Singaporean Journal of Scientific Research

International Journal of Human Computer Interaction (IJHCI)
Singaporean Journal of Scientific Research(SJSR)
Vol.6.No.6 2014 Pp. 320-326
available at:www.iaaet.org/sjsr
Paper Received :02-10-2014
Paper Accepted:19-12-2014
Paper Reviewed by: 1. Dr. P. Surya Prakash 2. Dr. U. Renga Reddy
Editor : Dr. Binod Kumar

SURVEY OF CUSTOMER SATISFACTION IN SOFTWARE DEVELOPMENT

C. SenthilMurugan
Research Scholar
Dept of Computer Science & Application, SCSVMV University, Kanchipuram
Dept of MCA, Thiruvalluvar College of Engineering and Technology, Vandavasi.
dessenthil.c@gmail.com

Dr. S. Prakasam
Head & Associate Professor
Dept of Computer Science & Application, SCSVMV University, Kanchipuram

ABSTRACT

Software development and maintenance is the activity that is used to satisfy the user requirement and make the error-free Software and also concentrates on time-consuming and complex activity. The software quality management is used to evaluate the quality of a software product and to keep its level high is much more difficult than to do them for the other industrial products. The software quality is maintained to make the software as customer satisfied one. The Software quality assurance is planned to make the best quality software that is fully satisfied the requirement of the customer and it may be flexible for the developer to complete its work in time and in budget. This paper analysis the customer satisfaction in software development process.

Key words: Software Quality Assurance Software development, Software quality management, customer satisfaction.

1. INTRODUCTION

The Software has been used for commercial purpose. With every aspect of software development, Software engineers have been tasked to solve the large and complex programs and in a cost effective and efficient manner. Also the development and maintenance of the software product has become an important criterion. The Software quality assurance is implemented in the early stages of software development life phases, this is used to make the error free software and reduce the rework of development.

In the early years, engineers faced many problems, without having a better knowledge in the software fields, such as “Late delivery of software, Development team exceeding the budget, poor quality, user requirement are not completely supported by the software, difficult maintenance and unreliable software and lack of systematic approach, this problem are overcome by the implementation of software quality activities.[3]

To develop a software product, the following criteria has to be satisfied:

- User needs and constraints must be determined and explicitly stated.
- The source code must be tested thoroughly.
- The product must satisfy the user needs
- Supporting documents such as user guide, installation procedures and maintenance documents must be prepared.

1.1 Software Engineering

Software engineering is an emerging discipline that focuses on the creation, development, operation and maintenance of cost effective, reliable, correct and high quality solution to software problems and it is the application of a systematic, disciplined and quantifiable approach With the development, operation, and maintenance of software.

The software engineering is useful

- To acquire skills to develop large programs
- Ability to solve complex programming problems
- Learn the techniques
- To acquire skills to be a better programmer

The primary goal of software engineering is to improve the quality of software products and to increase the productivity and job satisfaction of software engineers.[5]

1.2 Software Quality Assurance (SQA)

A systematic, planned set of actions necessary to provide adequate confidence that the software development process or the maintenance process of a software system product conforms to established functional technical requirements as well as with the managerial requirements of keeping the schedule and operating within the budgetary confines.

The difference between the quality control and quality assurance should be recognized. Quality control activities are done to sort the products that do not qualify for the qualified products to not deliver the customer or to not sell in the market. [1]

A Software process is a set of activities and associated results which produces software products. These activities are mostly carried out by software engineers. There are four fundamental process activities which are common to all software process.

These activities are

1. Software Specification:
The functionality of the software and constraints on its operation must be defined.

2. Software Development:
The software to meet the specification must be produced.

3. Software Validation
The Software must be validated to ensure that it does what the customer wants.

4. Software Evaluation
The software must evolve to meet changing customer needs.[6]

Common Process Framework

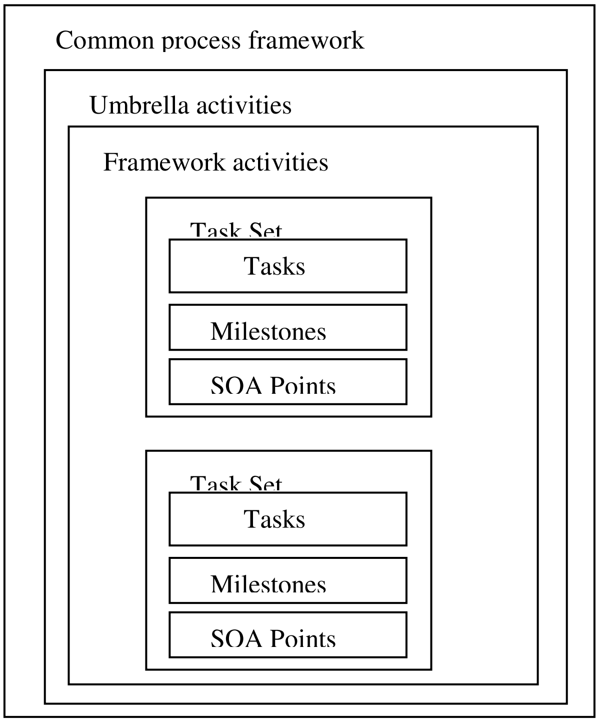

Figure-1: Software Process framework

Task Sets: A number of task sets-each a collection of software engineering work tasks, project milestone, software work products and deliverables and software quality assurance points.

Framework Activities: The task sets enable the framework activities to be adapted to the characteristics of the software project and the requirements of the project team. The generic process framework activities are

1. Communication: The customer requirement information gathering is done by communication.
2. Planning: The planning activity defines the engineering work plan, describes technical risks, list resource requirements, work products produced and defines work schedule.
3. Modeling: The modeling activity defines the requirement analysis and design.
4. Construction: The construction activity implements the corresponding coding and testing.
5. Deployment: The software delivered for customer evaluation and feedback is obtained.[15]

Umbrella Activities

Umbrella activities –such as quality assurance, software configuration management and measurement. Umbrella activities are independent of any one framework activity and occur throughout the process. The umbrella activities are

1. Software project tracking and control: The activity helps to access the software team progress and take corrective action to maintain schedule.
2. Risk management: This activity analysis the risk that may affect the quality.
3. Software quality assurance: This activity maintains the required software quality.
4. Formal technical reviews: This activity helps to analyze the engineering work products to uncover and clear errors before they proceed to the next activity.
5. Software configuration management: This activity manages the configuration process when any change in the software process.
6. Work product preparation and production: This activity defines the model, document forms and lists to be carried out.
7. Reusability management: This activity defines the work product reuse.
8. Measurement: This activity defines the project and product measure to assist the software team in delivering the required software.[6]

2. PROBLEM FORMULATION

Software Errors: The error resulting from bad code in some program involved in producing the erroneous result.
Software faults: Due to the Software errors the Software fault occurs.
Software failures: Due to Bug/defect/fault consequence of a human error.[1]

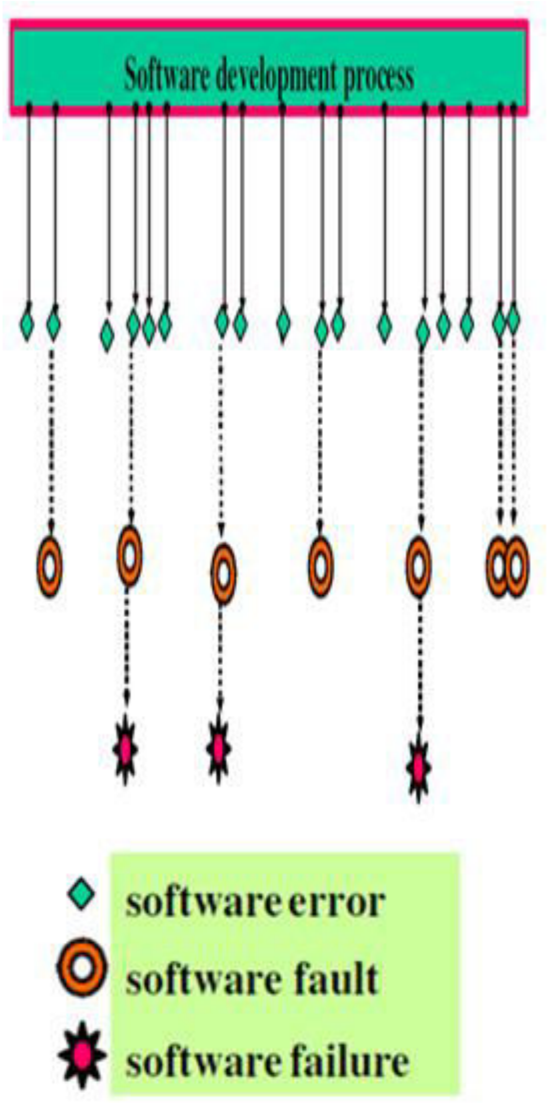

Figure-2: Software Errors, software faults and software failures

Nine Causes of Software Errors

1. Faulty requirements definition
2. Client developer communication failures
3. Deliberate deviations from software requirements
4. Logical design errors
5. Coding errors
6. Noncompliance with documentation and coding instructions
7. Shortcomings of the testing process
8. User interface and procedure errors
9. Documentation errors

Development process relating to defects:

Majority of defects is introduced in earlier phases

Phase	Percentage of defects	Effort to fix defects
Requirements	56	82
Design	27	13
Code	7	1
Others	10	4

Table-1: Majority of defects

Relative cost of fixing defects

Phase in which found	Cost Ratio
Requirements	1
Design	3-6
Coding	10
Unit/Integration testing	15-40
System/Acceptance testing	30-70
Production	40-1000

Table-2: Cost of fixing defects[1]

3. PROCESS OF SQA SYSTEM

The processes are classified into two main groups:

1. Organizational processes, and
2. Departmental/Project processes.

For example,
Galin explains an SQA system as dividing its components five major classes
1). Pre-project quality components,
2). Project life cycle quality components,
3). Infrastructure error preventive and improvement components,
4). Software quality management components,
5). Standardization, certification and SQA assessment components,
6). Organizing for SQA-the human components.[16]

4.REASONS FOR SOFTWARE BUGS

Microsoft Chief Executive, Steve Ballmer said that any code of significant scope and power will have bugs in it. And only 1% of bugs in MS Software is causing half of all reported errors.

Find and fix 1% of your software bugs, and 90% of your system problems go away, say experts.

The term "Software Crisis" is used in the software industry to emphasize the complexity in developing quality software. There are five common problems in the software development process. They are miscommunication, software complexity, programming errors, changing requirements and unrealistic schedule.

- *Miscommunication*: There is widespread miscommunication of information during all the phases of software development, because humans tend to assume and misinterpret a lot of things when communicating.
- *Software Complexity*: Any software, that's developed to serve some useful purpose, is enormously complex and no single person can fully understand it [2].
- *Programming Errors*: Software is created by people, and people are inherently prone to making errors. So, software bugs are also created due to programming errors.
- *Changing requirements*: Software functionality changes, when the requirements change. When we have a system with rapidly changing requirements, additional functionality that's added to the system, can affect the already existing modules in unforeseen ways. High level of interdependencies between the modules, makes the system error prone.
- *Time pressure and deadlines*: The software development industry is highly competitive, and schedule slippages are not acceptable. Some projects have unrealistic schedules, which make the development methodology far from perfect and the developed software lacks quality.

Given these problems, it's apparent that software bugs are very common. One is surely left wondering, "Did anyone do anything to reduce software bugs?" and make software more reliable.

5. FIVE PROBLEMS IN THE SOFTWARE DEVELOPMENT PROCESS

Poor requirements – if requirements are unclear, incomplete, too general, and Not testable, there will be problems.

Unrealistic schedule – if too much work is crammed in too little time, problem Are inevitable.

Inadequate testing – no one will know whether or not the program is any good until the customer complains or system crash.

Featuritis – requests to pile on new features after development is underway; extremely common.

Miscommunication - if developers don't know what's needed
or customers have erroneous expectations, problems can be expected.

6.CUSTOMER SATISFACTION

Customer satisfaction is the ultimate validation of quality. Product quality and customer satisfaction together form the total meaning of quality. The customer satisfaction is essential to the survival of businesses. If the customer is not yet satisfies means the project is not completed and it may fail.

The satisfaction of container is called as the success of the project. If any inconvenience of the customer means, the project may send back to rework, the rework cause lot of difficulties in software development like to exist in budject, time delay etc., leads to software failure.

There are three ways to gather the survey information about the customer satisfaction

- Face-to-Face interviews
- Telephone interviews
- Mailed Questionnares

Face-to-Face interviews:

The personal interview method needs the interview to ask questions based on the pre-constructed questionnaire and record the answers. The primary advantage of this method is that the high degree of validity of the data. The major limitations are the costs and factors concern in the interviewer. Errors in recording the responses could also lead to erroneous results.

Advantages of Face-to-Face interviews

- Ability to build rapport and hold the respondent longer
- Queries can be answered
- Show cards can be used
- Good response to open ended questions
- Can ask respondent to self complete tedious scalar response

Disadvantages of Face-to-Face interviews

- Expensive for a geographically dispersed population
- Takes longer to carry out the fieldwork because of the logistics

Telephone Interviews:

Telephone interviews were once considered an inferior substitute to face to face surveys. The telephone interviews are a widely used method of gathering survey data including the monitoring of customer satisfaction survey levels. This method specialises in low volume, high quality telephone interviews with senior executives and use them effectively for customer satisfaction survey studies.

Advantages of Telephone interviews

- Low cost
- High control of interviewer standards
- High control of sample
- Easy to ask for ratings using simple scales
- Quick turnaround of fieldwork

Disadvantages of Telephone interviews

- Tedious for respondents when there are dozens of attributes to rate
- Some consumers are hard to access by phone
- Cannot show explanatory visuals

Mailed Questionnaires:

Mailed Questionnaires are less expensive than face-to-face interviews. This types must be kept short and impersonal to maintain the interest of the respondent. The limitations of this method are the lack of directions observations, the lack of exhibits of explanation, and the limited group of potential respondents those who can be reached by telephone.

Advantages of Mailed Questionnaires

- Easy for a DIY researcher to administer
- Low cost
- Respondents can complete in a time to suit them
- Easy to complete scalar questions
- Visual explanations can be provided

Disadvantages of Mailed Questionnaires

- Low response rates
- Poor response to open ended questions
- Misunderstanding of questions can not be rectified by an interviewer
- Attracts response from complainers or the very satisfied

REFERENCES

[1] Galin, Daniel (2004), Software Quality Assurance – From theory to implementation, Pearson –Addison Wesley, England.

[2] Abdel-Hamid, T.; Madnick, S. E. (1991), Software Project dynamics: an Integrated

Approach, Prentice-Hall Software Series, Englewood Cliffs, New Jersey, USA.

[3] Ambler, Scott W. (2002), Examining the Agile Cost of Change Curve, Ambysoft Inc.

[4] ARENA Rockwell Automation (2007), Forward Visibility For Your Business,

[5] Balan, S. (2003), A Composite Model for Software Quality Assurance

[6] Charulaths, Software Enineering and Quality Assurance

[7] Bertsekas, Tsitsiklis (1996), Neuro-Dynamic Programming, Athena Scientific.

[8] Boehm, B. W. (1981), Software Engineering Economics, Prentice-Hall, NJ, USA.

[9] Brennecke, Andreas; Keil-Slawik, Reinhard (1996), Position Papers from Dagstuhl seminar 9635 on History of Software Engineering August 26-30, 1996, Germany.

[10] Chikofsky, E.J.; J.H. Cross II (January 1990), Reverse Engineering and Design Recovery: Taxonomy in IEEE Software, IEEE Computer Society:13–17.

[11] Pressman, Roger S. (2000), Software Engineering - A Practitioner's Approach, European Adaptation by D. Ince, 5th Edition, McGraw Hill International, London

[12] The Linux Information Project (2004 – 2007), Bellevue Linux, Bellevue, WA, USA.

[13] Warden, R. (1992), Software Reuse and Reverse Engineering in Practice. London, England: Chapman & Hall, 283–305.

[14] Zave, P. (1997), Classification of Research Efforts in Requirements Engineering, ACM Computing Surveys, 29 (4): 315-321

[15] Dr. B.Chandramouli, K.P.Pradiba Software quality management

[16] Omer Korkmaz - System Simulation For Software Quality Assurance

[17] N. Gross; M. Stepanek; O. Port; J. Carey (December, 1999), Software Hell: Glitches cost billions of dollars and jeopardize human lives. How can we kill the bugs? Business Week Online –International,

[18] NITS (2006), NIST/SEMATECH e-Handbook of Statistical Methods,

[19] Ntourntoufis, Panos (2002), From Software Quality Control to Quality Assurance, UPSPRING Software, UK.

[20] Nuseibeh, Bashar; Easterbrook, Steve (2000), Requirements Engineering: A Roadmap, Future of Software Engineering, Limerick Ireland ACM 2000 1-58113-253-0/00/6

[21]Any software code will have bugs- Microsoft (Source:http://www.ciol.com/content/ news/repts/102100302.asp)

[22]Software Quality Assurance-Principles and practice-Nina S Godbole [22]The Software Crisis (Source: http://www.unt.edu/benchmarks/archives/1999/july99 /crisis.htm)