

## International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology



ISSN 2320-088X

IMPACT FACTOR: 5.258

*IJCSMC, Vol. 5, Issue. 5, May 2016, pg.703 – 707*

# Comprehensive Study on Aspects of Software Testing for Quality Assurance

**Neha Bhateja<sup>1</sup>, Nishu Sethi<sup>2</sup>**

<sup>1</sup>Department of Computer Science & Engineering, Amity University Haryana, [bhateja.neha@gmail.com](mailto:bhateja.neha@gmail.com)

<sup>2</sup>Department of Computer Science & Engineering, Amity University Haryana, [sethinishu@gmail.com](mailto:sethinishu@gmail.com)

---

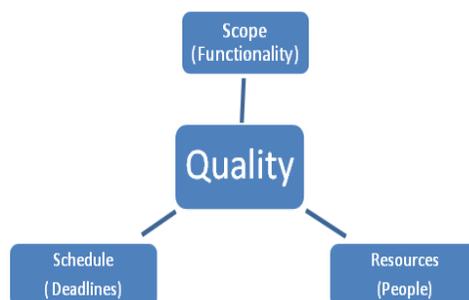
**Abstract:** *Software testing is the process of testing the implementation of software with some test data as input and observing the output of the software. It is a method to evaluate the attributes or usability of a program or a system to measuring that the developed system or product meets its quality. Software testing is an important technique for evaluating the quality of a software product. High quality software has more value than the lower quality product. This paper presents few attributes of the software testing to improve software quality.*

**Keywords:** *Software Testing, Quality, Functional Testing, Structural Testing*

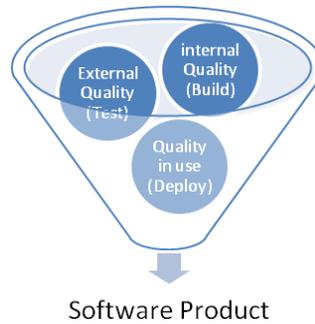
---

## I. INTRODUCTION

Testing is an important part of software development life cycle. Software testing process helps to reduced maintenance efforts, reduce errors and the cost of the software. Testing process is performed by the testers as well as by the end users before releasing the product. Testing can be done at the developer end and the customer end to predict the behavior of the software [1]. With the help of testing, the quality of the software can be measured. Quality refers to the quality of product, process, system, service, and information. It means quantifiable characteristics are achieved from the users need. Quality refers to those features of a product which meet the need of customers and provides product users satisfaction [2].

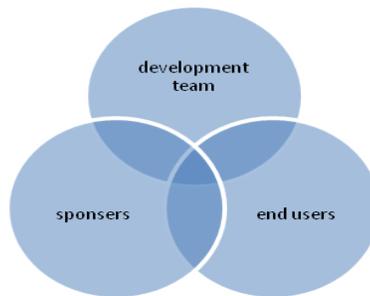


Quality is the important factor of any software engineering project. Without measuring, we cannot be sure of the level of quality in software. Software Quality Assurance (SQA) is the combination of the software development process, that includes software design, coding, source code, code review, configuration management ,change management, and release management. Quality assurance guarantees that the project will be completed based on approved specifications, standards and functionality [3]

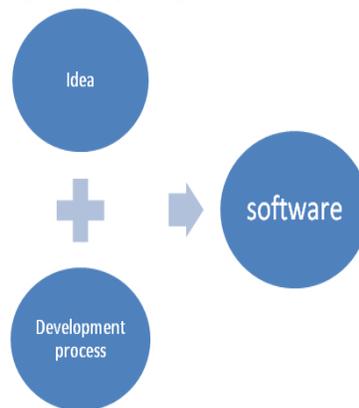


## II. ASPECTS FOR THE QUALITY

To achieving quality, the value of the software should be. Software quality depends upon the three groups of users: the developers, the end users and the sponsors. The aspect of the quality in these groups isn't the same but they all are responsible to achieve it in their own way [3,4].



- The software's users defined as the person who applies the software to some problem to achieve the goals.
- The development team is responsible for creates the software. the development process converts the idea into software
- The sponsors of the project are the people who paying for the creation of the software.



### **A. FUNCTIONAL ASPECT**

Functional quality means how well the software conforms or complies with requirements, based on its functional specifications. It also expresses how well the software can perform the tasks requested by the users [5, 6].

The achieve quality at functional level following attributes are considered:

#### *1. Minimizing the defects:*

It is being observed that over 30% defects found in the software come from the requirement phase which reduces the software's reliability, security and functionality. Achieving software with zero defects is a complex task, but it could be improved with re-requirement phase.

#### *2. Achieving specified requirements:*

Getting the right requirements plays a crucial role. This could either be achieved from sponsors, users, visual requirement editors, social features or online review. This requires implementing correct requirements not only which is initially defined but can be reconsider after moderation.

#### *3. Efficiency:*

The state or quality of being efficient. It means to be able to put in place the training, support and structures to continue to manage and improve efficiency. Efficiency is degraded if rework or change is not done. it also means seeing beyond the way things are now.

#### *4. User Friendly:*

The software user interface should be simplified and easy to learn. By providing simplified interface it becomes easy for the user to access and hence an important consumer application.

#### *5. Customer Satisfaction:*

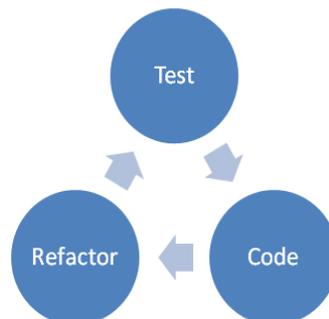
Quality Speaks. If the customer is satisfied with the quality of product, he comes back for it every time and gives good referrals to others. Customer loyalty comes with quality and the service you provide. And it is spread via Social media. Similarly poor quality, dissatisfaction is also communicated quickly.

### **B. STRUCTURAL ASPECT**

Structural aspect is related to the internal structure of the software or system where testing purpose is to get the information about how and what is happening inside the software or system. Structural testing is also referred as White –box testing. To perform the structural testing, the tester should know about the internal knowledge as well as testers also required knowledge about how the software or system is implemented [8,9]. The following attributes are considered at the structural aspect:

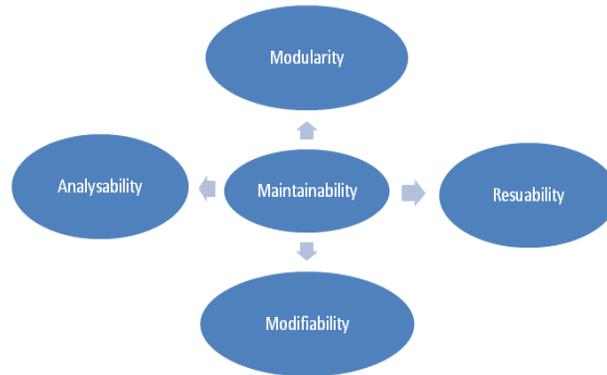
#### *1. Code Testing:*

Start with what you need to test and then write the code for that, at this point if found something is not testable which means it requires refactoring (modification). After the refactoring this process is repeatedly done until the coding becomes testable.



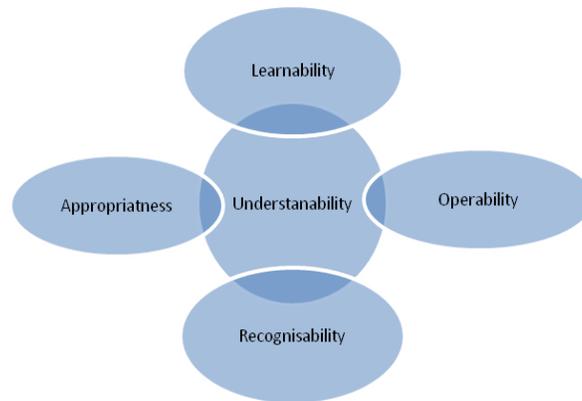
#### *2. Code Maintainability:*

It is related to the simplicity of the structural and design of the software in terms of whether new code is easily introduced or not, how easy is to change the existing code without any bug?, the modules of the software are loosely or tightly coupled? What is the cyclomatic complexity of the software in terms of independent path?



**3. Code Understandability:**

It is related to the readability; learn ability and operability of the code in terms of how easy to understand the code by the new developer to working on the existing system.



**4. Code Efficiency:**

It defines the amount of work in terms of behavior and capacity that the software can do with limited number of resources and in defined timeframe.

**5. Code Security:**

How secured the software is in terms of integrity, authenticity and confidentiality. Does the software handle the situation like buffer overruns or SQL injection?



### **C. PROCESS ASPECT:**

Software process factors also influence the quality of the software. Customer feedback is essential in the process of quality. Regular customer feedback helps the organizations to achieve the quality by detect and solve problems [10]. Following are the important features that are considered from the process aspect of the software:

1. Time and efforts taken to complete an activity or process
2. Resources required in terms of efforts in person-days for processes or activities.
3. Number of defects discovered during the software development
4. Whether the software is delivered within the defined budget

### **III. CONCLUSION**

The objective of software testing is to locate and fix the problem to improve quality. Quality is the main concern of software development process. Without measuring the quality, we cannot be sure of the value of the software. High quality product means better valuable software from the user's point of view. Only testing methods are widely used to ensuring the quality of the software. Quality of the software cannot be measured in a single way; it depends on the various aspects like functional, structural and process attributes.

### **REFERENCES**

- [1] Beizer, Boris, "Black-Box Testing Technique for Functional Testing of Software and System" New York Wiley, ISBN: 0471120944 Physical description: xxv, 194 p. ill.; 23cm (1995).
- [2] J. A. Whittaker, "What is Software Testing? And Why Is It So Hard?" IEEE Software, January 2000, pp. 70-79
- [3] Barber, Scott "Software Testing: An Introduction", PerfTestPlus (2006).
- [4] G. V. Bochmann and A. Petrenko. Protocol testing: review of methods and relevance for software testing. In Proc. ACM/SIGSOFT Int. Symp. Software Testing and Analysis, pages 109–124, 1994.
- [5] H. Do, S. Elbaum, and G. Rothermel. Supporting controlled experimentation with testing techniques: An infrastructure and its potential impact. *Empirical Softw. Eng.*, 10(4):405–435, 2005
- [6] Y. Le Traon, B. Baudry, and J.-M. Jez'equel. Design by contract to improve software vigilance. *IEEE Trans. Softw. Eng.*, 32(8):571–586, 2006.
- [7] Lytz, R., Software Metrics for the Boeing 777: A Case Study, *Software Quality Journal*, 4, 1-13 (1995)
- [8] Nachiappan Nagappan, Brendan Murphy, and Victor Basili, "The Influence of Organizational Structure On Software Quality: An Empirical Case Study", January 2008.
- [9] L. S. Maurya et al, Comparison of Software Architecture Evaluation Methods for Software Quality Attributes, *Journal of Global Research in Computer Science*, 1 (4), November 2010
- [10] Mark Kevitt, "Best Software Test & Quality Assurance Practices in the project Life-cycle", April 2008.