Accuracy of Software Reliability Prediction from Different Approaches

R. Sachin vasudev
Ashok reddy vanga

Department of
Computer Science
School of Engineering
Blekinge Institute of Technology
Box 520
SE – 372 25 Ronneby
Sweden
This thesis is submitted to the Department of Computer Science, School of Engineering at Blekinge Institute of Technology in partial fulfillment of the requirements for the degree of Master of Science in Computer Science. The thesis is equivalent to XXX weeks of full time studies.

Contact Information:
Authors:
Sachin Vasudev. R
E-mail: svra06@student.bth.com
Asoka Reddy Vanga
E-mail: asrv06@student.bth.com

University advisor:
Dr. Mia Persson
mia.persson@bth.se

Department of Interaction and System Design
Blekinge Institute of Technology
Box 520
SE – 372 25 Ronneby
Sweden

Internet : www.bth.se/tek
Phone : +46 457 38 50 00
Fax : + 46 457 102 45
PREFACE

We would like to thank the following individuals for their contribution in one way or another to our thesis.

Mia Persson

Vickey Wadhwani
ABSTRACT

Many models have been proposed for software reliability prediction, but none of these models could capture a necessary amount of software characteristic. We have proposed a mixed approach using both analytical and data driven models for finding the accuracy in reliability prediction involving case study. This report includes qualitative research strategy. Data is collected from the case study conducted on three different companies. Based on the case study an analysis will be made on the approaches used by the companies and also by using some other data related to the organizations Software Quality Assurance (SQA) team. Out of the three organizations, the first two organizations used for the case study are working on reliability prediction and the third company is a growing company developing a product with less focus on quality. Data collection was by the means of interviewing an employee of the organization who leads a team and is in the managing position for at least last 2 years.

Keywords: black box testing, white box testing, analytical model, data-driven model
## CONTENTS

**Abstract** ........................................................................................................................................... 1

**Introduction** ................................................................................................................................. 5

**Chapter 1: Background** ............................................................................................................. 7
  - Related Work............................................................................................................................. 7
  - Purpose of Study....................................................................................................................... 8

**Chapter 2: Methodology** ............................................................................................................. 9
  - Research Method...................................................................................................................... 9

**Chapter 3: Research Approach** ................................................................................................ 10
  - Data Collection....................................................................................................................... 10
  - Data Analysis Procedures...................................................................................................... 11
  - Validity of the Qualitative study............................................................................................ 11
  - Expected Outcomes............................................................................................................... 12

**Chapter 4: Case Study** ............................................................................................................... 13
  - Case Study 1........................................................................................................................... 14
  - Case Study 2........................................................................................................................... 20
  - Case Study 3........................................................................................................................... 23

**Chapter 5: Case study analysis** ................................................................................................ 25

**Conclusion/ Future work** ......................................................................................................... 29

**References** ................................................................................................................................ 30
Introduction

The use of software is increasing everyday. Software products are very complex in nature therefore; they need good processes to produce quality software products [15]. To be a lucrative and qualitative in software industry, an organization must be able to predict the future of their software product. Reliability prediction helps management of an organization to know about the performance of their system in advance. Reliability prediction is also an important activity to measure the quality of the software. Reliability prediction gives a picture of maintenance, cost and error in advance stages. This kind of information is useful for the organizational management to decide whether to release the software at a particular time or they require more time to release it. What is the right time to release software in terms of cost and quality?

This problem is addressed by many researchers and many modules have been proposed to solve this problem [9]. All prediction models can be classified into two approaches namely analytical and data driven approaches. In an industry practice, generally software quality threshold is set and predicted value from reliability is compared with threshold [9]. Good reliability prediction helps an organization to save cost, time and resources [8]. But this kind of reliability approach may lead to a badly allocated time, cost and resources [8]. Therefore, good prediction is very important to achieve the quality of software product and save organization resources.

Many models have been proposed for software reliability prediction [6, 10, 13] and research is matured in this field. There are many existing models, but none of these models could capture a necessary amount of software characteristic [10]. And all these software reliability models have made some assumptions [10]. The other common approach is the data driven which uses past failure data for the calculation of software reliability prediction [6]. Xie [14] proposed a model for the data driven approaches based on NHPP (Non-Homogeneous Poisson Process). That paper explains the usage of failure data from similar previous projects for the calculation of reliability prediction in NHPP [14]. Factors like environment, human developers are changing and without using these parameters we cannot make a good prediction. The analytical model like architecture based software performance and reliability prediction uses the architecture of software for the reliability prediction [12]. Another model, stochastic model of fault introduction and removal during software development process incorporates human errors for predicting the software reliability [6]. These both models do not focus on the past failure data. The human and software architecture depends on many things and they are highly variable. So these variables also vary the software reliability prediction in different environments. The main drawback of applying this model is that it depends on uncertainties. And the calculation of these uncertainties is subjective, which may lead to different results.

CMM is an abbreviation used for Capability Maturity Model and it is probably one of the most well-known model-based SPI standards [19]. This model is basically highlighted by standardizing the contents of processes according to a predefined number of practices. CMM level ranges from level 1 to level 5 with level 1 being the initial disciplined process and the level 5 is for the optimization (continuously improving process) [19]. “CMMI is an integration of several CMM versions and it was developed as the one model to replace use of multiple versions” [19].

Chapter 1 explains the background, related work and purpose of the study. Research method and research questions with respect to research method are explained in chapter 2. Chapter 3 details the qualitative research strategy along with data collection, data analysis, validity of the study and expected outcomes. Then the chapter 4 explains about the case study which is
being conducted by us on three different companies. Out of these three organizations two are CMM level organizations and the third is a growing organization. In chapter 5, we have given our analysis on the case study and have also included 5 new factors which will have an influence in developing a new approach for achieving good reliability prediction. Finally the chapter 6 provides conclusions.
Chapter 1: Background

Reliability can be defined as “the probability that an item can perform its intended function for a specified interval under stated conditions” [2, 3] and reliability engineering is “the technical discipline of estimating, controlling and managing the probability of failure in devices, equipment and systems” [2, 3]. Reliability prediction is a useful tool for forecasting the relationship between cost and quality of software. It helps the management in making correct decisions. Early reliability prediction provides an idea to the management about software quality in advance. Managers can take a decision on software release date based on this data. Researchers have proposed different reliability prediction models in literature and all of these models either follow an analytical or data driven approach [6]. Analytical models like NHPP (Non-Homogeneous Poisson Process) focus on software failure process (effect of software development life cycle on reliability prediction) [6]. Data driven models use failure data of previous projects for the reliability predication [6]. An observation from the literature review shows that both the existing approaches have shortcomings and need modifications for better software reliability prediction. Human role is considered as an important factor in any analytical model [5, 6]. Human developer is an important actor behind the software failure process [5, 6]. This research on reliability prediction focuses on the human uncertainties in software failure process.

1.1 Related Work

“Analytical software reliability growth models (SRGMs) are stochastic models to describe the software failure process with essential assumptions to provide mathematical tractability” [6]. And as per the above mentioned article the most practical and popularly used analytical models are Non-homogeneous Poisson process (NHPP), SRGMs.

“Data-driven models are developed from historical software failure data, following the approach of regression or time series analysis” [6]. Lack of sufficient data for learning or parameter estimation has made the traditional models non-useful for accurate prediction in the early phase of testing [6]. This is related to both the models analytical and data driven models used for reliability prediction

In analytical models, human developers play an important role. In data driven models, there are many other factors such as failure data from previous projects that are considered apart from the faults/failures caused by human developers [13]. There are few mixed approaches proposed to improve reliability prediction. Xie [6] has described a model for early software reliability prediction with artificial neural network and data-driven approach. Software failure depends on many factors like human developer, software development process [11, 16]. Early software reliability prediction is calculated in early phase of testing and testing is used to find defects from the software artifacts as it takes time for collecting defect data of the current product from the testing phase. Therefore, if we do not use the defect data collected from the similar previous projects then there will be some difficulty in reliability prediction. Software reliability prediction can not calculate a good value without incorporating human uncertainties, software development process and previous data. To increase the accuracy of reliability prediction, we have proposed the mixed approach of data-driven and analytical models. This proposed approach uses parameters from both the approaches (analytical and data driven) and explains how these parameters from both the approaches help in knowing the impacts on reliability prediction.

1.2 Purpose of the Study
The purpose of this study is to analyze the impact of parameters from analytical and data driven approaches on software reliability prediction. This study proposes a mixed approach using parameters form analytical and data driven approaches. This study also compares the analytical and data driven approaches with mixed approach and finds the impact of all these three different approaches on reliability prediction.
Chapter 2: Methodology

The research questions are divided into two categories based on the methodology that answers the research questions. The research questions that can be answered by qualitative research methodology are as follows:

1. What are the vulnerabilities in the existing software reliability prediction frameworks?
2. What are the factors that effect software reliability prediction?
3. How can a model be designed (with the different parameters) based on the vulnerabilities in the existing software reliability prediction frameworks and factors that effect software reliability prediction?

The research questions that can be answered by case study are as follows:

1. What is the impact of analytically based approaches on software reliability prediction?
2. What is the impact of data driven approaches on software reliability prediction?
3. What is the impact of the customized mixed approach on software reliability prediction used by different organizations based on the factors selected after analysis of the case study?

2.1 Research Method

We have selected the qualitative and research methodology involving case study [1]. This research proposes a new approach which uses the parameters from existing analytical and data driven approaches along with a set of derived parameters from qualitative analysis for software reliability prediction. The solution for this problem can be obtained by using case study method.

The main purpose of this study is to evaluate the impact of existing reliability prediction approaches and proposed prediction approach. An experiment can test the accuracy of different approaches [4]. This study compares the three different approaches for software reliability prediction namely analytical, data driven and mixed approaches. Every model uses different parameters for software reliability prediction. All the necessary parameters will be extracted from software development process. The following sections explain case study and planning. The experimental part is kept as a future study.
Chapter 3: Research Approach

When we are exploring an activity, case study would be the ideal solution to conduct qualitative research [7]. Vulnerabilities in reliability prediction approaches are found by combining literature study and case study. Different books, papers and journals are studied. Based on this literature survey, flaws in both analytical and data driven approaches are found. In case study, three different companies are selected and their selection criterion is based on the usage of the software reliability prediction approach in SDLC (Software Development Life Cycle). Our qualitative study involves two iterations with two steps in each iteration first step being the formal and the other informal. The two iterations are mentioned as follows:

1. Finding weaknesses or flaws in both the approaches (analytical and data driven models).
2. Factors affecting reliability prediction according to the approaches used by different organizations.

As mentioned earlier, two steps were followed namely formal and informal. In the formal approach, we have framed some set of preplanned questions and present them to the organizations during our case study. Here, we take the necessary solutions from the organizations while interviewing a member of the organizations. This process is followed for all three organizations during the interview process. The second step is an informal way of interviewing the member or developer of the organization. In this case, questions are not pre-planned instead they are framed during the conversation based on the reply given by the member of that organization. The member is also allowed to frame his own questions during the conversation and therefore sharing of information takes place. Organizations have used their own way of maintaining the data, finding the vulnerabilities and factors effecting the reliability prediction. These observations made during the conversation are noted down and maintained for future usage. This information sharing helps in finding weaknesses, flaws, strengths and many other factors effecting the reliability prediction based on the different models being used by different organizations. This process of undergoing two steps in each of these iterations will be in a cyclic manner until the required information is gathered for further analysis.

3.1 Data Collection

The data collection procedure in this case study consists of interviews with the selected three organizations and observations are drawn from the interviews. The interviews are done by the researchers for the purpose of gathering information about the reliability prediction approaches (analytical or data driven) that are currently used by the organizations, vulnerabilities occurring with those approaches and accuracy in their reliability prediction. The observations from the answers given by the organizations include reliability approaches, strengths and weaknesses of different reliability prediction approaches and accuracy of reliability prediction according to the practitioners’ view. While collecting the data from the three organizations, a group of staff with in each organization is selected that includes a software developer, a member of management team, and a software quality analyst. The software developer helps in getting the information about the failure data. Manager and the software quality analyst help in finding the accuracy of the software product as well as the vulnerabilities by using the current reliability prediction model. This type of case study that includes interviews and observations facilitates the researchers to collect the required data easily and assist in forming a mixed approach by justifying the observations. This case study that we conducted will also provide the organizations to improve their reliability prediction
by understanding the observations made by researchers. Hence, we believe that this type of qualitative approach is beneficial to the researchers as well as the participating organizations. The data during the interviews and the observations is voice recorded and documented perfectly. Voice recording the conversation between the researchers and the organizations is done for not missing any of the data revealed by the three organizations. Documentation helps the researcher to further analyze the data and assist in proposing a new approach. However, a high level of interaction is needed between the researchers and the organizations to collect the required data.

3.2 Data Analysis Procedures

The data collected through voice-recording will be copied out into a document in text format. This textual format will facilitate the researchers to analyze the data and find the observations. With this type of documentation the organizations can also be benefited by receiving the feedback and manipulating their reliability prediction strategies. The case study with in each organization will be divided into parts that represent the perspectives of the participants in the organizations (software developer, manager & software quality analyst). The conclusions are drawn by comparing the three case studies in three different organizations. The analysis from the interviews and observations leads to find the missing information and the impact of parameters of the alternative approaches on each other. For example, if an organization uses a reliability prediction model with only analytical approach, they may not use past failure data which is a parameter from data driven approach. The analytical approach parameters such as cause of software failure and human factors (skills, experience) are uncertain and may lead to uneven prediction. In that case, inclusion of parameters from data driven approach seems like a good solution. This type of analysis is used to propose a mixed approach that contains parameters from both the analytical and data driven approaches. If a new set of parameters that are not included in both the existing approaches (analytical and data driven) while conducting interviews and observing the conclusions, then those parameters will also be included in the mixed approach that is to be proposed by the researchers. The data collected by interviews and observations are analyzed by the researchers separately and the final findings confirmed by the group are put together to find the vulnerabilities in the approaches, derive the missing information and prepare a mixed approach. The data analysis procedure is also useful to draw the necessary conclusions with a proper solution for a better reliability prediction.

3.3 Validity of the Qualitative Study

A threat to validity in this study is possible for this type of qualitative approach. This can be possible by contradicting the opinions among the researchers without a proper case study. This problem is solved in this qualitative study by using case study in presence of practitioners in the real world. A case study is conducted in three real time scenarios so that the researchers will not be misguided. The validity of this qualitative study is generalized by combining the practitioners view with researchers view as the researchers’ group is small. As there are at least three practitioners from each organization involving in this research, the threat to validity is reduced by increasing the group size. We have selected three organizations with three different environments like gaming industry, financial organization and server based Software Company. Hence, the results obtained from this case study are validated making it more generic.

3.4 Expected Outcomes

The expected outcome of this study is a greater understanding of vulnerabilities in different software reliability prediction approaches that are currently being used by the organizations. The level of impact using different parameters from the existing approaches (analytical and
data driven) on software reliability prediction is studied with in this research. A set of parameters that are included in a mixed approach will be derived from the case study. The observations and conclusions are derived from case studies. Finally, a mixed approach which includes the parameters from both analytical and data driven approaches will be proposed. Even though case studies may have influence of individual scenarios with in the organizations, the observations and conclusions will be generalized to some extent by involving practitioners from three different organizations and the researchers into the case study.
Chapter 4: Case Study

As we explained in the section 3.1, we have collected data from 3 different organizations. The data collection was through case studies and in this case study we have formulated around 40 to 45 questions. These questions were raised during the interview with one of the employee from the organization. We framed some questions during the conversation and those were used for analyzing the case study and we did not mention those in this chapter. This is because they did not want us to present the raw data in our paper instead they allowed us to have an overall analysis and describe in as a text.

Blackbox testing is basically an external way of testing an object to derive the test cases with theses tests being functional or non-functional, mostly the test cases are functional. “The term black box refers to testing which involves only observation of the output for certain input values; that is, there is no attempt to analyze the code which produces the output” [17].

Figure 1.1 Black Box Testing

“White-box testing uses the internal structures (such as control flow or data flow) of programs. Black-box uses an external interface” [18]. For doing this the developer should have access to the source code as he has to do modifications and analyze the coding practices, data flow, control flow, exception and error handling and, information flow within the system to check if the system is working as intended. It is also suggested that white box testing is performed during the unit testing phase [18].
Case Study-1

In this section we will discuss the data gathered from Case Study-1. The data was gathered as follows.

We posed 42 questions to a company (see Appendix 1 for the questions), which is in the area of online gaming. They are based in Hyderabad, India. We choose to include this company in our case study since we knew that they were a well established and a mature company.

The designation of our contact person at this company under consideration works as an Assistant Manager in the QA team. The responsibility of our contact person at this company is that he primarily works with a team of 20, with one Team Leader and two Group leaders (one for every 10 team members). It is our contact person responsibility to report to the manager and this is done with daily basis. A final, monthly report with all the data collected from the testing phase is to be delivered to the manger. As this company works on online gaming platforms servers are the main resources for their applications.

We further asked about the domain of the company’s applications and got the answer that they work on various domains mainly on gaming applications. Furthermore, this company has to deliver their product on various platforms like mobile, Palmtop, Laptop and Desktops (for regular usage from World Wide Web).

An interesting question for us is whether this company possesses or has any international standards. Here we explicitly asked about which CMM Level and the answer we got from this question was that this company, which is a premier provider of software products and solutions to the global online gaming industry. They also mentioned that the company is a fully-owned subsidiary of a gaming company, and one of the world’s leading online gaming companies which owns and operates a poker company, the world’s largest online Poker business. They further mentioned that they are a CMM Level 4 recognized company.

They also pointed out that they have a remarkable management team with young, enthusiastic and passionate individuals who possess rich global experience and hail from renowned Institutes across the world. They pride themselves in retaining and rewarding the best talent in the industry. Currently, their employee strength is over 1000 people.

This company firmly believes in and strives to live up to the spirit of their principles:

- Get the best
- Provide the best
- Expect the best
- Keep the best

We also posed the question about which methods of development this company is using and we got the answer that this organization uses the iterative model for their products. Iterative development is described as a cyclical software development process developed in response to the weaknesses of the waterfall model. It is an essential part of the Rational Unified Process, the Dynamic Systems Development Method, Extreme Programming and generally the agile software development frameworks.
Another question, which was a follow-up question on the aforementioned answer, was which part of Software Development Life Cycle (SDLC) that consumes most of the resources. Here we were told that their application availability is 24*7 and that testing being a major part for their online products and that they have concentrated more on testing their product 24*7. They also pointed out that their clients play regularly and the servers need a constant check. They are tested using some tools, and even manual testing is done.

We further inquire about the major factor behind success of the organization, from their point of view. They replied that they have good process of development and also sufficient resources for maintaining the quality and reliability of their product. The team constantly tests the product internally (server software) and externally (online constraints and difficulties). When a player is playing online the major occurrence of failure will be due to some of the problems like data overflow, data loss etc. During these situations a backup is vital and restarting the server and running it under normal mode is important. In order to overcome this constant testing and inspection should be conducted on the software. They upgrade their software once in every 15 to 20 days and sometimes they also upgrade their product early if they find any security threat to their players who are playing online as they use their financial information while depositing or withdrawing their funds. And this has been the major success for the product and for the organization.

We also asked whether this company verify and validate their products and got the answers that they normally do it by testing and inspection. Inspection is done on a regular basis as the company does not have a separate team for inspection. Their quality assurance team performs inspection only before the release of the upgraded software to find faults and remove them as soon as they can.

Here we were interested to find out more about their inspection routine. Therefore, we asked them which kind of inspection techniques they are using, i.e., if they use inspection techniques like requirement validation by checklist, and etc, in the Software Development Life Cycle.

They said that as their product is 8 years old they have made a checklist for validating their product before releasing the upgraded software. This is based on the requirement given by the company’s clients after receiving some suggestions from their online players while using the software. This checklist is updated by the manager based on the requirements and the updated modules of the product.

The next question was whether this company has any SQA department in their organization. They told us that they have a software quality assurance team for this company and that he himself was the assistant manager of this team.

We were also interested in how many persons that were currently working as software quality assurance team and further, how much budget the company allocates for the SQA activities. The answers to these two questions were that this team has 22 members, including a manager, an assistant manager and a team leader. Regarding the budget, they told us that this will be decided by the managers, as per the requirement. This is because their product is for online gaming software which is available on the website of the company. As people play online many promotions are introduced and removed for a certain period of time and for this
a constant inspection and testing will be needed thereby it increases the usage of resources from the software quality assurance team. Normally 20% of the total development cost is assigned for SQA.

A follow-up question on the aforementioned question is whether the company has a separate department for testing or if it is a part of SQA department and we got the answer that testing is a part of the software quality team and as they mentioned before, according to the requirement, a special team will be made for testing and inspection for verifying and validating the product.

We also investigated how many resources that are allocated for the software testing and they told us that regularly, 40% of the SQA team works on software testing.

We continue this discussion by asking our contact how many testers that actually are allocated for a particular application His reply on this question was that the SQA team has 22 participants and around 6 to 8 of them are allocated for testing. And in them, 2 to 3 are allocated for every application.

We were further interested in the competence of the testers so we raised the question about the educational qualifications and experience of the testers. The answer we got here was that testers should be able to know about the product and should be experienced in this area, as we have to respond immediately for a failure occurrence in the servers. The testers of this company are educated with a minimum four years graduation in computer science or software engineering stream. They are given six months of training before handling the product. Experience with at least 2 to 3 years in this field is considered as an entry for this position.

A natural question here was whether the organization actually provides any kind of training to their testers and the answer to this question was that the company actually does this. Their testers are given regular 3-6 months training before using the product, as they have to know the usage of the product and possible failure occurrences while the software is used by the players.

On the question about how this company tests their products we were told that they have their customized tools and that they test the product using those tools. Also manual testing is done for the external part of the product.

We also got the information that their product is tested both internally (White Box Testing) and externally (Black Box Testing). Internally testing is done by the use of tools. Modifiability is needed whenever they have any security threat and for this the product needs to be tested manually and therefore they allocate some resources before starting the development process.

We further asked about the average number of faults that they find during testing. They told us that it is proved that faults found during testing phase increases when an upgrade is made to the current software product due to the addition of modules or features to the current modules. Testing time allocated is 45 to 60 days and the number of faults is 10-15 per week (approximately). They try to repair 80% of the faults during this allocated time and the remaining 20% faults are not critical.
A follow-up question on this was if they could tell us about the average number of faults they found after deployment of application, and they replied that this number of faults, found after deployment of the application, is found to be 15% of the total number of faults.

On the question about how much time that is needed to repair faults, they said that they have a standard time to fix or repair the faults and it depends on the level of up-gradation and development of another application. For a new product they allocate 50% of the resources from the SQA and 15% to 20% of the total development cost on repairing the faults.

We were further interested in the major reason behind the faults. The answer we got here was that as this product is based on the online customer, it is mainly due to server crashes. It might be due to data overflow or security issues. This is mainly due to lack in requirements gathering. And this is due to the lack in human experience and skills in domain knowledge. Our product being a generic one, we have difficult to satisfy the requirements of each and every customer who is using the application.

We also discussed whether an application created by an experienced developer contains the same number of faults as by an inexperienced developer and very naturally, their experience from this was that an experienced developer has mostly developed a less faulty product in comparison to the inexperienced developer.

The following two questions are important for our research. First, we tried to find out whether this company is using data from past application in the new application. They actually do, data from the previous application is used by the company in developing a new application or product as it helps the developers in overcoming the difficulties in developing software. By using the data from the past application they can build a product with some the factors like good cohesion or coupling which helps the future developers in upgrading or modifying the product.

Second, we investigate if this company is using past data of testing as well and they actually do so, since they are using the past data of testing for their current testing phase. We explicitly said that they are using the data-driven approach for developing a reliable product.

It is important for us to follow up this discussion, in order to perhaps extract more important information for our research, and therefore we explicitly asked the company which different approaches (i.e., analytic, data driven, or mixed) they actually apply. Here they told us that they deploy the data-driven approach, by the failure data of previous projects.

We were further interested in how they test the reliability of their products and their answer here was that, as they mentioned, the company is using the data-driven approach for developing a reliable product. This further helps in testing the reliability of the product.

We also raised the question about the educational qualification and experience of the reliability engineers of the company. Here we found out that the case was that the reliability engineers are the same engineers who are involved in testing mainly who deal with verification and validation. They are educated with a minimum four years graduation in computer science or software engineering stream. They are given six months of training before handling the product. Experience with at least 2 to 3 years in this field is considered as an entry for this position.
During our discussion with our contact, we found out that this company actually is following a specific technique for the measurement of reliability and this is the Data-Driven technique. On the question if there any specific reason for the selection of this particular technique our contact replied that since this is an online gaming product past data is necessary in predicting the reliability of the product. And data-driven justifies this approach in making a reliable product by using past data as the nature of products are similar in nature.

In this study, we were also interested in hearing whether this company experienced any problem in applying this technique. Here they told us that data collection or past data collection is a major criteria for this approach and that this company has started maintaining the previous data since few years back. And they are in need of more data, so our contact told us that he got a feeling that in the future, they will be able to increase or improvise in reliability prediction. On the question whether the company is satisfied with the results that they are getting from this technique, they said they are since they have used the data-driven technique, which used past data. By using the past data they were successful in making reliable software with almost 75% reliability.

We also inquire what constraints they were facing during SDLC. Their experience here was that the main constrain is with the system analysis and design phase as it happened before. Approximately 10 to 12 months back they had developed a software product for a game named backgammon, which had a fault in the design phase. They had to spend more than the expected percentage of funds, which was allocated to the testing team on verification and validation.

We further asked whether the aforementioned constraints do have any impact on the process of reliability and from the company’s perspective they consider that reliability is affected due to these constraints, as it directly affects the quality of the product.

On the question about what role they thought the company’s manpower had in the product’s quality, we got the answer that since they were always looking for experienced developers, and if they are not experienced, proper training is given to them and this affects the product’s quality. Thus, after proper training the employees are given the responsibility for developing the product. As they see it, the company does not depend more on human as they have good processes.

We also discussed whether the company was asked to achieve some level of quality and reliability before delivering their product and their answer was yes, in fact, they are asked to develop a product with good quality.

We were also interested in their opinion about which critical factors there are for failure of products. Their experience here was that when more number of player login to the software and play on multiple tables, their tactic was to include more number of servers. And when have less number of players, they then have to remove them, as the client cannot use the extra resources when not needed.

We also inquire how the company overcomes with these problems and they told us that in order to overcome these problems, a testing team will be constantly monitoring the traffic and they will be testing. When this testing team finds that the number of players is exceeding the limit for a particular server to withstand, they introduce another one in hand to avoid this.
During the discussion with our contact, we also raised the question whether their clients face any severe problems because of application failure. Their experience here was that as our client’s end customer are using their funds online, any application failure will create a severe problem to the customer. But an instant recovery will make them do their work without any difficulty. This is because of the requirements engineering phase, which was not analyzed correctly. Therefore, requirements are tested to overcome any severe problems.

Before we ended up our discussion with this company, we posed the question whether the company actually uses reliability prediction for software release and if so, they consider this as helpful information for software release. Our contact replied here that as he explained earlier, reliability prediction is used before software release. They explained that here they use the help of past data and try to predict the reliability of the product and thereby release the software. By following reliability prediction, the company is able to predict the possible failures in the software product and the possibilities for those failures. This in total helps in attaining a threshold for quality and thereby the manager can take a decision on software release.
Case study-2

We now present the 41 questions (see Appendix 2 for the questions) posed to the company which has its market in EUROPE, United States of America, Asia (financial organization). For each question answer follows directly after the posed question.

We interviewed the Team Lead in Software Quality Assurance Team and his responsibility is to look after a team of 43 including 4 group leaders. His main responsibility was to deliver a product with 93% quality. And the domain of their applications is financial enterprise systems.

This company is a CMM level 5 company and they follow Rapid Application Development (RAD) Model for their development method. They have mentioned that business modeling consumes most of the resources mainly time and manpower. Normally RAD model works on possibility of reusing the existing program components or create reusable components whenever it is necessary. RAD model also assumes that the use of the RAD tools like VB, VC++ rather than creating software using conventional third generation programming languages. And in most of the cases, they have used automated tools to facilitate construction of the software. Since the RAD process lays emphasis on reuse of the existing program, many of the program components have already been tested. This minimizes the testing and development time and it forms the major factor in success of the organization and they verification and validation in our company is done by testing the product. They do not use and kind of inspection technique but maintain a checklist which they normally use for every application as a primary phase after deployment of the product.

They have the SQA team specially trained with a team of 43. As per my knowledge, they also mentioned how the team is split with 4 group leaders. The 43 team member team includes 4 Group Leaders and excluding 1 Team Leader, 1 Assistant Manager, 1 Manager. The Manager has allocated 25% of the budget to the SQA department. And there is no separate department for testing. Since the RAD process lays emphasis on reuse of the existing program, many of the program components have already been tested. This minimizes the testing and development time and it forms the major factor in success of the organization. Because of this, members of SQA work on testing phase. They also mentioned that normally 60% of the SQA team is allocated for Software Testing but whenever there is a need they allocate more than 70% for Software Testing. Number of are allocated for a particular application depends on number of applications available as they divide the total strength out of the 60% which they allocate for the Software Testing Phase. On a regular basis, they assign 3 to 4 for a particular application for a period on 3 to 4 weeks depending on the time availability before software release.

They consider both fresh and experienced graduates for their projects. If they have experienced graduates they train them for a period of 3 to 4 weeks and put them on the project. For inexperienced candidates they have a training session of 3 to 6 months based on the platform. And for the education qualifications they have made mandatory for every candidate to have first grade with 4 years of minimum study in computer science or software engineering. Testers are assumed to be well versed in the field of software development. They scrutinize candidates before taking them into our organization. However, they provide training whenever they have a new application being introduced in the list of company’s products. They use their own customized tools and most of the time manual testing is done on their applications. Internal testing is done on their application as they have to modify and update the application regularly.

During testing they find the number of faults to be less in comparison to the previous year as they have changed the model to RAD. Testing time allocated is 5 to 7 weeks with 8 to 12
faults found weekly and they repair 70% to 75% of faults and the remaining are not the critical ones.

Average number of faults after deployment is less as they are removed in the testing phase. On an average they have found 15% after deployment of the application. They have a standard time set for repairing of the faults as 5 to 7 weeks. It might vary based on the number of faults found before deployment and after deployment of the application. They also mentioned that Time frame for repairing the faults after deploying the product reduces as they have less number of faults.

If they work on the same platform which they used for their previous application, there was no difference found between both experienced and inexperienced developers in creating faults. But when a new application is developed then experienced developer was found to develop the module with less number of faults in comparison to the inexperienced developer. They use data from the past application and RAD itself is used for reuse purpose. Here they reuse data from the existing or previous applications. They use past data of testing as this helped them in maintaining a checklist for undergoing the testing phase for their current application.

Out of the three types of approaches (i.e., analytic, data driven, or mixed) they have used the mixed approach with major emphasis on the past data. They follow the usage of past data of the previous applications which is a data-driven approach. They do not have any special team for reliability engineers. But it is the same for them as they consider for our developers. They consider both fresh and experienced graduates. If they had experienced graduates they train them for a period of 3 to 4 weeks and put them on the project. For inexperienced candidates they have a training session of 3 to 6 months based on the platform. And for the education qualifications they have made mandatory for every candidate to have first grade with 4 years of minimum study in computer science or software engineering.

They have taken the factors from both the models data-driven and analytical driven approach and formulated their own model or approach for predicting the reliability of our application. Sometimes they felt difficult to judge if they have less data collected from the previous application as it becomes difficult to analyze the result. They are satisfied with the result as they are able to deliver our products at the right percentage as per the requirement given by our clients in terms of quality. Constraints faced by them during usage of RAD are like some of the decisions made by the management based on the data available from the previous application may not be comfortable for the development team in developing the application. It might take more time by the developers to work on a particular module and they cannot have a flexible approach. And these constraints have any impact on the process of reliability, as it will be delivered with more time frame and the budget increases as the time frame increases as the manpower increases.

they told us that data is a major part in the development cycle and if a particular data lacks information for the current application later developers or testers may face serious problems. They are asked to develop the product with 93% quality. And the end client has not given a particular mark of percentage for reliability but has asked for a good reliable product. A critical factor behind the failure of product was design layout as it is an interactive application and failure in interaction with the customer might cause block of account and might lead to an extent of loosing the end customer of our client. This is all dependent on the developer who designs the application. To overcome with these problems, a good testing team is assigned the responsibility to check the design layout of the application and more budget is being allocated before starting the application. Their client did not face any severe problems because of application failure as they try to recover the failure in time and make the end customer feel comfortable and make him work normally. They use reliability prediction for software release as it helps in delivering an application or a product on time
and with good quality standards. They mentioned that it provides helpful information for software release and makes the higher management take a decision on software release.
Case study-3

We now present the answers as a case study for the 41 questions (see Appendix 3 for the questions) posed to the company which has its market in Asia (Server Based software).

Our interview for the third company was with a Quality Analyst in Lead position. He was involved in handling a project with a team of 17 and our team works on testing and inspection. Their domain of applications is Server software. The company does not possess any CMMi level and they follow the traditional System Development Life Cycle (SDLC) Model for their product development. The part of Software Development Life Cycle (SDLC) which consumes most of their resources is software requirement analysis phase. They spend most of their resources on the requirement phase and they do not have to go back to any of the phases for faults and errors and this forms the major factor behind success of the organization. They verify and validate their products by testing.

They don’t use any kind of inspection techniques in SDLC. They do not have any SQA Team at present. They are a growing company and the company is planning to expand and put one SQA Team in 3 months. But at present they are having a team of 25 working on an application. After the product is developed they assign 30% of the team for improving the quality and make the product reliable. No special SQA team is set in the company but 7 developers are working on testing and mostly on quality maintenance. Budget allocated for these activities is 30% of the total. They do not have a separate team or department for testing. They have allocated 25 to 30% of their resources based on the requirement for testing. This will be around 6 to 7 of the total team members. Number of testers allocated for a particular application is 2 and in need they assign another for testing.

Graduation in computers science stream with at least 3 years of specialization in software engineering or information technology field is considered as an entry to their organization. As it is a growing company they cannot hire new graduates and they have a deadline and they cannot train them. They hire experience candidates with minimum of 2 or 3 years of experience on the platform which they are currently working. They are currently not providing any training to the testers. They verify and then validate their products. They conduct only internal testing as per the client requirement. In the development cycle, average number of faults found during testing is 17 to 20 per week. And after deployment they found 8 to 10 faults in the application. For this, they do not have a fixed time based on number of faults they start working on it and it exceeds more time they try to optimize their work.

Major reason behind the faults is mainly due to the lack of data from the previous applications. As they are a growing company and their applications are the ones which needs to be started from the beginning they tend to make some mistakes or errors while developing the software. When we asked them about the application created by experienced developer contain the same number of faults as by the inexperienced developer they replied saying that it does matter when an experienced developer handles the project he tends to produce less number of faults when compared to an inexperienced developer. They normally hire experienced candidates in their company and they have found that a developer with more experience performs well.

They currently do not have past data and therefore they analyze and make a reliable product. They do not use past data for their current applications. They do not use any specific approach for reliability prediction and as they do not have any previous applications from where they can use the data. They use human experience and analyze the product for getting a reliable product. They do not have any special method for testing the reliability of our products. But they almost follow the analytical approach for their applications due to the lack of data from the previous applications.
Graduation in computers science stream with at least 3 years of specialization in software engineering or information technology field is considered for a tester. As they are a growing company they cannot hire new graduates and they have a deadline and they cannot train them. They hire experience candidates with minimum of 2 or 3 years of experience on the platform which they are currently working on. They do not have data available from the previous applications they prefer to have a model with which they can analyze the reliability of the product. It is difficult to acquire an accurate reliability prediction as they lack data of previous applications and this forms the major constraint. They are not totally satisfied with the results that they are getting from this technique as they are unable to get an accurate reliability prediction. The constraint that they are facing during SDLC is lack of data from previous applications and they affect the process of reliability as it becomes very difficult to predict software release date.

Manpower is very much needed in a software organization as they are the most vital resources for the success of an organization. As it is a low budget based applications they are asked to develop the product in fewer time frames and a minimum quality of 85% is required by the client. Lack of resources is one of the critical factors behind the failure of the product. Scheduling and timely interaction among the team members for the improvement in the modules of the application helps in avoiding failure in the product. Their client has not faced any severe problems because of the application failure. They use reliability prediction for software release. Based on this reliability prediction and quality analysis the manager takes his decision on software release.
Chapter 5: CASE STUDY ANALYSIS

We have conducted our case study on three organizations, out of them two are CMM level companies and the other one is a growing company. The first two organizations on which we have conducted case study are working on reliability prediction and the third company is a growing company developing a product with less focus on quality and reliability. In our case study we have laid more than 40 questions to the employee of the organizations so that we can produce an answer to our research questions. We have interviewed managers and Team leads of the organizations or companies who manage a group of team members. All these are from either SQA or Testing department. We have investigated in all the 3 companies to see if they are using the analytical, data-driven or the mixed approach for reliability prediction. In our analysis we have tried to see if the companies are facing any difficulty in order to maintain a good reliability prediction and introduce new factors which could improve their prediction.

In our investigation we have seen that the first two companies have used the data-driven approach for reliability prediction. From our case study, we have seen that these two companies are having their reliability prediction around 65% to 75%. Answer from the below questions of the case study illustrates this clearly.

From: Case study 1 and 2 Question 20
Case 1

“Q1: Can you tell us the average number of faults that you find during testing?

A1: Faults found during testing phase increases when an upgrade is made to the current software product due to the addition of modules or features to the current modules. Testing time allocated is 45 to 60 days the number of faults are 10-15 per week (approximately). We try to repair 80% of the faults during this allocated time and the remaining 20% faults are not critical”.

Case 2

“Q: Can you tell us the average number of faults that you find during testing?

Answer: During testing we find the number of faults to be less in comparison to the previous year as we have changed our model to RAD. Testing time allocated is 5 to 7 weeks with 8 to 12 faults found weekly and we repair 70% to 75% of faults and the remaining are not the critical ones”.

Therefore, there is a scope for improving the accuracy in reliability prediction. The third company which does not have any CMM level is using analytical approach for reliability prediction. And this organization is having a low reliability prediction below 60% which is less in comparison to the first two companies.

After analyzing the set of questions for the first and second company we have found that the major factor behind the loss of accuracy in reliability prediction is due to lack of experience in the same company, lack of domain knowledge and human skills of interpreting or analyzing different problems. For example, when an experienced employee joins a new company he will face difficulty in the initial stages as he has to have knowledge on the
current running products of the company. On the contrary, an employee who is working in
the same organization for a period of time has more practical knowledge on the product and
can perform better.

From: Case study 1 and 3, Question 23

Case study 1

“Q: What is the major reason behind the faults?

Answer: As this product is based on the online customer it is mainly due to server crashes. It
might be due to data overflow or security issues. This is mainly due to lack in requirements
gathering. And this is due to the lack in human experience and skills in domain knowledge.
Our product being a generic one, we have difficult to satisfy the requirements of each and
every customer who is using the application”.

Case study 3

“Q: What is the major reason behind the faults?

Answer: It is mainly due to the lack of data from the previous applications. As we are a
growing company and our application are the ones which needs to be started from the
beginning we tend to make some mistakes or errors while developing the software”.

From the third case study we have seen that they have followed analytical approach for
reliability prediction. Due to the lack of data from the previous applications they were
mainly dependent on their developers or human experience. As it is a growing company it
was not strong enough in process maintenance. As per the literature, we know that reliability
can only be tested from a large test set therefore previous data is very much necessary for
good reliability prediction. Good documentation is needed for future use when the testers or
developers reuse the application. This organization was not having a standardize
documentation because of which they had to modify the design phase regularly. After
analyzing the case study of the third company we infer that two factors will help in
improving their reliability are standard documentation and easy searchable documents. Ease
in searching document is not that the system should be fast enough in getting the searched
document. It is that when a developer is searching for a particular document he should be
able to find the exact document which he is looking in the organization database.

Factors included in the new approach ANADAT

Factor 1: Experience (within the same company) and Factor 2: Domain knowledge (For the
current application)

From the first case study we have seen that the company with reliability engineers is the
same engineers who are involved in testing mainly who deal with verification and validation.
They are educated with a minimum four years graduation in computer science or software
engineering stream. They are given six months of training before handling the product.
Experience with at least 2 to 3 years in this field is considered as an entry for this position.

The second company says that they take both fresh and experienced graduates for their
projects. If they have experienced graduates they train them for a period of 3 to 4 weeks and
put them on the project. For inexperienced candidates they have a training session of 3 to 6
months based on the platform. And for the education qualifications they have made
mandatory for every candidate to have first grade with 4 years of minimum study in
computer science or software engineering. Testers are assumed to be well versed in the field of software development. They scrutinize candidates before taking them into our organization. However, they provide training whenever they have a new application being introduced in the list of company’s products. They use their own customized tools and most of the time manual testing is done on their applications. Internal testing is done on their application as they have to modify and update the application regularly.

The third company answered saying that they needed graduation in computers science stream with at least 3 years of specialization in software engineering or information technology field is considered as an entry to their organization. As it is a growing company they cannot hire new graduates and they have a deadline and they cannot train them. They hire experience candidates with minimum of 2 or 3 years of experience on the platform which they are currently working. They are currently not providing any training to the testers. They verify and then validate their products. They conduct only internal testing as per the client requirement. In the development cycle, average number of faults found during testing is 17 to 20 per week. And after deployment they found 8 to 10 faults in the application. For this, they do not have a fixed time based on number of faults they start working on it and it exceeds more time they try to optimize their work.

When we asked them about the application created by experienced developer contain the same number of faults as by the inexperienced developer they replied saying that it does matter when an experienced developer handles the project he tends to produce less number of faults when compared to an inexperienced developer. They normally hire experienced candidates in their company and they have found that a developer with more experience performs well.

First two companies have good percentage of reliability prediction and they have experienced developers and they train them according to their company products. From the informal approach we have also seen that domain knowledge is also very important as it forms the basic requirement which helps in meeting deadlines and having left with time for phases like testing. Even if the developer has less knowledge about the current or previous product, he has to be given training on the current and previous products.

Factor 3: Proper Documentation and Factor 4: Ease in Searching

We have laid stress on two attributes cost and time as they are the main reasons for the success of many products developed by companies. When we have conducted the interview with the help of case study there was a formal and informal approach. In informal approach questions were formed dynamically during the case study. Some of the questions were like which part of the phase consumes more amount of resources like cost, time and tools. Here are the answers given by them in the formal approach.

From case study 1, normally 20% of the total development cost is assigned for SQA. We also investigated how many resources that are allocated for the software testing and they told us that regularly, 40% of the SQA team works on software testing. SQA team has 22 participants and around 6 to 8 of them are allocated for testing. And in them, 2 to 3 are allocated for every application. Testing time allocated is 45 to 60 days and the number of faults is 10-15 per week (approximately). They try to repair 80% of the faults during this allocated time and the remaining 20% faults are not critical. Number of faults, found after deployment of the application, is found to be 15% of the total number of faults. They have a standard time to fix or repair the faults and it depends on the level of up-gradation and development of another application. For a new product they allocate 50% of the resources from the SQA and 15% to 20% of the total development cost on repairing the faults.
From case study 2, The Manager has allocated 25% of the budget to the SQA department. And there is no separate department for testing. Since the RAD process lays emphasis on reuse of the existing program, many of the program components have already been tested. This minimizes the testing and development time and it forms the major factor in success of the organization. Because of this, members of SQA work on testing phase. They also mentioned that normally 60% of the SQA team is allocated for Software Testing but whenever there is a need they allocate more than 70% for Software Testing. Number of are allocated for a particular application depends on number of applications available as they divide the total strength out of the 60% which they allocate for the Software Testing Phase. On a regular basis, they assign 3 to 4 for a particular application for a period on 3 to 4 weeks depending on the time availability before software release.

From Case study 3, after the product is developed they assign 30% of the team for improving the quality and make the product reliable. No special SQA team is set in the company but 7 developers are working on testing and mostly on quality maintenance. Budget allocated for these activities is 30% of the total. They do not have a separate team or department for testing. They have allocated 25 to 30% of their resources based on the requirement for testing. This will be around 6 to 7 of the total team members. Number of testers allocated for a particular application is 2 and in need they assign another for testing.

From our informal approach during the questions raised with regards to the success or failure of the organization we have inferred that proper documentation of all the allocated resources and time and cost spent on every resource is important. Developers spend a lot of time in searching the required data from the previous applications or the current application and this in turn affects the cost and time of the total project availability. Hence we infer that third and fourth factors of our proposed approach are Proper Documentation and Ease in searching.

Finally, we would like to propose a mixed approach name ANADAT as the future work and the major factors to be included in the approach are Experience (within the same company), domain knowledge (for the current application), proper documentation and ease in searching.
Conclusion has been drawn from the Case study and the case study is conducted on three different organizations. The first two organizations are CMM level companies with the third one being a growing company. First two companies have used data-driven approach for reliability prediction and the third company has its own approach which is similar to the analytical approach. After analyzing the case study, a better understanding of vulnerabilities in reliability prediction by qualitative analysis have been obtained and based on these vulnerabilities we have proposed a mixed approach which is considered as future work. Qualitative analysis has also helped us to propose a model based on the vulnerabilities found in the existing reliability prediction approaches. In qualitative analysis, existing approaches (analytical and data driven) is compared with the proposed mixed approach named ANADAT. The main factors included in the proposed new model are Experience (within the same company), Domain knowledge (For the current application), Proper Documentation and Ease in Searching.

The case study involves three different organizations with three different environments reducing the external validity threats. Hence, the results can be generalized to industrial need. This model needs more factors in order to be used in an organization and we leave it as the future work. In order to conduct an experiment or use the proposed new mixed model (ANADAT) we have to introduce the factors mentioned in the case study analysis in an organization and study its effects as well as include the factors from the two approaches analytical and data-driven. Therefore, we propose this mixed model (ANADAT) as the future work for Reliability Prediction. Using the ANADAT model a company will be able to use Reliability Prediction for their software products like software release date. As we have conducted our case study on three different companies and all of the companies are in 3 different domains (gaming, healthcare and financial enterprise) makes it more generic. Therefore ANADAT can be used by most of the organizations.
REFERENCES


Appendix 1

Case study – 1

1. What is your designation in the company?
2. What is your responsibility in the company?
3. What is the domain of your applications?
4. What is the domain of your applications?
5. Which methods of development are being used by the organization?
6. Which part of Software Development Life Cycle (SDLC) consumes most of the resources?
7. What is the major factor behind success of the organization?
8. How do you verify and validate your products?
9. Do you use any kind of inspection techniques (like requirement validation by checklist and etc) in SDLC?
10. Do you have Software Quality Assurance department in your organization?
11. How many persons are working as SQA?
12. How much budget is allocated for the SQA activities?
13. Do you have separate department for testing or is it a part of SQA department?
14. How many resources are allocated for the software testing?
15. How many testers are allocated for a particular application?
16. What are the educational qualifications and experience of the testers?
17. Does the organization provide any kind of training to the testers?
18. How do you test your products?
19. Do you test application externally (Black Box Testing) or internally (White Box Testing)?
20. Can you tell us the average number of faults that you find during testing?
21. What is the average number of faults you found after deployment of application?
22. How much time is needed to repair faults?
23. How much time is needed to repair faults?
24. Does application created by experienced developer contain the same number of faults as by the inexperienced developer?

25. Do you use data from past application in the new application?

26. Do you use past data of testing as well?

27. Which of these three types of approaches (i.e., analytic, data driven, or mixed) does your organization applies?

28. How do you test the reliability of your products?

29. What is the educational qualification and experience of your reliability engineers?

30. Do you use any specific techniques for the measurement of reliability?

31. Is there any specific reason for the selection of this particular technique?

32. Do you face any problem in applying this technique?

33. Are you satisfied with the results that you are getting from this technique?

34. What constraints are you facing during SDLC?

35. Do these constraints have any impact on the process of reliability?

36. How do you see the role of your man power in the product’s quality?

37. Are you asked to achieve some level of quality and reliability before delivering your product?

38. What are critical factors behind the failure of product?

39. How do you overcome with these problems?

40. Does your client face any severe problems because of application failure?

41. Do you use reliability prediction for software release?

42. Does it provide any helpful information for software release?
Appendix 2

Case study-2

1. What is your designation in the company?
2. What is your responsibility in the company?
3. What is the domain of your applications?
4. Does the company possess/have any international standards (for e.g. CMM Level 1-5)?
5. Which methods of development are being used by the organization?
6. Which part of Software Development Life Cycle (SDLC) consumes most of the resources?
7. What is the major factor behind success of the organization?
8. How do you verify and validate your products?
9. How do you verify and validate your products?
10. Do you have Software Quality Assurance department in your organization?
11. How many persons are working as SQA?
12. How much budget is allocated for the SQA activities?
13. How much budget is allocated for the SQA activities?
14. How many resources are allocated for the software testing?
15. How many testers are allocated for a particular application?
16. What are the educational qualifications and experience of the testers?
17. Does the organization provide any kind of training to the testers?
18. How do you test your products?
19. How do you test your products?
20. Can you tell us the average number of faults that you find during testing?
21. What is the average number of faults you found after deployment of application?
22. How much time is needed to repair faults?
23. Does application created by experienced developer contain the same number of faults as by the inexperienced developer?
24. Do you use data from past application in the new application?
25. Do you use data from past application in the new application?

26. Which of these three types of approaches (i.e., analytic, data driven, or mixed) does your organization applies?

27. How do you test the reliability of your products?

28. What is the educational qualification and experience of your reliability engineers?

29. Do you use any specific techniques for the measurement of reliability?

30. Is there any specific reason for the selection of this particular technique?

31. Do you face any problem in applying this technique?

32. Do you face any problem in applying this technique?

33. What constraints are you facing during using RAD?

34. Do these constraints have any impact on the process of reliability?

35. How do you see the role of your man power in the product’s quality?

36. Are you asked to achieve some level of quality and reliability before delivering your product?

37. What are critical factors behind the failure of product?

38. How do you overcome with these problems?

39. Does your client face any severe problems because of application failure?

40. Do you use reliability prediction for software release?

41. Does it provide any helpful information for software release?
Appendix 3

Case study – 3

1. What is your designation in the company?

2. What is your responsibility in the company?

3. What is the domain of your applications?

4. Does the company possess/have any international standards (for e.g. CMM Level 1-5)?

5. Which methods of development are being used by the organization?

6. Which part of Software Development Life Cycle (SDLC) consumes most of the resources?

7. What is the major factor behind success of the organization?

8. How do you verify and validate your products?

9. Do you use any kind of inspection techniques (like requirement validation by checklist and etc) in SDLC?

10. Do you have Software Quality Assurance department in your organization?

11. How many persons are working as SQA?

12. How much budget is allocated for these activities?

13. Do you have separate department for testing or is it a part of SQA department?

14. How many resources are allocated for the software testing?

15. How many testers are allocated for a particular application?

16. What are the educational qualifications and experience of the testers?

17. Does the organization provide any kind of training to the testers?

18. How do you test your products?

19. Do you test application externally (Black Box Testing) or internally (White Box Testing)?

20. Can you tell us the average number of faults that you find during testing?

21. What is the average number of faults you found after deployment of application?

22. How much time is needed to repair faults?

23. What is the major reason behind the faults?
24. Does application created by experienced developer contain the same number of faults as by the inexperienced developer?

25. Do you use data from past application in the new application?

26. Do you use past data of testing as well?

27. Which of these three types of approaches (i.e., analytic, data driven, or mixed) does your organization applies?

28. How do you test the reliability of your products?

29. What is the educational qualification and experience of your reliability engineers?

30. Is there any specific reason for the selection of this particular technique?

31. Do you face any problem in applying this technique?

32. Are you satisfied with the results that you are getting from this technique?

33. What constraints are you facing during SDLC?

34. Do these constraints have any impact on the process of reliability?

35. How do you see the role of your man power in the product’s quality?

36. Are you asked to achieve some level of quality and reliability before delivering your product?

37. What are critical factors behind the failure of product?

38. How do you overcome with these problems?

39. Does your client face any severe problems because of application failure?

40. Do you use reliability prediction for software release?

41. Does it provide any helpful information for software release?