High I.T. Failure Rate: A Management Prospect

Author: Mazhar Tajammal Hashmi
mtha05@student.bth.se

Supervisor: Dr. Peter Stevrin
pst@bth.se
Acknowledgement

First and foremost, I would like to express my sincere gratitude to my (Allah) for blessing me with the potential and ability to work on this master thesis. Then I would like to thank my beloved parents without whom I would have never been blessed with the opportunities that I have been given in my life.

I would like to thank my supervisor Prof. Dr. Peter Stevrin for giving me an opportunity to work on this interesting topic and for providing me great guidance and support in this research work.

I would like to express my special thanks to my Program Manager Anders Nilsson for being so helpful in times of need.

I am grateful to Univ. Asst. MS. MSc. Zeeshan Ahmed for working with me as my external advisor and I learned a lot from his experience and guidance.

I am thanking my family and friends for their endless love and corporation especially from Salman Khan and Haseeb Saleem. Without their support, this work would not have been possible.

Mazhar Tajammal Hashmi
Abstract

Software industry is growing day by day and software is going more complex and diverse with increase in cost and rate of failure. This increase in size and complexity of software projects has negative impact on the software productivity, schedule and effort. Organizations are demanding high quality products to increase their productivity and profits. It is common that they are facing some serious problems even after spending a large sum of money. So, its alarming situation and the concerned parties should take effective steps to resolve software project failure problem. Buyers have some options available in market with low budget but in this case no guarantee for the product quality. Above all this, we are facing a high rate of software failure putting software industry on stake.

This study revolves around the core issue of finding the root causes of software project failure with respect to organizational factors. In this I have tried to find the organizational factors contributing towards the failure of software projects. I have done this study with the help of literature review and questionnaire survey. There could be one or several factors responsible for the software projects failure, which are mentioned in chapter two. I have slightly touched the Information Technology for digging deep into the failure and for understanding this phenomenon. Information Technology is playing a very vital role in today’s organizations for competing on world level. Software failure is the biggest challenge faced by IT as well as business people. There is strong need to find the root causes of software project failure and mitigate them. For controlling this failure problem management can perform its role. I have discussed the role of management in defining, measuring, controlling and implementation of software projects. I have a comprehensive discussion in literature review chapter that with an effective coordination between IT and Management can lead us towards achieving organizational goals. Better results can be achieved through both teams coordination and managing the software projects effectively.

A project is considered failure when it is not able to show the anticipated results and it is happened when team is not able to fulfill the requirements of the project e.g. overruns time, overruns resources, lack of conformance with initial requirements specifications etc. I have tried to find out the answers of my research questions through literature review and empirical study. In finding the answers of research questions chapter 2, 4, 5, and 6 are very important. Root causes of software project failure are presented and validated through literature review, data analysis, discussion, and findings. A comprehensive analysis of empirical data and discussion in the finding chapter on research questions will give you the insight into the problem and my effort to sort out them in a precise way. For the purpose of knowing the solution of this study, I will refer you towards the conclusion and recommendation chapter 6. In chapter 6, I briefly discussed the recommendations for avoiding the software project failure. These recommendations are drawn on the basis of learned knowledge through literature review and analysis on the root causes software project failure in chapter two.

The concerned or interested people can get benefit from this research study and definitely it will help them to avoid software project failure. The contribution of the
research is twofold. First, it will be helpful for the software making professionals/companies and secondly, it will be helpful for decision makers/users (Organizations), when they are going to buy or implement a software project for enhancing their productivity.

**Keywords**: Failure, Software Project Management, I.T. Management, Organizational Factors, Causes of Software Project Failure.
TABLE OF CONTENTS

TABLE OF CONTENTS..........................................................................................5
LIST OF FIGURES.................................................................................................7
LIST OF TABLES ...................................................................................................8
CHAPTER 1: INTRODUCTION .............................................................................9
  1.1 BACKGROUND .........................................................................................9
  1.2 AIMS AND OBJECTIVES .................................................................10
  1.3 RESEARCH QUESTIONS ......................................................................10
  1.4 SIGNIFICANCE OF THE STUDY ......................................................11
  1.5 RESEARCH METHODOLOGY .........................................................11
  1.6 STRUCTURE OF THE THESIS ............................................................12
     1.6.1 Chapter 1: Introduction ..................................................................12
     1.6.2 Chapter 2: Literature Review ......................................................12
     1.6.3 Chapter 3: Research Methodology .............................................13
     1.6.4 Chapter 4: Data Analysis ..............................................................13
     1.6.5 Chapter 5: Findings ......................................................................13
     1.6.6 Chapter 6: Conclusion and Recommendations ..........................13
     1.6.7 Chapter 7: References ..................................................................13

CHAPTER 2: LITERATURE REVIEW.................................................................14
  2.1 INTRODUCTION ..................................................................................14
  2.2 SOFTWARE PROJECT FAILURE .....................................................14
  2.3 RELATIONSHIP B/W MANAGEMENT AND SOFTWARE PROJECTS ....16
  2.4 MAJOR CAUSES OF SOFTWARE PROJECT FAILURE ..................18
     2.4.1 Lack of Project Management .....................................................18
     2.4.2 Lack of Resource/Budget Management .....................................20
     2.4.3 Lack of Project Planning .............................................................21
     2.4.4 Communications Failures ...........................................................22
     2.4.5 Lack of Top Management Involvement & Support ....................23
     2.4.6 Technology Illiteracy (User Involvement and Training) .............23
     2.4.7 Lack of Stakeholders Involvement ..............................................24
     2.4.8 Improper Software Documentation .........................................25
     2.4.9 Requirements Engineering Crisis (Specifications) ....................25
     2.4.10 Lack of Quality Management ..................................................26
     2.4.11 Lack of Proper Organizational Structure ..................................26
  2.5 SUMMARY ............................................................................................28

CHAPTER 3: RESEARCH METHDOLOGY .........................................................29
  3.1 DATA COLLECTION ............................................................................29
     3.1.1 Primary Data ................................................................................29
     3.1.2 Secondary Data ............................................................................30
  3.2 QUESTIONNAIRE ...............................................................................31
     3.2.1 METHODOLOGICAL APPROACH ..........................................31
     3.2.2 DESIGN OF QUESTIONNAIRE ..............................................31
  3.3 SELECTED SAMPLE (ORGANIZATIONS) .......................................32
     3.3.1 National Survey and Cadastre Denmark ....................................32
LIST OF FIGURES

FIGURE 1: THESIS RESEARCH PLAN ................................................................. 12
FIGURE 2: RELATIONSHIP B/W BUSINESS & IT PERSPECTIVE .................... 16
FIGURE 3: ORGANIZATIONAL STRUCTURE FOR SMALL TO MEDIUM COMPANIES ................................................................. 17
FIGURE 4: DIFFERENCE B/W PLANNED AND NON PLANNED WORK .......... 22
FIGURE 5: GENERAL DYNAMICS PARTIAL ORGANIZATION STRUCTURE .... 27
FIGURE 6: SOURCES OF PRIMARY DATA ....................................................... 30
FIGURE 7: SOURCES OF SECONDARY DATA .................................................. 31
FIGURE 8: WHAT IS THE CORE BUSINESS OF YOUR ORGANIZATION ....... 36
FIGURE 9: PARTICIPATION IN SOFTWARE PROJECT DEVELOPMENT .......... 36
FIGURE 10: WHAT WAS YOUR ROLE IN SOFTWARE DEVELOPMENT PROCESS 37
FIGURE 11: IN HOW MANY PROJECTS YOU HAVE PARTICIPATED? .......... 37
FIGURE 12: DEFINE SOFTWARE PROJECT FAILURE ................................... 38
FIGURE 13: RATIO OF SUCCESSFUL VERSUS FAILED PROJECTS ............. 39
FIGURE 14: PARTIAL OR COMPLETE PROJECT FAILURE ........................... 40
FIGURE 15: EFFECTS OF SOFTWARE PROJECT FAILURE ............................ 40
FIGURE 16: SKILLS AND ABILITIES NECESSARY FOR TEAM MEMBERS .... 41
FIGURE 17: CAUSES OF SOFTWARE PROJECT FAILURE ............................. 42
FIGURE 18: PARTICIPANTS OF SOFTWARE PROJECT TEAM ....................... 43
FIGURE 19: ROLE OF HIGHER MANAGEMENT IN ORGANIZATIONAL GOALS 43
FIGURE 20: ROLE OF HIGHER MANAGEMENT IN YOUR INDIVIDUAL GOALS 44
FIGURE 21: TECHNICAL KNOWLEDGE VERSUS MANAGEMENT KNOWLEDGE 45
LIST OF TABLES

TABLE 1: HALL OF SHAME ................................................................. 16
TABLE 2: STANDISH GROUP SUCCESS CRITERIA ...................................... 23
TABLE 3: PERCENTAGE OF RECEIVED QUESTIONNAIRE ........................ 35
CHAPTER 1: INTRODUCTION

This chapter introduces the background, objectives and the structure of this research thesis. This section also discusses the selected research questions and research methodology.

1.1 Background

Software industry is growing day by day. This increase in scale and complexity of software projects sometimes results in partial or complete failure. There can be one or several reasons for this failure of software projects. On the other hand, demand for high quality products is increasing, as organizations are spending huge sums and their core areas mostly depend on it. With the passage of time some products become larger and more complex. This increase in size and complexity can have a negative impact on software development with respect to cost and schedule. In this situation, one can think about some other low budget options available in market. Then, there is no guarantee for product quality. For maintaining successful rate in software industry, it is a right time, when we need to focus on product quality and try to improve productivity of software development process.

Software is involved in almost everything like cars, telephone and ATM machines etc. The difference being in some products, software is playing a primary role and in others its role is secondary. In both cases software is very crucial for the success of any business venture. If we try to focus on literature review, we will find hundreds of examples of software failure. For the evidence, I am giving couple of examples here; in “Fortune 200 companies they have ratio of facing at least one software project failure in company’s history and if I am talking about fortune 200, you can easily imagine how big those projects are and how big will be the loss” by (George, W.R. 1988). It means software projects failure is very important area and we need to focus on it. According to another famous (KPMG, 1997) survey, “35% of all major software projects exceed budget and schedule in months and some time years”. This is not a small figure when we are talking about major projects. In this way, software failure is very important issue for software development firms as well as buyer and user firms.

The things described above are the main reasons to select this unique topic of high IT failure rate with respect to management. You can easily see from literature review that the software failure is becoming a big problem for most of the organizations. There can be several reasons of software failure and I would give a fair try to find out all those reasons. Furthermore, I can suggest some improvements or recommendations for controlling the software failure. For this purpose, I need to focus on important areas with respect to my research study scope like project management, planning, budgeting, and total quality aspects of software projects. As, I am considering high IT failure with respect to IT management and that is why it is very important that management should play its role in finding ways to overcome this problem. This can control software projects failure in an effective and efficient way.

In early days, when IT was in its initial phase, managers were buying software for their organizations. Their basic aim was to increase productivity and profitability but it was not so simple to attain success. Actually when they started implementing software in the organizations, they faced lots of problems like coordination with organizational goals, ignorance of the staff and unsuitable structure of the organization for software etc. Still we are not able to remove these problems completely and are struggling for the solutions. (George, W.R. 1988)

Management has a key role in defining, controlling and implementation of software projects. With effective coordination between IT and Management can lead us towards achieving
organizational objectives. Software development team can work with coordination of management team for better results and being effective in managing software projects.

It is necessary to define organizational goals those that are aimed to accomplish through the development and successful implementation of software project. For accomplishing any project, every organization need to focus and follow management strategies like planning, budgeting, organizing, leading and controlling and some others. Organizations are always facing many problems and opportunities at the same time. It can be difficult for organizations to decide which project they have to proceed with. For selecting perfectly matched software with organizational needs, we need to define key factors which are critical for the success of a project. It is important to mention that strategic planning and quality management can be good techniques for selecting the best from the available options. Proper IT Management can improve the software project development process along with the accomplishment of the needs of users.

In this research study, I will slightly touch some technical aspects of software project development. There are many approaches for developing software projects e.g. Product Life Cycle Management, Waterfall system development approach and Agile development process etc. These all software system development approaches have their positive and negative points and all organizations are independent to choose any one according to their requirements. But all development techniques have almost same phases of development life cycle e.g. feasibility, data analysis, requirement engineering and testing etc. “Software development refers to the sequence of different phases of product goes through and strategies followed by management as product goes through its different completion phases” by (Zeeshan, A. and Gerhard, D. 2007). I am here only concerned with finding the root causes of software project failure due to sloppy management practices instead of going into the details of technology.

1.2 Aims and Objectives

The focus of the research is limited to find out the root significant causes of most software projects failure and try to present possible recommendations on the basis of learned knowledge during this research study.

For achieving my objectives, I will go for literature review and questionnaire survey. It will bring me in a position to draw final conclusion about the factors responsible for software project failure. Further I can also try to find how and to what extent organizational management can decrease or remove software failure problem?

1.3 Research Questions

Following are the research questions (RQ) along with their motivation based on the objectives that have to be addressed in this research study.

RQ 1: What is the state-of-art of software project failure due to inefficient management operations?

RQ 2: What are the possible factors of software project failure and how each is responsible for the failure?

RQ 3: How can we overcome on all the factors of software project failure identified in RQ2?
RQ1: has been devised to provide the underline concept of software project failure. This research question will bring forth a concise study of software project failure which has been discussed in literature i.e. definition of software project failure.

RQ2: There can be number of factors involved in the failure of software project i.e. “Lack of project planning and management, Inefficient cost management, Inefficient time management, lack of resource planning, Lack of stakeholder involvement, Lack of controlling and quality management, Communications failures within teams, Inadequate conflict management, Culture gap existing between IT and business professionals” etc by (Zeeshan, A. & Gerhard, D. 2007; Parker, M.M. 1995; OASIG, 1996; Standish, 1994). As all of the above described factors are in a disperse form, so there is a huge need to gather all of these factors which are occurring due to incompetent management operations under one umbrella.

RQ 3: Once I have identified all the factors of software project failure then there is a need to provide some useful recommendations which will try to mitigate the effects of those factors. This question has been devised to fulfil the stated requirements.

1.4 Significance of the Study

According to the scope of my research, I am interested in dealing with the software failure problem with the prospect of management.

It is very important to understand and realize the need of developing productive software projects and role of management for getting better results from the efforts of software development teams. This study is important as I am trying to address the contribution of management in the success of software projects. Obviously management will have some positive impact/influence on the development of software projects. But I will try to find out that to what extent proper implementation of management can bring success to the software projects.

1.5 Research Methodology

Following matrix briefly explains the research questions and what methods have been used to answer those questions and what outcome should be expected by applying those methods.
Answers to the research questions are sought through

- Literature Review (Chapter 2)
- Empirical Study (Chapter 4)
- Findings (Chapter 5)
- Conclusion and Recommendations (Chapter 6)

These activities and their goals in relation to the research questions are described below.

Research Question (1) (“What is the state-of-art of software project failure rate due to inefficient management operations?”) brings forth a comprehensive study of software project failure concept from literature.

Research Question (2) (“What are possible factors of software project failure and how each is responsible for the failure?”) is more related to the above question where a comprehensive literature review and survey questionnaire will be done to study the involved factors in software project failure.

Research Question (3) (“How can we overcome on all the factors of software project failure identified in RQ2”) will provide some recommendations to mitigate the effect of software project failure factors by analyzing survey data and results.

1.6 Structure of the Thesis

1.6.1 Chapter 1: Introduction

In this chapter I will present the introduction and include background, aims, objectives, research questions, significance, research methodology and structure of the thesis.

1.6.2 Chapter 2: Literature Review
In this chapter, I will deal with literature review and include background and introduction of IT and management, software project failure, relationship b/w IT and management. I will also present all major causes of software project failure and relevant data.

1.6.3 Chapter 3: Research Methodology

In this chapter, I will explain the research design and structure of the thesis. The major sources of data collection and all details relevant to survey questionnaire and short introduction of subject organizations and finally the limitation of the research will be presented.

1.6.4 Chapter 4: Data Analysis

In this chapter I will do the analysis of data gathered and present you the key points and correlate it with literature review.

1.6.5 Chapter 5: Findings

In this chapter I will correlate the results of data analysis and literature review with the research questions. It will try to summarize the whole thesis in this chapter.

1.6.6 Chapter 6: Conclusion and Recommendations

In this chapter, I will conclude the results from the study and highlight the area for future work.

1.6.7 Chapter 7: References

In this chapter, I will include the complete list of references used in this research thesis.
CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

The aim of this chapter is to collect relevant data available on software project failure problem. There is enough literature available to prove that software failure problem is faced by many organizations. In the past few decades, the business environment has been changed and facing rapid changes like technological advancement in sense of doing business and due to this almost everything partially or completely dependent on technology. Also the span of business is changing and shifting from local to global and now due to technological advancement, we are operating on global level and facing tough competition. “As the era of globalization proceeds and enterprises begin competing for the same customers”. (Parker, M.M., 1995).

Today, IT is a major department in most of the organizations and its impact is felt throughout the company. As (Grossman, I., 2003) said in a report that “IT departments now consume, on average 50% of all corporate capital expenses. Yet, it is widely accepted that over 70% of IT projects fail, with software development representing most of them are over age”. Further supported by (Saur & Cuthbertson, 2003) that an Oxford University survey reported that “only 16% of IT projects was successful, around 74% were ‘challenged’ and 10% were abandoned”. In this situation it is important to examine some fundamentals, like “what is software failure?”

The concept of a software failure is defined by (J.C. Laprie, el al., 1992) as “deviation of the delivered service from compliance with the specification”.

In current era as the software business reaches new customer groups, the tradition of a specification has changed. Often products don’t have detailed specifications, nor would the customer expect them. What is important to the customer is that the product does work and they are satisfied. In this situation customer expectation largely determines whether a failure has occurred or not. As (Chillarege, R. 1996) said that” If there were well defined specifications, it could be argued whether the product worked as designed or not”.

When I started examining the failure phenomena, I came to know that how little is truly understood regarding failure which matter the most. It is further a problematic that there is a lack of clarity as to what is software project failure? When a product/software is unable to perform, there can be countless reasons and on the other side some times difference between correct operation and failure is not clear. IT experts are agreed that such failures occur far more often than they should. Furthermore these failures are now well into the billions of dollars a year and that’s why it is most important and urgent matter for an organization to primarily find its causes and secondly try to avoid them. So, I need to dig deep into this failure problem and look for its main causes and their solution. But due to limited time and scope of the research project, I will just try to determine the root causes and present recommendations for avoiding software failure with the prospect of management.

2.2 Software Project Failure

After giving a brief introduction of my research topic, now I am in a position to start with the main research problem and that is “software project failure”. According to (George, W.R, 1988) in “Fortune 200 companies they had a chance of at least one software project runaway in their history and you can imagine for fortune companies what would be the cost of failure”. In (KPMG, 1997) they mentioned that “about 35% of all major information systems
projects become runaway”. In words of (Ackoff, 1994) “A mistake is an indicator of a gap in one’s knowledge. Learning takes place when a mistake is identified, its procedures are identified and it is corrected”.

“As Software industry is growing along the time; the software is going more complex, diverse and costly. We still have some low budget software options available in market but in this case software quality suffers. On the other side maintaining successful rate in software industry is on stake” said by (Zeeshan, A. and Gerhard, D. 2007). In another words said by (Konstantinos, G. et al., 2005) that “the design and development of large scale software projects is a complex endeavour, often facing problems like cost and schedule overruns as well as low quality”.

A (KPMG, 1997) survey on software project shows the results that “over 61% of the projects those are analyzed were failed”. In the (CHAOS, 1995) report “only 16.2% of software projects were completed on time and on budget”. According to (OASIG, 1995) report” IT project success rate revolves around 20-30% based on its most optimistic interviews. Bottom line, at best, 7 out of 10 IT projects fail in some respect”. According to (Trevor woodhapper, and John McManus, 2003) in the last decade a significant number of project failures have been reported. Research also indicates that in case of software projects there are many problems like “there is a tendency to jump to the project implementation phase as quickly as possible and put through a Return on Investment (ROI) validation”.

It is very important to present you some concrete examples of software project failure in our economy. For this purpose I have selected a table which presents some examples of software project failure. This table covers the time span from 1992 to 2005 and different industries as well as their amount of loss. After reviewing this table you will realize how big is this failure problem and that’s why we need to find out its real causes and focus on their solution.

<table>
<thead>
<tr>
<th>Year</th>
<th>Company</th>
<th>Outcome (Costs in US $)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2005</td>
<td>Hudson Bay Co. [Canada]</td>
<td>Problems with inventory system contribute to $33.3 million loss.</td>
</tr>
<tr>
<td>2004</td>
<td>Avis Europe PLC [UK]</td>
<td>Enterprise resource planning (ERP) system cancelled after $54.5 million is spent.</td>
</tr>
<tr>
<td>2004</td>
<td>Hewlett-Packard Co.</td>
<td>Problems with ERP system contributes to $160 million loss.</td>
</tr>
<tr>
<td>2003-04</td>
<td>AT&amp;T Wireless</td>
<td>Customer relations management (CRM) upgrades problems lead to revenue loss of $100 million.</td>
</tr>
<tr>
<td>2002</td>
<td>McDonald’s Corp.</td>
<td>The information purchasing system cancelled after $170 million is spent.</td>
</tr>
<tr>
<td>2001</td>
<td>Nike Inc.</td>
<td>Problems with Supply chain management system contribute to $100 million loss.</td>
</tr>
<tr>
<td>1999</td>
<td>Hershey Foods Corp.</td>
<td>Problems with ERP system contribute to $151 million loss.</td>
</tr>
<tr>
<td>1998</td>
<td>Snap-on Inc.</td>
<td>Problems with order entry system contribute to revenue loss of $50 million.</td>
</tr>
<tr>
<td>1997</td>
<td>U.S. Internal Revenue Service</td>
<td>Tax modernization effort cancelled after $4 billion is spent.</td>
</tr>
<tr>
<td>1996</td>
<td>Arianespace [France]</td>
<td>Software specification and design errors cause $350 million Ariane 5 rocket to explode.</td>
</tr>
<tr>
<td>1995</td>
<td>Toronto Stock</td>
<td>Electronic trading system cancelled after $25.5 million.</td>
</tr>
</tbody>
</table>
Table 1: Hall of Shame  
Source: Robert, N. Charette, 2005

<table>
<thead>
<tr>
<th>Year</th>
<th>Company/Project</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1994</td>
<td>Chemical Bank</td>
<td>Software error causes a total of $15 million to be deducted from 100,000 customer accounts.</td>
</tr>
</tbody>
</table>

2.3 Relationship b/w Management and Software Projects

The ability to use IT software’s effectively and efficiently is a success factor for the organizations today. Some organizations are really achieving greater progress with the help of IT and some on the other side are facing problem of failure. Mentioned by (George, W.R. 1988) that “the speed, cost, and capabilities associated with computers provide a wealth of highly attractive opportunities for using this technology to solve business problems or enhance current ways of doing business”. For this reason management should realize this fact that the use of IT supports corporate strategy. As mentioned by (Grindley, K. 1995) that “the successful implementation of software projects can be capable of producing better results for organization in less time and low cost”. The critical role ascribed by (Porter, & Millar, 1985) that “IT in both private and public sector settings is viewed as a strategic device which, if deployed effectively, can sustain and improve the competitive advantage of a commercial organization”. Following figure briefly explains the basic relationship between business and information technology. Furthermore I can analyze how both work and support each other.

<table>
<thead>
<tr>
<th>Business Prospective</th>
<th>I.T. Perspective</th>
</tr>
</thead>
<tbody>
<tr>
<td>+ Value</td>
<td>- Costs</td>
</tr>
<tr>
<td>Costs of Services</td>
<td>+ Recovery</td>
</tr>
<tr>
<td>Justification</td>
<td>Viability</td>
</tr>
</tbody>
</table>

Figure 2: Relationship b/w Business & IT Perspective  
Source: Parker, M.M. 1995
Explanation: The justification for IT is the balance of value and cost assigned to the business, measured by the business organization in business terms, whereas the basis of viability is the value of the IT to the IT organization, compared to the cost of using the technology for the benefits of the organization.

According to (George, W.R. 1988) “In early stages of IT the most of the managers were tend to adopt the technology first and later they tried to figure out what to do with the new system and how can they cope it with organizational implementation and coordinate with organizational goals”. In start IT was started with few computers and which were only in the use of head office at that time. But now the situation is changed and “at present almost every organization has a separate IT department and having separate department means, IT has a specific role and place in organizational structure and works with collaboration of other departments for the benefits of the organization” said by (Grindley, K. 1995). It is obvious that every organization has a unique structure with respect to the positioning and roles of IT in organization. Secondly all organizations are different due to difference in their functions, type and sizes. But with the help of following figure, you can have an idea that how an organization work with the collaboration of IT.

![Organizational Structure for Small to Medium Companies](image)

**Figure 3: Organizational Structure for Small to Medium Companies**

Source: Grindley, K. 1995

Explanation: For proper implementation of well established software in an organization, it is very important to study and understand organizational structure and culture of subject organization. After observing the above mentioned diagram you will have a basic idea that how IT works with the collaboration of different departments in an organization.

Like every other organization technological companies are also based on different departments for example management, marketing, production, accounts, quality and engineering etc. “In case of software projects the engineering department is most important as other all departments’ survival is dependent on its successful operations; especially in an IT company” (Zeeshan, A. and Gerhard, D. 2007). “Over the past two decades, the software industry has moved inexorably toward new methods for managing the ever increasing complexity of software projects. We have seen evolutions, revolutions, and recurring themes of success and failure. While software technologies, processes, and methods have advanced rapidly, but software engineering remains a people intensive process” mentioned by (Walker, R. 2001). As we are saying engineering is a people intensive and people are really important in developing, implementing and running the software projects. As (Trevor wood-happer, & John McManus, 2003) “studied 650 projects in the US and highlighted a number of factors and parties which contributed to the success of a project and the successful completion of
software project is not a matter of single department or person”. Due to this reason it’s
important to mention the parties involved in the development process of a software project.
Following could be the list of parties directly connected in software development process:

- Higher Management
- Team Leader
- Development Team
- Project Manager (Non Technical)
- Users
- Sales & Marketing Team
- Stake Holders

It is necessary to discuss that how we can promote a good relationship b/w management and
IT teams. For this purpose basically both teams need to understand the roles of each party
and try to accept them as a business partner. We need to lay down IT infrastructures,
recognizes that IT systems are a necessity. As “30% of companies now justify IT investment
by the achievement of necessary performance objectives rather than financial benefits”
(Grindley, K. 1995). In real life mostly the problems start growing as the projects are
growing in size and complexity. As said by (Zeeshan, A. & Gerhard, D. 2007) that failure
problems can happen due to “lack of control over engineering processes, rapidly increasing
data, lack of coordination among team members, unclear product configurations, loss of
experienced staff, conflicts among the teams, lack of suitable formal communications
between departments, bureaucratic and complex engineering change, control systems and
lack of project and resource management. As a result company can face unnecessary
additional increase in costs, delays in product completion, loss in quality and waste of time”.

As you find that some common reasons of software failure presented by an author in above
paragraph. Same as above I trying to find out some common reasons of software project
failure on the basis of reviewed literature with respect to management. So, from next heading
I will start with the core area of study and that is to find some common and most important
reasons of software project failure according to literature review and will also validate them
through questionnaire survey in chapter four.

### 2.4 Major Causes of Software Project Failure

There is so much data written on the causes of IS/IT project failure and several studies have
mentioned some common causes of software failure like “lack of project management,
organizational culture and structure, user involvement and training, lack of project planning,
scope and objectives setting” etc by (Boehm, 1991; Keil, et al., 1998; POST, 2003; OASIG,
1996; CHAOS, 1995). In BCS review by (Taylor, 2001) found that among other causes the
main cause was “the poor project management and other things those were not managed
properly were resources, organizational factors, project scope and planning” etc. In different
studies they have mentioned different causes and symptoms of software project failure. After
reviewing the available literature; I have selected some common causes of software project
failure with respect to inefficient management and which are mentioned below. Further, I
can discuss them one by one with reference to relevant literature. So let’s start with first
reason and that is “lack of project management”.

#### 2.4.1 Lack of Project Management

Experience has shown that Murphy’s Law prevails indeed; “every thing will go wrong unless
management forces it to go right”. Mangers must recognize the need to take an active role in
software development when a key business initiative depends on it. As the “Management has
a key role in overseeing the definition and implementation of software that meets the needs of the business” (George, W.R. 1988).

Projects are often defined as having particular characteristics (APM, 2005; Cotterell, & Hughes, 1999). A project is likely to:

- Be a unique set of coordinated activities
- Have a few, clear, specific objectives
- Have defined budget, schedule and performance parameters
- Need a special team of people

The most important function of the IT project manager is to allocate resources to various activities like project planning, organization, quality management, communications, and human resource management. “Project management decisions are often tricky precisely because they involve tradeoffs based on fuzzy or incomplete knowledge. Furthermore, bad project management decisions, such as hiring too few programmers or picking the wrong type of contract can put an organization in poor position” (Robert, N. Charette, 2005). In article titled “Software’s Chronic Crisis”, by (Gibbs, W. 1994) identified that “for every six new large scale software systems those are put into production, two are cancelled and schedule overshoots by half, with larger projects doing worse. 75% of all large systems are “operating failures” that do not function as intended”. Most of the time projects with major problems terminated at some point. As mentioned by (Keil, M. 1995) that these “failing projects are eventually terminated or significantly redirected”.

Information systems projects continue to fail at an alarming rate, according to a (Standish, 2001) study “only 16% of all projects are fully successful, and 84% are still failure. Failed IT projects cost US companies an estimated $145 Billion per year”. Project Managers are usually held responsible for the success or failure of their projects; therefore, most studies revolve around “PMs Skills”. “The development and design of a project requires skills that range from soft people skills to sophisticated technical skills” said by (Soraya, J. NetoAlvarez, 2003). Most Project Managers have the technical skills required to manage a particular project, but not everyone has the interpersonal skills required to manage people successfully. As (Ewusi, M. 1997) states that “IS projects are unique in that they are conceptual in nature and require the intense collaboration of several different groups of stakeholders including IS staff, users and management”. According to (Fortune, J. & Peters, G. 1997) once again, it is on the PM to inspire team members to take ownership of the project and do their best to lead the project to a successful completion.

Software project management is important to support managerial decision making during the software lifecycle. “Software project management can help in controlling cost, quality and project duration” (Norman, F. et al., 2004). We need to keep records of previous projects; and especially failed projects. Investigating failed projects would be a good effort to control the software failure to some extent. As “Studies suggest that most IS project disasters are avoidable” (Heekens, G.R. 2002). Most of the time warning signals occur way before the actual failure. “History has shown that software projects are far more likely to be successful if they are highly focused and built upon well understood technology” said by (Heekens, G.R. 2002).

One of the major difficulties with investigating software development is collecting detailed and accurate quantitative data. Survey results from the Software Engineering Institute suggest that “most companies do not collect quantitative data” mentioned by (SEI, 2003). “Unfortunately, many software projects do not collect quantitative data in many projects. Overall, the use of qualitative, subjective data is less desirable than quantitative data” mentioned in (Austen, R. & Tracy, H. 2004). Reliable quantitative data on progress is most desirable but some times or in some projects it is not available. In fact, results from the (SEI,
2003) suggest that “in this situation most feasible approach is to exploit qualitative data naturally produced by the projects”. The solution present by (Cook, et al., 1998) that “it is often possible to generate quantitative data from other kinds of data that are collected within a project. For example, they were able to ‘recover’ quantitative data on conformance to process from events occurring in a defect fixing process”.

Finally according to (Lingberg, K. 1999) that “struggle and challenges are a part of learning process and many organizations have used an IS project failure as a method to improve the next version of software or on a completely different project. If you lose with an IS project, do not lose the lesson and especially if lessons can be learned and applied”. In project management, a project is considered “failure” when results don’t match initial objectives; common reasons for project failure are budget overruns and time overruns. Understanding why projects are not completed on time and budget can help correct the problem; as mentioned by (Lytinen, K. & Robey, D. 1999).

2.4.2 Lack of Resource/Budget Management

Not having control over resources can cause a project to fall behind schedule and force it towards failure. A survey by IT Cortex (Lytinen, K. and Robey, D. 1999) on “Reasons for Project Impaired Factors” conducted among 365 IT managers from companies of various sizes and in various economic sectors showed that “lack of resources” is one of the main reasons for project failure and also the organizational skills play a very important role in managing resources successfully. Organizations are investing large sums in software projects for getting strategic advantages but running their computers and getting various benefits from these software’s is only possible by making and implementing them successfully. According to (Grindley, K. 1995) in late seventies, price water house had established a worldwide panel of 5000 IT executives, pledged to volunteering their opinions on important IT issues. They respond that each year they are expecting an increase in their company’s investment in computers as compare to last year.

Project failure is typically defined in three ways: “Overrunning the budget by 30% or more; overrunning the project schedule by 30% or more; or failing to demonstrate the planned or intended benefits” by (Whitaker, B. 1999). IT projects have a bad reputation for going over budget and schedule, not realizing the opportunity and for providing poor return on investment (Clegg, et al., 1997; Keil, et al., 1998; Fielding, 2002; Register, 2002; Jaques, 2004). PAC mentioned in (Telegraph, 2005) that “far too often major IT enabled projects in government departments are late, well over budget, or do not work at all; an enormous waste of taxpayers’ money”. In the words of (Feld, and Stoddard, 2004) that “the average organization spends around 20% of their total budget on IT procurement, maintenance, operation and development and approximately $500 billion is wasted worldwide each year on IT purchases that fail to reach their objectives”.

Following examples may help you to illustrate the nature and scale of software project failure due to inefficient resources management.

- An estimate of the combined annual cost of cancelled IS projects in the late 1990s were $81 billion by (Liebowitz, J. 1999).

- (Dosani, F. 2001) Claimed that “When the true costs are added up, as many as 80% of technology projects actually cost more than they return. It is not done intentionally but the costs are always underestimated and the benefits are always overestimated.”
• A study of 50 government projects over 20 years found that those involving technical innovation and systems development historically ran over budget by up to 200% and over the original contract duration by 54%. This led the treasury to issue a warning that the public sector is over optimistic when evaluating IT project costs and timescales mentioned by (McCue, A. 2002).

• (Fortune, J. and Peters, G. 2005) mentioned a project that was to replace the British Army, Navy and Air Force inventory systems with a single system was brought to a close down after £130m had been spent. Hardware worth just over £12m was used elsewhere but the remaining £118m was written off.

• According to a report billions of pounds are wasted every year on badly managed IT system deployments. “The UK public sector alone spent an estimated £12.4 bn. on software in the past year and the overall spend on IT is projected to be 22.6bn”. Further this study suggests that only about 16% of projects can be considered successful and even conservative estimates put the costs of failures into the tens of billions of pounds across the EU.” By (Jaques, R. 2004).

• Many projects overrun because of poor estimates. Obtaining good estimates is a prerequisite for winning business. According to (Trevor wood-happer, and John McManus, 2003) from a cultural perspective, organizations must be prepared to walk away from a project rather than commit to unrealistic dates or budgets.

2.4.3 Lack of Project Planning

It is major and most common problem faced by many software projects and become abundant on some stage of its completion. You will get the clear ideas after reviewing this literature review. As “most of the time in past, business didn’t even include the IT group in strategic planning sessions and as an alternative, most of the time they inform to the IT department on a need to know basis and that is only after completing the strategic plan” said by (Parker, M.M. 1995). But (Annie, I. Anton, 2003) said “If you don’t know where you are going, you will probably end up somewhere else.” A complex project will likely fail without a plan. As again said by (Annie, I., Anton, 2003) for a software developer who wants to satisfy customers’ needs, that plan is a complete, consistent, and correct expression of the stakeholders’ requirements. For supporting statistically sound planning. Both “quantitative” and “qualitative” information must be captured. Any organization with even modest discipline in software development will have this information. Like the quantitative elements, outlined by the software engineering institute (SEI) in (Grossman, I. 2003), include “schedule, effort, functionality and quality. The qualitative elements include skills, tools, methods, complexity, interfaces,” etc.

Planning can be a good way to achieve a goal, because without planning we don’t have a specific path to follow and our efforts can leads us towards undesired objectives or results. We need to do strategic planning as every organization face hundreds of problems and opportunities for improvement. Due to scarce resources managers have to decide which of one out of several projects they are going to begin, this process could be called “strategic planning” (George, W.R. 1988). Most of the time projects are failed due to “lack of proper planning” and mostly it is happened due to project sponsors; as they want the results of a projects as soon as possible and believe that “planning represent unwarranted delays” (Trevor wood-happer, and John McManus, 2003).

For getting some clear idea about the planned and non planned work you have to see the following diagram that; in first planned diagram it is clear that most probably work will finished as planned and on time but in unplanned diagram there could be the chances of
some delays and even extra cost due to rework. Following figure could provide you more clear and precise view of this idea.

**Diagram For Planned Work**

![Diagram For Planned Work](image)

**Diagram for Unplanned Work**

![Diagram for Unplanned Work](image)

Figure 4: Difference b/w Planned and Non Planned Work

### 2.4.4 Communications Failures

In (Register, 2002) survey they suggested that “the reason for projects failure is poor communication between company managers and those who develop and manage new IT infrastructure. Project delivery requires cooperative effort, good communications and early warning systems”. Projects managers need to keep the team and other parties well inform about the project status at all time that what is going on. They should also be the person that tells everybody, if any changes occurring in the schedule or plan. The concerned parties should be consulted before things are put into action. Another superior approach suggested by (Trevor wood-happer, and John McManus, 2003) is “ask for information in writing. It is more productive, after being given some information, to send the giver a memo, which summarizes that, what was said. This serves to check the accuracy of communication and to remind them of any commitments they have given”.

![Diagram](image)
2.4.5 Lack of Top Management Involvement & Support

Changing technologies often threaten the interests of the employees and some other groups. “Organizations are political constructs and will always have a certain level of internal political conflict as the organization strives to meet its goals and objectives. There are times when IT projects become instable due to lack of higher management involvement” (Fincham, R. 2002).

In (Standish, 1994) report they identified 10 success criteria that could be used to forecast the potential success of a project or to evaluate its failure. They presented the idea that the projects can be rated according to their importance and overall scores. Following table briefly describes those success criteria and points given to each criterion. With the help of this table you can easily see that how important is the executive management support in the success of a project.

<table>
<thead>
<tr>
<th>Success Criteria</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. User Involvement</td>
<td>19</td>
</tr>
<tr>
<td>2. Executive Management Support</td>
<td>16</td>
</tr>
<tr>
<td>3. Clear Statement of Requirements</td>
<td>15</td>
</tr>
<tr>
<td>4. Proper Planning</td>
<td>11</td>
</tr>
<tr>
<td>5. Realistic Expectations</td>
<td>10</td>
</tr>
<tr>
<td>6. Smaller Project Milestones</td>
<td>9</td>
</tr>
<tr>
<td>7. Competent Staff</td>
<td>8</td>
</tr>
<tr>
<td>8. Ownership</td>
<td>6</td>
</tr>
<tr>
<td>9. Clear Vision &amp; Objectives</td>
<td>3</td>
</tr>
<tr>
<td>10. Hard-Working, Focused Staff</td>
<td>3</td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 2: Standish Group Success Criteria
Source: Standish, 1994

Explanation: The above table indicates that “Executive Management Support” is one of the top factors in success of a project. With the help of this diagram I can say that in developing and running the project successful, we need full and sincere efforts from higher management. I earlier mentioned that mostly the organizational factors are major causes of software project failures and it appears that most of the obstacles in the way of success are tend to be social and people oriented nature.

2.4.6 Technology Illiteracy (User Involvement and Training)

According to (Brown, T. 2001) the “United Kingdom Passport Agency System Project began to fail when the software was deployed without thorough testing and adequate training for the staff. The problem was gone worse when newly enacted legislation required that all children should have their own passports”. In (Standish, 1994) study they indicated that for
successful implementation of a software project, we should make sure about the
“involvement of end user” in the process.

There is still a big gap existing between complexity of technology and our ability to manage it effectively. As said by (Allen, B. 2007) “A lot of what were seeing is the support staff in the IT infrastructure in many cases are not trained well enough, they don't know the operational programs well enough and so, I think that is where we're seeing a shift, is that the governance model is weak and that is there is a huge risk from the people side.” A global study conducted by (Athabasca, 2001) on project failure and a large scale online survey was posted to 40,000 executives, project personnel, and consultants involved in managing or directing public and private sector projects via different internet sources and finally researchers received 3,156 responses (7.0% response rate). Results indicate that project failure dominates all sectors, and have little or no formal training to deal with the complexities of today’s projects. So it is all about training of people who are managing or handling software projects.

In (Accidental Managers, 2001) “75 % reported that projects consistently came in late and over budget, and that management tools, techniques, and methodologies were rarely applied to projects”, A (Standish, 2001) survey shows “User Involvement” as the number one reason for successful projects. A example of successful project referred by (Soraya, J. NetoAlvarez, 2003) that Hyatt Hotel implemented a system that allows customers to dial from a cellular airplane telephone, and check into their room, courtesy bus for pick up, and have the keys ready at express desk. This new reservation system was ahead of schedule, under budget, with extra features. Because they had all the right ingredients for success starting with user involvement, management support, clear requirements, proper planning, and small project milestones.

Lack of user involvement deals mainly with soft skills because the Project Manager needs to schedule appointments to interview users, and get them to buy into the project. Sometimes project managers can face resistance from users who do not like changes. PMs have to use their interpersonal skills to assure users that they will benefit from the new system. In addition, it is important to make the users feel they are part of the project and their input is highly valuable for the overall success of the project. Besides this “Lack of Training” deals mainly with technical knowledge; Project Managers need to understand different models used for developing and designing systems; recurrent training is also important due to fast technological advances. As said by (Laudon, and Laudon, 1996) that “as many as 75 percent of all large systems may be considered to be operating failures. As in many of these systems the problems appear to be the lack of attention paid to the needs of people who had to use the technology”.

### 2.4.7 Lack of Stakeholders Involvement

Project delivery requires cooperative effort, good communications and early warning systems. Project mangers need to keep users aware of what is going on at all times. They should also be the person that tells users first of any changes in the schedule or plan. The project sponsor should be consulted before things are put into action. (Trevor wood-happer, and John McManus, 2003)

It is also problematic that most of the time user’s lack of understanding of their role in the project and it is necessary that they should have some knowledge about the business process. However, many projects have a strategic angle which is not always understood by the customers and this can result in poor business definitions (Trevor wood-happer, and John McManus, 2003). My personal believe is that the lack of stakeholder’s involvement can create problems in the way of successful software project development.
Project managers need to rigorously monitor requests for changes. Each request should be logged and assessed for impact on time and cost. It is important to remember who the sponsors are, and what their objectives are, managers should check with the sponsor first before changing anything significant. “Understanding the customer wants is crucial to any successful system development project. Put simply, everyone’s efforts are wasted if the product produced is not what the customer needs” by (Trevor Wood-Happer and John McManus, 2003).

2.4.8 Improper Software Documentation

Software engineering is the discipline that provides principles, techniques, methods, and tools to systematically develop software systems. The development of a software system encompasses the development of the executable program and the creation of appropriate documentation. Documentation is an integral part of a software system. It contains the information that is necessary to effectively and successfully develop, use, and maintain a system.

In practice, mentioned by (Joachim, B. and Dirk, M. 2006) that “the creation of appropriate documentation is largely neglected. It is a well known fact that software documentation in practice is poor and incomplete”. Though specification, design, and test documents among other things are required by standards and capability maturity models (e.g., SEI CMM), such documentation does not exist in a complete and consistent form in most organizations. “When documents are produced, they tend to follow no defined standard and lack information that is crucial to make them understandable and usable by developers and maintainers” by (Lionel, C. and Briand, 2003). “Banco Itamarati, a privately held Brazilian bank, attributes the success of its IT project to clear vision and documented specific objectives. The company produced an annual net profit growth of 51% and moved from 47th to 15th place in the Brazilian banking industry” by (West, D. 1998).

2.4.9 Requirements Engineering Crisis (Specifications)

“No one knows what they want. But when they see what you have produced, they suddenly know what it is they don’t want and 92% put this problem amongst their top three. 42% claims the reason is the imprecise nature of their corporate objectives and the consequent difficulty of specifying computer system requirements. It is often difficult to distinguish between failures in requirements engineering and problems elsewhere in the software development lifecycle” by (Grindley, K. 1995). Argued by (Leveson, 1986) that most software reliability models define failure in terms of deviations from the software requirements specification; most accidents involving software are due to errors in the software requirements specification.

According to (UK Health, 2003) they did a detailed review of the causes of software failures and “requirements issues accounted for 40% of the incidents, hardware failures for 26%, software bugs 11%, maintenance issues 6% and system use around 17%”. According to (Standish, 1995) CHAOS survey, the top two “project impaired” factors were “incomplete requirements” and “lack of user involvement”. On another place mentioned by (Annie, I. Anton, 2003) that “clearly, we are facing a requirements engineering crisis. Finding and fixing a software problem after delivery is often more expensive than finding and fixing it during the requirements and design phase”. As (Barry Boehm, and Basili, Victor, R. 2001) observed that major driver in focusing industrial software practice on through requirements analysis and design are the major sources of avoidable rework involve hastily specified requirements.
This study was carried out in 1968 and it led the company to identify systems analysis, and the specification of requirements, as the two key activities which governed programming productivity by (Grindley, K. 1995). For giving you some evidence of how much time can a delay takes due to inadequate specified requirement specification; needs to see the summary of a case study which is presented by an automobile manufacturer, in the US, that he analyzed how programmers spent their time and he produced the following summary: (Grindley, K. 1995)

- Writing of software 4%
- Testing of software 38%
- Delays (requirements specification queries, errors and changes) 48%

We must understand the problem before expressing the requirements for the solution. It is normal that small projects can succeed without formal requirements engineering, but any project of a considerable size and complexity requires proper requirements (Annie, I. Anton, 2003). As argued by (Nuseibeh, and Easterbrook, 2000) that the primary measure of success of a software system is the degree to which it meets the purpose for which it was intended. Broadly speaking, software systems requirements engineering is the process of discovering that purpose, by identifying stakeholders and their needs, and documenting these in a form that is agreeable to analysis, communication, and subsequent implementation.

### 2.4.10 Lack of Quality Management

Quality Management is necessary for preparing or choosing a best option. What is quality? Quality Management helps us in defining, refining and measuring the software development process. “Quality Management’s objective is to help system developers to deliver high quality systems those indeed meet the needs of their users” by (George, W.R. 1988). Customers are today demanding the right combination of product price and fair quality. Actually customers want value, value for their money. “Value means not overpaying for quality. Value is about real performance, informative advertising, and delivering on promises” by (Parker, M.M. 1995).

We have to make use of quality management systems in our documentation and try to reduce the error rate. It is only possible by developing or adopting quality standards to reduce error rate in software development process. A survey by (McManus, J. 1994) on staff attitudes to TQM undertaken within the IT department of a well known UK company and following were two major responses. First 98% of staff agreed that TQM is a not a just another management gimmick and secondly almost the same high percentage agreed that everyone could benefit from TQM. But there was a wildly varying response, when staff was asked that did TQM improved the way they worked, there was a 100% agreement that TQM is vital to the IT department’s future. It has been also noted that small pockets of resistance do exits, which can be attributed to the lack of awareness of the subject matter.

### 2.4.11 Lack of Proper Organizational Structure

It is important to discuss organizational structure for making necessary changes and improvement to avoid failure. “Government organizations are largely bureaucratic in nature and filled with counter productive rules and regulations. Many organizations often resist fundamental change however because the people within them wish to stick to a hierarchical model” by (Morgan, 1989). Even a small IT project team has a strict chain of command that is hierarchical in nature, the decision makers are different then the people doing the actual work. The managers engaged in systems planning must take into account the organizational
structure of the IT project team to ensure that communication is not hindered by layers of bureaucracy. (Morgan, 1989)

A common source of aggravation in software projects arises from ‘people’, but it is rarely admitted. “Personnel changes for whatever reason way through a project are clearly not desirable” (Trevor wood-happer, and John McManus, 2003). Organizations those have a track record for success in other areas are also at risk due to IT project failures. Focusing on previous successes will also result in harmonized structures, procedures, and personnel further restricting the organizations’ ability to adapt to change (Bernard, W. 2003). Most companies have some type of hierarchical organization, which may undermine the authority of Project Managers. Ideally, PMs should be able to control the company’s resources needed to accomplish their projects, and directly supervise their teams. Unfortunately, in the real world PMs have little or no authority, so they have to rely on the official process for communicating their needs across the organization. This cross-organizational communication can result in delaying project tasks, which can ultimately bring the project to failure.

The field study conducted in the General Dynamics Systems Integration Management Office (SIMO), Project Managers were asked to give the most common reasons for project failure. Survey results indicated that 75 percent of Project Managers are unhappy with the current structure of the company, and list it as the top reason for not accomplishing their projects on time. General Dynamics is broken down into departments; each department has its own manager, and the department is further broken down into sections managed by section supervisors. For this you can see following chart taken from this organization.

Figure 5: General Dynamics Partial Organization Structure

More than 50 percent of General Dynamics Project Managers attribute the success of their projects to the good personal relationships at work. In most cases, they don’t follow official communication channels. They establish good relationships with people from other departments, and approach them directly whenever they need to get the job done. This shows the importance of good interpersonal skills, but also the need for an organization structure that facilitates communication across the company. (Soraya, J. NetoAlvarez, 2003) and further Organizational structure also deals with soft skills because Project Managers need to adjust to the company structure, and be able to find effective channels of communication. In order to get all the key players involved, project managers need top management support to schedule meetings, and to use the company’s resources.
2.5 Summary

In this chapter, I worked with the aim of collecting and analyzing literature review as per my research targets. Now, I am going to present you the summary of this chapter in detail.

Today, IT is a major department in most of the organizations and its impact is felt throughout the company. With the increase in IT spending, IT projects failure is increasing alarmingly. First of all I need to discuss the question that what is a project failure? Why it is happening so often? What could be the causes of this failure? Is there any role could be played by effective management in this regard? What is relationship between IT and management for completing a software project successfully?

When I started examining failure phenomena, I came to know that there is little understanding regarding software project failure and it is further a problematic that there is a lack of clarity as to what is a software project failure? When a software product/project is unable to perform, there could be countless reasons. IT experts are agreed that such failures occur far more often than they should. Furthermore these failures are now well into the billions of dollars a year and that’s why it is most important and urgent matter for an organization to primarily find its causes and secondly try to avoid them.

First of all, I tried to present you the introduction of the scope of the study and move towards the brief introduction and literature available regarding to this problem. Further, in this regard I tried to present important literature review, reports and case studies on the issue of software project failure existed in our economy. The ability to use IT software’s effectively and efficiently is a success factor for the organizations today. In this respect, I tried to present enough data to show the relationship between IT and management. It is also proven with presented literature that effective use of management for proper implementation of well established software in an organization is very important. After that, I move towards the main area of research which is “the major causes of software project failure”. In different studies they have mentioned different causes and symptoms of software project failure. After reviewing the available literature; I have selected some common causes of software project failure which are mentioned in this chapter in detail. As you get noticed that all the time I was talking about the failure of software projects. It was necessary to draw your attentions on the failure of software projects and as well as on the major causes of software failure. Some extend, I am satisfied that I tried to come up with as much literature review as I can to convince you that there is more or less failure in almost every company and this problems is growing day by day. We need to get rid of this failure problem or at least reduce it. As it is proven by the literature review that software project failure is an avoidable problem. In coming chapters, I will validate these reasons on the basis of empirical study.
CHAPTER 3: RESEARCH METHODOLOGY

In this chapter, my basic aim is to tell you in detail about the processes adopted in answering all the research questions mentioned in chapter one. This chapter will summarize my work from beginning to end. This research work is aimed at sourcing and analyzing data. I will try to explain all the steps to reach my objectives and problems encountered in this research process.

3.1 Data Collection

There are two sources available for data collection for the purpose of research. These data sources include the following:

- Primary Data
- Secondary Data

For the purpose of my research thesis, I used both primary and secondary sources for data collection. I collected secondary data from books, articles, journals and internet etc. This secondary data helped me in answering the questions and also provided a better understanding about the research topic. Once there is enough data available on a certain research topic, a researcher needs to collect empirical data which is called primary data. This primary data can be collected through certain ways which could be observation, questionnaire, and communication and through experiment.

I tried to present and verify my research work through both the secondary sources and primary sources. In primary source of data collection, I used questionnaire survey as in this way I could be able to validate my theory with practice. I collected this primary information from the people who are practically taking part in different information technology projects. Their hands on experience testified my theory in a good manner. Thus, with a mixture of both primary and secondary data source, I was in a better position to present my results.

3.1.1 Primary Data

There are many choices available regarding primary data collection i.e. Observations, Experiment and Communication. As mentioned by (Ghauri, P. & Gronhaug, K. 2005); “that there are some advantages and disadvantages of primary data like long time and cost involved; further difficulties in getting access to respondents, careful application of tools and methodology for analysis, and at last the quality of information can be affected due to the ability of respondents”. In my research study, I have used survey questionnaire to collect required information from respondents. In the figure below, you can get more clear idea about the sources of primary data collection:
3.1.2 Secondary Data

According to (Hussey, J., and R., Hussey, 1997); secondary data is “the data which already exists”. Secondary data is useful and is easily available information to solve the research problems as it is pre-published information. My secondary data is extracted from different sources ranging from books, journals, electronic book databases, surveys, publications, and dissertations etc. We can easily understand the problem with the help of secondary available data and then can explain it according to our research problem for achieving our objectives.

It is obvious that secondary data has some benefits for researchers in sense of saving money and time. It is also reliable and can provide quality information for our research purposes. It is important here to mention as said by (Churchill, 1999:215) that “Do not by pass secondary data. Begin with secondary data and only start with primary data when the secondary data start showing diminishing return”.

Secondary data can be divided into two main sources which are:

- **Internal Data**
- **External Data**

Both sources of data can be explained in detail in the following:

**Internal Data:** It is the available information from organization’s internal resources like their internal studies on the problem and different issued reports and done analysis on relevant problem.

**External Data:** These belong to the resources which are relevant to the company but available publically or privately like company annual reports published through stock exchange etc.

Following figure can give you a clearer picture of above mentioned two sources of information:
3.2 Questionnaire

3.2.1 Methodological Approach

I have three main research questions which are listed in chapter one. In chapter two I tried to cover my first two research questions in detail and also chapter two gives me the necessary insight and necessary background information on my research issues.

In the next chapter, I will try to investigate and analyze my research questions through empirical study. Further, I will also intend to cross check and validate my selected root causes of software project failure through this empirical study.

3.2.2 Design of Questionnaire

I basically designed my empirical study questions by focusing on main three research questions presented in chapter one. I included both close ended and open ended option in some questions for fulfilling my research purposes. One of the main objectives of this empirical study is to ensure the causes of software project failure. For this purpose, the empirical data is acquired from three Danish firms and some disperse people working in different small, medium and large scale organizations and have some sort of link with IT software projects. The reason behind selection of these three firms and other personals is that I can get my required information from these sources easily on the basis of available time and resources. Furthermore, I have selected one organization from each level depending on their employees strength level, like first organization is from large scale (The National Survey and Cadastre), Second is of Medium scale (Vopium A/S), and third one (XStream Media Solution APS) is from small scale and along with this, there will be some other people from other organizations as well. In this way, I will be able to cover as much as I can for getting best results from this empirical study.
The survey questionnaire has been made according to the needs for both the Technical people and Management people. Some questions focus upon the technical persons who have sound technical knowledge and mainly work in the software development teams and as well as upon the management persons who have more managerial knowledge and experience. These are the persons whose work is to organize the work and guide the software development team according to project plans. I basically based my survey questions on my research questions. I made most of the questions close ended, with addition of few questions with open ended options where it was unavoidable. This helped me to collect genuine and accurate answers from respondents. In addition, I also presented some descriptive type of options, which allowed the respondents to express their personal point of view.

3.3 Selected Sample (Organizations)

3.3.1 National Survey and Cadastre Denmark

**Mission:** The National Survey and Cadastre is Denmark’s public authority for the management and distribution of geographic information.

**Vision:** The National Survey and Cadastre's vision is to provide Denmark’s public sector, private companies and general public with accurate and updated geographic information about the territories and waters of Denmark, Greenland and the Faroe Islands.

Responsibility and Tasks: The National Survey and Cadastre is Denmark's central authority for surveying, topographic- and hydrographical mapping, land registration and the authorization of surveyors.

Since 2002, the National Survey and Cadastre has been responsible for Denmark's national geographic infrastructure. The country is moving towards the implementation of a broad E-Government strategy in which geo data will serve a central role. The National Survey and Cadastre promotes Denmark's goals for more efficient public administration.

**History:** Although the National Survey and Cadastre was not officially established until 1989, the authority's roots extend back to 1757, when the Royal Danish Academy of Sciences and Letters undertook the first comprehensive triangulation survey of Denmark. The surveying work was handed over to The Danish Survey in 1816, while cartographic responsibilities came under the Danish Army's Topographic Mapping Department in 1842. These two institutions were merged in 1928, forming the Danish Geodetic Institute.

The history of systematic charting of Danish waters reaches at least as far back as mapping of the country's territories. The country's first centralized nautical chart archive was established in 1784. Hydrographic surveying, chart production and distribution became the archive's official responsibilities. The history of Denmark's modern cadastral mapping system began in 1804, when the Danish government recognized the need to maintain accurate property records for taxation purposes. The cadastre was- and remains- a concrete tool for protecting land owner's property rights.

The National Survey and Cadastre itself was established in 1989 with the merger of Denmark's Geodetic Institute, Hydrographic Department and Cadastral Department. The authority operated under the Ministry of Housing until 2001, when it was transferred to the Ministry of the Environment. Until 2004, the National Survey and Cadastre functioned both as a mapping authority and as a geographic sector research institute. On July 1, 2004, responsibility for seismology was transferred to The Geological Survey of Denmark and Greenland (GEUS). Six months later, on January 1, 2005, all research activities related to cartography and geodesy were transferred to the Danish National Space Centre.
3.3.2 Vopium A/S

**Mission:** Vopium’s aim is to focus on global market of communication, voice and data.

**Vision:** The vision of the company is to create a global mobile internet Protocol (IP) network and deliver next generation voice, video and data services to international consumers; transcending cultural, technological and geographic barriers to bring people and business together through superior communications at exceptional value.

**History:** It is a mobile Telecommunication Company with new solutions for their customers founded in 2006. They want to radically change global communication from fix line and PC to the mobile phone. Vopium specializes in the development of software solutions for mobile telephones intended to reduce the cost of international telephone calls. Vopium is listed on NYSE Euronext Paris in 2008. For our questionnaire we just focused on head office as it’s a multinational company with head office situated in Copenhagen, Denmark.

3.3.3 XStream Media Solution APS

XStream is developing and selling complete broadcast solutions for distribution of streaming media via the Internet, mobile and TV platforms. Since 2000, XStream has developed and provided communication solutions based on streaming technology. Their head office is based in Copenhagen, Denmark. Today, their main focus is on the development and sale of the company’s market leading media assets management system for managing and distribution of streaming media.

**MediaMaker:** Is used by many large corporations and major media a company in Scandinavia, MediaMaker has become the market leading media assets management system in the region.

They are basically selling products to their customers and for making and maintaining these products. Their project managers and staff are working on all these products as a project. These all products are their core business so they are also providing services and solutions to their partners and customers in this respect.

3.4 Size of the Sample

The questionnaire is distributed to 30 people related to software projects development and management in some respect. In this there are some people who have direct role in developing and maintaining the software projects but others are involved in managerial duties. Further, I distributed 20 questions among different people who belong to different organizations. I believe that this questionnaire will serve best for the purpose; if filled by selected people, who have experience of dealing with different kind of software projects and their experience can enrich my research in positive way.

3.5 Method for Submission of Questionnaire

For the purpose of this research thesis, primary data will be gathered using communication which includes the use of surveys i.e. questionnaire which will be distributed personally and through online survey software to get the necessary data. I will try my best to assist the respondents in case of any ambiguity in questionnaire. I am very optimistic to get efficient and effective response from selected respondents.
3.6 Limitations

3.6.1 Limitations of the Questionnaire

The questionnaire was sent to above mentioned number of people in those three selected organizations and other individuals as well. It was not feasible for me to meet them in personal as they were at scatter places and they are very busy people. There is a possibility that I may face a less rate of response due to the lack of time and interest in answering the questions. It is also a possibility that in my selected sample of population, some will be more experienced while others may not. I have the same probability that there would be diverse opinions from the respondents on the basis of their knowledge and experiences. Finally there could be some missing points or aspects in this survey questionnaire which I would come to know during or after the completion of the survey.

3.7 Data Analysis

The next chapter will cover the empirical study done through questionnaire survey. I will present you the results gathered from the respondents and its analysis. At the end you will find the comprehensive discussion on the empirical study linking with the literature review and presented causes of software project failure.
CHAPTER 4: DATA ANALYSIS

This chapter is comprised of data analysis and discussion. I already mentioned in research methodology chapter, that enough secondary data has been used to find the main causes of software failure. In this chapter, I will try to cross check these causes and validate them with the help of questionnaire survey. Due consideration and care has been taken to ensure that the data presented is from reliable sources.

4.1 Questionnaire

First I distributed 30 copies of questionnaire among three selected organizations. I got 100% results from two organizations (KMS and Vopium) and the response rate from XStream was 40%. Furthermore, I distributed 20 more copies of questionnaire and got 100% response rate from individuals working in different organizations. You can see an overall picture of distributed questionnaires and their response rate from the following table:

<table>
<thead>
<tr>
<th>Organizations</th>
<th>National Survey</th>
<th>Vopium A/S</th>
<th>XStream APS</th>
<th>Individuals</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Distributed Questionnaires</td>
<td>10</td>
<td>10</td>
<td>10</td>
<td>20</td>
<td>50</td>
</tr>
<tr>
<td>Received Questionnaires</td>
<td>10</td>
<td>10</td>
<td>4</td>
<td>20</td>
<td>44</td>
</tr>
<tr>
<td>Response in %</td>
<td>100%</td>
<td>100%</td>
<td>40%</td>
<td>100%</td>
<td>88%</td>
</tr>
</tbody>
</table>

Table 3: Percentage of Received Questionnaire

4.2 Analysis and Results

The results of questionnaire are divided into three sections:

- Introductory Questions
- Questions related to the Causes of Software Project Failure
- Management’s role in Software Project Development

I developed 14 questions for the purpose of survey and analysis. In section one I have few questions which are of general nature like asking about respondents experience and the number of projects they had worked on as a technical or non technical person. In section two, I asked the respondents point of view on the causes of software failure. In third section, I included few questions regarding the role and importance of management regarding software projects. This survey includes both areas of my research interest (IT and Management).

- In first section, I have selected first four questions which are (1, 2, 3, and 4). These are for the purpose of introduction and to know about the people involved in this survey. It was necessary to know that how many people are taking part with pure IT knowledge and how many with pure knowledge of management. There would be some people with little knowledge about the software development process and for them; I have a section with the name of “others”. Respondents who don’t have much knowledge of management or IT and it will have some impact on analysis and results.
Q.1: What kind(s) of software development services your organization provides?

![Diagram showing percentage distribution of software development services provided by organizations: Long Term Projects 44%, Short Term Projects 9%, Both 38%, Others 9%]

Figure 8: What is the core business of your Organization

In figure 8, I have 44% respondents who are working on “long term projects” and only 9% working on “short term projects/products”. 38% are with the knowledge and experience of working on both the long term and short term projects. 9% of respondents are those who are not working on either long term or short term projects. 9% are not involved in any of the above two categories. I have majority of respondents working on long term projects or either on both same time. I have enough technical people to for the survey to know the true causes of software failure.

Q.2: Have you ever been participated in the development of software project/product development?

![Diagram showing participation in software project development: 80% as a Technical person, 9% as a Non Technical Person, 7% through Literature, 4% not participated in the process]

Figure 9: Participation in Software Project Development

The purpose of Q.2 was same as Q.1 to know about the respondent’s area of expertise and their knowledge about the software projects. In figure 9, I have 80% respondents with technical knowledge and experience. I have very little ratio of 9% respondents with no technical knowledge and may be they have good knowledge of management. It is due to the selection of IT firms and it is obvious that majority of their employees are with experience.
and knowledge of IT. It is the reason; IT firms are managing everything with the help of IT professionals rather than taking help and incorporating professionals with pure management knowledge. There were 7% respondents who have the knowledge about the process of software development through literature review and 4% respondents involved in our research were without any knowledge about the development process of software projects.

Q.3: If you have participated in software development then what was your role in that team?

![Figure 10: What was your Role in Software Development Process](image)

In above figure, 52% of our respondents are software designer/developer. Second highest ratio is 25% and these respondents are working as a team leader. On a third level, there are people who never fall under any specified fields but majority of them were working on some IT related fields like IT designer, software tester, PM technical and application support etc. In last, there are respondents who are working or worked as a project manager (Non Technical) and their ratio is 10%. Project managers are very important for my study as their input will make the survey more balanced in the sense of managerial people involved in it. In this figure, I have enough respondents with technical background and they will help my to know the technical factors important in the success or failure of a software project.

Q. 4: In how many software projects/products have you been involved?

![Figure 11: In how many Projects you have participated?](image)
In this figure, I asked my respondents to tell me in how many software projects they have took part. I have 9% respondents who worked on 20 or more projects. Some of them worked in around 30 to 50 projects and it is a great number of projects. The highest number of respondents worked on “10 to 15 Projects” and this ratio is 30%. Second highest ratio is 23% and they are who worked on “5 Projects”. Further, there are some respondents who had worked on less than five projects and it also includes those who just worked on one or even never worked on a single software project at all. The ratio of respondents with (0 to 4) projects is 16%. I have many respondents who have good knowledge and experience on the basis of their participation in different software projects. It means they are the right people with right knowledge to leads me towards achieving the objectives of the survey.

- In 2nd section, I have couple of general questions and also couple of specific questions related to my research study. In this section, I will try to know that how much my respondents have knowledge about the software project failure and how they define it. I have aim to collect the point of view on major causes of software failure and how these factors are responsible for such failure. I included question number (5, 6, 7, 8, 9 and 10) in this section to know their point of view on software project failure.

Q.5: How do you define/describe software project/product failure?

![Diagram of Define software project failure](image)

Figure 12: Define Software Project Failure

It is one of my core questions. I have made it with the help of mentioned definitions in literature review chapter. In this question I used some areas which are important with respect to the software project failure. I have made some concise options which are mentioned in above figure and those are (Project Requirement Specifications, Project Time deadline, Quality Aspects and Budget Parameters etc). 26% respondents are agreed that a project is failure when it never comes within defined time. Time is an important factor as I also mentioned that “lack of project management and planning” make the projects impossible to finish within defined time. Planning and Management are useless if they are not able to manage the timely completion of a project. In the second category, 32% said that they consider a project failure when it never fulfills the quality aspects of a project. Quality is a necessary element as company pay to get high quality projects. Third important factor is budget and 15% respondents are agreed that a project is not successful if it never completed within defined budget. For checking the importance of budget, you need to read out the literature review chapter. At the end, there was an option which is inverse of above three and
that is “A project is not according to specifications but fulfils time, quality and budget parameters” and 27% were agreed that a project is defined failure when it never fulfils basic requirement specifications. I have mentioned in my literature review chapter about the importance of requirement specification, time, budget and quality aspect in detail with evidence that these factors are very crucial for the success of a software project.

Q.6: What was the ratio of successful versus failed software projects/products in which you were involved? (Divide 100% in A & B)

<table>
<thead>
<tr>
<th>Successful</th>
<th>Failure</th>
</tr>
</thead>
<tbody>
<tr>
<td>80%</td>
<td>20%</td>
</tr>
</tbody>
</table>

Figure 13: Ratio of Successful versus Failed Projects

It is a general question to know the experience of respondents regarding the ratio of successful versus failed software projects. Most of the respondents said that they have minimum failure rate in the projects they worked on. Some respondents even answered that they have 100% success rate and other said that they have minimum rate of failure in their worked projects. Only five to six respondents said that they have 30% to 40% failure rate. It is important to note that I have never asked them to mention the rate of failure separately for short term and long term projects. According to my presented literature review in 2nd chapter, there is very less rate of failure in short term projects/projects and in case of long term projects the rate is very high. According to some studies and research work it is more than 70%. In this survey response, I got totally opposite experience comparing to my literature review. There could be two reasons; one is that respondents are sometimes mixing short term projects with long term projects and sometime they never feel good to accept that they have done something wrong.

Q. 7: When you consider a software project/product partial or complete failure? (Please specify all those apply)
When you consider a Project Partial or Complete Failure?

- Project has not been delivered with in the defined time: 21%
- Project does not meet the user requirement specification: 34%
- Project does not completed within defined budget: 16%
- Project does not meet the defined quality standards: 20%
- Others: 9%

Figure 14: Partial or Complete Project Failure

In this question, I asked the respondents same question as I asked earlier in Q. 5. This time, I simply asked them to mention the causes of software project failure without any connection to requirement specifications. Their response rate for requirement specification is 34%, for time factor 21%, for Quality 20%, for budget 16% and 9% gave other reasons for software project failure. According to response rate, it is clear that they are giving importance to the “requirement specification”. I have comprehensive discussion on literature review chapter to show the importance of requirement specifications. Project should have those features which are demanded by the clients. The second highest percentage is for “time” factor and “quality” is on third level with slightest difference as compare to time. In Q.5, respondents gave 32% to the “quality” aspect which is highest in that figure. In this figure majority of respondents gave high importance to the requirement specification, quality, and budget following with time factors. Only 9% gave some other reasons for software failure different than mine.

Q. 8: What can be the effects of software failure for the organization as a whole? (Please specify all those apply)

- Financial loss: 22%
- Time loss: 22%
- Lowering of Performance/Moral: 17%
- Depletion of assets: 6%
- Loss of shareholder confidence: 14%
- Job losses: 8%
- Bad press/media publicity: 11%

Figure 15: Effects of Software Project Failure
In figure 15, I asked a general question to show you the possible effects of software project failure for an organization. In case of software failure, organizations could face one or more effects at the same time. Respondents mentioned “financial loss” 22%, “time loss” 22%, and “lowering of performance” 17%. These are few main effects of software project failure for the organization but some respondents also mentioned few other reasons. In my literature review chapter, financial loss and time loss are the two major and common effects of software project failure and I got the same highest ratio of these two factors in this figure.

Q. 9: What are the basic skills and abilities a team member should have for successful project completion? (Please specify all those apply)

![Skills and Abilities necessary for Team Members]

Figure 16: Skills and Abilities necessary for Team Members

In figure 16, I presented the data collected through questionnaire survey against above mentioned three heads. 30% respondents are agreed that “education of team members” is very important for the success of a software project. It is necessary for the project team members to have knowledge about the tools and technology used in project execution. Second highest response rate is in favor of “all above” option and it is 29%. These respondents are agreed that all first three options given in this figure for supporting the skills and abilities of the project team members are equally important. On third level I received 22% response rate to the “specified role and responsibilities” for them. It is the duty of management to provide them specific roles and responsibilities according to their skills and knowledge. It will be good for the organizations and as well as for the projects. Then, we have option of “coordination among the team members” and response rate is 19%. It is easy to see that most of the respondents are giving the value to the technical aspects of the projects and less value to the managerial aspects. It is due to the role and experience of our majority of respondents participated in this survey. According to literature review presented in 2nd chapter the both technical and managerial factors are crucial for the success of a software project.

Q. 10: What are the main causes of software project failure according to you? (Pleas specify three or all those apply)
In figure 17, I choose the same causes as I had mentioned in chapter two. My survey respondents gave 18% vote in favor of “lack of project planning” and “communication” equally. 14% respondents selected “lack of project management” as a second major cause of software project failure. 11% selected “lack of quality management” as a third major cause of software project management. 10% selected “lack of user involvement” in the process of project management as a major cause of software project failure. Further, they gave 8% to the “lack of budget management” and 9% to the “lack of top management involvement”. Overall, all the factors are important in the success or failure of a software project management but with the help of respondent’s response, I can see which factors are more important in respondents point of view. I am happy that respondents are totally agreed with my selected reasons of software project failure. Some reasons are more important to them and some are less but as you can see that only 2% respondents presented other reasons. It means, only 2% respondents gave some other reasons then presented by me.

- In third section, I have few questions to see the importance of people involved in the process and what role they can play. It is obvious that without technical people we can not complete a software project. Similarly, it is important for us to know other important factors for the successful completion of a software project. I have asked my respondents four questions in this section, which are (11, 12, 13 and 14). With the help of these questions, I can get the respondents view point that what kind of personals should be involved in the process. How much knowledge they should have about the IT and Management for working on a software project and making it successful. Further, through couple of questions, I tried to know that what kind of support and role higher management play in making a project successful.

Q.11: What kind of people should be involved in software project team? (Please specify all those apply)
What kind of people should be involved in Software Project Team?

Figure 18: Participants of Software Project Team

This question is again made on the basis of data presented in literature review chapter. With the help of collected data and presented results in above graph, I can know the ideal team members for the development of software project. It will tell us the important participants and their role in the success of a software project. Few researchers mentioned the above parties important for the successful development of the software project in literature review. Let us see which party is more important according to survey respondents. Respondents gave 22% to the “technical staff/developers”, 21% to the “project manager”, 20% to the “team leader”, 18% to the “users of application”, and 10% to the “stakeholders” and finally they gave 7% to the “sales and marketing team”. I am also agreed that above mentioned personals are very important for the successful completion of a software project. In addition it is necessary that we should involve the “users” and “stakeholders” in this process as users have to use this software for the benefit of stakeholders. I think “team leader” and “technical staff” is necessary to develop a project but along with them a software “project manager” can play a vital role in the development and success of a software project. An important function of the “project manager” is to allocate resources to various activities, and manage project planning, estimation, control, quality, risk management and communication etc.

Q. 12: Do you think proper attention of higher management can play a vital role in Software Project/Product success?

Figure 19: Role of Higher Management in organizational goals
In figure 19, I tried to know about the “role of higher management in performing collective goals”. According to literature review higher management involvement can play a vital role in the success of organizational goals. In this regard, 91% were agreed with the statement that “Higher management attention and interest can play a vital role in the success of a software project” and 9% of respondents disagreed with above statement. It is clear that higher management involvement is very crucial for project success. It is also mentioned in Standish group report that “higher management involvement” is very important for the software project success.

Q. 13: Do you consider higher management’s attitude can play any role in achieving your goals assigned by the company?

![Higher Management attitude is Important in achieving your individual goals](image)

Figure 20: Role of Higher Management in your individual goals

In previous figure, I asked the respondents about the role of higher management involvement in the success of their collective goals. Here, I am asking about the “role of higher management in performing their individual goals”. 98% respondents are agreed that higher management support can help them in performing their individual goals and only 2% said that it has no effect and concern with performing their individual duties and responsibilities. Again I will refer you to my literature review that “Higher Management Involvement” is supportive for overall organizations goals and as well as individual goals.

Q. 14: How much technical knowledge is sufficient for a software project manager?

![How much Technical knowledge is sufficient for a Software Project Manager](image)
In figure 21, I asked my respondents to tell me that a “project manager” should be the part of software development team or not. 21% said that a “project manager” should be involved in that team and it is the second highest ratio in that figure. So, it is obvious that a “project manager” should be involved in the team and I also mentioned in literature review the list of people who should participate in the process of software project development. I asked the respondents to response about different given percentages of the technical and managerial knowledge of a software project manager. It is contradictory issue that how much IT and Management knowledge a software project manager should posses. 33% respondents are in the favor that a project manager should have “technical 50% and mgt. 50%” knowledge. In second highest ratio is 31% and they said that he should have “technical 70% and mgt. 30%” knowledge. On the third level, 29% said that “technical 30% and management 70%”. In “others” I got 7% response rate and they have given their personal ratios for the mixture of knowledge of software project manager. It is good that respondent’s high ratio agreed that a project manager should have equal knowledge of both areas. I got 2nd highest response that a project manager should posses 70% technical and 30% management areas. But in my opinion a project manager should have more knowledge of management or at least equal and for the reason you can see explanation given in figure 18.

4.2 Discussion

In this section, I will try my best to discuss and correlate questionnaire survey results with the literature review presented in 2nd chapter. Here, I will only discuss the few core questions which are important with respect to my study. This discussion will try to cover all those core areas which are either covered or not above and the purpose of this discussion is to summaries the important areas here. In first four questions, I tried to get basic information about the respondents, their organizations, and their role in the projects etc. These things will help us to know about the respondents. I have most of the respondents with enough knowledge and experience regarding software project development. In Q.5, I tried to validate the definition of software project failure presented in literature review. My respondents are agreed and they have given high importance to the requirement specifications. It is second highest ratio of respondents those agreed that a software project is a failure if never comes according to requirement specifications. Others are those, who gave almost equal importance to the time, quality and budget for considering a project successful. Once again in Q.7, I asked the same question from the respondents with slight changes and in a simple way. This time I got the same results as I was expecting. It conformances with my results received in Q. 5. Here 34% respondents were agreed that a project is failure, if it never telly with initial requirement specifications. It is a significantly high percentage of respondents. Again some respondents are agreed that “time, quality and budget” are also important elements for deciding a project is successful or failure. So, final conclusion from Q.5 and Q.7 is that requirement specification is the very important factor in case of software projects along with time, budget and quality factors. After discussing Q. 5 and Q. 7, I will move towards Q.10. I basically made this question to validate all my major causes of software project failure presented in chapter two. In the result, I got highest response in favour of “lack of project management”, “lack of project planning”, “communication failure”, “and technology illiteracy”, and “lack of quality management” etc. On above mentioned causes respondents were agreed with me that these are the most important and common factors. Same factors are the example of few core areas responsible for the software project failure. Now I validate them with empirical study. Few respondents mentioned some other reasons of software failure which were different than my presented problems. But this percentage was only 2%. It shows that all our respondents were almost agreed with the causes of software failure mentioned by me in literature review chapter. In Q.11, I tried to ask my respondents about the parties should be involved in software project development process. In answer of this,
they were agreed that a “team leader”, “project manager”, “technical staff” and “user of application” should be the part of this team. A high ratio of respondents mentioned that “team leader”, “project manager” and “technical staff” is very crucial parties for this process. Further, they have given fourth ranking to the “users of application”. It is very important as the all products will be used by the users. So, they should be the part of this process and they can guide us that what is required by them. Further, on the basis of reviewed literature involvement of “stake holders” is equally important. They can be helpful in this process and their participation will give us the clear picture and support will help to make us a successful software project for their benefits. With the help of collected data through Q. 5, 7 and 10, respondents have given high percentage to the “Quality assurance manager” for take caring of all quality matters in the development of software projects. A quality professional is needed, who have extensive experience and knowledge about the quality related matters. In case of Q.12 and Q.13, almost all respondents are agreed that “higher management involvement” is important and helpful for accomplishment of “organizational” as well as their “individual” goals assigned by the organization. So, the effective management involvement is important in defining, developing and implementing the management strategies to meet the technology challenges and business needs. It was a useful survey which helped me to know the point of view of experts on the software project failure and other concerned issues. A strong point about the survey is that majority of respondents involved in this survey were experienced people. Most of them are currently involved in managing and developing software projects in different organizations.

4.3 Limitation

Unfortunately my survey has limited number of Project Managers due to the lack of time and resources. Further I never know that those who have taken part in my survey are equipped with well enough Project management training and knowledge or not. A larger sample is required to include other companies for making this survey more productive and enough number of project managers from different companies. I have more respondents from technical side, especially developers. It can affect the results and analysis of my survey. I personally think it is sufficient for this study, as I had covered much in my literature review. There will be no failure and no learning if everything is perfect. I made this survey for supporting my literature review and some extends I accomplished my objectives.
CHAPTER 5: FINDINGS

In this chapter, I will try to discuss about the research questions. I will try to describe about the research questions and their presented answers in different chapters. Here, I tried to cover my research questions in relevant chapters in very detailed way. I will try to link my research questions with the presented answers in this study.

- Here, I am going to present a short summary of my thesis in a very concise way.

This study revolves around the core issue of finding the root causes of software project failure with respect to organizational factors. In this study, I have tried to find the organizational factors contributing towards the failure of software projects. Study is comprises of literature review and questionnaire survey. There could be one or several factors responsible for the software projects failure. I have finalized some important causes of software failure on the basis of literature review and empirical study in chapter two. Further these finalized causes of software project failure are again validated with the help of questionnaire survey in chapter four. I have presented a comprehensive analysis of the gathered data from respondents. For avoiding any aspect of the analysis, I have further added a detailed discussion on data gathered through survey. I have slightly touched the Information Technology with respect to management’s role in software project development. Information Technology is playing a very vital role in today’s organizations for competing on world level. Software failure is the biggest challenge faced by IT as well as business people. In this way, software failure is very important issue for software development firms as well as buyer and user firms. There is strong need to find the root causes of software project failure and mitigate them.

In currant age, the effective use of IT is a success factor for any organization. It is only possible if we link IT with organizational goals. Business and IT managers need to learn that how they can measure, manage and justify technology as a business matter. The example of ideal organization is that which gives value to the collaboration, openness, and communication.

The insight gain through this research is the basis for describing the solution for software failure problem and it is presented in chapter six (Conclusion and Recommendations) briefly. The concerned parties will be able to get the benefits from this study to avoid the failure problem. The contribution of this research is twofold. First, it will be helpful for the software making professionals/companies and secondly, it will be helpful for decision makers/users (Organizations). Especially, when they are going to buy or implementing a software project for enhancing their productivity.

5.1 Limitations

Here, I can only give you a very brief overview connecting to the answers of the research questions (RQ). I can only summarize the answers of the research questions in this chapter. In case you want to see the full description of the answers, you need to read the particular chapters made for providing you the answers of the research questions.

5.2 Research Question 1

RQ 1: What is the state-of-art of software project failure due to inefficient management operations?
I have presented the answer of RQ 1 in chapter one (Background) and in chapter two (Literature Review) precisely.

Different authors presented different definitions for software failure. Some of them said that a project is partial or complete failure if it never meets the initial requirement specifications. Others are considering time, budget and quality a key factor for the failure. Everyone has his own definition of software failure based on the requirements of the customers. Similarly, respondents of empirical study are agreed that “requirement specifications, budget, time and quality are the key factors responsible for the software project failure but still there were some respondents who have different point for defining software project failure. In this way, I can conclude that there is no fixed definition of software project failure existed. Every organization have different objectives attached to these software projects and in same why different factors are important for them to consider a software project failure or successful. It could be requirement specifications, time, quality, budget or anything else. Due to this uniqueness of objectives and important key factors, I am not in a position to generalize any definition of software project failure in the industry. Everyone has different perceptions of failure according to their needs and objectives. It is very important for the organizations to define key factors for their software projects and as well as the parameters of success and failure for the software projects in advance. In this way they can know that where they are going and how they have to reach there which left no ambiguity in the aim and objectives of the organization.

In second part of RQ 1, I am considering high IT failure rate in the economy is due to efficient management practices. For the answer of this portion, I will refer you towards chapter two (Literature Review), where I had presented a brief introduction of management and its possible role in controlling the high IT failure rate.

In this, my emphasis is to provide you the evidence that effective role of management can help us in controlling the failure problem. Different studies and authors placed the responsibilities of software project failure on sloppy management practices. I have tried to prove it with the help of literature review and empirical study that effective management is very crucial for the successful completion of any project. Without effective management and planning we will end up no where. Even more than 95% respondents were agreed that the role of executive management is very important for project success. I have also asked them about the role of project manager for effective implementation of management tools and techniques for handling the software projects. Again majority is agreed with the role and importance of management in this matter. Respondents were also agreed with the role and importance of management. It means effective management has a key role in defining, measuring, implementing and controlling the software projects. The role of management is unavoidable and it is the right route to achieve organizational objectives. Software teams can work with coordination of management teams to get better results and effective management of software projects.

5.3 Research Question 2

RQ 2: What are the possible factors of software project failure and how each is responsible for the failure?

In RQ 2, I have tried to find out the root causes of software failure with respect to management. The answer is presented in chapter two (literature Review) and chapter four (Data Analysis).

In RQ 2, I tried to find the root causes of software project failure. There could be the number of factors responsible for software project failure i.e. “Lack of project management and
planning, Inefficient resource management, Inefficient time management, Lack of users
training, Lack of stakeholder involvement, Lack of quality management, Communications
failures within teams, Cultural gap existing between IT and business professionals etc. Same
way I have also selected above mentioned and few others on the basis of presented literature
review. In second part of the chapter two, I have presented the root causes of software
failure. So, it is important to read their full description in literature review chapter for having
idea about their role in software project failure. For knowing about these factors and why
they are responsible for failure you have to read the presented literature review in chapter
two. I earlier mentioned that their could be several reasons responsible for software project
failure but my selected and discussed factors in literature review chapters are almost
common in reviewed case studies and reports. For providing more strength to my presented
causes of software failure I have validated these reasons with the help of empirical study. I
have tested these causes with the help of few questions in my empirical study. Respondents
were pretty much agreed with me on the causes of software project failure. Some
respondents had given more importance to some causes according to their experiences.

5.4 Research Question 3

RQ 3: How can we overcome on all the factors of software project failure identified in RQ2?

- In RQ 3, I was intended to present you with my findings and recommendations for
  avoiding software project failure. In Chapter six (Conclusion and Recommendations), I tried my best to present you the possible recommendation for
  controlling or avoiding this failure problem. Conclusion and Recommendations are
drawn on the basis of literature review, empirical study, and data analysis. It is the
portion which based on my personal insight into the failure problem and contribution
  to avoid it as much as we can.

It is important to analyze the problem for reaching to any conclusion or suggesting a
solution. For this purpose we need to study the problem and reach to the roots of the
problem. The better you will understand the problem; the better will be your position to
solving it. In this chapter, I utmost tried to cover almost all causes of software project failure
presented in chapter two. In chapter five, I tried to present you the all possible
recommendations as a solution of the problem for avoiding software project failure in future.
Further, this chapter reflects my insight and point of view drawn on the basis of literature
review and data analysis.

5.5 Conclusion and Recommendations

In next chapter, I have presented you the conclusion and possible recommendations drawn
on the basis of done literature review and insight gained on the problem. This will be a
successful endure to save our readers from the usual factors of software project failure.
Majority of discussion in next chapter is relevant to the common causes of software project
failure which are faced by almost every organization today. For getting more precise idea
you need to give a look to the next chapter.
CHAPTER 6: CONCLUSION AND RECOMMENDATIONS

In this chapter, conclusion and recommendations will be drawn on the basis of literature review and survey analysis. This chapter is divided into the following parts.

- Conclusion (On the basis of presented research study)
- Recommendations (Some useful recommendations on the basis of literature review and questionnaire survey)
- Future study (Possible areas for future research)

6.1 Conclusion

IT software project failure seems like an intractable problem, but the research indicates that with an understanding of the problem and effective management strategy, one can control the existing situation. This will ultimately help us in bringing the projects within budget and on time with the specified requirement specifications.

The main purpose of this research was to outline the main causes of software project failure. At an abstract level, it is difficult to say how much money is wasted through software project failure. If I consider a conservative failure rate of 5% to 10%, then billion of dollars are wasted due to software project failure on yearly basis. Yet it is true that there will be some level of software failure existent even after doing every effort to control it. The increased social and business dependency on IT could be a serious issue in case of failure. We need to make these technologies more reliable as our future is dependent on them. High IT failure rate represents the existing gap between the current used technology and our knowledge to handle it effectively and efficiently. IT software development process is very fragile in nature and a mistake of one line can lead us to project failure. Sloppy development practices can be one of the major causes of software project failure. Here, I will emphasize on the use of effective management role. Secondly, the engineers should realize their responsibility in case of failure and try to approve the project only when they have tested it and checked all aspects of project/product quality. Every engineer should know all the technologies and procedures adequately to handle and avoid any problem.

A critical role is given to IT in private and public organizations today. It is due to the strategic role played by IT; if deployed efficiently and effectively to gain and sustain competitive advantages associated with it. Commercial organizations are now competing on global level and IT could help them in reaching existing and potential customers successfully. This increase in scale will increase the complexity and failure rate. We need to develop strategy and aligned IT with organizational goals for competing on global level. For achieving the available opportunities and benefits associated with technology; we need to consider the role of management. We can conclude that management has a key role in defining and implementing IT solutions to meet the business needs. It is famous that effectively working on right path is better than efficiently working on wrong project. So, the key to success is effective involvement of management in developing clear vision for supporting organizational goals and strategies. For achieving all the benefits associated with IT, it is necessary to teach the people involved in software projects about the relationship between technology and management. All this is possible if we incorporate management to boost this relationship. It is learned from the study that IT is always beneficial for the business, depending on how perfect and well developed the (Solution/Project) is according to the business requirements and implemented well.
The goal of this research was to analyze failure problem and come up with suggestions and recommendations, which can be helpful for the software practitioners and user firms. In current age, the effective use of IT is a success factor for any organization. It is only possible if they use IT with the strong linkage with organizational goals. Business and IT managers need to learn that how they can measure, manage and justify technology as a business matter. The example of ideal organization is that which gives value to the collaboration, openness, and communication. In this way, we will be able to resolve most of the failure problems faced by the organizations. Now, I will move towards the recommendations for avoiding software project failure.

6.2 Recommendations

- Project Manager should be the part of a software development team. Project manager’s skills are very important as software projects require a range of skills varying from soft skills to technical skills. These skills are necessary for managing projects and allocating resources, for planning, estimation, quality control, risk management, and handling manpower etc. So, project manager has a key role in software project team.

- A project manager should have formal training to deal with large and complex projects. In many organizations, projects managers are lacking with formal skills and training required for handling projects. It could be the one reason of software project failure, that management techniques, methodologies and tools were rarely applied in project execution.

- Planning is very critical for success, every organization is facing many problems and opportunities for development but the resources are scarce. In this situation, for addressing these problems and opportunities, we needed a strategic planning. In presence of strategic planning, we will be in a better position to decide which problem should be solved on urgent basis and which opportunity needs to proceed with.

- Projects should be started with an adequate budget. If projects don’t have required budget, the developers will try to decrease the scope of the effort and sometimes take shortcuts in gathering requirement specifications, review and testing phase. Inadequacy of resources can lead us towards bad quality projects and some times total disaster.

- Sloppy development practices are the major source of software project failure. This kind of practices can cause a project failure at any stage of development. The CMM model is introduced by SEI for attaining best practices in the industry. SEI, is expecting from developers to understand the quantitative variations in the processes for control and success in the development of a software project.

- Risk management is the least practiced discipline in IT project management. Effective risk management is crucial for IT projects. Without risk management, software development team has little insight about what could go wrong and how they can control the risk factors. If you never know the potential risks to the project, in turn you will not be able to protect your project from those unknown and unseen risk factors. It is necessary to conceptualize the potential risks related to the initiated projects and understood by the participants.
- Large scale software projects are more complex and often facing more problems as compare to small scale projects. Large scale projects result in budget overruns, schedule overruns or low quality etc. In order to manage large software projects; it is advisable to manage and complete them in small parts/stages. On completion of one stage, it needs to be analyzed in comparison with the completed part of the project and compared with the defined standards. Any deviation from standards should be identified, solved for the completion of current stage. Project should move to the next phase on the completion of one phase successfully.

- Cultural gap between IT staff and users of the application can lead us towards the lack of collaboration and communication. Building the cultural gap between IT users and IT knowledge workers is a major problem for organizations. We need to educate the users of technology about the relationship of IT with their performance and progress. A Hybrid management can bridge good working relations among technical people, users of technology and organizational management. Hybrid manager can help us in decreasing the cultural gap and integrating IT with corporate objectives successfully.

- The benefits of quality are known by the whole world. To maintain certain level of quality in all projects is necessary. It is the degree of attributes; which enables the producer of services and products to meet the standards and needs of its customers. It helps us to define and measure the quality of projects in process. It ultimately helps the development team to produce the desired products and services according to the user needs. We need to apply the quality standards on every phase of software development process e.g. feasibility, analysis, design and implementation etc.

- The software project managers should be aware of the “Escalation Commitment Theory”. It refers to the human tendency of adherence to the course of action; even in case of negative signals are coming regarding the viability of the project in process. It is human tendency to stick to the earlier decisions, even in the availability of latest information proving that the earlier decision was wrong. It is due to the over confidence of the project manager in outcomes or either he is not willing to accept his earlier made bad decision. Other reasons of escalating commitment are the internal political support from the higher management and competitive rivalry with competitors.

- It is common that requirement specifications are changed during the software project development process. If project team is forced for any midway changes in large and complex projects, it demands to revise the whole project plan. In case of any midway changes in projects will not serve with “Linear Thinking”. It means a project should be revised in sense of updating the project deadline, budget, staff and effort required for completing it successfully. There could be some other factors needed to revise depending on the project.

- During forecasting for planning purpose, be careful from “Delusional optimism” or “Planning Fallacy”. On the basis of delusional optimism, projects are started with unrealistic analysis by underestimating losses and overestimating the profits. In results of this; most of the projects are over budgeted and never come on time. Project managers should make their decisions based on true probability of gains and losses.

- Most of the organizations are facing internal and external political pressure or involvement in their projects. These groups and individuals influence the projects for attaining their political objectives. These political groups and individuals try to
manipulate the projects due to their attached interests. Higher management needs to take care of these political factors, which sometime results in project termination.

- Organizational structure can play a very important in the success of a software project. Public organizations are bureaucratic in nature and it makes a project impossible to complete on time with defined budget. Similarly, private organizations can face this kind of difficulties in carrying out their projects on timely basis. Sometimes, organizational structure never adopts quick changes. Certain organizational structures are a hurdle in the way of quick communication and decision making. Before, starting any project, the project team should study and understand the organizational structure.

- Project “goals and objectives” should be realistic and attainable; otherwise it is difficult to achieve them. We should have a clear idea about the status of the project on every stage of its completion. It is only possible by true comparison between achieved and forecasted performance. Realistic goals are easy to achieve and a motivational factor for the project development team.

- Use of “immature technology” can increase the chances of software project failure. Software Projects needs to be built on tested technology and should be highly focused. Companies should avoid using immature and untested technology, especially for large scale projects. It is good to divide the project into small parts for successful completion and further, they need to test the technology before going to apply on large scale projects.

- Software project team should collect both the “qualitative” and “quantitative” information for developing strategy and achieving their objectives. Qualitative elements are skills, tools, methods etc and Quantitative elements are schedule, functionality and effort etc. So, both type of information is very important for project teams to collect for successful software project completion.

- “User’s training” is important for effective and efficient use of IT system in organizations. Without effective training users would not be able to use and maintain the implemented systems in the organization. With the help of user’s training, organizations can easily drive the projected benefits from implemented software projects.

- Every project should have a “contingency plan” prior to its start. It helps us in handling any unforeseen situation, and even we can enhance our knowledge about the projects, and will be helpful in removing any fault.

- Successful completion and implementation of a software project requires the collaboration and communication among different participants like Stakeholders, Management, Users, and Development Team etc. Every “team participant” has to play his role and responsibilities for making a project successful. We need to include all required participants in the process for successful completion of a project.

- We need to define “key factors” which are critical for the success of a project. Organizations need to clearly define their objectives from these projects. On the basis of these objectives, the software project development team needs to point out these critical factors and try to fulfill them for attaining success in the projects.

- As we are saying engineering is a “people” intensive and people are really important in developing, implementing and running the software projects. A common source of aggravation in software projects arises from people, but it is never ever admitted that
it could a big factor to analyze in case of any failure. Personnel changes during the software projects are not desirable.

6.3 Future Study

In this research study I tried to find the main causes of software project failure. For this purpose, I have done a comprehensive literature review. Keeping in mind the same objectives, I had developed a questionnaire survey to find and analyze the failure problem. I have finalized some root causes of software project failure on the basis of literature review and questionnaire survey. I presented a conclusion and recommendations learned through this research study. These presented recommendations covered the major causes of software failure and as well as some other important issues.

There is massive data written on software project failure. According to my analysis and findings, software project failure is an avoidable problem. It is also important that failure give us the opportunity to think, analyze and learn from our mistakes and help us to avoid them in future. The critical analysis and research in this area could save us from the failure.

For future study, I can suggest one thing which I felt during my research study. In case of any future research work on software project failure, first of all we need to define failure, that what is software failure? In what context we are talking about failure? As there could be multiple reasons of software failure like software project is not according to time, requirements, and quality etc. During my literature review and survey analysis, I was not able to find a single proper definition of failure. Everyone is defining a software project failure according to their circumstances. In simple words, everyone has their own definition of software project failure. My survey respondents are also considering a software project failure by giving importance to different factors like (Quality, Time, Requirement Specification, and Budget). So, we need to develop a standard definition of software project failure?
CHAPTER 7: REFERENCES


Barry, B. & Basili, V.R. (2001). Software Defect Reduction top 10 list, software management,


Computer, Weekly, (2003). *Courts Libra system ‘is one of the worst IT projects I have ever seen.’*


Lionel, C. Briand, (2003). Software Documentation: How Much is enough? Software Quality Engineering Laboratory, Carleton University,


McCue, A. (2002). “Treasury warns of over-optimistic IT projects”. Accountancy Age,


APPENDIX I – COVER LETTER

This survey will help me in my Master dissertation on “High IT Failure Rate: A Management Prospect.” The data collected will be strictly used for study purposes and only for my dissertation in the MSc. Business Administration program at Blekinge Institute of Technology, Sweden (www.bth.se). It will hardly take a few minutes, try to complete as much questions as you can. Please feel free to ask any question. I am available at mazhar.hashmi@hotmail.com

I will be extremely thankful to you for your extended cooperation.

Yours faithfully,

Mazhar Tajammal Hashmi

mazhar.hashmi@hotmail.com
School of Management
Blekinge Institute of Technology
Sweden
APPENDIX II - QUESTIONNAIRE

Name__________ Organization____________ Designation/Position____________

Experience: (In Years)_____________ Email Address______________

Q.1: What kind(s) of software development services your organization provides?

A. Software projects with the specified time span (Short term)
B. Software products developed over the years (Long term)
C. Both
D. Please specify if others____________________

Q.2: Have you ever been participated in the development of software project/product development?

A. Yes, I have participated in software development as a Technical person
B. Yes, I have participated in software development as a Non Technical person
C. No, I have not participated in software development but I know most of the procedure through literature survey

Q.3: If you have participated in software development then what was your role in that team?

A. Team Leader
B. Project Manager (Non Technical)
C. Software designer/developer
D. Stake holder (Investor)
E. Please specify if others_____________

Q.4: In how many software projects/products have you been involved?

A. 5 Projects
B. 5 to 10 Projects
C. 10 to 15 Projects
D. 15 to 20 Projects
E. 20 or More Projects
F. Others (0 to 4)Projects

Q.5: How do you define/describe software project/product failure?

A. Software project/product delivered according to specification but was not on time.
B. Software project/product delivered according to specification but lack some quality aspects
C. Software project/product delivered according to specification but was over budgeted.
D. Software project/product was not according to specification but fulfills other requirements like time, quality and budget

Q.6: What was the ratio of successful versus failed software projects/products in which you were involved? (Divide 100% in A & B)

A. Successful_____%
B. Failure______%
Q.7: When you consider a software project/product partial or complete failure? (Please specify all those apply)

A. Project has not been delivered with in the defined time  
B. Project does not meet the user requirement specification  
C. Project does not completed within defined budget parameters  
D. Project does not meet the defined quality standards  
E. Please specify if others____________________

Q.8: What can be the effects of software failure for the organization as a whole? (Please specify all those apply)

A. Financial Loss  
B. Time Loss  
C. Lowering of Performance/Moral  
D. Depletion of assets  
E. Loss of shareholder confidence  
F. Job losses  
G. Bad press/media publicity

Q.9: What are the basic skills and abilities a team member should have for successful project completion? (Please specify all those apply)

A. Every Team member should be well educated/trained for tools/technologies used in project/product  
B. Every team member should have the ability to coordinate with other team members  
C. Every Team member should have a specified role and he must have the ability to fulfill the responsibilities of his/her role  
D. All above

Q.10: What are the main causes of software project failure according to you? (Pleas specify three or all those apply)

A. Lack of project management  
B. Lack of project planning  
C. Communication failure  
D. Technology Illiteracy (User Involvement and Training)  
E. Lack of Resource/Budget Management  
F. Lack of Top Management Involvement & Support  
G. Lack of Stakeholder Involvement  
H. Lack of Quality Management  
I. Lack of Proper Organizational Structure  
J. Please specify if others________________________

Q.11: What kind of people should be involved in software project team? (Please specify all those apply)

A. Team Leader  
B. Software Project Managers  
C. Technical staff/Developers  
D. Stake holders  
E. Sales & Marketing people  
F. Users of application  
G. Please specify if others___________________
Q.12: Do you think proper attention of higher management can play a vital role in Software Project/Product success?

A. Higher management involvement can play a vital role in the success of a software project/product (Agree)
B. Higher management involvement does not have any impact on the success of software (Disagree)

Q.13: Do you consider higher management’s attitude can play any role in achieving your goals assigned by the company?

A. Yes
B. No

Q.14: How much technical knowledge is sufficient for a software project manager?

A. Technical 50% Mgt. 50%
B. Technical 70% Mgt. 30%
C. Technical 30% Mgt. 70%
D. Please specify if others____________