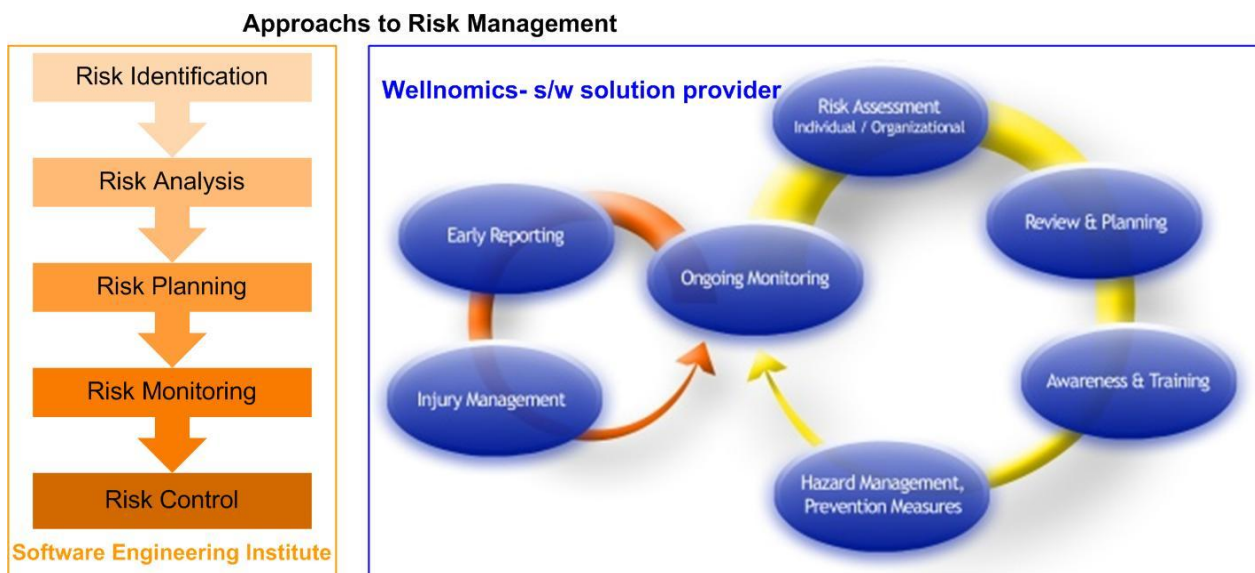


Figure 2.6: Different approaches to Risk Management by SEI & Wellnomics

This method provides a proactive approach, allowing the organization to identify and assess risks. This approach aids the project team in eliminating, transferring or minimizing the risk they are likely to encounter. Let us look at the risk management process in detail.

2.6.1 Risk Identification

As the title implies, this step involves identifying the possible risks the software project is expected to encounter in the future. In the case of identifying software risks, it is recommended that a project audit (identifying the key value attributes within the project) or a software threat audit (identifying the threats to the project) is conducted. Generally, the process is conducted as a brainstorming workshop involving key decision makers. According to NASA (2004) a reliable technique should be adopted in order to alleviate stress on the team which allows discovery of numerous risks. In this function, project management tools such as 'cause and effect' or 'fish bone' diagrams are suitable.

There are some software packages available in the market in order to carry out this function, such as Wellnomics Risk Management tool built by Wellnomics Pvt. Ltd. (n.d.) and Fault Tree Analysis by NASA. NASA is one such organization which uses multiple approaches depending upon the situation and complexity. In an article published in September 2004, the chief of Information System Division at NASA, Hennessy spoke about the importance of identifying risk to "comply with operating requirement including risk management". He describes the entire process, where organizational risks are considered equally important as process risks. The (NASA) risk identification process is summarized below.

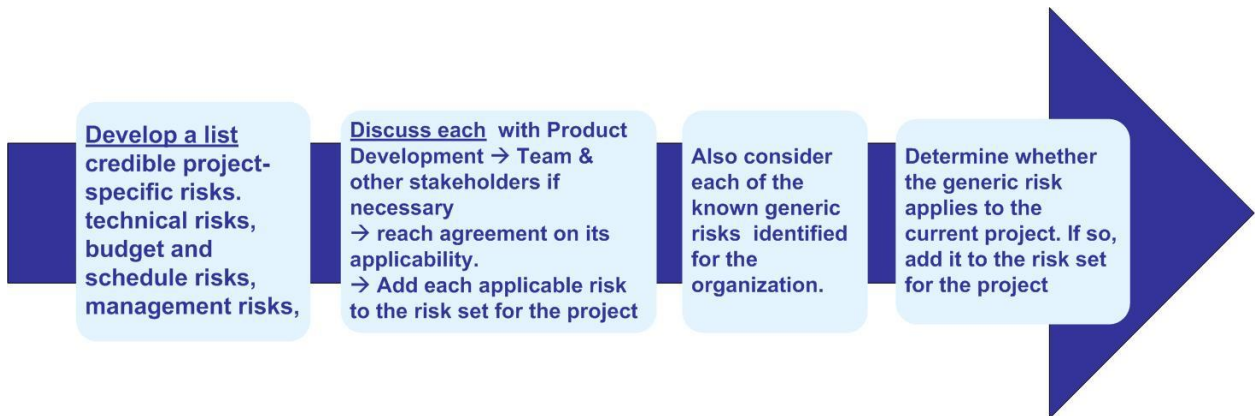


Figure 2.7: Risk Identification approach Summarized

2.6.2 Risk Analysis

This stage involves taking an in-depth study of every risk identified. According to IBM's corporate governance and risk management strategy (2007), one should study the full background of the identified risk by going beyond likelihood and impact and treating the background and source, then planning a response.

Change Source Pvt Ltd. (n.d.) in Johannesburg suggests a unique technique of measuring risks by preparing a template and asking all the participants during a workshop to individually complete the risk analysis by scoring seven risk criteria (Figure 2.9). This is the focus of this stage. The primary goal of this stage is to understand the impacts of various risks and develop a method of dealing with them if it were to occur. Questions such as how to avoid, minimize, control and remove the risks are addressed.

A good measure of assessing the magnitude of risks is, to plot them on a chart with frequency on one axis and impact on the other, as shown below.

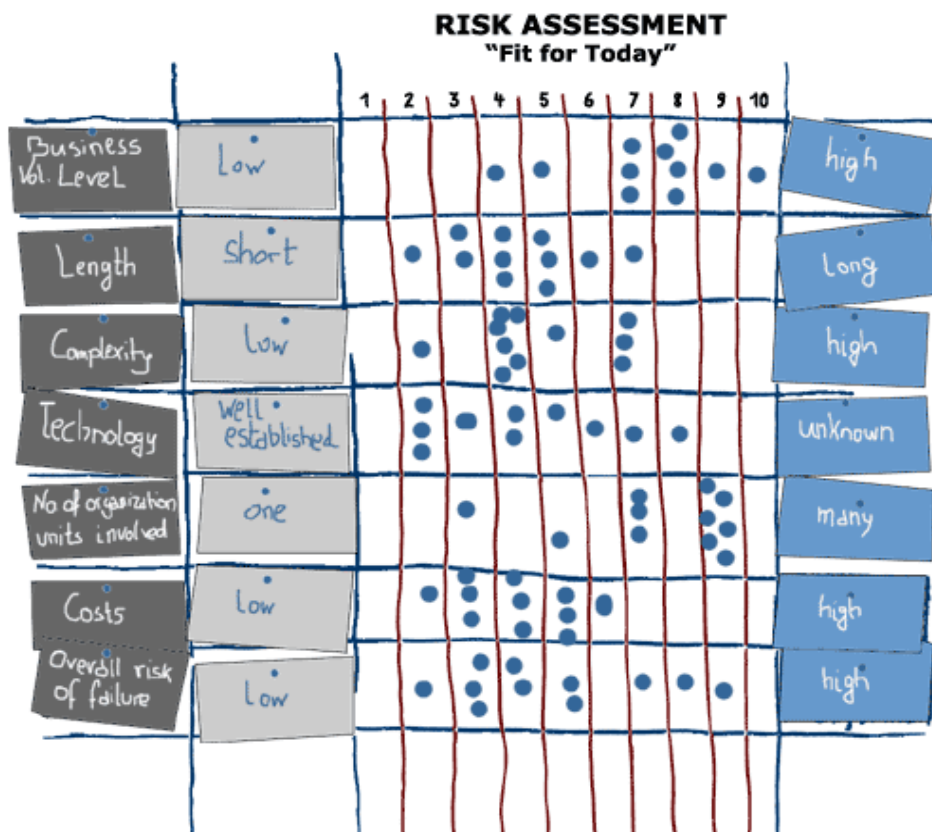


Figure 2.8: Software Risk Analysis Chart by Change Management Tool Book (Change Source Pvt. Ltd)

2.6.3 Risk Planning

According to Jones (2003), The Institute of Quality Assurance proposes five ways an organization can plan to deal with risks, these are:

2.6.3.1 Ignore it and hope that it will not happen. If the risk materializes, the plan is to face the consequences even if this proves beyond the resources of the organization, which may then cease to exist.

2.6.3.2 Share the risk with others - particularly clients. This is where the IT Company may decide to develop partnering arrangement to contract out the associated risks to the client. The company will make a commitment to the client and in return; the client will take some or all of the risks associated with late or non-compliant delivery.

2.6.3.3. Transfer the risk to a third party through insurance. Premiums for this approach are related to the insurance companies' assessment of the risk. Typically the organization protects itself for a given level of claim that theoretically can occur several times during the period covered

2.6.3.4. Reduce the risk by careful management of high-risk activities. This usually involves quality assurance, hazard analysis, FMEA (Failure Mode Effect Analysis) and other quality-related tools. The approach is not exclusive and can be used together with other approaches. The advantage is that possible counter-measures can be identified and deployed if the risk materializes.

2.6.3.5. Cease the risk-making activities. In the case of product liability this may mean stopping the project development where there is high level of known risk. Points to consider at this stage are as follows:

- Be careful not to avoid too many risks. As mentioned in point one, hoping the risk doesn't materialize could result in the organization ceasing to exist.
- Pay attention to root causes or series of risks.
- Don't delay the planning activities.
- Involve all risk owners.

2.6.4 Risk Monitor and Control

The final stage leads to a very important factor. It is vital to assign responsibility for the risk management plan. Once the steps have been established by the organization, the next move is to designate an individual or a team for developing and implementing the organizations' risk management program. While the team is principally responsible for the risk management plan, a successful program requires integration of risk management within all levels of the organization. "Operation staff and board members should assist the risk management committee in identifying risks and developing suitable loss control and intervention strategies". (Nonprofit Risk Management Center, 2007)

Summarized below are Risk Management methods employed by IBM and Microsoft.

Tools Employed For Risk Management	
• Risk Reassessment. • Risk audits. • Variance and trend analysis. • Reserve analysis • Status meetings Microsoft, 2002	• 24x7 security infrastructure monitoring • Maintain audit steady posture • Proactively detect and respond to tests • Monitor trends for emerging threats. IBM, 2007

Table 2.1: Summary of Tools Implemented by Microsoft and IBM

2.7 Place of Risk Management in an Organization

Current methods of risk management are very complex. The approach taken by companies has improved and they are now “taking a more proactive stance in developing and implementing systems to manage risk throughout their organizations” (Fraser, 2003). Nearly all large organizations have risk assessment or audit teams within their company, such as IBM and Microsoft. If one wishes to join their Risk management team (RMT), they have to go through a formal procedure of series of tests from three main departments – audit, human resource and information systems. This becomes essential for the reason that just in case the senior management does not admit and plan around the vision held by the RMT, the risk management is all but insignificant. Thankfully, the times are changing, “most executives would probably concur that risk management is an integral part of their jobs.” (Barton, 2003) In order to summarize the analysis of this chapter, the following figure has been drawn in order to demonstrate, how the risk management process aptly fits into the overall business environment and how important it is in order to reap project benefits.

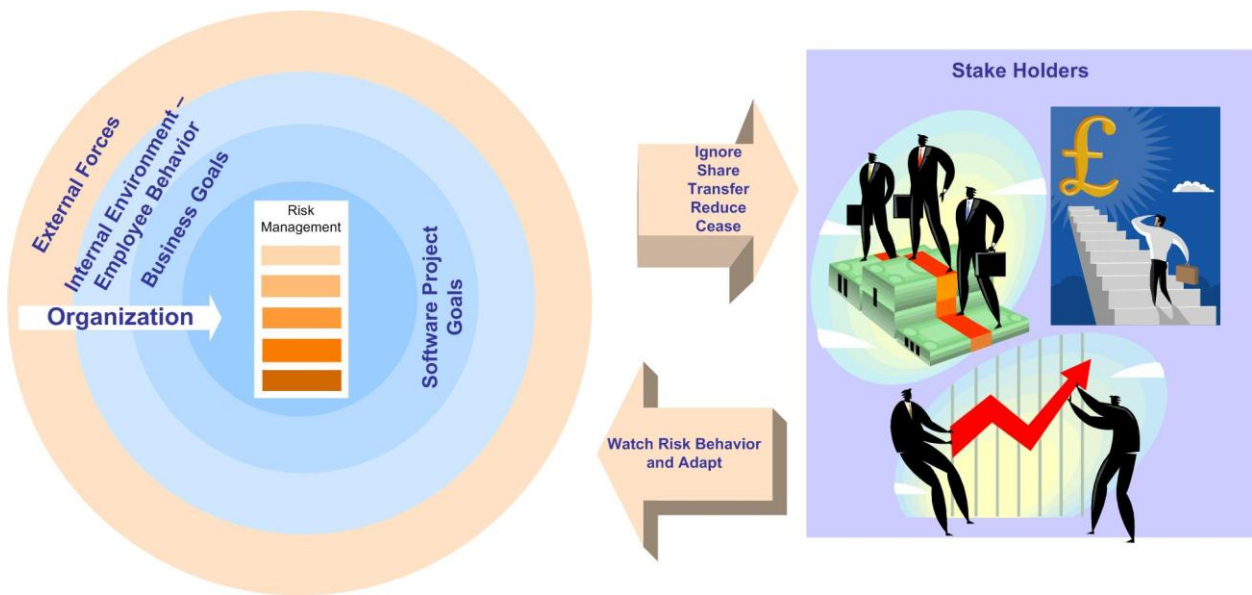


Figure 2.9: Place of Risk Management in an Organization

IT systems give life to modern business. Hence, development of new software applications and maintenance of existing systems are critical to productivity and profitability. Evolution of software technology over past two decades has gradually allowed more complicated business solutions to be developed, which helps companies to offer their customers interesting and comforting services and products. Despite the technologies being used, software development projects are haunted by same old evils and embody same old characteristics, which they had been suffering from more than a decade ago. Next chapter tries to unfold the causes behind these risk characteristics and discusses various good and poor cultural practices in an organization which influence the project outcome.

Chapter 3

Literature Review - II

Organization Culture – Components, Formation and Role in Software Risk Management.

**“Poorly designed and mismanaged cultures
are the silent killers of the corporate landscape.”**

-- Neil Postman and Charles Weingartner, 1969

3.1 The Current Climate

Most of the organizations presume risks as something which has only tragic effects on the organization; however this might seem a very narrow focused approach, as a risk can come in many guises. This is further supported by Debenhams and Hutchins (2003) who state that “the term risk management has a periphery which limits to nuclear, chemical, explosive, poison and occupational health and safety type risks and this perception sees only one facet of risk”. Supporting this view, this chapter explores that different facet of Software Project Failure. The study focuses on software project risk management from organization culture perspective. It studies the style of living and working in one’s second home and understanding how it can affect the project performance.

Looking at some of the facts cited by IEEE Spectrum (2008), an average company spends about 6 to 9 percent of its revenue on information technology, with those that are highly IT dependent, such as financial and telecommunications companies, spend more than 10 percent on it. In other words, one of the biggest commercial expenses apart from employee costs is incurred on IT applications. Major part of that expense goes into hardware and software upgrades, software license fees, and so on, but a significant share is kept aside, especially for new software ventures intended to create a better future for the organization and its customers.

When a project fails, it jeopardizes an organization's prospects. If the failure is large enough, it can steal the company's entire future. According to Goldstein (2002), in one stellar meltdown, a poorly implemented resource planning system led FoxMeyer Drug Co. a \$5 billion wholesale drug distribution company in Carrollton, Texas, to plummet into bankruptcy in 1996. The way pilots never intend to crash, software developers do not aspire to fail. After a commercial plane crash, investigators try to examine many factors, such as weather, maintenance log, the pilot's temperament and training, and ambience factors within the airline. Similarly, an organization needs to look at the weather of the set up, which implies business environment, previous performance records, project team disposition and training, and culture within the team and the organization, to get hold of the roots of software failures.

In the previous chapter, the discussion was carried out on the subject of software project management and risk management in general. Now let us understand how culture can influence these two factors. In order to study these cultural factors, it is important to understand the meaning, emergence and significance of culture in an organization.

3.2 Defining Organization Culture

Earlier research (Schein, 1992) demonstrated that an organizational culture is defined as a pattern of shared basic assumptions that the group learned as it solved its problems of external adaptation and internal integration, that has worked well enough to be considered valid and, therefore, to be taught to new members as a correct way to perceive, think and feel in relation to those problems. According the Schein, organizational culture is the learned result of group experiences, and it is to a large extent unconscious. Schein considers culture to be a three-layer phenomenon as shown below.

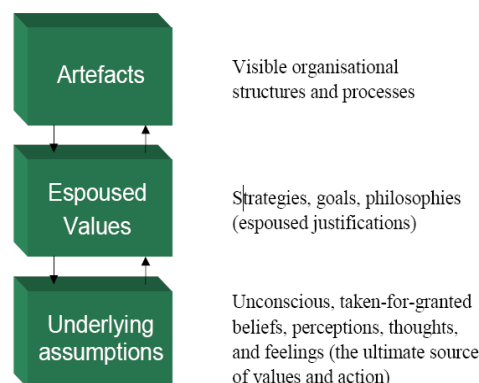


Figure 3.1: Organization Culture (Schein, 1992)

3.2.1 How does the Organization Culture Emerge?

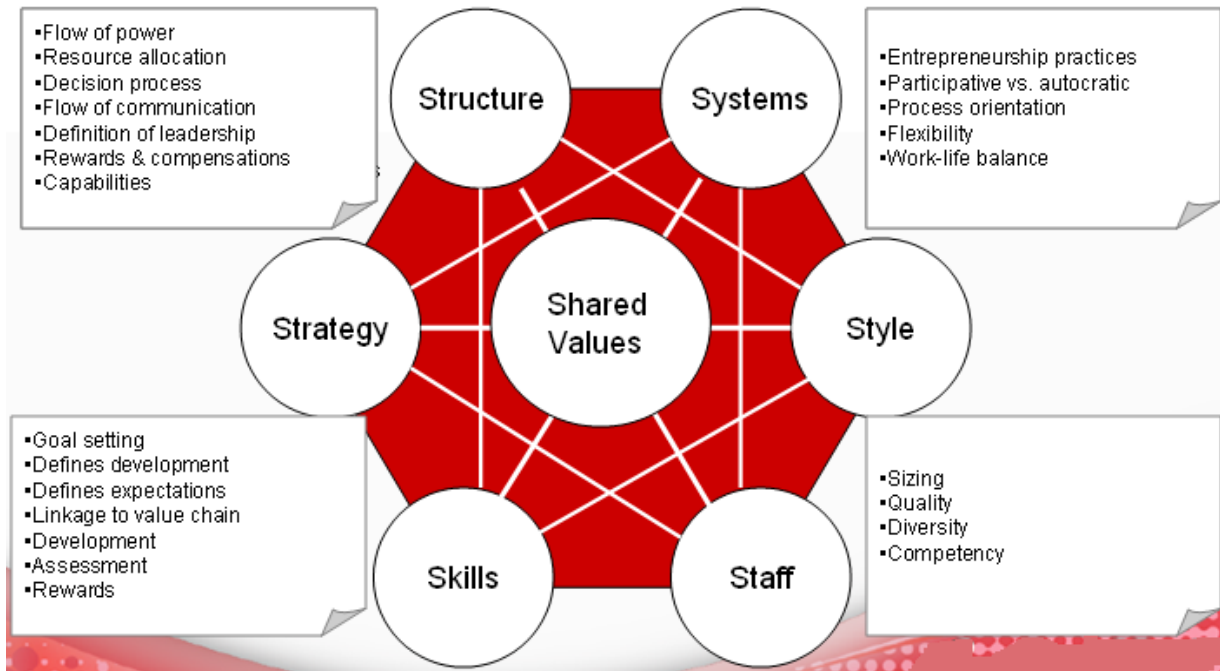


Figure 3.3: Seven Determinants of Culture in an Organization (Malia, 2006)

Malia (2006) in his speech says that “a successful organization in a given paradigm is always in a state of cultural equilibrium with the 7 determinants of culture (Figure 3.3). When paradigms shift, everything is reduced to zero and a balanced state of culture becomes imbalanced”. Through diagnosis, we need to test each vertical ‘S’ (shared values, systems, structure, skills, style, staff, strategy) and its impact on the behaviors required for the new culture by performing interventions accordingly to arrive at the new equilibrium.

In order for an organization to build a healthy culture, it does not have to train the staff in above seven determinants; rather it essentially needs to implement a flourishing environment where its employees are able to perform for the benefit of the project as well as their personal development.

3.3 Essential Elements of Risk Culture

According to Pritchard (2007), Senior Consultant, Cutter Consortium, to know what is missing from an organization's risk culture, one must first determine which elements are required. If, for example, an organization has the financial resources to endure nearly any risk, then setting up cost thresholds should be the decisive factor. One should recognize which elements are necessary. An organizations' risk culture is composed of consistent terminology, thresholds, triggers, mandatory practices, and controls. Within these elements, a unique set or practices determines the treatment of these elements.

3.3.1 Consistent Terminology and Risk Language

Risk culture flourishes on clear definitions of risk terminology. In building a risk culture, it is critical to share the glossary with all decision makers and reach common terms of usage agreement by defining them in a way that makes sense to all and is simple to understand. This will avoid differences and deviation while implementing risk mitigation plans. Some of the key terms as described by Pandian (2007) that need definition for clear understanding and usage, are summarized in the following figure:



Figure 3.4: Risk Terminology summarized

Pandian (2007) encourages that each organization should publish its own definitions of these terms and make them known to all stakeholders. To check the existence of a risk language within an organization, the test has to be full of similar terms, involving questions like: Can staff members define "risk" consistently? Can they explain the organization's risk strategies? Can they define risk management procedure in consistent terms? Can they tell the difference between the probability and impact of these risks? If not, a common risk language may not exist. Simple terminology might carry different meanings for different individuals in the same set-up in the absence of a risk culture.

Whenever a risk culture becomes an organizational necessity, developing a common language is one of the priorities so that those who serve the organization can talk about it smartly. Typical examples can be found in US Government Accountability Office which elucidates day-to-day terms such as "assessment" and "hazardous" that can have different meanings when risk is at the center of the discussion.

3.3.2 Tolerances, Thresholds, and Triggers

Tolerances refer to limits of organizational behavior. This is the boundary an organization jointly will not cross. For instance, a big company not accepting very small projects.

Thresholds are probably the most universal and easily developed element of a risk culture. It decides the extent to which an organization will go during odd times. Organizations generally draw these limits on costs, schedules, employee conduct, cultural knowledge, community participation, and a herd of other concerns. Strangely enough, these thresholds are rarely announced, they are instead passed on from one employee to another as part of an organization's inherent tradition. New members of an organization learn culture of risk handling by virtue of unpleasant experiences. On encountering a trouble and being denounced for it, new employees are introduced into the culture. Absence of documented thresholds may bring in two concerns: difficulty in communication about them and lack of consistency in values inherent in the form of thresholds. For instance, maximum amount a company is ready to invest in a project.

Triggers are the caution signs that point to a threshold about to be violated. For instance, project cost during development is going to reach the threshold amount, which was set while planning the functionalities.

3.3.3 Mandatory Processes

Mandatory process refer to the “must do” list of an organization. It is important for some critical situations, where individual style of dealing may not be enough. They also make sure that communication takes place on a regular basis among involved parties in case of experiencing any risks. Good example can be – a developer producing a daily report of his completed task list or looking at similar previous projects before starting a new one, in order to find similarities and deal accordingly with project complexity.

3.3.4 Risk Reporting and Controls

In good practice risk organizations, internal risk management practice many times involves measuring the performance. These practices being in place ensure the existence of a consistent behavior, for instance providing management with timely updates and documentation of potential problems. On the contrary, a backup plan of risk responses will act in the absence of these practices.

Risk controls act as "thermostats" of the risk management procedure, which can be set for problem detection and forgotten, provided there are some pre-existing thresholds in the process. This helps in getting used to a consistent practice of “how to deal with risks”. In an organization where risk controls run successfully, staff is capable of defining risks effectively and describing organizational risk strategies efficiently and it can also manage to handle risks efficiently.

3.4 Importance of Organizational Culture – IBM Vs Microsoft

In order to clear the mist of doubts regarding the importance of organization culture in a software development project, let us discuss the cultures of two superpowers of IT world, which are none other than IBM and Microsoft. One of the great all time examples discussed in almost every software risk discussion. In this example, IBM did not even necessarily have a horrible organizational culture; however it got entrapped for several reasons and became stagnant for a while.

This example comes from the late 1980s when Microsoft was doing quite well and making tens of millions but nothing as compared to what they are today. That time IBM had the largest market share with 80% mainframe market. Also IBM spent large amount of money and time in building a system that was supposed to “take over everything”. The OS/2. Everyone had great expectations and looked OS/2 the only thing in the software world, but it could not happen for the very reason, that Microsoft did what IBM was expected to do. Microsoft focused more on building a “light memory” system unlike IBM's OS/2 which failed to meet the memory requirements.

This happened for the only reason that IBM was engrossed in keeping track of number of coding lines finished and ignored the system performance aspect. This was IBM's over confidence. Meanwhile, Microsoft's entire organizational culture did not focus on bureaucracy, rather on getting things done. Microsoft took advantage by building a product meant for customers, but not for internal specification. As a result, they completely overtook the computer market. IBM had a series of setbacks which affected their market share and they had to hire a complete outsider to re-establish the company.

This is an example of falling into the trap of bureaucracy and becoming overconfident and loosing all what a company has against another company who "had no business competing with IBM" according to most sources. Though IBM today is one of the sought after names in the IT world, however it carries this painful experience of rising. (Adapted from Organization Culture^{101♦}, 2007)

This presents a wonderful example of how one can become “the ruler” from an “underdog” by just keeping eyes and mind open to its surroundings.

♦ A website which discusses only on various cultural aspects of an organization.

3.5 Building Risk Management Culture – How to Begin?

Risk management can bring along many reward-and-recognition ideas. Organizations commonly acknowledge the performance of their employee "stars" who do wonders with their extraordinary endeavors to maintain a client relationship or repair a sinking program. By contrast, individuals who simply prevent approaching negative events (or who work in a way that management intervention isn't necessary) are rarely acknowledged. The way firefighters are awarded but fire preventing experts are ignored. In order to come up with a healthy organizational culture, the rewards and recognitions must be defined and established along with terminology and processes, as discussed in previous section. They must be nurtured, and preserved. Following are a few fundamentals which play a great role towards building a healthy risk management culture.

3.5.1 Evaluating Organization Culture

Before embarking on building a risk culture, the existing risk environment needs to be evaluated. In order to evaluate that environment, the questions to be asked are relatively simple to answer. Pritchard (2007) lists some of these questions as following:

When a team member identifies a significant potential problem, does he or she know what to do next?

When senior management identifies a significant potential problem, how is it dealt with?

Who determines what constitutes a "significant" event?

Who determines which resolution approach will be applied? How are these approaches implemented?

As studied by Pritchard, for organizations without a risk culture, the answers to above questions would be circumstantial. If the response to three or more of these questions is 'It depends on a case-by-case basis', the organization has no existing risk culture. It does not mean that it does not have a risk management process, but it shows that the management is inconsistent about its principles.

This is also indicative of the fact that many projects may be running on the shoulders of one single person, the one who defines or identifies the risks. The simple way to study a culture of an organization is to question the staff on various levels, which has been carried out in the study with the help of a survey and an interview questionnaire. It finds out the work culture of an organization and how effectively it handles risks.

3.5.2 Measuring Risk Value

Many software organizations do not consider indicators of danger on software development projects seriously as they arise. This might be due to poor project management maturity and living with the belief that project management is just a misuse of time and money. Another reason is Project Managers are reluctant to report any trouble during the project and assume that they will manage when time arrives.

A universal oversight in software project management believes that an over scheduled project during initial phases of implementation will still be able to catch up in the later stages and finish on time (personal experience). This is practically never the case. Ferle (n.d.) cites that DOD conducted a study which observed historical data on almost 700 projects since 1977, it has been confirmed that when a project is at least 15% complete, any acquired overruns will persevere until the end of the project and may increase towards its completion, they will unquestionably not reduce. The explanation of this finding says that if the project team underestimates the initial stages of the project it is very liable that it underestimates the entire project.

The figure below shows an example of a real life project, closely similar to the one, of which I had been a part. It was consistently running later from the beginning and was adding more delay as it advanced. In spite of the noticeable delays encountered in the initial weeks, the management did not turn serious in order to do something about it. This resulted in a delay of 8 months.

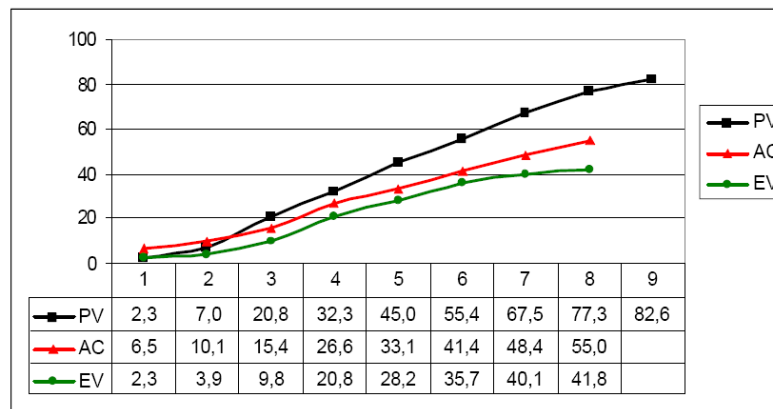


Figure 3.5: EVM of an Over Scheduled Project (Ferre, n.d.)

There are a variety of reasons why a company should measure the value of its software Project, the main reasons being are as follows:

- ⇒ In order to judge how much room for improvement there is.
- ⇒ To assess the effects of actions the company is currently taking. (e.g. using new framework, new hardware, and so on.)
- ⇒ To assess the potential risk they could face and the effect they could have on the company.
- ⇒ To inform your strategy going forth.
- ⇒ To assess its value as a company asset. (say IT project credibility)

3.5.3 Understanding the Key Value Attributes

The notion of "value" has a history that predates the software business by several millennia. In the Athenian Academy, in the fourth century B.C., Aristotle (384-322) realistically argued that the value of an object was based on the need for it. Similarly, "a software product isn't just an executable program; it can also refer to a software service, system, or process. Value is expressed as a price and is measured by the revenues that flow to the producer and consumers of the software over its lifecycle". (McKenna, 2005)

IBM (2006) rational has always been a foremost supporter of the practice of software economics, the main strength of which is in estimating project cost. The following equation, taken from COCOMO II, is well known throughout much of the rational community:

$$\text{Effort} = (\text{Personnel}) (\text{Environment}) (\text{Quality}) (\text{SizeProcess}^*)$$

Where, "*" reflects process effectiveness.

Above equation captures some key factors where Effort refers to person-months required to complete the project, Personnel implies factors considering the abilities of the team, Environmental factors consider tools and techniques, Quality takes care of considering the required product quality, Size refers to the number of human-generated source instructions building the end product and finally Process is the formula based on the effectiveness of the process used to produce the end product.

Software project valuation tools and methods offered by IBM cannot create value by themselves, but they can help stakeholders understand which projects promise to create profitable value. With this view, project managers and business leaders can work with their projects with more elasticity and compliance, which will add value to the overall software venture.

3.5.4 Organizational Suitability of Risk valuation techniques

IBM (2008) published a book, which talked about software success stories of 100 different established companies with up to 1,000 employees who have implemented solutions from the five IBM software brands mainly Information Management, Lotus, Rational, Tivoli and WebSphere. The main ideas behind these softwares are compiled below.



Figure 3.6: IBM Software Brands Summarized

The way IBM tools enabled successful implementation of over 100 projects, there are various methods available to foresee risks and inculcate risk management in the day to day plans of software development. However, the trickiest part here is; the team should be able to figure out the best way of identifying risks followed by an ideal way of dealing with them. This entirely depends on the competence of an organization and the enormity caused by not being able to identify risks correctly. Following is a far-fetched compilation of most popular techniques of evaluating risks depending upon the organization type project nature, because evaluating project's financial worth automatically leads to the evaluation of imminent risks.

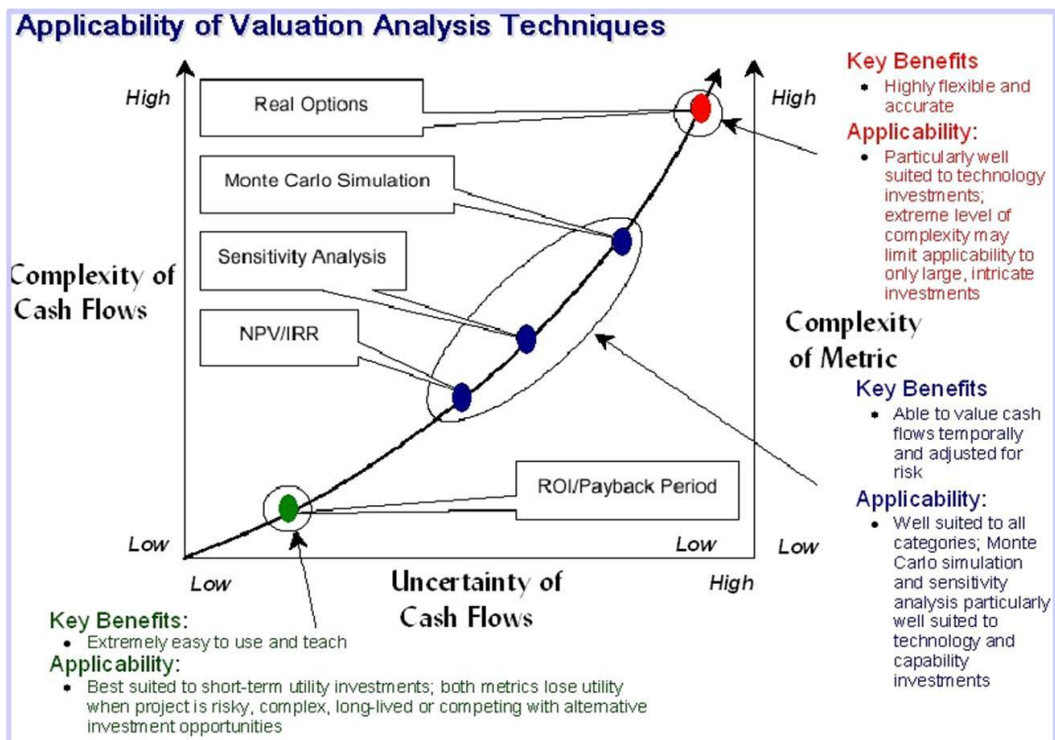


Figure 3.7: Project Value Measurement Techniques (Burchfield, 2002)

The first set of valuation techniques are quite well known:

Return on Investment (ROI)
Net Present Value (NPV)
Internal Rate of Return (IRR)

3.5.4.1 ROI or Payback Period

Prospective project value can be easily calculated in terms of ROI which is comparatively a straightforward investment metric that finds out how much time is required (the payback period) to recover the original investment. It can also be calculated in percentage -- ROI%. ROI is calculated as:

$$\text{ROI} = \frac{\text{Cost of Project}}{\text{Annual Cash Inflows}}$$

The best investment is with the shortest payback period. There are three main problems with the ROI/payback period method. Firstly, it skips the benefits that may occur after the payback period, and so does not measure long term profitability. Secondly it ignores the time value of money -- the discount rate. Thirdly, it does not consider the risk, when risk of failure in project work of any kind may exceed 50 percent. For these reasons, other methods of capital budgeting like NPV, IRR are generally preferred. However the "simple" ROI should not be discounted. It is widely used and, with a tendency to ask for a shorter payback period for software investments, it does have value and currency.

3.5.4.2 NPV (Net Present Value)

NPV is a tool used in capital budgeting in which the present value of cash inflows is subtracted from the present value of cash outflows. It measures the profitability of a project by comparing the value of a dollar today to the value of that same dollar in the future, taking inflation and profits into account. NPV analysis is responsive to the consistency of future cash inflows that an investment or software venture will yield. Here is the formula for NPV calculation:

$$\text{NPV} = \sum_{t=1}^T \frac{C_t}{(1+r)^t} - C_0$$

C = Cash | T = time (no. of years) | r = rate in %

If the NPV of a potential project is positive, it should be accepted. However, if NPV is negative, the project should perhaps be rejected for the cash flows being negative.

3.5.4.3 IRR (Internal Rate of Return)

This measure is often used in capital budgeting. IRR is the interest rate that makes net present value of all cash flow equal to zero. In order to calculate IRR, NPV is assumed to be zero and the IRR necessary to produce this project is computed. This internal rate of return (IRR) for the project is then compared to a minimum required rate of return for projects with similar risk. If the IRR for the project is greater than the minimum required rate of return, the project has positive net economic benefit for the company.

The second set of techniques is usually applied when a software product is developed or enhanced to fulfill a perceived marketplace need:

**Sensitivity Analysis
Monte Carlo Simulation**

Before explaining the next two valuation techniques, let us understand what a *value model* is. A value model for a software product can be seen as a formula or an algorithm, which helps in estimating the number of licenses that can be sold at a particular price over the product lifecycle. It deals with uncertainty related to value of market for the product and the market share the product can capture. These estimates are based on best, worst and most likely value scenarios. Not going deep into the value model, let us look briefly into the above two techniques.

3.5.4.4 Sensitivity Analysis

A technique that can determine which ambiguities in the inputs to a value model will produce the greatest impacts on the output, which implies the software product value. Say small change in technology growth rate results in reasonably large changes in the value, then this particular input needs to be measured accurately and tracked closely as the outcome is apparently "sensitive" to that particular input.

3.5.4.5 Monte Carlo Simulation

A Monte Carlo Simulation involves the use of random numbers and probability to find solutions to complex problems. The term was first coined by Ulam and Metropolis (1949) in reference to games of chance, a popular attraction in Monte Carlo, in the Kingdom of Monaco.

If enough data is available, and the model is realistic, the final result of the simulation is prediction of project value (often expressed as NPV) along with some measure of discrepancy (one standard deviation of the NPV) that expresses the risk of the project. The estimate of risk is calculated from the distribution (curve) of value (NPV) estimates that are generated by the simulation.

The third technique, Real Options, I will examine in the light of IBM Rational software development principles and practices, which come under the general heading of "options analysis."

3.5.4.6 Real Options

"Real Options" is a term coined by Myers in 1977 and refers to the application of options pricing theory to the valuation of non-financial or "real" investments.

Software development projects are pricey, precarious, and usually entangle the sponsoring party with uncertainty in the commercial lifecycle. In general, a manager can gain by waiting as long as possible before entrusting funds to a project or before locking in to a set of features. In real terms, delaying commitment to a project can reap double benefit. As it can protect the pool of investment for the right project and hold the capital flow until it finds the best feature set.

On the other hand, delay may invite the risk of a missed opportunity. Balancing these risks with benefits is the key to a good culture. Today, software engineering projects based on options models are gaining better reception. For example, Boehm and Port refer to "design for change" and the "information hiding" approach. (McKenna, 2005)

3.5.5 Organizational Approach to Risk Management– Proactive and Reactive

3.5.5.1 Proactive Software Risk Management

Proactive risk management means; the project team has a visible, transparent, measurable, and repeatable process for managing risks using the tools and techniques as discussed in previous sections. This approach to risk management emphasizes creating an environment in which the team *proactively* examines what can go wrong, on an ongoing basis, and then makes proactive choices about which risks need to be addressed, and properly addresses them.

The team will carry risks forward and deal with them until the risk impact, or probability is reduced to zero, or until the risk probability has become 100 percent or has occurred, which means that there is no longer the possibility of loss but now the guarantee of loss. Handling these issues involves minimizing the amount of that loss. One of the best examples of this approach is Lean software development, which will be discussed further in detail.

3.5.5.2 Reactive Software Risk Management

By contrast, in this traditional approach to project management there is hardly any usage of the term “risk management”. Here the team deals with problems generic to all software projects systemically as and when they arise. Team assesses risks only once during initial project planning. Also, identifying and addressing major risks are rarely explicitly reviewed again. This approach produces initial plans for the risks visible at the project initiation, but does not help the project team in responding to the changes it might encounter throughout the project lifecycle. A good example of this approach is IBM’s OS2 failure, which has been discussed in the previous section.

3.6 Successful Techniques of Software Development – A Closer Look

Having sufficiently discussed about software failures and their reasons, SDLCs, and culture issues, it is also necessary to learn about a few good work practices, which still keeps project team’s spirits high as there are many examples of successful projects as well. A technique Toyota started in the 1980s to transform the automobile industry called the approach of “Lean manufacturing”. Later on Poppendieck (2003) transferred principles and approach from the manufacturing to the software development environment.

Lean movement is a new development method that tries to detect and eliminate all the loop-holes and disabilities of old reactive approaches like Waterfall. Lean places main focus on people and communication which involves respecting the people who develop software and providing them with freedom of approachability, which encourages better chances of quality product delivery and better customer satisfaction. 3Q Solutions is one of those companies which develops wealth management systems and uses Agile methods extensively. Three organizations (in case studies) closely studied with respect to culture; IBM, Microsoft and Infosys also adopt many ideas from Agile methodology.

According to Agile 2008 survey results 61% of developers think that their organizations are implementing agile methods. Agile success rates are 82% for co-located teams, 72% for near located (people in different cubes, on different floors, working from home) and 60% for significantly distributed (planes would be involved to get people together). Lean Software Development consequently gave birth to Agile Software Development methods and its main branches like **Scrum** or **Crystal Clear** or **XP**. Agile is another word for Lean or Lightweight. Next sections summarize Agile technique in general and one Agile methodology (Lean) developed by Poppendiecks’ with the help of a figure.

3.6.1 Agile Methods – Encourages Team Empowerment

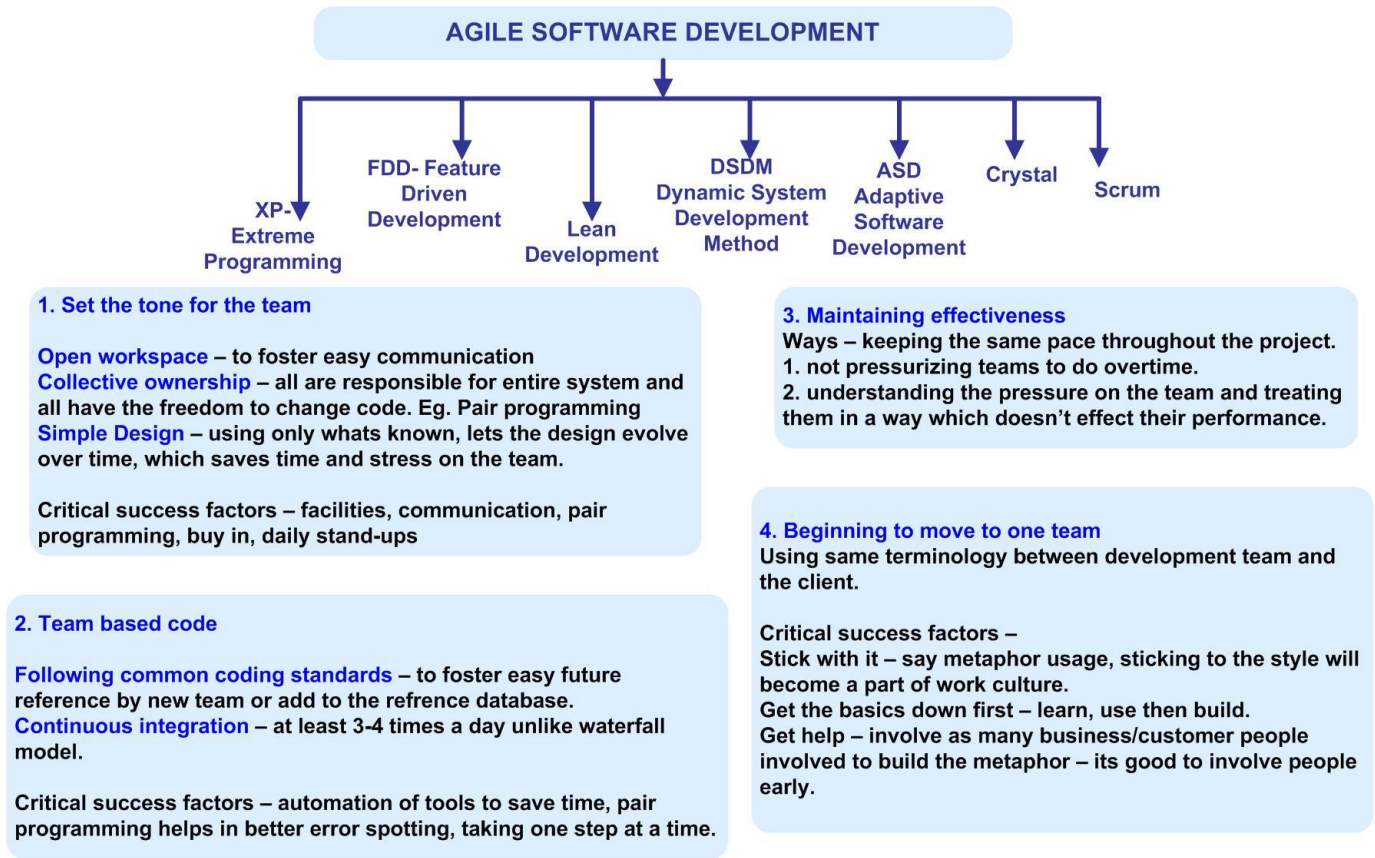


Figure 3.8: Agile Software Development Principles Summarized

Here the developer works on a tight feedback loop. Group practices are key to Agile, as they show how the team can work together more efficiently and make technical decisions together. Swan (2005) looks at this evolution of the team in four iterations, mainly setting the tone, team based code, increasing and maintaining efficiency and the first moves towards a "one team" approach, which also involves those outside the immediate development team. While many processes advocate teamwork, Agile embeds the practices which encourages teamwork and brings teamwork in daily chores by employing methods like Xtreme programming and Scrum. Using agile makes following possible: Delivery of time-boxed increments; Client can release the software anytime; Client can add, delete, and reprioritize features anytime; Allows schedule commitment because of consistent customer feedback and allows stopping anytime but still using what has been built.

3.6.2 Lean Software Development – Promotes Team Empowerment and Waste Avoidance

One of the agile tools discussed in detail – The 7 Lean principles

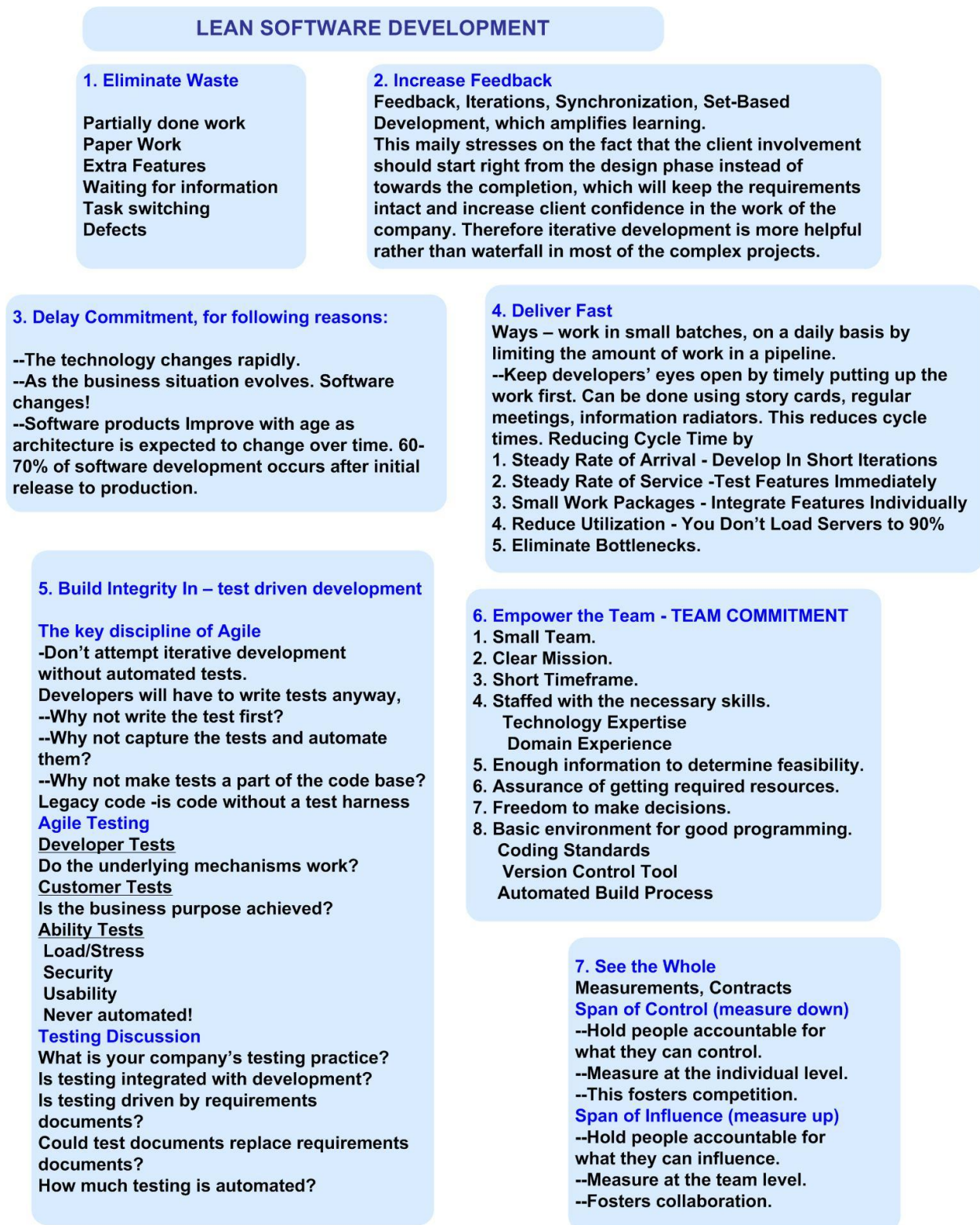


Figure 3.9: Lean Software Development Principles Summarized, one methods of Agile

3.7 Human Factor in the Domain of Risk Management Culture.

Risk involves technology, process, people, business/domain issues and competitive pressures, but also the proactive cross-functional communication between various disciplines. At the end it is none other than people who make the rules, define the requirements and form a culture to work. They are the rulers of the project destiny. There is no super-technology but people, who develop risk management plans, who set targets, who define scope. Perhaps they are completely dependent on each other for the success of any business. Symmetry between business needs and IT processes carried out by people is necessary for all the phases of a software implementation. This becomes possible only by minimizing software risks and ensuring maximum return on technology investments.

IT projects are too often about technology when they should be about people, say Dann, Broome and Joyce (1998) who share their experience of projects which put people first. Having discussed different aspects, methods, current situation, one fact is very clear that it does depend on how efficient and knowledgeable the software development team is or how sure the client is about the requirements he provides, but it also strongly depends on the right people meeting the right way of working. Let us look at three different scenarios to understand this view point.

Scenario 1: A big project handed over to a small company in order to save costs, which finally results in schedule overrun and causes losing market share or delay in the business operations.

Scenario 2: A project with vaguely defined objectives, handed over to a very efficient team but not so efficient management, where requirements keep changing as the project progresses, again causing schedule overrun.

Scenario 3: On a positive note, a very complex project contracted to a team with average skills yet technical enough to build the system. The project manager is fully equipped or highly experienced to handle such projects, hence successfully implements the project with an average team.

Therefore it entirely depends on the right people working at the right time, with the right attitude and right resources/methods. It practically seems difficult; however there can also be a few chances of following the best available practices, yet not being able to build the software on time. However, here the winning edge would be: project might delay only by few days or weeks instead of overrunning by several months or years.

Hence, all projects are different. Existence of two identical projects is impossible - even if they are geared towards the same result with the use of same tools and built under similar environments. Projects are brought to life by people, animated by people with different backgrounds and are immersed in a specific corporate culture. Research by ITcortex (2005) further strengthens this outlook that "more than everywhere else, the human factor is decisive in projects".

Chapter 4

Research Methodology

“Risk comes from not knowing what you`re doing”.

-- Warren Buffett, 1930, American Investment Entrepreneur

4.1 Research Framework

With a view to attain an in-depth understanding of software project risks and how they are handled or perceived because of good/poor organization culture, it is necessary to espouse a variety of research methods. This chapter summarizes various methods of research on the grounds of why they were adopted and an explanation of how each method was developed. The chart below provides a short description of each phase of the study and the research techniques that were implemented during these phases.

Research Phase	Topic under Discussion	Method
Phase One	Understand the area of software risk, risk management and culture in an organization	Secondary research Case Studies
Phase Two	Identify the software risks and study their behavior with respect to organization culture	Secondary Research Case Studies Interviews Surveys
Phase Three	Provide a framework to help in building a healthy risk management culture in a software organization	Secondary Research Interviews Surveys Case Studies

Figure 4.1: Methodologies Adopted

The research techniques used in this study were mainly qualitative. There are two main reasons for this. Firstly, desk research shows that in spite of enough statistical studies conducted on the types of software project failures, reasons of failures, number of failures and software risk management, there has been very little discussion on the topic of risk management culture, both by organizations and also collectively (industry wide).

Secondly, as there are numerous methodologies and suggestions by various experts in the field of software project risk valuation, it was obvious that an attempt to gain first hand statistical research would have provided inconsistent measures.

With the collapse of *Toyota Prius* and other large software projects, the issue of corporate governance has started to gain pace recently. This meant that the area of software risk management is still developing, and attempts to find quantitative data were deemed to be outdated or worse still misleading. The survey questionnaire found to be of real value, which has been used as one of the basis for developing a suitable support framework.

There were four main types of research carried out which are outlined below:

Method of Research	Phase of Study	Grounds for choosing the method
Desk Research	  	Involved consistent study of existing work throughout the dissertation (using books, online articles-journals, magazines, youtube, presentations, speeches, software methods, case studies, discussion boards, blogs, e-mails). In the first phase, it was used to gain a clearer understanding of the current issues within software risk management. For phase two it was used to develop the issues found in phase one and finally it was used for developing and verifying the recommended framework for phase three.
Interviews	 	The interviews were conducted with top managers who closely encounter software project risk management issues resulting from poor/good organization culture.
Case Studies	  	Many case studies were employed, some in-depth, while others were used with a view to validate a point. They provided an industry-wide perception of the software risk issues concerned with organization culture and solutions which are currently popular.
Surveys	 	The survey referred to is a compilation of many surveys: Standish Group; AFSN Organization; ITcortex; Repario; Loon. Questions are compiled after researching on various surveys available on "organization culture effecting software development", also it was supported by the dissertation supervisor who himself had been a part of many software development projects.

Table 4.1: Description of Research Methods

Following pages provide further analysis on each research method. They also talk about how the research progressed, what were the problems encountered and how the success was accomplished.

4.2 Case Studies

The initial aim of the case studies was to help gain understanding of best practices of an organization and how it influences software project development. The organizations taken into account were IBM, Microsoft and Infosys, which had demonstrated more than 80% success in software development business for past couple of years (Standish Group, 2008). The case studies were chosen in order to provide a variety of different issues related to the topic of study.

Company	Main Focus	Reason
Microsoft	What makes it the best amongst many?	Microsoft has been investing in not only being the best but also protecting their image. The case study provides an analysis of how they conduct this, and the risks they encounter while not being able to manage the culture properly.
Infosys	Wonderful environmental facilities which keep employees always motivated.	Focuses on the different areas which the management feels can effect the perception of employees towards their work and believes in continuous development of employee along with organization. Business Week ranks it among top 3 IT services companies in the world.
IBM	Studying the project failure reasons of such a big name. Also its present indispensable status quo.	Previous chapter talked about the reasons of its failure due to improper management culture, also its 100 success stories. Next chapter on “findings” focuses on its multiple tools, techniques and ways of promoting a healthy culture.

Table 4.2: Aim Behind Choosing Case Studies

4.2.1 Progress

Information was gathered by a variety of means. These included books, online articles, journals, magazines, youtube, online power point written and speech presentations, software methods, case studies, discussion board, blogs and (sending) e-mails. A variety of views from experts were gained from software project related publications and websites. The book called “Software project development” by Joel Henry provided a great help in understanding the importance of team culture.

4.2.2 Problems Encountered

As the information required was based on the company culture, which was not easy to find on the websites, for it could have published online either if the failure went public or the manager himself talked about the failure. The actual company websites provided information on the current scenario of software development, but very little information on the cultural failures. However, annual reports and website articles were very useful.

4.2.3 Accomplishment

As the information available on risk culture is very qualitative, case studies deemed to be a very good option, as they allowed views and opinions on different issues. Furthermore, the companies discussed within the case studies were actual cases of blue chip organizations; they provided assessment for existing views to be put into the context of industry.

The companies chosen were judged appropriate for different reasons, some were chosen after seeking advice from interviewees and supervisor of the university. The case studies on Microsoft and Infosys were chosen because they had strong views on the limitations of current risk culture. The case studies on IBM and Microsoft were chosen because they provided clearer understanding as to software development risk is currently assessed.

4.3 Desk research

Desk research was conducted at every phase of the project. At each phase, research was conducted under four headings, mainly literature reviews, surveys, expert views and current practices.

4.3.1 Progress

The survey on Software Project Risk management was the result of desk research, however wider search provided more data for survey on areas of software project risks handling and its importance. This method was employed at every phase of the research in order to justify the need for effective risk management culture. It also helped in verifying some conclusions derived from the analysis of interviews, surveys and case studies. The approach taken at each phase of the project is outlined below:

Phase	Field	Approach
I	Identify	There were three research areas. Mainly, most prominent and common software risks, measuring risk value and good/poor risk management cultures in different organizations. Largely, online articles, books, personal observation and experiences were analyzed in order to gain an understanding of these areas. Surveys were also looked at in order to obtain statistical data.
II	Appraise	Once an understanding of various topics was gained, key issues started emerging. These were researched further and are presented in the findings section. This area was developed using white papers and articles, mainly from the internet.
III	Wrap up	Desk research was used in two ways in this field. Firstly, in order to find the issues which required analysis and secondly, to seek best practices at present in the different focus areas. This was later used in order to develop an effective framework to help in embedding software risk management in the everyday work of an organization.

Table 4.3: Desk Research Rationale

4.3.2 Problems Encountered

The main problem with desk research was obtaining relevant information on the topic area. Since the subject of “software risk management culture” has not been developed extensively, there were limited articles which were specific to the study. Another issue was that the information provided in most of the articles was very vague. An attempt was made to contact the authors. They were only willing to provide limited information.

4.3.3 Accomplishment

Although the information gained from this method was not all specific to this study, it did provide some valuable insights and views. Furthermore, a lot of the research provided avenues for additional data in the form of interviews with key figures in the field of software project risk management with respect to organization culture. This method was particularly relevant for obtaining a firm understanding of risk culture and the key issues relevant to this field. Secondly the lack of research on the chosen area provided a basis for the key questions to be formulated for this study.

4.4 Interviews

Interviews were conducted in order to gain a better understanding of the area of study. They were mainly questionnaire based. Same set of questions were asked in order to spot the variations between various organizations' cultures. The interviewees are outlined below: (also see appendix)

Interviewee	Position	Significant contributions
Fraidy Pinto	Regional IT Manager, Rezidor Hotel Group.	Mr. Pinto was contacted in order to gain a better understanding of the current methods and views on s/w project management sector and how it is implemented in their multinational organization with universal procedures and standards. He was contacted after reading an article on new openings of Radisson, Park Inn, Regent hotels in Middle East and Africa.
Iqbal Madakkatel	Section Head, Database Warehousing, Commercial Bank of Dubai.	Mr Madakkatel was able to provide an in-depth analysis of how his organization takes care of its employees and has various plans for keeping their spirits high in order to get the work done. He has been a part of this industry for over 10 years.
Salama Aldhadheri	IT Manager, Institute of Applied Technology, Abu Dhabi.	Miss Salama was able to throw light on the procedures followed and difficulties encountered in order to receive support from the management regarding various needs related to software project development.
Abeer	IT Supervisor, RTA, Dubai.	Helped in studying the culture of a government organization of Dubai, Road Transport Authority, Dubai, which is successfully running a well organized operation in Dubai.

Table 4.4: Overview of Interviews

4.4.1 Progress

The interviewees were very helpful in not only providing an understanding of the topic areas, but also in providing additional information and contacts. Mr Madakkatel provided additional information in the form of presentations, which helped in developing progressive tool for identifying risk factors.

4.4.2 Problems Encountered

Interviewees mainly provided information related to their own domains. However, two of them had very little understanding of the methods employed outside their team, which could have helped more in understanding their organization culture; for example, Ms Abeer had very little understanding of risk culture in general and knowledge about what practices are carried out in the other departments of her organization. Although this made it difficult to gain views from the same person on different aspects, it did strengthen the findings on the need for experts in certain fields to gain knowledge of other departments that could have an impact on theirs. In spite of Mr Pinto's expressed interest in the study, his integrity made it very difficult to gain specific information from him. As he belonged to the Hotel Industry, where software need is limited only to CRMs and some well defined requirements, as compared to big software companies which deal in progressive development of a software project.

4.4.3 Accomplishment

The interviews proved very helpful; particularly at phase one, for it was important to gain a strong understanding of organization culture in a software project company, the risk management process and the relation between the two. The interview with Mr. Iqbal was significantly useful as it was conducted face-to-face and enabled discussing intricacies of procedures adopted in his organization, as his bank had been successfully implementing software solutions for a long time. The other three interviews were via phone and email. Yet the information obtained was very useful.

4.5 Survey

There was only one survey used throughout. This was a compilation of the studies by Standish Group 2006-2008, and a thorough analysis of various surveys conducted on software development by various organizations in the past (Loon, 2004; ITcortex, 2006; Repario, 2006; AFSN, 2008). The research for the study consisted of 52 employees of 27 international and 5 local firms who responded to an emailed survey on software project risk management. Some specific questions on risk management and organization culture were embedded with general questions, so that the person filling the survey does not feel that he/she is being asked for any such information, which he is not allowed to disclose.

4.5.1 Progress

The survey titled “**Risk Management Survey of your Latest Software Project**” (see appendix) consisting 83 questions, came into light through research conducted on various software project failures. The study had three key objectives:

- ⇒ To gain insight into the current state of software project risk management across large and small corporations rather than just knowing the number of failures and successes on the basis of projects undertaken in a particular organization.
- ⇒ To evaluate the impact of organization culture on the risk handling, using embedded questions related to cultural aspects.
- ⇒ To assess the contribution of organization culture on employee performance on the whole.

4.5.2 Problems Encountered

The only problem with the survey was that it was quite long to answer; therefore 20% of the surveys were half-filled which resulted in the cancellation of their inclusion. In order to cover all the areas of the “very vast” field of software project development and organization culture, it was necessary to have the minimum length of survey as adopted. However the research conducted using other means showed that the issues discussed were current and still underdeveloped.

4.5.3 The Success and Significance of Outcome

The views and findings expressed in the surveys were consistent with the discovery of other more recent findings. The results of this survey, together with other views, have been used as the foundation for developing phase three of this study (developing a solution).

4.6 Research assessment

There were a variety of research methods investigated and adopted. Although all did not materialize the information obtained, yet provided a very strong platform for this study. An overall summary of the research conducted is discussed in this section.

4.7.1 Types of Research Techniques

There were two types of research methods employed – informative and dogmatic research. The informative research methods refer to the interviews with Mr. Madakkattel and studying methods like Agile, Lean, companies like IBM and Infosys for phase one. These methods were used in order to gain a better understanding of issues related to risk valuation, software project risk, risk management and organization culture. The dogmatic research refers to the interview conducted with Mr. Pinto, surveys at phases two and three, brief details of number of software failures and their reasons, which greatly helped in tracking down the history of particular company culture. These methods were undertaken in order to strengthen the findings of this study and provide alternative views to verify a point made in other findings.

4.6.3 Overall Limitations of the Study

The main problem was that the area of study was very subjective. In order to make it specific at few places and uphold the interest in the study consistently, special diagrams had to be drawn. There is satisfactory quantifiable data available on the topic of software risk management but the area of culture is yet to be explored. This could possibly be due to the fact that companies are reluctant to discuss problems and failures resulting from their decisions. As studied by Standish Group, in one such problem faced by Robbins-Gioia, who in 2001 demoted their marketing director due to ERP failure.

Another issue, which limited the research and findings of this study, was the inconsistency of studies and measurements by organizations, for example, there are a variety of measurement techniques for risk management and valuation. An adaptable industry standard is required in order to provide easier assessment, for which a national study is necessary which could provide quantifiable results based on common variables.

The next chapter provides the results acquired with the help of above discussed research methods.

Chapter 5

Research Findings

**“Poorly designed and mismanaged cultures
are the silent killers of the corporate landscape”.**

-- Repario Ltd, 2006

5.1 Findings

In this chapter, a close study at the research conducted over seven months is performed. The findings are illustrated in a variety of different formats depending upon the results of the research. The approach adopted is outlined below.

Research Findings	Presentation Method
Desk Research	Bubble Diagram
Case Studies	Table Format
Interviews	Table Format
Survey	Bubble diagram

Table 5.1: Presentation Methods to Illustrate Findings

5.2 Desk Research

Variety of information sources were explored. Desk research was used as one of the methods of findings. Although not common in the findings section, the views expressed were, by people with vast expertise in both risk and Software Project Management. Furthermore, their opinions were seen as necessary to be considered while developing a support tool for Effective Software Risk Management Culture. The method used to present the findings was deemed to be the most effective method as it helped in identifying issues or series of issues. An outline is provided below to help further understand this approach.

Note: Analysis supported by is taken from references.

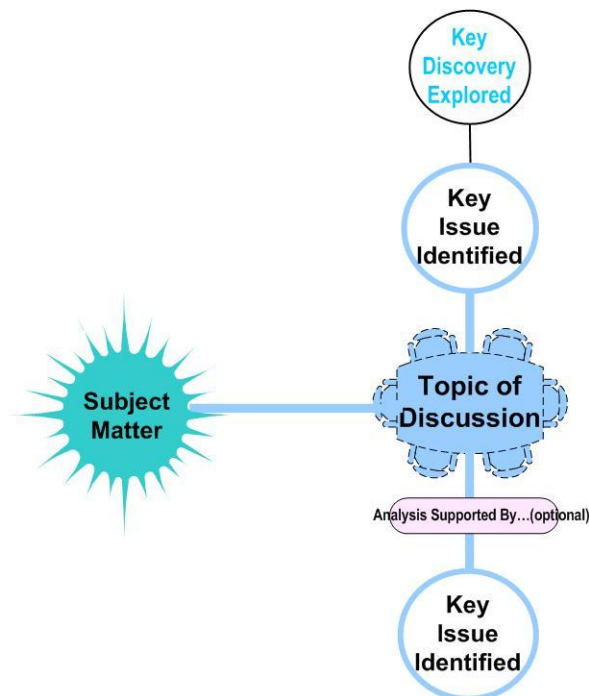


Figure 5.1: Bubble Diagrams Presentation

5.2.1 Software Project Risks | Barriers and Problems within an Organization

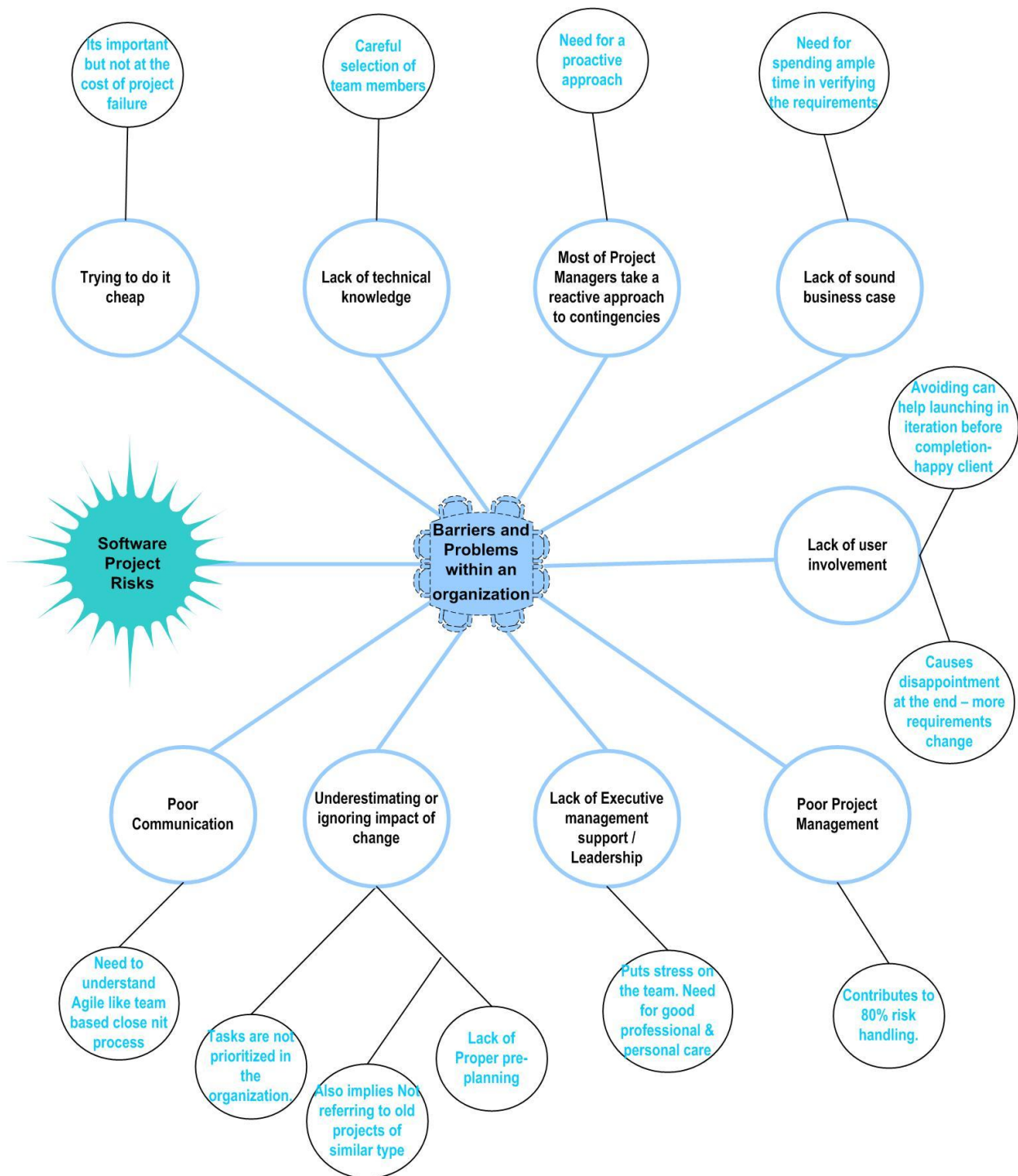


Figure 5.2: Software Risks – Barriers and Problems

5.2.2 Software Project Management | Risk Management Components

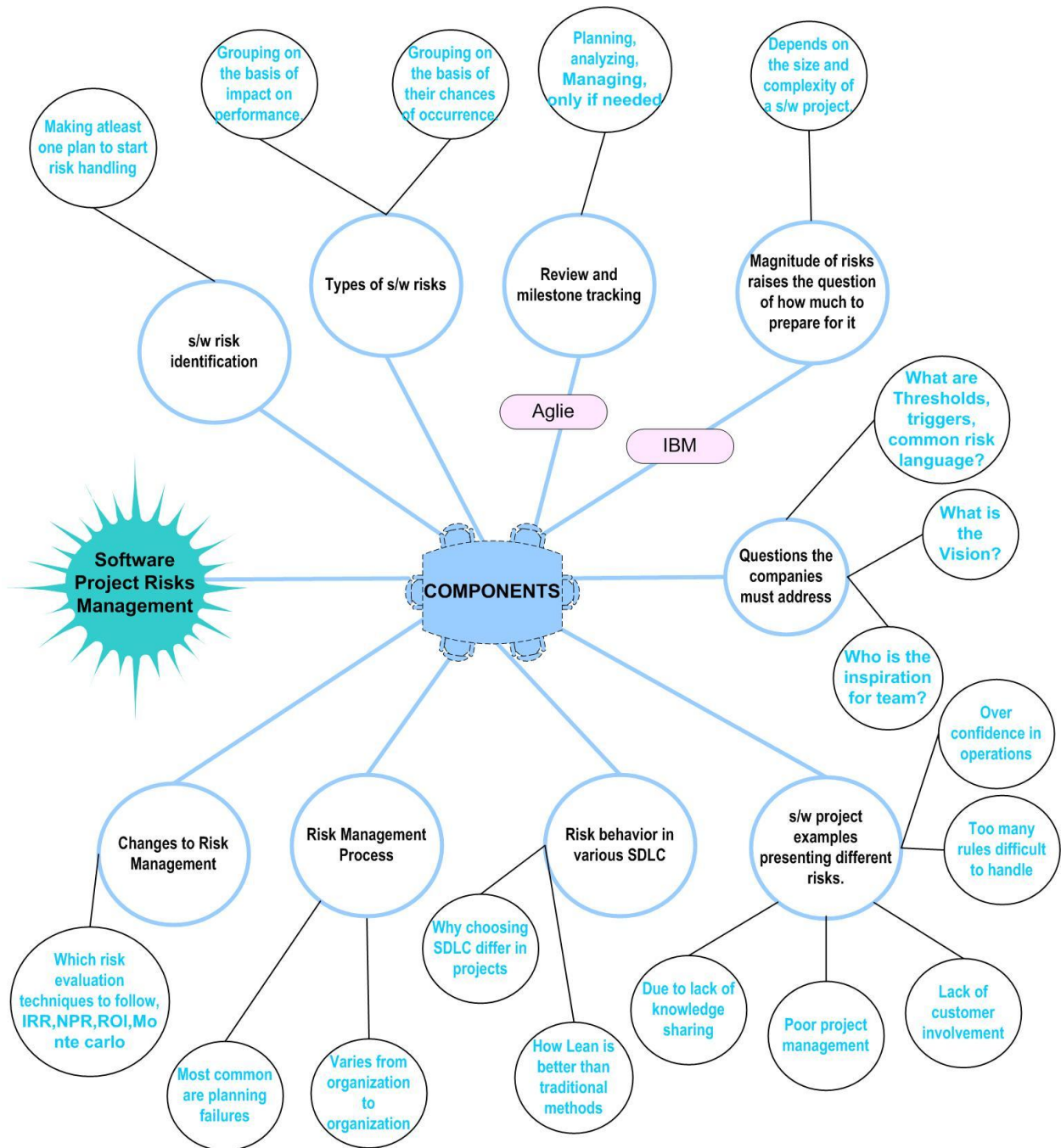


Figure 5.3: Software Risk Management – Components

5.2.3 Organization Culture Components which Impact Risk Management

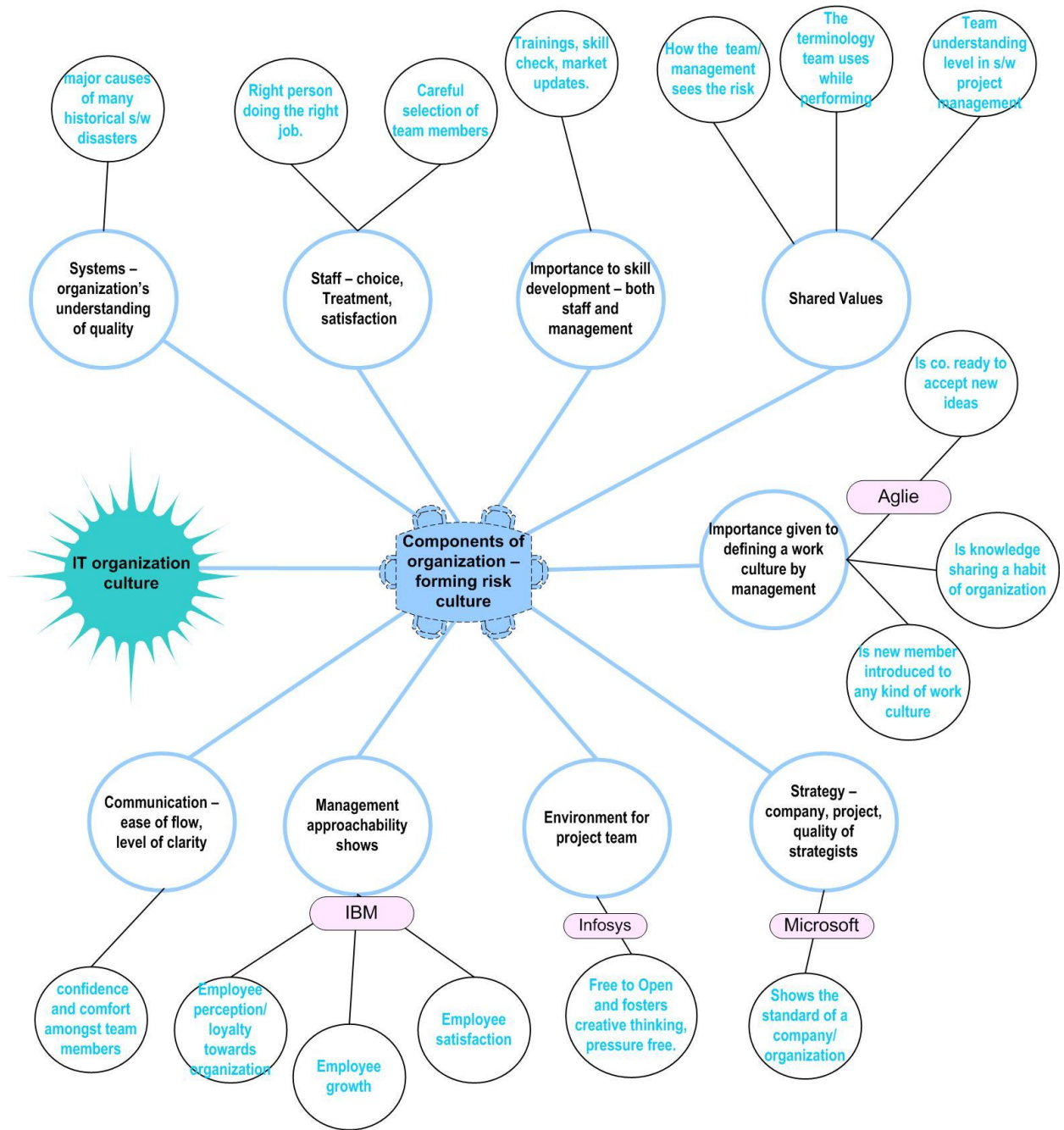


Figure 5.4: Organization Culture Components Impacting Risk Management

5.3 Software Risk Management Culture Interview Findings

Interviewee ⇒ Topic of discussion ↓	Fraidy Pinto Regional IT Manager, Rezidor Hotel Group. (8 Years)	Iqbal Madakkatel Section Head, Commercial Bank of Dubai, Database Warehousing. (4 years)	Salama Aldhadheri IT Manager, Institute of Applied Technology, Abu Dhabi. (3 years)	Abeer IT Supervisor, RTA, Dubai. (3 years)
1. Risks Encountered	Delay in setting up the network, everyday intranet issues, 1 st time software installation issues in the new properties.	Cost overrun, time overrun, implementation delay.	Majority of projects fail on quality for not meeting user expectations which causes speeding up the development process, leading to poorer quality. This becomes a chain reaction.	Delay in implementation as RTA (Road Transport Authority) does not want to compromise on quality. Which causes time overrun.
2. Issues with Risk Management	Communication issues because of various nationalities performing under one roof, which leads to ending up writing down the Risk management process in details and providing training at various levels.	→Inability to define and describe in detail, the end deliverable of the project in the requirement analysis stage of the project. →Poor assessment of vendor, technologies, consultants, & so on. →Inability to develop comprehensive test cases in the beginning & execution phases of the project.	Since the organization plan is not clear, according to Salama, they cannot internally plan their work very well.	→Because of the complexity of projects, requirement specification doesn't cover all the aspects which causes schedule over run. →Testing sometimes gets stuck which further causes delay and some times even more errors.
3. Methodology of Risk Management	Planning, scheduling, task allocation, timely check on the tasks accomplished.	Management is very eager to support any efforts by the employees to improve the process, the operational efficiency and there by reducing the costs. Suggestions from employees are carefully studied by management team for the feasibility of implementation. Management feels they need to support the efforts by adequately providing resources. Enquiries are conducted to learn lessons from the failure.	Only requirement specifications are documented, as such no separate plan carried out to handle risks.	Spending good time in planning, timely meetings, recording the errors, making test plans, checking up with the time flow.
4. Environment & Human Resource Management	World class environment for employees, best standards and infrastructure has been employed.	Decent working hours, no harassment for mistakes, paying for overtime, no discrimination on any basis, attention to health & safety.	Satisfactory work environment, no need of improvement on the environment front.	World class facilities. Promotion to employees without any delay.
5. Human Resource Development	→Trainings on latest skills for employees. →Staff parties, events, reward functions, special awards for exceptional performance is awarded to keep the spirits high, as its hospitality. →Once in a week meeting to discuss personal issues of employees.	→ On the job and online training programs in the areas of product knowledge, regulations & compliances, techniques, technologies, soft skills, Project Management Professional (PMP) certification. → Special courses to improve soft skills - leadership, supervisory, time and stress management trainings. - →employees Suggestions accepted.	→Few trainings are approved by management for the new staff to be trained by old employees to work on their skills. →Inclination is more towards getting results than improving the employee standard.	Staff is timely updated by giving trainings in order to inculcate latest technologies.
6. Tools to bring clarity of risks and risk handling by team and the interviewee.	Intranet, personal discussions, emails, timely meetings.	Meetings on various milestones, intranet portals, emails etc. are used to keep employees informed about overall business objectives and processes, running projects and their results. Once in two weeks client feedback.	There is no direct communication. The new projects or results of running projects stay with the higher management and few people. The organization has no perfect planning for its projects and doesn't communicate regularly with its people.	Meetings on the completion of a task and verifying it with the specification, as the quality are of utmost importance.

Table 5.2: Interview Findings

5.4 Organizational Culture - Case Studies Findings

Organization Topic of discussion	⇒	Microsoft	Infosys	IBM
↓ (1) Views on software project risks		Risk represents a possible event or condition that would have a negative impact on your project. A problem waiting to happen. Characteristics: Inherent in every project, neither intrinsically good nor bad, not something to fear, but something to manage.	RISK stands for Rate, Innovate, Share Knowledge (Srividhya, 2007) The whole idea of undertaking a project is to risk. Risks provide opportunities for innovation. To differentiate from competition, project managers and organizations must see opportunities in risks and add value to client services.	Risk is an ongoing or upcoming concern that has a significant probability of adversely affecting the success of major milestones.
(2) Method/Tool of software project risk handling		Microsoft Solutions Framework (MSF). Latest version is MSF4 (earlier MSF-2,2.5,3) Key goals of MSF: ⇒ Drive business success through business & technology alignment ⇒ Ensure high quality solutions; handling many facets of quality as defined by multiple stakeholders. ⇒ Accelerate delivery, reduce costs, and minimize risks. ⇒ Improve team effectiveness. ⇒ Results from project teams and product groups are analyzed. ⇒ Analyzed results are contrasted with industry practices and methods. ⇒ Combined results are then organized and consolidated into "people and process" guidance.	Along with Microsoft Blueprints. Infosys has introduced 360 degree Risk Management Model that helps rate and innovate and also exploit opportunities. It focuses on enabling project managers and organizations to discover and seek the silver lining in the clouds of risk. Periodically identify and mitigate the negative consequences of risks. Utilize the learning from dealing with the risks to enhance the competency of managers. (Srividhya, 2007)	Challenge: --Link IT processes and data to business strategies. --Optimize IT processes, including planning, implementing and delivering IT systems. --Optimize IT resources to capitalize on business opportunities and gain competitive advantage. Benefits: → Better risk management. → Increase in investor and shareholder confidence. → Corporation wide standardization of all IT-related risks. Solution: → Developed IT governance framework to establish governance processes and steps to monitor regulatory compliance. → Created a process maturity model for IT governance processes.
(3) Approach to software risk handling		Application Lifecycle Management (ALM) approach based on the traditional software development focuses on a single aspect of the application life cycle, ALM covers not only development, but also the full range of the project life cycle, including business requirements, modeling, build, testing, maintenance, and operations.	Seeing risk as opportunity to learn for future projects. Risk by itself is not bad. The secret lies in striking the right balance between its negative consequences and potential benefits of associated opportunity.	IBM's unified approach to IT governance and risk management. The material impact of risks on IT have reached a threshold that justifies investment in an enterprise architecture and structured process approach for enabling IT governance and risk management.
(4) Organization principles which helps in software risk management		⇒ Foster open communications. ⇒ Shared vision. ⇒ Empower team members. ⇒ Clear accountability and shared responsibility. ⇒ Focus on business value. ⇒ Stay agile, expect change. ⇒ Invest in quality. ⇒ Learn from all experiences. ⇒ Partner with customers. ⇒ Always create shippable solutions.	LEAN manufacturing principles across IT organization. ⇒ It's a change management tool for continuous improvement. ⇒ Eliminate non-value added wastes that incur in any process. It can be applied on four areas ○ Cycle time ○ Inventory ○ Value-added content ○ Through put	Philosophy: Companies that fail to reduce the complexity of IT, spend 30% more on finance operations and 18% more on human resources

Topic	Microsoft	INFOSYS	IBM
(5) Components of existing culture influencing software development projects (if any)	An investment in quality becomes an investment in people, as well as in processes and tools. Recognize this and incorporate quality into the culture of the organization. It encourages: ⇒ Accountability ⇒ Knowledge sharing ⇒ Innovation (if encouraged then failures are also accepted, as innovation doesn't bring success alone) ⇒ Creating a vision statement. A short, coherent statement that concisely describes the purpose of building the new or improved system.	⇒ Offers its IT professionals challenging assignments, competitive salaries and benefits and one of the first stock option plans adopted by a public Indian company. ⇒ Attract and motivate IT professionals by offering: an entrepreneurial environment that empowers them; programs that recognize and reward performance; challenging assignments; a continuous updating of skills; ⇒ a culture that emphasizes openness, integrity and respect for the employee.	IBM's unified offerings for IT governance and risk management Standard process model with supporting software and services for helping clients consistently apply governance and risk policies to IT. Center of Excellence(COE) For clients to access consulting experts, education and information. IBMTTools: Information Management, Lotus, Rational, Tivoli and WebSphere (discussed in chapter 2)
(6) Personnel Development Tools and techniques w.r.t risk handling	Microsoft Visual Studio® Team System development system Various role-based editions of Visual Studio Team System make it possible for team members to take on more specialized roles and to integrate the software with the tools they are already using. Environment: Establishing an atmosphere that promotes knowledge sharing and collaboration. Focus: Prioritizing the areas in which knowledge sharing is most valuable Tools: Providing the digital tools that make knowledge sharing possible. Motivation: Rewarding people for contributing to a full flow of knowledge (Gates, 2008) Employee benefits such as pool tables, free fruit and drinks, working outside and the areas for relaxing.(MS,2009) Through the Employee Involvement Program, Microsoft aims to offer employees with an additional opportunity for personal and professional development, to apply and develop their skill set through cross-group collaboration, teamwork with colleagues and partners in the community, to expand personal horizons outside the MS environment.	→ Its main facility in Bangalore (India), which spans five acres, encompasses not only 160,000 sq. ft. of office space but also 150,000 sq. ft. of landscaping, a cafeteria, outdoor sitting area, library and gymnasium as well as tennis, volleyball and basketball courts. →Through this campus-like environment, the company fosters a collegial atmosphere and informal culture, which is further promoted by its "open door" operating philosophy where communication and ideas flow freely irrespective of title or tenure. → Invests heavily in training, including 14-week training sessions for newly recruited IT professionals as well as a variety of two-week continuing education courses in technology and management skills conducted by a 33- person faculty.	Training: A variety of in-house localized training and IBM Rational external. Five days training per person where twelve people on a project attend an on-site course. The training is defined relative to the project's specific needs. Mentoring: Involves carrying out predefined activities for knowledge transfer from person to person. It is the key to identifying future mentors within the organization, who can be trained on the projects, who will ultimately become future mentors. Dependent on skills being taken on board, suggests three days support for the first two weeks, then two days support for the next four weeks.
(7) Views about developing a culture	Culture of innovation Microsoft has an innovative corporate culture and a strong product development focus that is designed to keep them on the leading edge of the industry. It believes that its employees are the company's most important asset. They are the source of its creative ingenuity and success so it empowers each staff member to take initiative in solving problems, coming up with new ideas and improving the organization. (Daniel, 2008)	An effective system in place to ensure creation, documentation and dissemination of experiential knowledge. The backbone of this system is a user friendly, searchable database known as the "Body of Knowledge (BoK)", comprising of knowledge components contributed by employees of the company. Incentive schemes are in place to encourage a knowledge sharing culture in the organization.(SetLabs Briefings,2007)	Whether the organization has an internal development team, or outsources development projects to a third party, it is common for the relationship between the business team and the development team to be adversarial. The two groups often develop an "Us versus Them" attitude towards working together. It focuses on ensuring to have good, working relationships and communication. Encourages creative thinking.

Table 5.3: Case Studies Findings

More than 80% success rate of above discussed organizations validates the fact that failure provides a great learning opportunity and should be viewed as lifeblood of success. If you give people freedom to innovate, the freedom to experiment, the freedom to succeed, then you must also give them the freedom to fail. Sloane (2008) states that, “it is only by trying lots of initiatives that we can improve our chances that one of them will be a star”.

According to Deshowitz (2007), KPMG International's survey of 600 organizations across 22 countries revealed that 86% of respondents reported the loss of up to a quarter of their targeted benefits across their project portfolios. Nearly half of respondents reported at least one project failure in the past year, an improvement from KPMG's 2003 survey where 57% experienced one or more project failures in the previous 12 months. Gumn, a partner in KPMG's IT advisory group, was surprised to find that 59% of organizations had no management process to measure benefits.

“Project Management Errors tend to be associated with 100% of the cancelled and seriously delayed software projects noted over many years”. (Jones, 2005)

5.5 Survey findings

The survey titled “Risk Management of your Latest Software Project” was conducted across different software organizations, involving 52 employees who contributed many surprising facts about some important aspects. Not only are the findings directly related to this project, but also many see the credibility of the work to be very important to this field of study.

5.5.1 Main Findings of the Survey

Unexpectedly, the results of survey were very positive. Out of 52 surveys 37 were positive. This indicates a very good project success rate as compared to the studies in the past. Since the aim of the survey questionnaire was more to find out “the cultural reasons and procedures adopted in order to deliver successful projects” and less to acquire a failure statistics, which made the research basis stronger, as the survey questions were answered more for successful projects; hence it was not so difficult to analyze various cultural success factors.

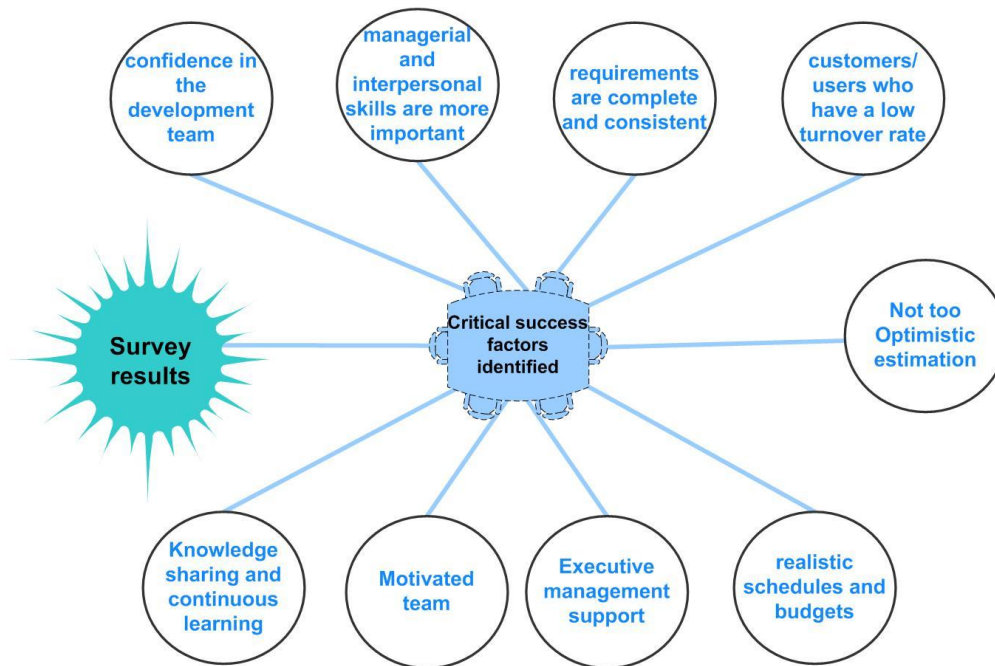


Figure 5.5 Survey Findings

1. 80% of the employees feel that it was easy for them to arrange a (project related) meeting with their respective project managers, which implies that the management is easily approachable. This further leads to the presence of process clarity in the minds of employees.
2. More than 85% of Project Managers did not conduct meetings with their team members on personal issues, which implies less attention towards employee behavior and more towards their performance results.
3. 82% think that there is no major need for any change in the organizational environment, which implies the presence of satisfactory work environment.
4. 40% are not sure about the procedures of the organization. Still it can be concluded that thresholds would be well defined which led to the success of projects.
5. In more than 75% cases, the management approach throughout continued to ensure co-ordination of sub-projects, communication among different sub-project teams and addressing shared horizontal issues.
6. 64% say that they use iterative development; yet the major part of integration testing takes place towards the completion phase. This implies that the mere adoption of a quality processes can lead to success in spite of leaving major testing on the completion phase.
7. 70% feel the need of recreational activities to release stress of coding work, which means, there are not enough stress buster activities practiced in the organization.
8. Almost 80% of the employees surveyed were sure about the system functionality right at the beginning of the project. Which implies the clarity of what user wants.
9. More than 60% Project Managers were able to list the top ten risks of projects they were expecting to face. This implies clarity of system requirements and pre-defined task lists, which enabled successful risk identification.
10. For more than 65% project teams, it was reasonable to expect financial stability for the duration of the project, which implies they were able to measure financial value of the projects, hence foresee upcoming risks.
11. In more than 60% cases, the organizational changes were clear to its employees; this implies healthy communication flow in the organization.
12. In more than 70% surveys, Project sponsor took the responsibility and accountability for keeping the project within scope, which means the clients did not leave the responsibility of system's functionality completely on the shoulders of project development team. (see Appendix C)

5.6 Overall Findings of the Study

Generally, a mismatch exists between the importance organizations place on their software projects and the level of responsibility assigned to protect them. Predictably, different organizations and corporate functions perceive very different qualities as central to software risk value. There is a broad agreement as to the most damaging scenarios to a software project and the relative importance of various risk avoidance measures. For those (relatively few) firms which have carried out risk valuation exercises, one of the key drivers has been the employee satisfaction. It is evident from the number of surveys that, small projects do not need as much care and attention as large ones. There is a clear shortfall in the market supply of effective employee culture solutions. On one hand software project crises can damage shareholder value severely and on the other, firms with strong learning attitude and strong communication values can outperform the market by over 100%.

Chapter 6

Discussion

Software Project Risks -- Issues Identified From Culture Perspective

**"Real motivation comes from within.
People have to be given the freedom to
succeed or fail."**

– Gordon Forward (n.d.)

As studied by Standish Group (2009), project management organizations are losing their perspective on what is important. It is not about the process or the methodology; project success is achieved when smart, capable project managers are allowed to focus and spend time on what is important to get their projects delivered.

6.1 The Issues Identified

Analyzing the results of survey and interview questionnaires, there are a number of issues which need to be addressed if risk management culture is to gain momentum in the software industry, which is necessary for every organization, big or small, because akin to a human being, no organization is perfect. Say IBM and Microsoft, they became superior with every passing day. Like an individual, every company learns with time and experience, and some smart ones from other company's mistakes.

Interview with Ms. Salama indicates that a certain amount of ignorance has been developed within industry as to the need of protecting the company culture, where management focuses more on product development and less on the employee performance. For example, studies as those conducted by McManus and Wood-Harper (2008) show one notable causal factor in project's abandonment was the lack of due conscientiousness at the requirements phase. An important factor here was a low level of skill in design and poor management judgment in selecting software engineers with the right skill sets. However, not every organization acknowledges this fact.

Furthermore, the point above is highlighted by Ewusi-Mensah (2003). In his book he discusses that there are two dimensions to organizational environment in which the work of a team takes place. The social, which is amongst the team members and the organizational, which deals with management authority and position. The organizational issues involving behavioral and opinionated authority, the management influence, all come in line while shaping the project aims and objectives, and guiding the project to victorious completion.

Case studies' analysis, along with industry statistics sufficiently validate that software implementation projects are burdened with difficulties and often end up in failure. Perhaps, the bigger the project, the more likely it is to fail. Software development projects are a complicated and challenging interplay of people, techniques, cultures and technology, and their understanding shows that there are a multiple reasons why these projects can become a 'hot potato' that no one is willing to touch. As the FBI's Virtual Case File fiasco has shown that the \$170 million VCF system, a searchable database proposed to allow agents to "connect the dots" and follow up on distinct pieces of intelligence, ended five months ago without even touching the system deployment stage.

In the simplest terms, a software project mostly fails when the rewriting surpasses the value-added work that has been assigned to a budget. In a study by Goldstein (2002), similar situation was experienced by Sydney Water Corp., the largest water provider in Australia, when it attempted to introduce an automated customer information and billing system in 2002. According to an investigation by the Australian Auditor General, among the factors that doomed the project were inadequate planning and specifications, which in turn led to numerous change requests and significant added costs and delays. Sydney Water aborted the project midway, after spending AU \$61 million (US \$33.2 million).

Carefully looking at the findings of surveys, it is not very difficult to list down the reasons of a project to be a success, reason being the large number of positive surveys. On the other hand, interviews with various IT heads, threw light on many cultural aspects which are necessary for a successful project implementation. Also case studies of three finest organizations by themselves speak on various reasons for producing a large number of successful projects in a row.

Optimistic findings of research methodologies encourage the need for a strategic tool to aid effective organizational culture which smoothen the otherwise anarchy of entire software development process. This has been the main aim of the study. The following sections provide an in-depth look into the issues which need to be addressed in order to develop a satisfactory support solution. (See Appendix D for graphical analysis)

Let us closely look at the ten cultural issues identified, which cause software project failure. The diagram below shows the issues identified from the analysis of all the aspects discussed in previous chapters.

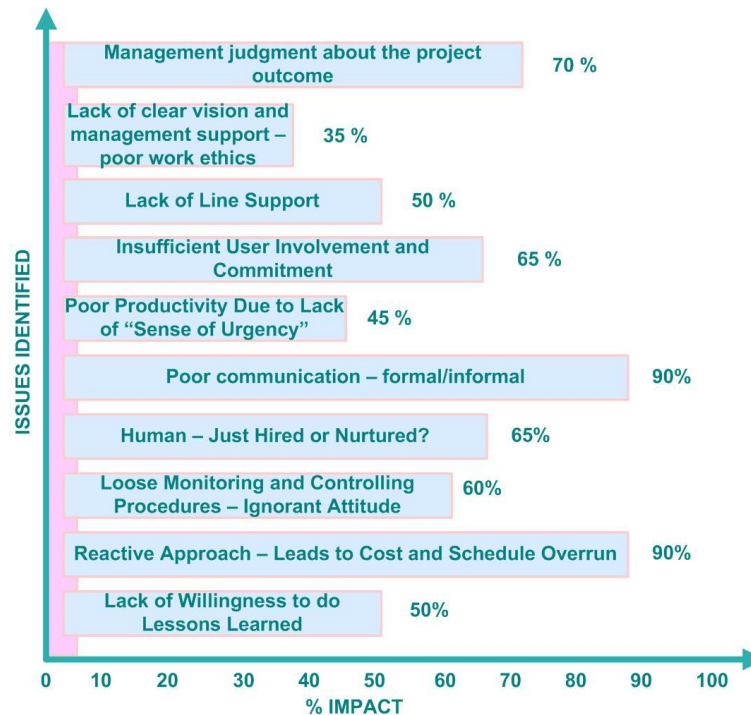


Figure 6.1: Software Project Failure - Main Issues Identified

6.1.1 Management Judgment about Project Outcome

According to the survey results, almost 80% of the employees were sure about the system functionality right at the beginning of the project, which led to the successful completion of their respective projects. On the contrary, this also implies that the project stakeholders have a tendency to make a judgment long before in relation to the relative success or failure of projects. Inspection of various (other) project failures simply shows that many project managers plan for failure rather than success.

According to McManus and Wood-Harper (2008) if we consider the inherent complexity of risk associated with software project delivery, it is not too unpredictable that only a small number of projects are delivered as per the original time, cost, and quality requirements.

Survey results showed, in more than 75% projects, clients took the entire responsibility of scope change. However, sometimes because of the pressure from clients and their changing requirements, the management holds the team responsible for it. Therefore this issue calls for seeking clarity of goals in the mind of management, then the correct procedure of transferring it to the team. This consequently brings more clarity of project development process, in the minds of management, and the developers.

6.1.2. Lack of Clear Vision and Management Support – Poor Work Ethics

This implies, “lack of active participation of corporate management in monitoring progress in a project and in making decisions at critical junctures is a major concern”. (Ewusi-Mensah, 1997).

Unlike Microsoft (in the case study), one difficulty many organizations encounter is lack of a clear vision on how success will look like when the software application is deployed. Also highlighted by Ms. Abeer in her interview, many organizations focus absolutely on technology performance metrics of the new application (say cycle time of batch processing jobs, number of “seats” they will obtain with a software license, increased functionality compared with the legacy system, and so forth). These measures are important but they are not the only measures to count. Following image depicts how the organizational culture and the structure go hand in hand in order to successfully run a venture.

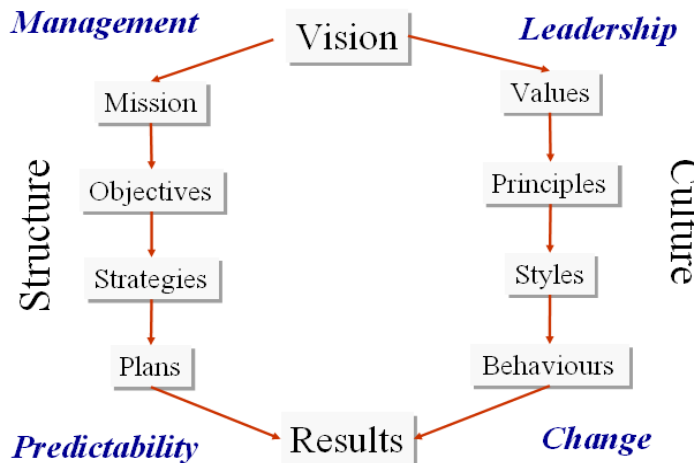


Figure 6.2: Culture vs. Structure (Stanciu, 2006)

The Standish group (1995) also confirms this as a major contributor because of the varied negative influences “lack of executive support” may have on the overall project outcome. The KMPG study (Cole 1995) lists this factor as “insufficient senior staff on the team”, which can largely seem to include lack of executive promise to keep an eye on progress of the project, at the same time be attentive in taking important decisions. Not every organization thrives on encouraging a concept like “shared vision” – clarity of vision in the minds of all the project members and possessing a shared and known terminology. As supported by Mr. Madakkattel, knowledge sharing atmosphere/environment implies; good infrastructure, the team members’ comfort while asking doubts and giving suggestions to the project manager, so that its absence does not give rise to mistakes in the later stages which may eat up the project time.

6.1.3 Lack of Line Support

More than 80% of employees (in the survey) believe that team leaders do not sign off on cost and schedule estimates and they are not held to budgetary constraints. If the project is meant to be headed predominantly by the Chief Information Officer (CIO) of the IT department, at the same time making CIO or any other higher management executive the project sponsor, might lead to its failure. While, it is not a new concept, many have been practicing it since long. In fact the tough decision to be made by CIO or IT executive is; making a line member interested to take the responsibility of sponsoring the software implementation, also educating them on what is necessary to accomplish this. The best way to find a prospective line campaigner for software implementation is to spot the best of all the line people who are expected to gain most from the successful completion of the system. Frequently, this is the manager or the executive whose department or division can visualize or see the extent of benefits it can reap from the implementation of the system. This will help the team to work on it with complete dedication and passion. Moore (1991) sees these promising sponsors as early ‘Adopters or Visionaries’. Their characteristics are:

- ⇒ Keep an eye on fundamental breakthroughs.
- ⇒ Business goals drive them. (more than technology goals)
- ⇒ Interested in project direction. (pilots)
- ⇒ Work towards earning a substantial return on investment.

6.1.4 Insufficient User Involvement and Commitment

More than 50% of the times (in the survey results), the client was not involved during the development. The Standish group (1995) lists this factor separately as “lack of user involvement” in the project development because of which team is not able to compete with the changes in the project. Disappointing or inadequate user participation and commitment hamper the project team’s ability to frame the requirements that are complete, reliable, and capable of successfully meeting the needs of the system as expected by users. As the project advances sequentially, more and more unidentified estimates and features start emerging that endanger previous estimates and schedule plans. Whereas, adopting methods like Agile enables accepting changes as and when they arise.

6.1.5 Poor Productivity Due to Lack of “Sense of Urgency”

Microsoft, IBM and Infosys (in case studies) promote the use of Agile-like techniques, which encourage “sense of urgency”. Due to lengthy schedule plans, very often team loses the sense of urgency causing time loss during the initial phases, which is difficult to regain in later stages.

Parkinson's Law identified in Agile states that: “Work expands to fill the time available” and Student Syndrome: “Given a deadline, people tend to wait until the deadline is nearly close before starting work”. Short iterations, choosing right personnel, training for development (if necessary), can prove to be of great help in retaining the sense of urgency.

6.1.6 Poor Communication – Formal or Informal

Failed projects signify that the project managers in those projects were often unconscious of dormant volcano which was going to erupt anytime. Unfortunately, the finding was; mostly there were some people in the organization who were aware of this eruption, yet did not inform the project manager of its existence. Ms. Salama (in her interview) also feels that it is imperative to implement risk communication in order to avoid these incidences to happen. In her organization, management is not really keen on putting efforts in developing communication and employee skills.

Effective communication is one of the key contributors of building unified teams and is vital to successful handling of key stakeholders. The probability of communication failure strengthens in a virtual environment. According to Standish group (2008) study, over 90% of project issues could be tracked down from communication problems.

As Microsoft encourages, a shared global-view, which involves a shared understanding of tools, terminology, culture, practices and principles, ensures effective communication within a team. A shared understanding can be implemented by project manager’s thorough analysis of communication constraints at various milestones.

6.1.7 Human – Just Hired or Nurtured?

6.1.7.1 Human Nature Obstacle

Culture study of Infosys supports and encourages the rationale discussed by Kwak & Stoddard (2004), in which project team members may have the necessary skills to employ a risk management process; however, this does not promise that the team will use it during the project lifecycle. The authors also emphasize on the fact that effective risk management needs acquiring functional behavior, but not just following a method or having diverse means of information. Moreover, they observed that behavior is a result of organizational history, structure, practices, and reward system. For instance, software developers naturally tend to withhold technical information given that information is a source of power. Nevertheless, rewarding and holding them accountable for sharing knowledge can help overcome this tendency.

6.1.7.2 Lack of Incentives for Employees to Change their Behavior

As agreed by Mr. Madakkatel and Infosys (case study), not encouraging and rewarding employees to work in a high pressure environment can be threatening to a software development project. Employees sometimes need an incentive to “push the edge of the envelope” in working for heavy projects. Simple incentives like bonus, free movie tickets, lunch or dinner gift cards, or simple recognition can go a long way to encourage employees to work creatively under pressure, and then to share this creative learning with the mechanism they have established to communicate and leverage various outcomes.

6.1.8 Loose Monitoring and Controlling Procedures – Ignorant Attitude

In 80% of the projects surveyed, review sessions were held to assess the continued relevance of the project, project performance, and to raise concerns about actual/potential problems, which led to their successful completion. However, ignoring continuous monitoring deprives a team of identifying any deviations, suggesting corrections, forecasting future performance and absorbing the lessons learned into the work of the organization. If monitoring and control procedures are not in place to determine how project performance will be assessed, prioritized and corrected, this may further affect the quality control, Human Resources (HR), risk, time, cost and project management elements. Ever since projects have increased in scale and magnitude, the monitoring methods have become insufficient. Not revising the procedures to encompass the new demands and not having periodic reviews to gauge the performance of the projects can increase the chances of failure.

6.1.9 Reactive Approach – Leads to Cost and Schedule Overrun

This issue implies reacting towards risk only when it is visible. For instance, testing only at the end, emphasizing on the amount of coding rather than quality, tracking the development against requirements only towards the end, similar factors further expand the project time. Cost overrun and Schedule delay is generally symptomatic of the occurrence of any combination of factors discussed above, because those factors have the potential to add to the cost of the project and push the delivery date further back and make a chain reaction of uninterrupted risk occurrences. My personal experience while working on an economic portal for Saudi government (www.jegs.org) could strongly observe this issue. Over time the project faces critical shortage of resources, as the Standish Group (1995) revealed. IBM (in the case study) also supports that the constant changes in requirements will cause designs to be revised each time, and this will influence all the proceeding work based on the previous designs. The KMPG study (Cole 1995, 4) found a major contributor to this factor to be “bad planning and estimating,” which renders the original cost and schedule estimates inaccurate and thus presents a false impression of project cost escalation and schedule delays.

6.1.10 Lack of Willingness to do Lessons Learned

It implies, not doing a “lessons learned” of what worked and what did not work. More than 70% of the project teams (in the survey), did not carry out this task in spite of having a successful outcome. Doing this might have led to gaining additional knowledge on how to perform better in the next project. In almost every organization’s project management process, the last phase of system development should ideally include some type of lessons learned. Unfortunately, most of the times this activity hardly takes place for some reasons. The organization may lose valuable insights of good and bad experiences of development that can easily feed the next software project. Also, software development is just one of many types of change management initiatives that constantly happen in an organization. By developing a culture of habitually doing lessons learned at the end of major efforts, companies can gradually develop “implementation of initiatives” as a key organizational competency. This will help an organization in the future projects.

6.2 Summary

Established project management enjoying formal project status and performance reporting measures provides a sensible image of the project, the resources employed and the progress it is making towards its objectives. It helps in making project outcome more predictable, which further helps in reducing uncertainties. It contributes to treating the risk and giving feedback especially on tight scheduled projects. Periodic monitoring and reporting on the status of the project helps in recognizing the signs of the overall project health and red-signals, for the project may go off its planned course.

According to the Standish Group (1995), project management is most valuable when planning new projects or enlarging existing ones. Not all projects need formal project management techniques. Project management perhaps is not appropriate for small and simple projects when the overhead of the project exceeds the overall value. However, there should be no exceptions when it comes to stakeholder interaction, especially when status reports point to potential trouble.

The summary (Figure 6.3) of discussion consists of all the research findings which are taken into account, while developing an effective culture (particular style or way of working) tool for a better software project risk management. Also two research papers are being written, based on this and the next chapter of the research.

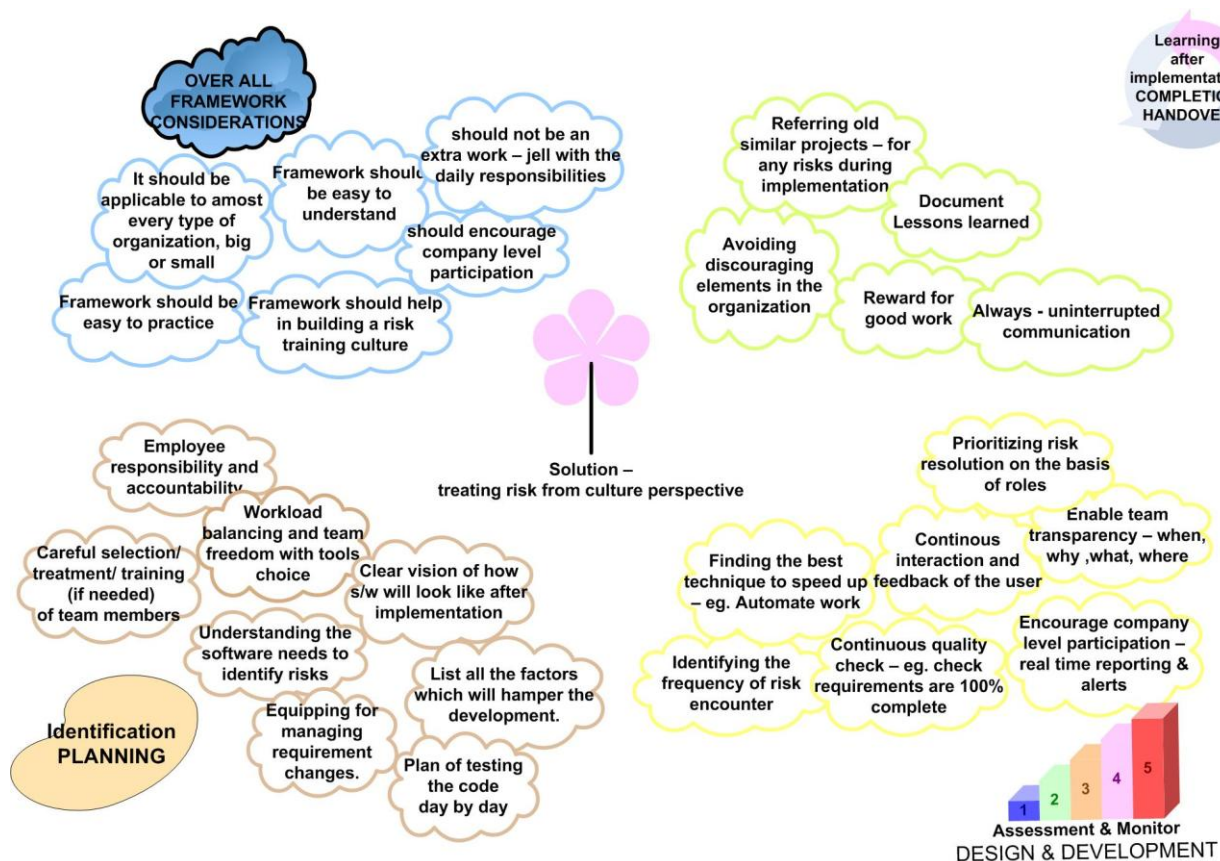


Figure 6.3: Various Factors Brainstormed for Solution Development

Chapter 7

Solution Development

“The golden rule is that there are no golden rules”.

--G.B. Shaw 1856-1950, Irish critic and poet

7.1 The Framework Foundation

Having all the understanding and analysis gained from the findings in relation to the significance of culture in a software development process, it is obvious that how momentous the culture is while deciding the fate of a project. The field of software development is very vast, as mentioned before; this study focuses on the culture aspect of software risk management. Therefore, the support framework tries to deal with cultural issues (in an organization) which hamper smooth project development. The figure[♦] below has been worked upon, towards developing a support structure, conducive to healthy risk management culture. A thorough analysis of findings and discussion has been carried out in order to identify key elements of an organization, which impact risk management process during software development. The following figure explains the elementary basics, which an organization needs to accept and understand before initiating a project.

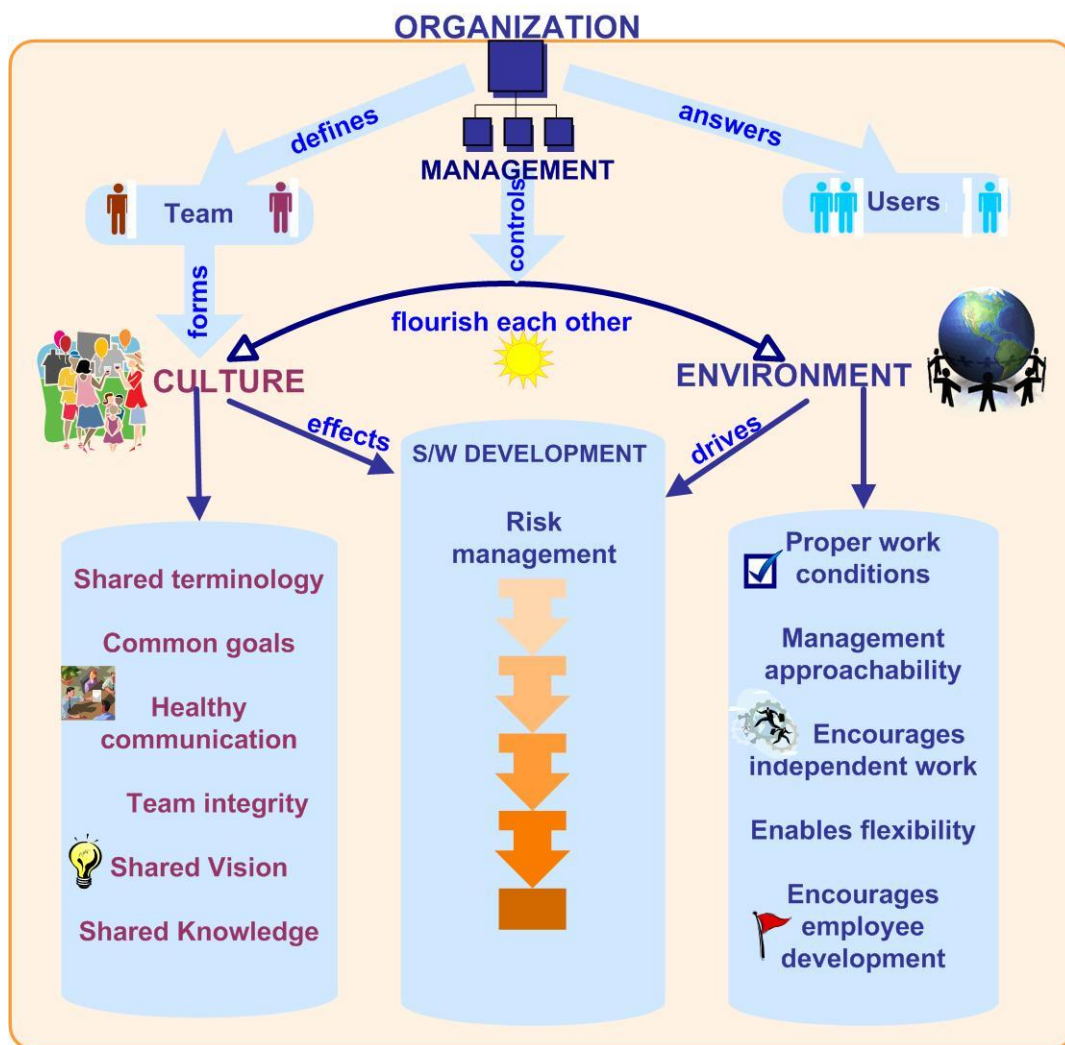


Figure 7.1: Software Risk Management - Dependence on the Organization Culture

[♦] All unreferenced figures and tables are self-created.

A project team, like a football team, is a cultural entity. Project culture is significantly influenced by organizational culture (as supported by Infosys). A key finding, also encouraged by Mr. Madakkattel in his interview that, if software engineers have little confidence in management, struggle with computer aided software engineering (CASE) tools, have no software process to follow, or see schedules as pure fiction, a project attempting to be strong in these areas faces an uphill struggle. As quoted by Henry (2004) “the first step towards shaping the culture if a software project is to understand the culture of your organization”.

In order to build the framework and discover not so perfect but a healthy culture, which smoothens the software development process by handling risks the way it handles coding, it is imperative to understand cultural requirements in the process of software development from conception till completion.

There might be organizations with not so strong infrastructure due to financial limitations, small set up or many similar reasons. However, the leader or the manager could still consider the framework and enjoy dealing with the risk (with a learning attitude) rather than struggling with it, because in any new project, especially software, *risks are inevitable*. After a comprehensive analysis of research findings, a support framework has been worked out in the following sections, which explain risk handling in a healthy culture during a software development life cycle.

Various subjects on software risk management and organizational culture were explored and analyzed till now. The diagram hereunder outlines the areas covered in previous chapters and the approach adopted in order to develop the solution. The aspects mentioned in the figure have been taken into account while developing the solution framework (in section 7.3).

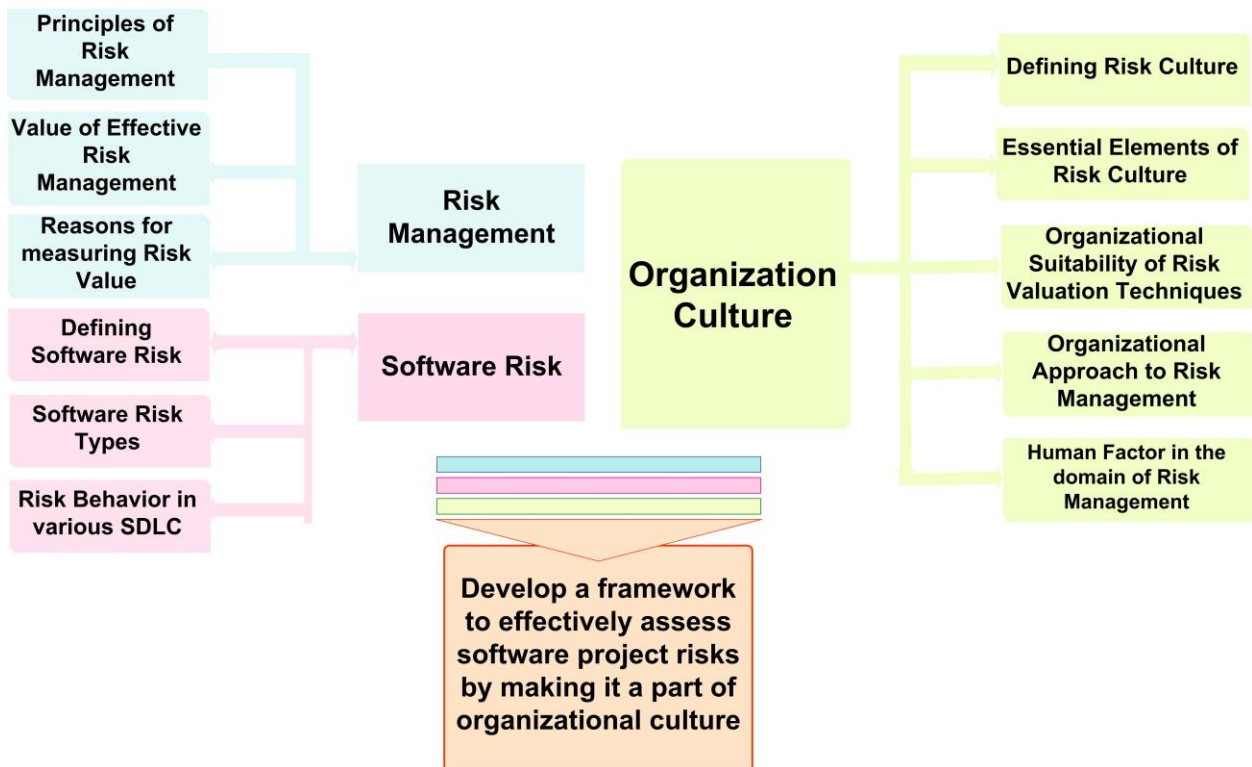


Figure 7.2: Approach to Solution Development

7.2 Software Risk Management Culture Development – A Mind Mapping

Before going ahead with developing the intended framework, following mind mapping has been performed after a thorough analysis of research findings, in order to envisage a culture, which helps in risk management during software development. Likewise, it is highly recommendable that an organization should spend a few hours in analyzing and brain storming these aspects of its environment, which will help in identifying very small mistakes which could cause big risks.

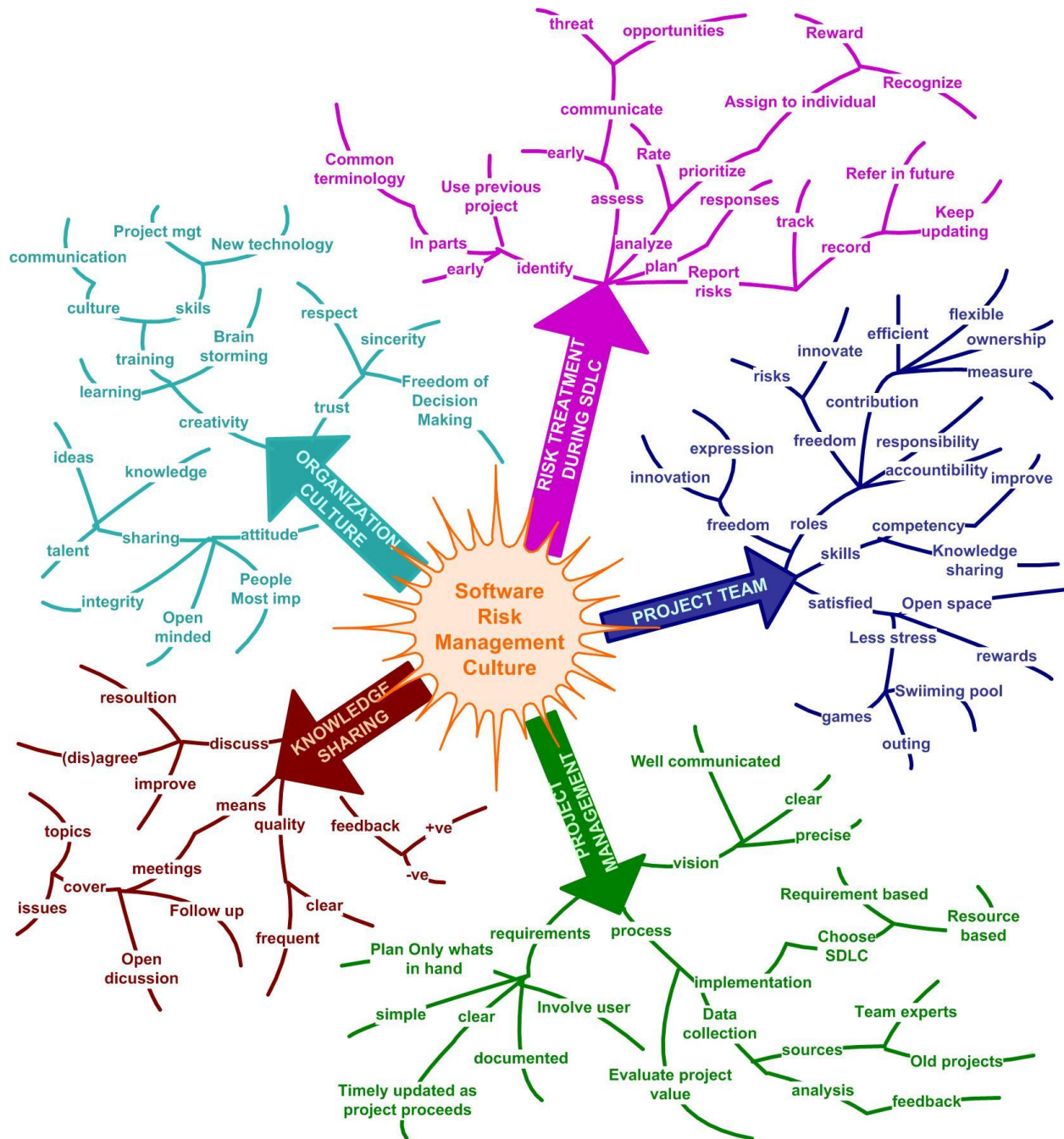


Figure 7.3: Aspects Considered while Developing the Framework

7.3 Solution Development

Discussion till now leads to the first derivation that *Planning and Design* phases in Software Development and *Identification and Analysis* phases in Risk management decide the fate of a project. Carefully looking at any SDLC, the development phase may involve same coding language, same process, however the successful completion of any software project is highly dependent on how the team understands it and how it plans for the tasks to be carried out and risks to be treated. Although there are numerous other factors which trigger risks, however the most of work gets completed when the team and the project manager feels confident about taking the project ahead by planning tasks from requirements and identifying risks from the tasks.

The second most important derivation is: risk management is not just one time function; it begins with the project planning and goes till implementation. The way successful projects suggest testing a piece of code at least 3 to 4 times a day, similarly Risk management needs to be carried out at least 4 to 5 times a week. According to Standish Group (2007), one of the greatest risks which has not been discussed in the previous chapter is the sales and marketing people crunching numbers due to sales pressure. However, this study focuses more on a project contracted by a client for specific requirement, rather than simply developing it for sale in the market.

According to a global study conducted by IBM (2008), majority of organizational change projects failed in 2008. This “Making Change Work Study” divulges that nearly 60 percent of projects aimed at achieving business change fail to meet their goals fully, but that the most successful organisations - the Change Masters – have cracked the code and succeeded in 80% of their projects. In sharp contrast, the bottom 20 percent of the sample, referred to as Change Novices reported a project success rate of only eight (8) percent. This data proves that people are reluctant to changes in operations. However an organization needs to identify the reception level of its staff before implementing a change. This also implies that the change is meant for the business, but not the business for the change. Therefore careful study of existing situation (in the organization) is required before implementing any change and same applies for a SDLC. Working in an uncomfortable environment will only lead to negative outcome. Also inducing unnecessary changes at the cost of team discomfort or project failure is an unwise decision. Consequently, the next section discusses various steps in order to identify an adaptable risk management process, depending upon project type, organization type and employee type. It is not necessary that implementing this solution will ensure a 100% success, but this definitely ensures a project not falling into a trap of failure, when sometimes it becomes impossible for the team to carry the project ahead for various insignificant factors. Following support framework aims for ‘eliminating many of these insignificant factors’ which in later stages of development may cause major risks. As the risk management in general is known to all, also it has already been discussed in previous chapters, the following sections will emphasize on the “working style” or the *Cultural Factors*, which will help in managing risks during software project development process.

Cultural forms facilitate how people make sense of their world (Ovaska, 2008). Consequently, this section discusses various steps in order to identify an adaptable risk management culture, depending upon project type, organization type and employee type. In order to develop a framework which supports a healthy culture, we need to work on the cultural aspects of risk management process during software development life cycle, irrespective of small or big organization, or be it a free lancer. This will be accomplished with the help of research findings discussed in previous chapter, also with the help of literature review. By looking at these two scenarios; the software development process is apt but the risks are not handled properly or there is a full proof risk management process with an inefficient software development knowledge, it is apparent that the chances of project failure are high. Therefore, a list of risk management guidelines (following page) has been discovered and encouraged to be followed all through the SDLC. Furthermore, a SDLC culture has been explored and discussed, in order for suggested risk management guidelines to be a success. As both are necessary for each other and complement each other.

7.3.1 Risk Management Guidelines – Superior Culture.

Risk Management Process

Step 1. Identify Risks

- Identify on the basis of complexity.
- Team brain storming – empowers trust.
- Study previous projects.
- During team discussion, include at least 1 person with experience in similar project.
- Identify through out SDLC.
- Spot risks during requirements gathering.
- Assign one “risk tracking” each to a team member.
- Use risk terminology familiar to all.
- Give freedom to team members while risk definition.
- Also spot success factors.

Step 2. Analyze Risks

- Analyze only what is most important at present.
- Analyze tasks which may cause risks.
- Don't spend too much time as time is precious.

Impact	Catastrophic	. Risk A	. Risk F	. Risk D
	Significant		. Risk C	
	Moderate	. Risk B		. Risk E
		Process risks	Platform	Business

Step 3. Plan and Resolve Risks

- Prioritize on the basis of current situation, don't over plan.
- Keep eyes open for time and resources needed, it should not out weigh the benefits.
- Discuss more and more about the issue, during informal meetings.
- Research on similar risk planning if the situation is difficult.
- Employ prototyping – can help foreseeing more risks.
- Take a look at staffing decisions – to be changed or revised.
- Always monitor quality, it should always increase the project quality.
- Focus on 20% of most likelihood risks which cause 80% of problems.

Step 4. Control Risks

- Clarify ownership issues and award for risk handling.
- Reward for risk avoidance.
- Build a work habit where technical lead & developers must be able to state their top 3 risks at every stage of risk identification.
- Use only those risk control measures which simplifies the already difficult development process.

Step 5. Treat Risks

- Make a log of risks solved for future reference and new members.
- Track risks with associated tasks and make sure they don't re appear. This may be of great help on the organizational level.

Figure 7.4: Superior Cultural Aspects of Risk Management Process

7.4 How a Good SDLC Culture Manages Risks

After working out a particular way of handling risks, let us further work on the software development process of a new project through implementation of the (above) risk management guidelines at every stage. The findings of case studies, surveys and the interviews have been of great help in defining these guidelines.

7.4.1 Project Planning and Requirements Gathering - Identify Risks

Three main sources of gathering requirements and identifying risks are people, pen and paper. People are the key team members who are equipped with previous experience and expertise. Other people to consult are experts outside the project, who have hands on experience with similar kind of work you are undertaking. Paper here is referred to, the project plan, business case and resource planning documents. Pen involves tools and the thought process which will help accomplish above two requirements.

Following steps, if followed in the first stage of project planning may possibly avoid several risks, right in the very beginning.

- When the project team starts gathering requirements, it should **focus on defining the purpose** of the system rather than the technical specifications. This improves system functionality understanding. (view supported by Mr. Pinto in his interview)
- **Freedom of decision making and participation** should be provided to the team in the beginning because it is the team who handles the risks during later stages.
- Project Manager or the Project team should have a **face to face meeting** with the user in order to get more clarity on requirements by dividing requirements into following three categories and begin with most important features. Better the understanding, easier is to deal with scope changes. My personal experience of a web development project also supports this view point, that whenever there used to be a team meeting, issues of 10 days used to get resolved in a few minutes.

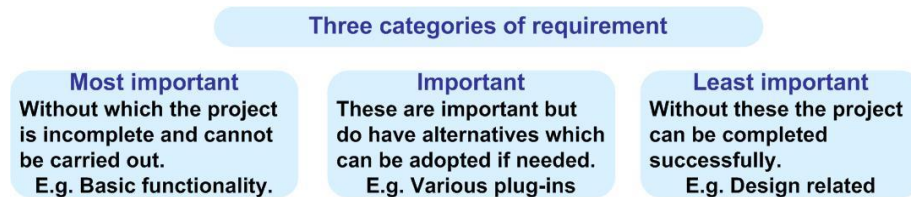


Figure 7.5: Requirements Categories Summarized

- **Planning for only what is in hand** helps to a great extent (Agile rationale). Many times a project team waits until all the requirements are documented, and then starts the process of designing, which causes schedule over-run.
- **Treating people as most important** is very crucial during this phase, which involves the user and project team, as the entire development process runs on their mercy. Therefore it is important to see them as most important asset, because sometimes it is never the method or the money, but the **man kind** who can prove dreadful to the project. As per the Infosys case study, it is beneficial to assign ownership of risks to people, so that they can help in timely optimization, moreover it is easy to track risks. Make arrangements for proper work conditions and ways of managing stress. (Agile Rationale)
- **Select an efficient Project Manager (PM)**, who plays a crucial role in the success of a project. For instance, say a PM is happy about choosing an efficient team, but not as technically sound, he may miss on very important steps due to lack of knowledge and can cause harm to the project. (View supported by Mr. Pinto in his interview)
- **Risk identification team**, (as adapted by IBM) depending upon the resources allocated to the project, may include whole organization or a small-assembled risk team. However, it is recommended that in order to gain a positive outcome, it should include department heads, the creative team, crisis team and the senior management (in case of highly complex projects).

- **Brainstorming requirements** with the team can expose some hidden **threats** one might encounter or some golden **opportunities** that are coming up on the way (IBM and Microsoft). Interviews, team sessions (risk brainstorming) are some common techniques to accomplish this. A helpful method is to put the participants into different teams to identify the project risks, prior to engaging in a group discussion. As suggested by Krigsman (2008), identifying risks in categories (Figure 7.6), will help in their better prioritization.

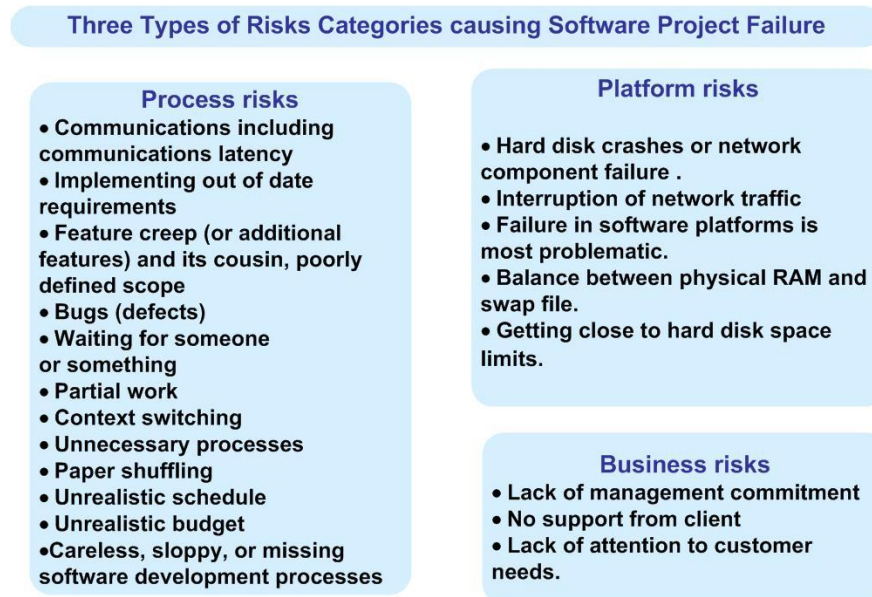


Figure 7.6: Risk Categories by Krigsman (2008) Summarized

- Planning **Work break down structure** is a good way of tracking tasks and therefore risks.
- **Project value** should be evaluated for complex projects with the help of ROI, IRR, and NPV, which actually measures the **risk value** for the project during requirements gathering stage.
- **Provide examples of the value of cultures**, using various academic and corporate sources (Articles and presentations available from websites) backed up by structured thinking on the subject of risk management by the team. They should be given equal opportunity to be part of risk identification process, so that during the development process, the member is still aware of the initial situation. The main aim of this stage is to develop a firm foundation, so that the risks being identified are based on structured thinking. (View also supported by MSN Framework from Microsoft)
- **Selection of a SDLC** should not depend on its “name”; rather it should depend on the level of completeness and understanding of requirements (personal experience). Say, requirements are complete and a project is not so complex, a team can choose to implement waterfall SDLC. However if requirements are expected to come in iterations, any Agile technique is the best.
- The requirements gathering process should be **kept simple, keeping in mind a vague idea of how your client visualizes the system**. For example, unnecessarily using difficult tools, UML, difficult architectural design may delay the project if any team member or the client is not able to understand.
- **Sound development principles** should be employed. Say high quality coding standards, organizational training, and training to increase knowledge of the developer. It will not only develop him as a professional, but the entire project and the organization will benefit from it. For instance implementing generic coding standards will make future reference easier by new members or just in case the existing member leaves for any reason. (Personal experience)

7.4.2 Project Design – Assess and Analyze Risks

During this stage of SDLC, following ways of performing tasks have been worked out, which will encourage better implementation of risk management.

- **Categorize identified risks** during project planning in such a way, which helps in prioritizing their treatment. For instance, a color should be assigned to each risk depending upon risk category on x-axis, and y-axis should measure the degree of complexity each risk carries. Each risk should be placed under the three risk areas identified as in figure 7.4. If a risk falls under more than one area both the colors should be assigned (see previous section, Analysis stage of Figure 7.4).
- **Brainstorm with the team**, by looking at their previous experiences, project plans, the company Intranet, and assistance websites, then design for basic functionality with the existing data.
- **Focus on quality** by encouraging team through offering rewards and titles. Quality design helps in treating almost 25% of risks. (Microsoft's perception of project risks)
- **Empower team** by providing them an opportunity to express views and suggestions, which again improves the 'process quality' and saves time.
- **Design with keen insight** and spend more time than development stage. This will **save time** by encouraging lesser testing. (Agile principle)
- **Design test plan** at this stage, so that testing doesn't drag till completion phase which avoids schedule over-run.
- **Involve software engineers** during architectural design stage, which will help them deal better if requirements change in future.

Understanding the nature of a risk is a prerequisite for a good response. Therefore take some time to have a closer look at individual risks and do not jump to conclusions without knowing what a risk is all about. Risk analysis occurs at different levels. If one wants to understand a risk at a particular level, it is most fruitful to think about the effects that it carries and the triggers that can cause its occurrence. Looking at the effects, one can describe what effects take place immediately after a risk occurs. A more detailed analysis (Figure. 7.7) may show the order of magnitude of an effect in a certain 'effect category' like costs, lead time or product quality. Another perspective to look at risks is to focus on the events that precede a risk occurrence. List different causes and the circumstances, which decrease or increase the likelihood of risk occurrences. The information gathered in risk analysis will provide valuable insights to your project and the necessary inputs to find effective responses to optimize the risks.

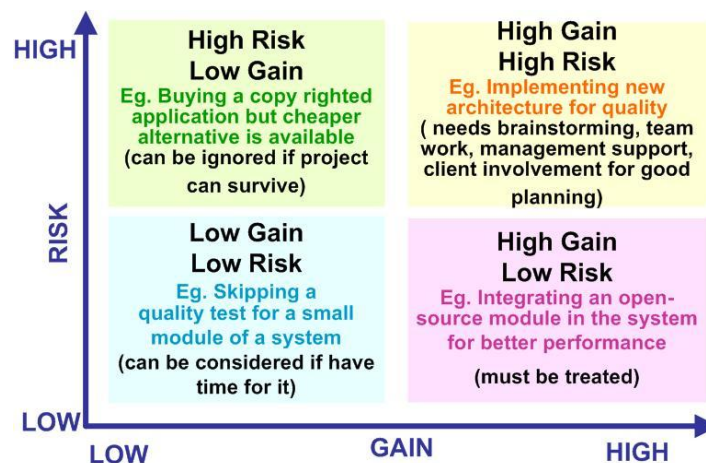


Figure 7.7: Analyzing Which Risks to Respond

7.4.3 Project Development and Testing -- Handle risks

This is the phase when project visibility improves and all the requirements planned and risks identified take a physical form. The testing phase has been fused with development phase because a good work culture is being discussed throughout the study, which helps reduce the risk occurrences in a SDLC. Let us discuss this stage for a better risk treatment.

- When a project enters the development phase, risks are also visible, it is very important to keep **communication on** about risks being encountered. Frequent meetings should be conducted in order to handle them. Make project risk discussion a part of the meeting agenda by default, which will further help in identifying new risks or opportunities. (survey findings strongly support this view)
- **Test, test and test** while developing. It is the key to quality and an effective time and cost management. Also most of the times, it is actually impossible to code without testing. Doing this automatically controls many risks and system maintenance starts much before its handover.
- Make **use of old projects** to check how similar risks were handled. (personal experience)
- **Prioritize risk responses**, keeping in mind that responses should not be handled at the cost of project budget.
- **Spot opportunities:** Risk management does not mean only avoiding risks but also helps in spotting opportunities for more financial gains (Agile technique). For instance, there might be times when a developer comes up with a functionality which carries an enormous financial value.
- **Report to the customer** on completion of every module (if the project is big), or at least 5 times during the entire project life (if small). Because there are many project managers or relationship managers (who communicate with the client regarding the project's progress and) who show up only after the project is complete or when it is bound to fail (personal experience).
- **Manage senior developers with lot of respect**, because they might be older in age than the project manager (in many cases) and may not listen to him because of their wider experience. (personal experience)
- **An informal meeting during this phase is** the key to success, as the developers are under a lot of stress of programming. These meetings sometimes turn out to be of great help in cracking problems. (Survey findings support this view)
- Always **track against requirements and earned value** to check deviation from the set targets. This is one step further to risk management, because of which most of the projects surveyed were successful.
- **Test with stability and integrity.** Which means the system should not only complete but run smoothly after deployment. Because risks will be more frightening if they appear after completion.
- **Do not over work**, as taking care of team members' ease is also necessary. For the simple reason of it being an intellectually challenging field, mind needs to rest. Therefore, focus should be more on completing the functionality and less on counting the number of lines coded. (personal experience)

7.4.4 Project Implementation – Risk Treatment for Future Reference

With the end of development phase, risk management is not over. The ‘good culture’ being discussed tries to deal with:

- ⇒ Discovering risks – for treating them.
 - ⇒ Playing with risks – for assessing, analyzing, controlling.
 - ⇒ Preserving risks – for future reference.
- **Tracking risks** and their associated tasks can be carried out with the help of the risk register. Incorporating risk tasks into daily routine is the key to implement responses. **Good system maintenance** ideally starts with the development stage, when the quality is kept in mind while performing tasks, where developing a running software is the aim of the project team as compared to just meeting the technical requirements. (proposed by Microsoft)
 - **Registering Project Risks** will enable the team to view progress and ensure that risks are not forgotten. It is also a good technique to inform team members and stake holders about what is going on.
 - **Maintaining a good risk log** clarifies ownership issues, provides risks descriptions, and enables carrying out some basic analysis regarding causes and effects. Some project managers do not want to record risks, as they may be blamed if anything goes wrong. However the opposite is true, as there is a precious possession with the manager which cannot match up with anybody else’s experience of risks. Hence he can refer to it anytime because doing projects implies taking risks.

Therefore risk management involves not merely preventing or controlling risks, but it is a way of widening the area of knowledge of one’s expertise in the task one is performing. Hence, the risks are not to be scared of because the synonym for “software project” is “risk undertaking”. Therefore, one should take pleasure in risk handling with a learning attitude.

7.5 Evaluation

The tools and methodologies used in order to demonstrate this framework are a sound analysis and compilation of existing tools and techniques from a variety of sources. The main sources of these tools are from the techniques developed and espoused by IBM, Microsoft, Infosys, NASA, ChangeSource private Ltd., Agile, various software development life cycles and literature review of various companies adopting risk management techniques in their own unique styles and most importantly the findings of surveys, case studies and interviews.

Unfortunately it was not feasible to develop a fully irrefutable method. This model covers the significance of culture (way of working) during risk management process of a software development project. More effort is vital to develop this area. Additional work needs to be carried out on the front end of the cultural model of a software setup (culture audits, dimensions and needs) and also at the end of the SDLC process discussed above. This involves developing measurable outcomes of the cultural risks and performance methods. Furthermore, cultural crisis management could be a beneficial area of study to find out better ways of working in a software set-up.

Chapter 8

Recommendations and Conclusion

"He, who doesn't risk, never gets to drink champagne."

-- Russian Proverb

8.1 Review & Recommendations

The primary objectives of this study were to identify the limitations of current risk management culture in software development organizations and develop a solution in order to come up with a better culture. Above discussed solution is expected to help in a way that even though project faces threats over and above the managing ability of the team, they should be able to handle the risks bravely and finish the project with minimum possible deviation.

Universal View

There is an unequivocal issue which needs to be addressed. We have some widely discussed facts about corporations accepting and believing in the importance of culture. Some of them, even if small, implement various techniques to have the best possible culture. There are some key players in the field of software risk management such as Microsoft, IBM which emphasize on risk management and culture development. Yet, many see it as an overhead and ignore it after struggling for a while. However, they ignore the fact that, accepting and introducing a good culture is a one time task, and it will keep benefiting till the organization is alive. Software industry is growing at least 10 times faster than any other industry. Statistics show that by 2015, the number of computers in the world will reach around 2 billion. This simply proves that we need to take software business as seriously as any other business, by carrying out a deeper study on failed projects, but this time also considering the aspect of “organizational culture” along with other key measures.

Organizations

For the only reason that software development hardly involves any field job, and employees sit at one place and perform their tasks, does not mean that they do not need the same care and attention as in other industries. When parents get their child admitted in a new school, they investigate and make sure that the child learns all the important lessons in his/her life to be a successful person, which also embraces good values and manners, moreover being cultured, because grades alone will not help in building a successful career and a satisfactory life. Similarly any new project is a child to its initiator or the stakeholder, who in most cases sees only the process aspect and ignores the ethics aspect during the development life cycle, which decides the quality of a software, hence the quality of an organization. For this reason, it is easy to figure out just by looking at a running software; how smooth or rough its development would have been.

Although, many software companies have raised issues in relation to culture within their organizations, they do not seem to be benefiting themselves, except for a very few of them. Organizations need to:

- ⇒ Seek help in the areas in which they possess limited knowledge.
- ⇒ Encourage internal communication, chiefly amongst culture and risk management. There might arise some conflicting outlook towards this issue, however this is the best way to enable both the groups to work collectively and develop a successful solution of their own, depending upon their company structure, beliefs and management ideas.
- ⇒ Create in-house understanding of measuring project value, hence the risk complexity. Make sure the staff easily understands the culture guidelines and management expectations.
- ⇒ Educate senior management about the value of culture and the need for risk management to be integrated with development process, right from day one.
- ⇒ Common to most projects is the lack of apposite and transparent communication. Team members (and other stakeholders) often do not share a common understanding of the project's goals and strategies. It is important to unveil these misunderstandings and hidden agendas from the very beginning.

Many senior executives clearly understand the significance of culture, but they are in minority. If senior management does not feel the need to invest in building a healthy culture, it is quite unlikely to expect success from poorly executed projects.

Risk Management

There have been many modifications to how risk management is conducted. The most remarkable is the emphasis placed on project risk management in recent years. However, it mostly takes into account the aspects like Information Practices (IP) and copyright issues. It is still limited to tangible product business. Although there is enough written about risk management, and risk in software projects where the main aspect of organizational culture gets sidelined. The creative sector should take the responsibility of treating this issue and developing a common guide, which can easily be manipulated and adapted by organizations depending upon their requirements.

Project Managers

The point made by Ewusi-Mensah (2003) is very interesting as it touches on two aspects which have been discovered repetitively within his study on software projects with respect to organizational environment - 'management perception, commitment and pressures'. Software development project managers need an approach, methodology, and vocalization to evaluate the economic value of challenging ventures in order to allocate resources properly because outsiders such as consumers, sponsors, shareholders, and auditors by and large believe that vital business decisions are institutionalized on valid and fair valuation techniques.

8.2 Conclusion and Future Work

While developing the solution it was important to understand, that unless the fundamentals are resolved, the solution will never be efficient or accredited. The creative team or the creative person of an organization needs to take control of these issues. Their expertise does not lie in the risk management or deploying a particular tool, but more in their deep understanding and knowledge of business needs and the knack of developing various measures which concern the project. Thinking of an old saying, "fail to plan, then, plan to fail", organizations need to overcome this vicious cycle of "not" planning with an open mind and open eyes. All what creative team needs to accomplish is – simplify communication, build a team spirit and provide organizations with thoughtful culture of working towards achieving a *win-win situation* (Covey, 1989). As discussed in the literature, there are numerous external risks to be monitored and fixed, so why not make life (within the organization) easier by correcting loop-holes, which are inherent in the organization and in the hands of people who cause them.

Considering the recommendations listed in the previous section, it is clearly understood that there are many other issues needed to be resolved before coming up with a culture in an organization which helps in successful implementation of software projects. During the initial researches on *software failure*, it was assumed that the main issue is lack of tools and techniques being implemented for risk management, however it is not the main issue. There are numerous tools available at a project team's disposal; it is their maturity while choosing them which makes a difference at the end. The perceptions and outlooks towards risk management activities become the root of tough challenges encountered while applying numerous available risk management strategies.

Due to time constraints, it was not easy to access the solution devised against a software project success measures as defined by Software Engineering Institute. Everyone is aware of how a project can be defined as a success, although not every project is successful. However, analyzing cultural requirements of an organization and finding a solution was the aim of the study. In order to conform to the cultural guidelines, the tool has to comply with Project Management Standard.

In order to develop a more effective tool for building Risk Management Culture in a software setup, a wider study is necessary. There are various underlying constraints which need to be addressed. Ngwenyama and Nielsen (n.d) are working with an objective and analyzing the assumptions based on organizational culture that are embedded in the CMM (Capability Maturity Model – A model developed by the American Software Engineering Institute (SEI) in cooperation with Mitre Corporation on November 1986). Future research should establish both convergence and variation between the recommended propositions and innumerable strategy measures of culture, through pragmatic analysis. It is however challenging to establish these aspects of validity, as the current scenario of software project validity of risk culture literature is hazy. Simply because in spite of extensive research, 1000s of books, tools and techniques on risk management, culture management and software management, there is still not a definite measure, which is accessible to the corporate world. However as underlined in the above mentioned recommendations, future researchers need to enlarge sample size, in terms of the number of organizations modeled and the number of employees studied within each organization, in order to boost the findings and incorporate numerous perspectives of an organizational culture.

References

AFSN, 2008. *AFSN Organisational Culture Survey*, [online]. Available at: <http://futuresearch.net.au/australian-future-search-network-organisational-culture-survey.html> [Accessed 4 Nov 2008].

Barton, L.Thomas, 2002. *'Managing Risk: Enterprise wide approach'*, Printice Hall.

Boehm, Barry & Port, Dann, University of Southern California and Kevin Sullivan, University of Virginia -- *White Paper for Workshop on New Visions for Software Design and Productivity: "Value Based Software Engineering,"* available as a MS Word document at: [http://www.isis.vanderbilt.edu/sdp/Papers/Barry%20Boehm%20\(Value%20Based%20Software\).doc](http://www.isis.vanderbilt.edu/sdp/Papers/Barry%20Boehm%20(Value%20Based%20Software).doc) [Accessed 3 Jan 2009].

Burchfield, Nancy, 2002. *Assessing the economic value of software projects*. IBM Library, [online article], 15 Nov. Available at: <http://www.ibm.com/developerworks/rational/library/nov05/mckenna/> [Accessed 5 Dec 2008].

Change Source Pvt. Ltd., (n.d.) *Risk Assessment for Projects*, [online]. Available at: <http://www.change-management-toolbook.com/mod/book/view.php?id=74&chapterid=65> [Assessed on 15 Dec 2008].

Charette, N., Robert, 2005. *Why Software Fails*, [internet], September. Available at: <http://www.spectrum.ieee.org/computing/software/why-software-fails> , ITABHI, [Accessed 17 Nov 2008].

Cole, Andy (1995). "Runaway Projects: Cause and Effects," *Software World* (UK) Vol. 26, No. 3

Covey, R. Stephen, 1989. *Seven Habits of Highly Effective People*. 1st Edition. Free Press: USA.

Dann, Jill & Broome, Annabel & Joyce, Wendy, 1998. Human Barriers to project success. *Oxford Journals ITNOW – British Computer Society*. Vol. 40, No.3, P:24-26

Debenhams, Mike & Hutchins, David, 2003. Barclay Simpson - Company Profile, *The Risk Factor*, [online]. Available at: http://www.barclaysimpson.com/about-barclay-simpson/?nodeid=29&session_id={65385ff6-e771-4045-97a7-e577f7554204} [Accessed 29 Nov 2009].

Deshowitz, Nachum, 2007. *Software Horror Stories*, [online]. Available at: <http://www.cs.tau.ac.il/~nachumd/horror.html> [Accessed 5 June 2009].

Etuzun, 2008. *Risk Management in Software Development Projects*, [online Article]. Available at: <http://www.withrisk.com/index.php?kat=4&yazi=3> [Accessed 2 Jan 2009].

Ewusi-Mensah, Kaweku, 2003. *Software Development Failures* . Cambridge:MIT Press
Folleth, Connie, 2002. *What is SDLC and what is it used for?* Essortment, [online article]. Available at: http://www.essortment.com/hobbies/sdlcsoftwaresy_svkr.htm [Accessed 11 Nov 2009].

Gale, Cengage, 2008, Culture Performance Research – challenges and Future Directions, *Journal of Academy of Business and Economics* [online library]. Available at: <http://www.thefreelibrary.com/Culture-performance+research:+challenges+and+future+directions.-a0192852612> [Accessed May 5 2009].

Goldstein, Harry, 2002. Who Killed the Virtual Case File? *How the FBI blew more than \$100 million on case-management software it will never use*, [Online Case File]. Available at: <http://www.spectrum.ieee.org/sep05/1455> [Accessed 3 Dec 2008].

Gotterbarn, Don, 2002. *Reducing Software Failures: Addressing the Ethical Risks of the Software Development Lifecycle*. Australian Journal of Information System Software Engineering Ethics Research Institute. East Tennessee State University, USA, Vol.9, No.2.

Griffiths, Mike, 2006. *The Top Five Software Project Risk*, [online article]. Available at: <http://www.projectsmart.co.uk/developing-a-high-performance-project-team.html> [Accessed 29 Jan 2009].

Hennessy, Joe, 2004. *ISD Software Risk Identification*, [online], 1 Sept. Available at: <http://74.125.45.132/search?q=cache:V8OL4ATXGJAJ:software.gsfc.nasa.gov/AssetsApproved/PA1.2.3.pdf+software+risk+identification&cd=1&hl=en&ct=clnk&gl=us> [Assessed on 25 Dec 2008].

Henry, Joel, 2004. *Software Project Management*. 1st ed. Montana: Pearson.

Higuera, P. Ronald & Haimen, Y. Yacov, 1996. *Software Risk Management*, [online book], June. Available at: <http://www.sei.cmu.edu/pub/documents/96.reports/pdf/tr012.96.pdf> , Pittsburgh, [Accessed 19 March 2009].

IBM, 2008. *Winning in Midmarket:100 Software Success Stories* [online book], June. Available at: <ftp://ftp.software.ibm.com/software/smb/casestudies/expresscasestudy.pdf> [Accessed Jan, 2008]

IBM Corporation, 2006. *PRJ480 Mastering the Management of Iterative Development, Post Project Analysis*, [online article]. Available at: http://www.zsk.p.lodz.pl/~mkwapisz/zppi/courseware/prj480/rp601qv2_slides_08.ppt#259 [Accessed 20 Dec 2008].

IBM Corporation, 2008. *IBM Global Study: Majority of Organisational Change Projects Fail*, [online], 15th October. Available at: <http://www.ibm.com/news/hk/en/2008/10/15/s921892u38298o66.html> [Accessed 5 May 2009].

ITcortex, 2001. *Failure Examples of IT projects*, [online article]. Brussels. Available at: http://www.it-cortex.com/Examples_f.htm [Accessed 5 Nov 2008].

Jiang, J.J., Klein, G., 1999. *Risks to different aspects of system success*. Information and Management 36 (5), 263–272.

Jones, Capers, 2005. *Software engineering : the state of the art*, [online Paper]. Available at: <http://www.compaid.com/caiInternet/casestudies/capers-stateofart2005.pdf> [Accessed 20 Jan 2009].

Jones, R., 2003. *Quality World* Emeralds Group Publishing. Vol.7, No.29

Pritchard, Carl, 2007. *Creating a risk culture in an IT environment*. Cutter Consortium Press Release. Vol.2, No.1

Krigsman, Michael, 2008. *Three Risk Categories that explain IT failure*, [online White Paper], 14th Feb. Available at: <http://blogs.zdnet.com/projectfailures/?p=598> [Accessed 5 March 2009].

Kwak, Y.H. & Stoddard, J., 2004. *Project risk management: lessons learned from software development environment*, [online White Paper], p4. Available at: http://home.gwu.edu/~kwak/Project_Risk_Management.pdf [Accessed 12 Feb 2009].

Lewallen, Raymond, 2005. *Software development life cycle models*, [online article]. UK. Available at: <http://www.codebetter.com/blogs/raymond.lewallen/archive/2005/07/13/129114.aspx> [Accessed 19 Nov 2008].

Loon, V.Han, 2004. *Initial survey results: Surveying the status of organizational culture, improvement and innovation in the SPICE and Knowledge Management communities in 2004*, Knowledge board, Nov 04 [online]. Available at: <http://www.knowledgeboard.com/cgi-bin/item.cgi?id=504&d=23&h=5&f=3> [Accessed 5 Nov 2008].

Malia, Adil, 2006. *The Role of HR in Organization Culture Building*, [online speech ppt]. New Delhi: FCCI. Available at: <http://www.ficci.com/media-room/speeches-presentations/2006/speeches-2006.htm> [Accessed 17 Dec 2008].

McKenna, Patric, 2005. *Accessing the economic value of s/w projects*, [online article], 15 Nov. Available at: <http://www.ibm.com/developerworks/rational/library/nov05/mckenna/> , [Accessed 23 Dec 2008].

McManus, John & Wood-Harper, Trevor, 2008. *A study in project failure* [Online article], june. Available at: <http://www.bcs.org/server.php?show=ConWebDoc.19584> [Accessed 21 Jan 2009].

McNamara, Carter, 2000. *Organizational Culture*, Free Management Library [online]. Available at: http://managementhelp.org/org_thry/culture/culture.htm [Accessed 8 May 2009].

Messmer, Ellen, 2007. *IBM spells out IT governance and risk strategy*, [online news article], 15 May, Available at: <http://www.networkworld.com/news/2007/051507-ibm-it-governance.html> [Accesses 19 Dec 2008]

Moore, A. Geoffrey, 1991. *Crossing the Chasm*. United States:Harper Business Essentials

Murphy, Cliff, 2000. *Reducing Risk and Increasing the Probability of Project Success*, [online article]. UK. Available at: <http://www.projectsmart.co.uk/reducing-risk-increasing-probability-of-project-success.html> [Accessed 18 Nov 2008].

Ngwenyama, Ojelanki & Nielsen, A. Peter, *Competing Values in Software Process Improvement: An Assumption Analysis of CMM From An Organizational Culture Perspective*, [online]. Hungary:Elsevier. Available at: http://www.download-it.org/free_files/file123456789101112131415161718192021222324252627282930313233343536373839404142434445464748Pages%20from%20Chapter%204.pdf [Accessed 15 May 2009].

Nonprofit Risk Management Center, 2008. *Risk Management Basics*, [online Publication], 30 Jan. Available at: https://www.riskinstitute.org/peri/index2.php?option=com_bookmarks&do_pdf=1&id=576 [Accessed 5 Jan 2009].

Non-profit Risk Management Center, 2007. *Risk Communication*, [online news article], 17 Jan. Available at: <http://nonprofitrisk.org/library/enews/2007/enews011207.htm> [Accessed 5 March 2009].

Organization Culture, 2007. *Sample Organization Culture*, *Organization Culture Journal* [online article]. Available at: <http://www.organizationalculture101.com/sample-organizational-culture.html> [Accessed 19 Dec 2008].

Ovaska, Päivi, 2008. *A Case Study of Systems Development in Custom IS Organizational Culture*. Springer Publication, US.

Pritchard, Carl, 2007. *Creating a risk culture in an IT environment*.Cutter Consortium Press Release. Vol.2, Noi.1

Poppendieck, Mary & Poppendieck, Tom, 2003. *Lean Software Development, An agile toolkit* [online ppt]. Poppendieck.LLC. Available at: <http://xpday3.xpday.org/slides/LeanTutorial.pdf> [Accessed 24 Dec 2008].

Pandian, Ravindranath, 2007. *Applied Software Risk Management*, Auerbach Publications:Newyork

Repario, 2006. *Assess Your Organizational or Team Culture!*, [online article]. Available at: <http://www.employeedevelopmentsolutions.com/cultureassess.htm> [Accessed 6 Nov 2008].

Schein, E.H. (1992). *Organizational Culture and Leadership* (2nd ed.). San Francisco: Jossey-Bass.

Scoy ,Van, Roger, L.. *Software Development Risk: Opportunity, Not Problem*, SEI, Sept 1992.

Simms, Jed., 2007. *Manage Your Project Risk Profile*, [online white paper]. Melbourne. Available at: <http://manageyourprojectriskprofile.com/> [Accessed 5 Nov 2008].

Sloane, Paul, 2008. *Lateral Thinking Skills*, [online article]. Available at: http://www.1000ventures.com/business_guide/crosscuttings/failure_freedom.html [Accessed 5 May 2009].

Stanciu, Adrian, 2006. *Managing organization culture*, [online ppt]. Houston. <http://humansyn.bg/conference/presentations/Adrian%20Stanciu%20-%20Managing%20Organizational%20Culture.ppt#271> , 4, Culture vs. structure [Accessed 27 Jan 2009].

Standish Group, 2009. *Standish Group Chaos Report finds increasing project failure*. [internet]. The Productivity Habit. Available at: <http://theproductivityhabit.wordpress.com/2009/05/04/standish-group-chaos-report-finds-increasing-project-failure/> [Accessed 15 May 2009].

Standish Group, 1995. *The Standish Group Report*, [internet]. Available at: <http://www.cs.nmt.edu/~cs328/reading/Standish.pdf> [Accessed 17 April 2009].

Swan, Brian, 2005. *Using Agile Software Development, part two*, [online article], 2005/03/15. Available at: <http://www.builder.au.com.au/strategy/projectmanagement/soa/Using-Agile-Software-Development-part-two/0,339028292,339180181,00.htm> [Accessed 24 Dec 2008].

Ulam, S. & Metropolis, Nicholas, (1949). The Monte Carlo Method, *Journal of The American Statistical Association*. Vol 44. No.247. (Sept), pp 345-351.

Verma, Charu, 2009. *Cultural Barriers to Effective Software Development*, *Software Development Journal* [online article], June. Available at: <http://www.devbazar.com/modules/news/article.php?storyid=820> [Accessed 23 June, 2009]

Wellnomics Ltd. Netherlands. *Wellnomics Risk Management*, [online]. Available at: http://wellnomics.com/pages/w_what.aspx [Accessed 5 Jan 2009].

Appendix A

RISK MANAGEMENT SURVEY OF YOUR LASTEST SOFTWARE PROJECT

(Answered by Mohammed Aasim, Senior Database Architect)

⇒ Name of the Organization (required)– Abu Dhabi Media Company

⇒ Organization was setup in (year) (required) – Year 2005

⇒ No. of projects Undertaken (approx.) – 4

⇒ No. of projects successfully completed (optional) – 4

⇒ Name of the last big project (required)– Seibel CRM

⇒ Size of the team (required) – 10 people

⇒ Note: simply write “y” or “n” for the following questions.

The Project Planning Phase

This questionnaire is a tool or checklist to evaluate an IT project proposal and to ensure that all factors influencing the success of a project have been taken into consideration in the planning of it.

Section I

Intuitively, how did you feel about this project?

Dangerous



Might succeed



Likely to succeed



Likely to succeed

Why?

(Ans). The managing director happily took the ownership of this project and was ready to support whenever and wherever required.

Section II - Support for the Business

Ask yourself:	Yes	No	Not sure
1. Was the project well aligned with the business plans and the Information Management Plans (IMP)?	Y		
2. Was an adequate business case analysis performed and the results well-documented and available?	Y		
3. Was the business case based on the full cost of the system from initiation through implementation and estimated annual cost of operation?	Y		
4. Did the Project Sponsor, representing the users take accountability and responsibility for project scope and definition?	Y		
6. Were there formal, documented plans in place to involve appropriate levels of users throughout the project (from requirements definition, through evaluation and acceptance of deliverables, product and integration testing to final acceptance and sign-off)?	Y		
7. Was the project justification based on an ROI with an attractive projected return?	Y		

Section III - Accountability

Ask yourself:	Yes	No	Not sure
1. Were the positions of project sponsor, project leader and project manager filled?	Y		
2. Were there clearly defined, documented and understood responsibilities, accountabilities and authorities for each of these positions?	Y		
3. Had the project manager identified adequate resources to allocate to the scheduled tasks at the scheduled time?			NS

Section IV - Corporate Project Manager Discipline

Ask yourself:	Yes	No	Not sure
1. Did the project manager (Crown's and/or contractor's side) have sufficient control over appropriate project resources?		N	
2. Was the Project Manager prepared to escalate issues when warranted, based on pre-determined criteria?	Y		
3. In the event of serious problems, were required decisions apt to be taken?	Y		
4. Were the project independent of other projects which are presently underway (i.e., not relying on the successful completion of other projects)?	Y		

Section V - Risk Management

Ask yourself:	Yes	No	Not sure
1. Was a formal process used to break down the work and estimate task duration?	Y		
2. Were formal mechanism or tools used to monitor the project schedule?	Y		
3. Were costs allocated in accordance with work breakdown structures?	Y		
4. Did team leaders sign off on cost & schedule estimates and were they held to budgetary constraints?		N	
5. Had all known management and technical risks been assessed and Were mitigation strategies in place for all identified risks?		N	
6. Did the project approach pass reasonable checks for what is to be accomplished?	Y		
8. Could the project manager and sponsor list the current top 10 project risks?	Y		
9. Was the technology being used well tested and had project staff sufficient experience in using it and knowledge of it?	Y		
10. Was it reasonable to expect financial stability for the duration of the project?	Y		

The Organization Culture

Ask yourself	Yes/no/notsure
1. Is it easy for you to arrange a meeting with your project manager?	Y
2. Does the Project manager hold regular meetings regarding the projects progress? (if yes then is it daily(d), weekly(w) or monthly(m))	Y
3. Does the Project manager hold meeting regarding personal issues with the team?	Y
4. Do you discuss your ideas in the meetings giving any suggestion for	Y

improvement?	
5. Is the organizational environment useful for the project?	Y
6. Do you consider yourself as a team worker?	Y
7. Do all team members share desire to meet	Y
8. Do you consider most of your team members as team worker?	Y
9. Are the procedures of any changes in project/ organization clear to you?	Not Sure
10. Does the management show commitment towards the team?	Y
11. Do you think that there is a major need for any change in the organization environment?	Not Sure
12. Do you think that your company environment positively helps in a successful project execution?	Y

The Project Implementation/Execution Phase

This questionnaire is a tool or checklist to evaluate the progress of an IT project and to ensure that all factors influencing its success are being closely monitored during its execution

- Support for the Business

Ask yourself:	Yes	No	Not sure
1. Had the Project Sponsor, representing the users, taken accountability and responsibility for keeping the project within scope?	y		
2. Had the Project Sponsor ensured user commitment throughout the project (from evaluation and acceptance of deliverables, product and integration testing to final acceptance and sign-off)?	y		
3. Had the business goals for the organization changed and if so, was the new system still aligned with the goals of the organization?		N	
4. Had the business case been reviewed and revalidated at each scheduled gate?	Y		
5. If project specifications had changed significantly, had these changes been well documented and approved by the appropriate stakeholders?	Y		

Section III - Accountability

Ask yourself:	Yes	No	Not sure
1. Did status/progress meetings occur regularly?	y		
2. Were issues raised and dealt with?	y		
3. Was this project on time and on budget?	y		
4. Were changes in scope being managed?	y		
5. Were adequate staff allocated to the scheduled tasks at the scheduled time?		n	

Section IV - Corporate Project Manager Discipline

Ask yourself:	Yes	No	Not sure
1. In the event of serious problems, had the necessary information been available to support decisive action?			Not Sure
2. Was the project manager avoiding premature, inordinate and inappropriate use of contingency funding?		N	
3. Was the project manager maintaining sufficient control over appropriate project resources?	Y		

4. Was the contractor providing complete information on project performance and progress?			Not Sure
5. Were project managers going to others in the organization for relevant knowledge, skills and experience for advice and support?	Y		

Section V - Risk Management

Ask yourself:	Yes	No	Not sure
1. Were the team leaders being held to budgetary constraints?		N	
2. Did the management approach continue to ensure coordination of all sub-projects, ensure communication among different sub-project teams and address shared horizontal issues?	Y		
3. Were internal/external peer reviews being held as scheduled?	Y		
4. Were oversight reviews by a senior steering committee being carried out at each gate?		N	
5. Were regular review sessions held to review the continued relevance of the project, project performance, and to raise concerns about actual/potential problems?	Y		

The Project Completion and Review Phase

This questionnaire is a tool or checklist to evaluate the success of a completed IT project and to ensure that all the factors have been evaluated and that any lessons learned are documented and fed to the continuous learning loop. It can provide valuable information during the project wrap-up session

Section III - Support for the Business

Ask yourself:	Yes	No	Not sure
1. Were the clients involved throughout the project from evaluation and acceptance of deliverables, product and integration testing to final acceptance and sign-off?	Y		
2. Did the clients carry through on their commitment to the level of effort required of them?			Not Sure
3. Was the client satisfied with the final deliverable?	Y		
4. Was the final product aligned with the goals of the organization?	Y		
5. Did the project remain within the predefined scope?	Y		
6. Were any changes to project specifications documented and approved by stakeholders?	Y		
7. Was the project consistent and compatible with the department's information and technology direction, strategies, architectures and infrastructures?	Y		
8. Was the business case reviewed and revalidated at each scheduled gate?	Y		
10. Was the business case reviewed and whenever there was significant change to the project or business function?	Y		

Section III - Accountability

Ask yourself:	Yes	No	Not sure
1. Was the project adequately staffed during all phases?		N	
2. Were issues with respect to weak or poorly performing personnel quickly and adequately addressed?			Not Sure
3. Was a project wrap-up session held to document 'lessons learned'?		N	
4. Did the project sponsor assume the responsibility for realizing the benefits predicted for the project?	Y		

Section IV - Corporate Project Manager Discipline

Ask yourself:	Yes	No	Not sure
1. Did the project maintain a positive image throughout?			Not Sure
2. Did project managers avoid premature, inordinate and inappropriate use of contingency funding?		N	
3. Did the department make use of an apprenticeship program to provide valuable experience to junior project managers?		N	

Section V - Risk Management

Ask yourself:	Yes	No	Not sure
1. Were all risks identified and managed properly?	Y		
2. Were the mechanisms and tools used to monitor the project schedule adequate?		N	
3. Were team leaders held to budgetary constraints?		N	
4. Was there adequate communication to all involved personnel throughout the project?	Y		
5. Did the contractor provide necessary information on project performance and progress?	Y		
6. Did the project reviews (including internal/external peer reviews) occur as scheduled?	Y		
7. Was there enough time, money and resources to get the job done right?			Not Sure
8. Were there adequate contingency plans for potential problems (i.e., no problems arose for which there were no contingency plans)?	Y		
9. Had the project complexity been accurately determined before start-up?			Not Sure
10. Were all changes analyzed quickly and brought to management's attention in a timely fashion?	Y		

 Signature –
 Name – Mohammed Aasim
 Designation – Senior Database Architect

Any Recommendations for a better Risk Management?

Theres a system architect tools along with Change & Requirement Management which helps significantly to the success of project with all the adequate precaution against the perceived risk.

Appendix B

Survey Questionnaire Report – Prepared in Excel

Corporate Project Manager Discipline					
The Project Planning Phase					
Support for Business					
1. Was the project well aligned with the business plans and the Information Management Plans (IMP)?	40	2	10		
2. Was an adequate business case analysis performed and the results well-documented and available?	36	0	16		
3. Was the business case based on the full cost of the system from initiation through implementation and estimated annual cost of operation?	35	0	17		
4. Did the Project Sponsor, representing the users take accountability and responsibility for project scope and definition?	38	3	11		
6. Were there formal, documented plans in place to involve appropriate levels of users throughout the project (from requirements definition, through evaluation and acceptance of deliverables, product and integration testing to final acceptance and sign-off)?	17	30	5		
7. Was the project justification based on an ROI with an attractive projected return?	32	8	12		
Accountability					
1. Were the positions of project sponsor, project leader and project manager filled?	40	1	11		
2. Were there clearly defined, documented and understood responsibilities, accountabilities and authorities for each of these positions?	39	2	11		
3. Had the project manager identified adequate resources to allocate to the scheduled tasks at the scheduled time?	37	3	12		
Corporate Project Manager Discipline					
1. Did the project manager (Crown's and/or contractor's side) have sufficient control over appropriate project resources?	38	2	12		
2. Was the Project Manager prepared to escalate issues when warranted, based on pre-determined criteria?	41	0	11		
3. In the event of serious problems, were required decisions apt to be taken?	38	3	11		
4. Were the project independent of other projects which are presently underway (i.e., not relying on the successful completion of other projects)?	32	11	9		
Risk Management					
1. Was a formal process used to break down the work and estimate task duration?	25	13	14		
2. Were formal mechanism or tools used to monitor the project schedule?	25	6	21		
3. Were costs allocated in accordance with work breakdown structures?	25	4	23		
4. Did team leaders sign off on cost & schedule estimates and were they held to budgetary constraints?	33	5	14		
5. Had all known management and technical risks been assessed and Were mitigation strategies in place for all identified risks?	22	9	21		

26	8. Could the project manager and sponsor list the current top 10 project risks?	33	7	12
27	9. Was the technology being used well tested and had project staff sufficient experience in using it and knowledge of it?	38	1	13
28	10. Was it reasonable to expect financial stability for the duration of the project?	39	1	12
29	The Organization Culture			
30	1. Is it easy for you to arrange a meeting with your project manager?	39	1	12
31	2. Does the Project manager hold regular meetings regarding the projects progress? (if yes then is it daily(d), weekly(w) or monthly(m))	35	3	14
32	3. Does the Project manager hold meeting regarding personal issues with the team?	10	42	0
33	4. Do you discuss your ideas in the meetings giving any suggestion for improvement?	34	18	0
34	5. Is the organizational environment useful for the project?	41	0	11
35	6. Do you consider yourself as a team worker?	48	4	0
36	7. Do all team members share desire to meet	44	3	5
38	9. Are the procedures of any changes in project/ organization clear to you?	32	3	17
39	10. Does the management show commitment towards the team?	39	4	9
40	11. Do you think that there is a major need for any change in the organization environment?	15	33	4
41	12. Do you think that your company environment positively helps in a successful project execution?	39	2	11
42	The Project Implementation/Execution Phase			
43	Support for the Business			
44	1. Had the Project Sponsor, representing the users, taken accountability and responsibility for keeping the project within scope?	37	3	12
45	2. Had the Project Sponsor ensured user commitment throughout the project (from evaluation and acceptance of deliverables, product and integration testing to final acceptance and sign-off)?	37	2	13
46	3. Had the business goals for the organization changed and if so, was the new system still aligned with the goals of the organization?	32	8	12
47	4. Had the business case been reviewed and revalidated at each scheduled gate?	34	5	13
48	5. If project specifications had changed significantly, had these changes been well documented and approved by the appropriate stakeholders?	37	6	9
49	Accountability			
50	1. Did status/progress meetings occur regularly?	27	18	7
51	2. Were issues raised and dealt with?	40	3	9
52	3. Was this project on time and on budget?	36	9	7
53	4. Were changes in scope being managed?	40	7	5
54	5. Were adequate staff allocated to the scheduled tasks at the scheduled time?	37	4	11
55	Corporate Project Manager Discipline			
56	1. In the event of serious problems, had the necessary information been available to support decisive action?	27	4	21
57	2. Was the project manager avoiding premature, inordinate and inappropriate use of contingency funding?	8	30	14
58	3. Was the project manager maintaining sufficient control over appropriate project resources?	40	0	12
59	4. Was the contractor providing complete information on project performance and progress?	37	2	13
60	5. Were project managers going to others in the organization for relevant knowledge, skills and experience for advice and support?	39	2	11

61	Risk Management			
62	1. Were the team leaders being held to budgetary constraints?	35	5	12
63	2. Did the management approach continue to ensure coordination of all sub-projects, ensure communication among different sub-project teams and address shared horizontal issues?	36	4	12
64	3. Were internal/external peer reviews being held as scheduled?	38	3	11
65	4. Were oversight reviews by a senior steering committee being carried out at each gate?	32	7	13
66	5. Were regular review sessions held to review the continued relevance of the project, project performance, and to raise concerns about actual/potential problems?	38	0	14
67	The Project Completion and Review Phase			
68	Support for the Business			
69	1. Were the clients involved throughout the project from evaluation and acceptance of deliverables, product and integration testing to final acceptance and sign-off?	31	2	19
70	2. Did the clients carry through on their commitment to the level of effort required of them?	38	3	11
71	3. Was the client satisfied with the final deliverable?	40	7	5
72	4. Was the final product aligned with the goals of the organization?	41	6	5
73	5. Did the project remain within the predefined scope?	32	14	6
74	6. Were any changes to project specifications documented and approved by stakeholders?	38	2	12
75	7. Was the project consistent and compatible with the department's information and technology direction, strategies, architectures and infrastructures?	40	1	11
76	8. Was the business case reviewed and revalidated at each scheduled gate?	22	3	27
77	10. Was the business case reviewed and whenever there was significant change to the project or business function?	32	5	15
78	Accountability			
79	1. Was the project adequately staffed during all phases?	40	9	3
80	2. Were issues with respect to weak or poorly performing personnel quickly and adequately addressed?	37	3	12
81	3. Was a project wrap-up session held to document 'lessons learned'?	22	25	5
82	4. Did the project sponsor assume the responsibility for realizing the benefits predicted for the project?	36	0	16
83	Corporate Project Manager Discipline			
84	1. Did the project maintain a positive image throughout?	36	13	3
86	3. Did the department make use of an apprenticeship program to provide valuable experience to junior project managers?	19	9	24
87	Risk Management			
88	1. Were all risks identified and managed properly?	37	3	12
89	2. Were the mechanisms and tools used to monitor the project schedule adequate?	34	6	12
90	3. Were team leaders held to budgetary constraints?	24	5	23
91	4. Was there adequate communication to all involved personnel throughout the project?	39	2	11
92	5. Did the contractor provide necessary information on project performance and progress?	40	1	11
93	6. Did the project reviews (including internal/external peer reviews) occur as scheduled?	28	3	21
94	7. Was there enough time, money and resources to get the job done right?	38	3	11
95	8. Were there adequate contingency plans for potential problems (i.e., no problems arose for which there were no contingency plans)?	24	7	21
96	9. Had the project complexity been accurately determined before start-up?	27	3	22
97	10. Were all changes analyzed quickly and brought to management's attention in a timely fashion?	29	1	22

Appendix C

Interview Questionnaire – Answered by Mr. Iqbal Madakkattel

1. Is the organization vision and mission lived by people? means are they loyal, and if they are how are they rewarded for their loyalty?
 - a. There are different kinds of people in an organization. Some people are very loyal to the organization just like how loyal a person to his/her family, while others are not. I have seen the organization vision and mission lived by people with and without loyalty. So loyalty shouldn't be a criteria to measure whether employees is really concerned about the mission and vision of the organization. It is the thought of a person which determines whether he/she should live for the vision and mission of the organization.
 - b. Normally loyalty to the superior is highly rewarded in the form of earlier promotions, better increments and bonuses. Those people are also shielded from unnecessary troubles in the organization. I believe it is loyalty to the superior that plays major role in getting rewarded rather than the loyalty to the organization.
2. How do you consider improving staff environment in your organization such as staff offices, working hours and facilities?
 - a. Employees expect decent working conditions such as they are not punished by requiring paying large sum money or harassed physically or verbally for their mistakes. They are also compensated for overtime duty. They must not be discriminated based on cast, race, religion, origin, disability, gender, sexual orientation, political affiliation, etc in the work place. Health and safety mechanisms must be available in the workplaces.
 - b. The best way in my opinion to improve the staff environment is to go for global standard like SA 8000 developed and overseen by the New York based organization Social Accountability International (SAI). Such standards ensure decent working conditions for employees. They also help employees understand their rights such as collective bargaining and freedom to form associations. They also protect people from discrimination based on cast, race, religion, origin, disability, gender, sexual orientation, political affiliation, etc.
3. Do you give your staff a chance to express their opinions regarding the project and the management?
 - a. Definitely. Regular project progress meetings are held to discuss the progress, the issues and delays in the project. These meetings with higher authorities such as project sponsor ensure the issues are escalated and addressed timely. The first and the most important thing in managing issues and conflicts is to give a chance to the stakeholders express their opinions before coming to a conclusion. A working atmosphere which allows any healthy expression of opinion is highly effective in managing projects.

4. To what extent are employees informed about overall business objectives and processes, running projects and their results?
 - a. I believe, we, to a great extent inform all the employees on what is going on in the organization, what organization needs to achieve and what are the processes in place. Major milestones of large projects are communicated to the employees time to time. The organization uses different tools to do this. Staff parties, meetings, intranet portals, emails, etc are used to keep employees informed about overall business objectives and processes, running projects and their results.
5. Are there activities in order to improve the capabilities of employees, for example through training, job rotation (personal development)?
 - a. Definitely. There are plenty of activities to improve the capabilities of employees. For instance, a number of training programs in various areas such as product knowledge, regulations and compliances, techniques, technologies, soft skills, etc are conducted. Employees are encouraged to take course in the areas of improvement. Training is not only provided as class room training but as on the job training, online training, helping to get certified such as Project Management Professional (PMP), etc. Special emphasis is given to courses to improve the soft skills such as leadership trainings, supervisory trainings, time and stress management, etc.
 - b. Suggestions scheme is implemented so that employees can suggest new ideas to cut costs, to improve the service levels, products, working conditions, operational efficiency, etc.
 - c. Employees with talents in a field other than their field of work are noticed and are given chances in those fields.
 - d. Employees are encouraged to use the libraries to improve their knowledge. Also, they are given holidays for examinations, etc.
6. How would you rate the commitment and active participation of personnel doing the work (for example, software practitioners, service staff) in defining and improving practices?
 - a. One of the methods is to check whether the employees meet the deadlines given for the tasks. Consistently succeeding in meeting the deadlines shows the commitment and active participation of personnel.
 - b. The quality of the deliverables also shows the commitment of personnel.
 - c. The attempt by the personnel to get things done within the allotted budget also shows commitment and active participation.
7. Is innovative behavior promoted/supported/rewarded? Are innovations directed from management or can they be proposed and developed by people at all levels of their organization?
 - a. Innovative behavior is promoted, supported and rewarded. There are various schemes through which they are promoted, supported and rewarded. For instance, the suggestion scheme allows anyone to express innovative ideas. People across the board participate in suggesting new creative ideas.

- b. There are also schemes like on the spot rewarding. This also keeps people motivated.
- 8. How easy is it to gain resources, support and privileges from management for improvement or innovative activities during a project?
 - a. Management is very eager to support any efforts by the employees to improve the process, the operational efficiency and there by reducing the costs. Suggestions from employees are carefully studied by management team for the feasibility of implementation. Management feels they need to support the efforts by adequately providing resources.
- 9. What risk does this organization regularly confront in a project?
 - a. There are different types of risks that the organization is exposed to in any projects. For instance, one of the common risks found is that the end products delivered by the project do meet all the expectations that were set in the beginning of the project. Also, some times there are cost overruns. Also sometimes, the vendor implementing the solution fails to complete the project effectively due to their internal problems.
- 10. Is risk of failure recognized and tolerated as part of behavior of people at all levels of their organization during a project execution?
 - a. To a great extent, they are recognized and tolerated, especially when all the efforts were put into the project and when the project turned out to be a failure thanks to things beyond the control of the personnel who executed. However, enquiries are conducted to learn lessons from the failure so that it wouldn't be repeated in the future projects.
- 11. If you could name three things that you believe hinder the organization from achieving excellence on a regular basis in a project, what would they be?
 - a. Inability to define and describe in detail the end deliverable of the project in the requirement analysis stage of the project.
 - b. Poor assessment of vendor, technologies, consultants, etc.
 - c. Inability to develop comprehensive test cases in the beginning and execution phases of the project
- 12. For management to act proactively, about what specific concern should it be vigilant?
 - a. Close monitoring of issues being reported during all phases of project and to provide immediate directions on those issues before things deteriorate.

Firm support to the project personnel during tough times in the project