

Study and Analysis of Factors Responsible For Failures of Software Project

Prof. Ashvini Shende¹ Dr. Janardan Pawar²

¹Assistant Professor, Computer Science Dept., Indira College of Commerce and Science, Pune

²Vice-Principal, Computer Science Dept., Indira College of Commerce and Science, Pune

Abstract

Most software projects can be considered at least partial failures because few projects meet all their cost, schedule, quality, or requirements objectives. Failures are rarely caused by mysterious causes, but these causes are usually discovered post-mortem, or only after it's too late to change direction. This research paper is based on survey carried out on different software consultants, developers and practitioners who were asked to provide reasons of failed projects with which they have been acquainted. Many factors have been considered in this research which generally leads to failure of software project and after analysis it has been proven that "Lack of user involvement" in the requirement phase leads more impact on software project failures. Hence, in this research, researcher has recommended that user should get involved completely in requirement gathering, designing phase.

Keywords: Software project, factors responsible for software project failures etc.

1. Introduction

21st century known for computerization of all manual works, that human being was doing so far. Computerization made man life easy and this computerization become possible because of integration of hardware and software. Software plays most important role in the automation of most of electronic appliances. Hence, in current market, demand for all types of software is increasing day by day. This demand leads to development of thousands of software applications in turn increase in software industries.

Every year many software industries are spending billion on IT application development. Statistically, 31% of projects will be cancelled before they ever get completed. 53% of projects will cost twice as of their original estimates, overall, the success rate is less than 30% [1]. Why did the project fail? From symptom to root cause - what are the major factors that cause software projects to fail? What are the key ingredients that can reduce project failure?

Project failure can be defined as one or a combination of cost overruns, late deliveries, poor

quality, and/or developing a product that does not get used. Regardless of their involvement during the planning stages, more often than not, software developers bear the brunt of the responsibility for such situations; after all, they're the ones who built the application. However, closer examinations of the projects do not always show evidence of incompetence [2].

In this paper various points has been considered. Following points are majorly seen impacting factors for software project failure. [3]

1. Lack of user involvement
2. Long or unrealistic time scale
3. Poor or No Requirements
4. Inadequate Documentations
5. Scope Creep
6. No Change Control System
7. Poor testing
8. Lack of foresight in building efficiency markets
9. poor managerial decisions
10. Cost overrun.
11. Lack of an experienced project manager
12. Well-defined Schedules.

2. Scope of the Study

The study is related to the study and analysis of failures in software project. Pune city has been considered for this research work. As this research mainly focused analysis of failures in software project, the scope of this research is decided to have software companies resides in following two areas of Pune City.

1. PMC area
2. PCMC area

Pune is the second largest city in Maharashtra and well known for educational facilities, research institutes and software industry. Due to the good educational facilities, Pune is called as "**The Oxford of the East**" and hence students from all over the world are getting attracted towards pune city. Due to big software industry, pune is transforming into vibrant modern city with bubbling activities in the IT and Hi-Tech sectors. Thousands of software companies can be found in pune city. And as there is software development industries, SDLC process surely gets followed by all software companies.

During the course of the present study the main focus has been given on the study and analysis of failures in software project. In this research, survey has been carried out for analysis of current scenarios which leads to software project failures in many software industries.

The geographical location of Pune city and software companies present in PMC and PCMC are indicated by the map 2.1, 2.2 and 2.3 as follows

Map 2.1 Map of Pune city

Map 2.2 Map of the software companies present in PMC area

Map 2.3 Map of the software companies present in PCMC area



Figure 1.Source:
<http://www.mapsofindia.com/maps/maharashtra/pune.htm> (23/7/2008) [4]



Figure 2.3 :Source : [https://maps.google.co.in/maps?hl=en-IN&gbv=2&ie=UTF-](https://maps.google.co.in/maps?hl=en-IN&gbv=2&ie=UTF-8)

[8&fb=1&gl=in&q=software+companies+in+pune&hq=software+companies&hnear=0x3bc2bf2e67461101:0x828d43bf9d9ee343,Pune,+Maharashtra&ci=av_nVKH6O86luwSvrIL4Bw&ved=0CB4QtQM&output=classic&dg=brw](https://maps.google.co.in/maps?hl=en-IN&gbv=2&ie=UTF-8&fb=1&gl=in&q=software+companies+in+pune&hq=software+companies&hnear=0x3bc2bf2e67461101:0x828d43bf9d9ee343,Pune,+Maharashtra&ci=av_nVKH6O86luwSvrIL4Bw&ved=0CB4QtQM&output=classic&dg=brw)[6]

3. Objectives of the study

The main objective is to study and analysis of failures of software project.

4. Research Methodology

This research study is related to study and analysis of failures of software project. It utilizes both primary and secondary data. The secondary data utilizes already available information both published as well as unpublished. For primary data however such a facility is not available and it has to be collected by using the survey method. The scope of research is limited; the survey is undertaken by obtaining a purposive and quota sample. The description of the research methodology required for the process of obtaining a sample as well as the nature and size of sample should be adequately explained. Purposive, quota and convenience sampling techniques involves the selection of respondents based on the important characteristics under study such as where they work, position in organization, specific knowledge related to the research problem etc.

4.1. Primary data

Primary data are obtained through a survey. Such data is first hand and original in nature.

Several methods are used for collecting primary data like telephone survey/e-mail survey, mail questionnaire, personal observation and interviews. Particularly in survey, the important ones are – observation, interview, questionnaire, schedules, e-mail survey, telephone survey etc. Each method has its advantages and disadvantages. The primary data collected by the researcher is explained in the following manner:-

4.2. Selection of the city

For the present research work Purposive sampling method has used to select the Pune city as Universe of the study. Pune city is also known as “The Oxford of the East” and a center of IT activity. In this research, Pune city is defined as a scope for the study of impact poor requirement gathering on software testing and designing of model to reduce software product failures.

The type of research is **Exploratory Design** in which the **Survey Method** will be used for data collection; the focus will be given to the aspects of tester’s problem in software companies.

4.3. Universe of the Study

For the present study Software companies located at Pune area has been treated as a universe of study by using purposive sampling method.

4.4. Unit of the Study

21 Software Companies considered as a sample for conducting review to understand the basic problems and technical problems while testing the software.

4.5. Sampling procedure

As total respondents are more than 1, 00,000 so according to K. Morgans Law minimum sample size should be 384 so here researcher has considered 400 respondents.

The study units (Hinjawadi , Hadapsar, Kharadi) approximately covers 400 testers.

4.6. Parameters of Development

- **Data collection :**

Primary Data Collection: Primary data for various samples will be collected in the following ways:-

- a. Information will be collected from testers QA Managers and end users through the structured interview schedule.
- b. Data will also be collected through personal field visits to companies and focus group

discussions with end users, who are using software for various uses.

Secondary Data Collection: The secondary data will be collected from books, journal articles and websites, newspapers and conferences souvenir.

- **Data Analysis**

The collected data will be analyzed by quantitative and qualitative ways. The SPSS (Statistical Package for Social Science Research) Version:8 is used for quantitative analysis.

4.8. Secondary Data

The Secondary data is used to study the awareness and usage of SDLC models and impact of software requirement gathering process on software testing with the help of earlier research studies made by others. It is also used to find out the merits and demerits and limitations of different SDLC models and awareness of poor requirement gathering process with the help of available data. It is helpful to study the objectives and hypotheses framed for the present study.

The secondary data is collected from reputed journals and magazines, newspapers, articles, internet websites and archives. For collecting this data the researcher has visited various libraries. A few of these libraries are Jaykar Library (Pune University), Yashada, Tilak Maharashtra Vidyapeeth Library, British Library, Indira College of Science Library and Indsearch Library.

5. Data Presentation, Analysis and Interpretation

Survey based research methodology has been used to carry out this research. This research is related to the study and analysis of failures of Software project with special reference in Software companies of Pune city. The researcher has tested positively the hypotheses of this research study, with the help of primary and secondary data. For

the purpose of the study, samples have covered all software companies present under PMC and PCMC area. Hence, the researcher has selected one sample viz. software companies present under PMC and PCMC area and collected data from the employees working in these software companies.

Strongly Agree (SA)-5, Agree(A)-4, Neutral(N)-3, Disagree(D)-2, Strongly Disagree(DS)-1							
Sr.No	Factors	SA(5)	A(4)	N(3)	D(2)	SD(1)	Avg
1	Lack of user involvement	275	91	34	0	0	7.39
2	Long or unrealistic time scale	201	199	0	0	0	7.23
3	Poor or No Requirements	237	156	7	0	0	7.35
4	Inadequate Documentations	158	198	44	0	0	6.88
5	Scope Creep	181	219	0	0	0	7.15
6	No Change Control System	172	191	37	0	0	6.97
7	Poor testing	231	169	0	0	0	7.35
8	Lack of foresight in building efficiency markets	222	141	37	0	0	7.17
9	poor managerial decisions	206	178	16	0	0	7.19
10	Cost overrun.	184	200	16	0	0	7.1
11	Lack of an experienced project manager:	141	236	15	8	0	6.87
12	Lack of methodology in the process	163	230	7	0	0	7.05
13	Well-defined Schedules	238	155	7	0	0	7.35

Table No. 1: Various factors responsible for failure of software project

It is quite important to understand the factors responsible for software project failure. Table No. 1 shows the various factors, which are responsible for software project failure. To meet the objective of this research a questionnaire has been designed by using various factors which define the various points responsible for failure of any software project. It is observed that for each document the average scale is in between 1 to 5 that is in between strongly disagree to strongly agree. In fact all the values are above 3.5 which mean that with respect to all the parameters much approval is observed. In a 5-point Likert scale, having categories like strongly agree, agree, neutral, disagree and strongly disagree clubbed into three categories. The reason for using Likert scale is that the responses by the respondents should not become monotonous while answering the questions. Hence researcher has also applied 5-point Likert scale and calculates weighted average value. There is very less difference between the comparative value of rank order average value and 5-point Likert scale value. [7]

It is seen that the highest average value is 7.39 for the ‘Lack of user involvement’ followed by ‘Poor or No Requirements’, ‘Poor Testing’ and ‘Well-defined Schedules’ which are 7.35. The average value for factor ‘Long or unrealistic time scale’ is 7.23 followed by ‘poor managerial decisions’ is 7.19. The average value of ‘Lack of foresight in building efficiency markets’ is 7.17, followed by ‘Scope Creep’ is 7.15. The average

value of ‘Cost overrun’ is 7.1 followed by ‘Lack of methodology in the processes is 7.05. The average value of ‘No Change Control System’ is 6.97 and ‘Inadequate Documentations’ is 6.88 followed by ‘Lack of an experienced project manager’ is 6.87.

As per most of respondents, it is clear that ‘Lack of user involvement’ followed by ‘Poor or No Requirements’ is most important factor responsible for the failure of software project. Of course, user means end user involvement is most important as end user only going to tell his demands or request to business analyst and if end user only unavailable in the requirement gathering meeting then there is no point to discuss anything, anymore. Average 7.39 respondents are agreed to have end user in the requirement gathering meetings or sessions.

Even if end user is available in requirement gathering meeting or session but requirements quality remains poor then also it leads to failure of software project. Poor requirements can get collected if business analyst having less domain knowledge. Also if end user does not have understanding what exactly he wants then also quality of requirement becomes poor. Poor or non-qualitative requirement can become base for any software project and it creates failure throughout SDLC process. Hence, in this research 7.39 responds recommended to have good quality of requirements.

If end user is available and quality of requirement is also good but if testing team executes test cases wrongly then also it creates failures in software project. As per 7.39

respondents, testers should always execute test cases based on the business functionality and requirements written in Functional Requirement Document (FRD). Hence, in this research recommendation needs to provide to testing team to follow testing best practices for the test case execution and in turn to reduce software failures.

Project management is the key factor for the success or failure of any software project. For qualitative project management, well-defined schedule is mandatory factor. If project will have well-defined schedule then all the teams like requirement team, development team, designer team, testing team etc. will follow the same time lines to meet the project success. Hence, 7.39 respondents agreed to have well-defined schedule to reduce failure in software project.

As we saw well-defined schedule for software project plays vital role in the success of project, but well-defined schedule mean short and realistic. If project schedule becomes long and unrealistic then it surely leads to failures in software project. About 7.23 respondents agreed to this point and hence researcher strongly recommends that project schedule time scale should be short and realistic.

In software project execution, many situations can come where managerial level people need to take decision and provide answer to client. If project goes in RED situation where customer is not happy and he is demanding software in very short period of time then in that case managerial decision plays very important role to keep customer calm and happy. But if managerial decision becomes poor then customer won't allow us to work and can take break deal with Software Company. Hence, as per average 7.19 people poor managerial decision leads to failures in software project and therefore, there is need to improve managerial decision skills to increase success rate of software project.

If we developed any product then for selling that product marketing plays vital role. But if we don't have foresight about our project efficiency then in the market product won't get sell. The average 7.17 respondents saying 'Lack of foresight in building efficiency markets' is most important factor and software companies need to focus on this point. Also there is need to have future knowledge about the market of developed software project.

Around average 7.15 respondents is giving importance to 'Scope Creep' factor. Scope creep means project scope should not get creep if we are dealing with success of software project. Scope creep generally happens if project schedule is long and unrealistic. Hence, as discussed above to avoid scope creep there is need to have well-defined, short and realistic project schedule.

If there is lost of changes in requirements or development team created faulty software component or taken too much time to developed

software product, also testing team could not finish testing within specified time then project cost can get overrun. The average 7.1 respondents agreed that 'Cost overrun' could lead to failures in software project. Hence, project management or team lead needs to focus on work status of requirement, development and testing team. Need to resolve all the issues coming throughout SDLC phase so that it cannot overrun cost of the software project.

For error free SDLC process there is standard defined by software engineering for each phase. Requirement team should follow the best standard practices for requirement engineering process, development team should follow best development practices for the coding of software components, and testing team should follow the best testing practices. But if there is lack of methodology present in these best practices then it will lead to project cost overrun and in turn lead to failure of software project. Average 7.05 respondents are agreed with 'Lack of methodology in the process' lead to failure of software project and hence this research recommends best practices and methodology should be followed throughout SDLC process.

To record the changes taken place in requirements by the end user or customer, software management should create change control system and update it as and when required. The average 6.97 respondents agreed that 'No Change Control System' always lead to failure of software project. If project management do not use Change Control System then it won't be possible to record changes made in requirements and it will miss out few important functionality in proposed software project. Hence this research recommends that change control system should be mandatory and gets updated as and when required.

Documentation throughout SDLC process plays vital role to transfer knowledge from one team to another. If requirement team does not provides adequate documents to development team then development team cannot come up with component design and specification documents appropriately and if development team does not provides FRD, component design and specification documents to testing team, then testing cannot come up with appropriate test cases. Hence, each team should provide adequate documents to other team. Because Inadequate Documentation can lead to the failure of software project and in survey around average 6.88 respondents agreeing that 'Inadequate Documentations' lead to software failures and there is need to have adequate documentation throughout SDLC process.

Experienced resources always play vital role in the success of software project. If resources are fresher or new joiners then they don't have product/domain knowledge and hence they cannot

understand business functionality easily and quickly. Experienced person can easily communicate with customer on the domain knowledge, business functionality issues etc. Hence, around average 6.87 respondents are agreeing that 'Lack of an experienced project manager' lead to software failure. Therefore, there is need to have experienced resources in the all the teams of software company.

Conclusion

In this research paper, different factors have been considered for the study and analysis of failures of software project. The factors of failures of software project have been documented and analysed. Analysis showed that due to these factors software project leads to failures and based on analysis result, researcher has recommended that user should get involved more in requirement engineering phase for clear understanding of customer requirements.

References

- [1]<http://www.enterprisecioforum.com/en/blogs/pearl/it-project-failure-symptoms-and-reasons>.
- [2]<http://www.infoq.com/articles/software-failure-reasons>
- [3] Jones, Capers, Patterns of Software Systems Failure and Success, International Thompson Computer Press, Boston, Mass., 1996.
- [4]<http://www.mapsofindia.com/maps/maharashtra/pune.htm> (23/7/2008)
- [5]<http://www.mapsofindia.com/pune/software-company-pune.html>
- [6] https://maps.google.co.in/maps?hl=en-IN&gbv=2&ie=UTF-8&fb=1&gl=in&q=software+companies+in+pune&hq=software+companies&hnear=0x3bc2bf2e67461101:0x828d43bf9d9ee343,Pune,+Maharashtra&ei=av_nVKH6O86IuwSvrIL4Bw&ved=0CB4QtQM&output=classic&dg=brw
- [7] Vidya Gaveakr, (2013) "A Study of Geographic Information System based computerized framework to enhance the water supply system in Pune City" Thesis of Computer Management Dept., Tilak Maharashtra University.