

# Policies, Processes, Procedures and Measurement in Software Quality Assurance: A State Of Art Survey

M.Pravarsha Reddy<sup>1</sup> and K L Raghavender Reddy<sup>2</sup>

<sup>1</sup>Department of CSE, Vardhaman College of engineering, shamshabad, Telangana, India

<sup>2</sup>Department of CSE, Vardhaman College of engineering, shamshabad, Telangana India

## Abstract:

To focus on time-consuming and complex activity and also to construct the error-free Software, Software development and maintenance is used. Software Quality assurance is a way of avoiding errors or imperfection in manufactured activity and keeping away from problems when distributing explanation or services to consumers. The expressions "quality assurance" and "quality control" are frequently used interchangeably to refer to ways of make sure the value of a product. The software quality assurance policy facilitates to normalize software quality and code management in association. A disordered software process can result in waste of time and exhausted developer assets. Processes and procedures are the set of activities which includes software development methodology, project management, configuration management, requirement management, estimation, software design, testing, etc. This paper reviews Policies, processes, procedures and Measurement in software quality assurance in software engineering domain.

**Keywords:** *Software quality assurance, quality control, software development methodology, project management, development methodology, configuration management, requirement management, software design, software testing*

## 1. Introduction

According to Webopedia, Software Quality Assurance, truncated as SQA, and also known as "software assurance". Software quality assurance is defined as "the intensity of assurance that software is at no cost from vulnerabilities, either purposely planned into the software or unintentionally introduced at any time throughout its lifecycle, and that the software role in the proposed manner". According to the DHS (Department of Homeland Security) SQA is Trustworthiness, Predictable Execution, Conformance [1] [4]. To broadcast paramount practices and methods that encourage security, integrity, and reliability in software code development, as well as processes and procedures

that reduce the possibilities of invalid code, malicious code, or trap doors that could be initiated during development, DHS will make possible a national public-private attempt. To identify potential security vulnerabilities there are open-source software tools for software assurance. According to United States Department of Defense (DoD) SQA relay on "the level of self-reliance that software functions as projected and is liberated of vulnerabilities, either on purpose or involuntarily considered or introduced as element of the software."

## 2. Literature survey

In the SQA Management plays a huge role. Mishra and Mishra[16] explained about addressed management and global software development process and quality assurance concern for providing global software inspection process with meticulous reference to the area of distributed software development. Follow-up stage, conference stage, modify, entity scrutiny stage and the complex stage various stages in global software development. To code churn, dependency and code testing earlier errors in the distributed software development can be associated. In terms of less cost, minor improvement time and improved quality GSD methodologies and practices tender a lot of advantages.

D. Nirmala, T. Latha Maheswari[17] explained about the guarantee liberation of elevated quality software to the end user of overall attempt of the software engineering. Successful production of test cases, prioritizations of test cases are number of primary issues in software testing which need to be engaged in. test strategy, test case generation, test execution and test evaluation are the four characteristics on which automated test framework mostly depends.

Antonia Bertolino in Software Testing Research: Achievements, Challenges, Dreams explained about Software engineering grasp about a number of

disciplines dedicated to stop and remedy malfunctions and to deserve sufficient performance[18]. Beside the improvement of cycle and ahead of, aimed at dissimilar goals software testing is a wide term surrounding a variety of behavior. Hence, a collection of challenges are faced by software testing research. A regular roadmap of the most significant confront are to be addressed here. By the marvelous research challenges the routes from the accomplishment to the dreams are covered.

Shivkumar Hasmukhrai Trivedi in Software Testing Techniques explained about reduction of errors, cut preservation and in general software costs [19]. Over the last few decades hopeful to improve software quality tools, frequent software development, testing methodologies, and techniques have emerged. How to get a appropriate set of cases to experiment a software system is one of the most important problems within software testing area.

### 3. Software quality management

Software quality management is a managing course of action that aims to develop and deal with the quality of software. Because of new skills in the software development process such as object-oriented development, CASE tools, etc. In accumulation, a sprouting importance of software quality management and the assumption of quality management techniques from industrialized can be observed [8]. The concept of quality generally used in manufacturing differs from software quality. The reasons for these differences are the uniqueness of the product that the customer desires should be reproduced by the software specification. Maintainability, usability and reliability are software quality aspects which cannot be exactly specified and measured. Defining a absolute software specification of software development is very complex at early stages. Quality assurance, quality planning, quality control are the activities of software quality management. On the software and software maturity process an independent check is afforded by Quality management.

#### 3.1 Process and product quality

In general if quality of development process is affected then the quality of delivered product also affects. To accomplish appropriate quality level the process is enhanced and the quality of the product can be calculated [3]. There is an unambiguous

connection between production process and product quality in industrialized systems in manufacturing systems. Quality of software is extremely influenced by the knowledge of software engineers. Maintainability, reliability, usability, etc., are software quality attributes which are complex to measure. Process quality has a major influence on the quality of the software exposed by the skill. Defining process principles, supervising the maturity process, reporting the software are the events integrated in Process quality management. Figure 1 shows quality based assessment.

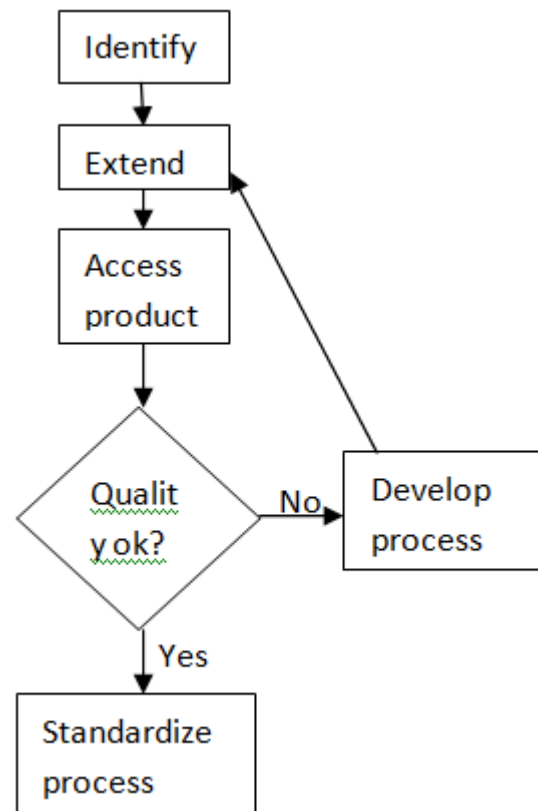


Fig1: process based quality assessment

In process and product quality we have discussed about result of end product, attributes of process quality management and steps to be followed in process based quality assessment and next we discuss about who is going to write about software quality policy, quality principles and their applications in any organization.

(i) Who writes the software Quality Policy?

By scripting a quality policy superior management exhibits requirement to authority and make sure that

everyone in the organization has read it and be conscious of its assertions [2]. ISO 900 is Quality Management System which is a necessary condition of documented quality policy.

#### (ii) Writing a software Quality Policy

Applications to an organization's business environment from basic quality principles, Quality policies are originated. They handle the following areas:

##### (a) Customer's hub

- Accepting existing and expectations of customer requirements
- gathering the customer's needs
- determined to go beyond the customer's outlook

##### (b) Leadership

- Producing a situation where one and all can contribute to excellence and supervise the liberation of vision

##### (c) Appointment with people

- To contribute to quality make one and all conscious of the want
- Make sure full commitment in actions that contact quality

##### (d) Process approach

- Mounting a obligation to illustrate all work in conditions of processes

##### (e) System approach

- Organizing the business as a set of cooperate processes

##### (f) Frequent upgrading

- to develop quality promote regular improvement of processes

##### (g) Fact based management

- Based on analysis of process data make assessment

##### (h) Equally advantageous trader relationships

- With suppliers develop close co-operative relationships.

#### 3.1.1 Examples of Quality Policy

On familiar quality principles and specific business processes Quality policy testimonials can be attentive.

##### (a) General

As initial priority the administration is dedicated to deliver quality software to customers. To business survival it is predictable that unswerving fulfillment of customer's need is crucial.

##### (b) Planning

Plans shall be created for all projects by Software Project Management. Describing the testing possibility and activities, software test plan is a document. For officially testing any software it is the source.

##### (c) Observing and scheming

At intervals approved with the project supporter project managers shall generate project presentation information.

##### (d) Manuscript requirements

Describing the functions, presentation and the interface requirements of the software product, all projects shall put in order a Software Requirements Specification

##### (e) Architectural intend

To establish that design inputs all projects shall conduct a Design Input Review such as Software Requirements Specifications are absolute, explicit and accurate and to sustain the development of a design solution acquire satisfactory quality

##### (f) Coding

With predefined coding principles projects shall develop software in conformity. To imitate any project definite practices coding standards shall be modernized

##### (g) Testing

For testing in a Software Test Plan all projects shall describe the approach. To unit, integration and acceptance testing, the STP shall describe the project's approach

##### (h) Organizing quality

All projects shall prepare and execute the software quality management actions obligatory to make sure that the customer's declared and implied needs are gathered and that the software product is urbanized in conformity with predictable best practice

##### (i) Managing people

Suitably skilled to carry out their assigned responsibilities and make sure those development team members conscious of their quality

responsibilities, Project managers shall be responsible.

## 4. Software Testing Metrics

A Metric is a quantitative appraisal of the scale to which a method, method component, or procedure possesses a given attribute [7]. To evaluate the quality of the project Software Metrics is used. Metric is a scale for measurement and entity used for describing an attribute. For computing the attribute “Weight”, “Kilogram” is a metric. In software for a program we write thousand lines of code here number of issues is one measurement and number of lines of code is an additional measurement. Metric is clear from these two measurements.

### 4.1 Software Measurement

Measurement is the quantitative proposal of degree, quantity, measurement, talent, or volume of some attribute of a artifact or process. For Software Test Manager Generation of Software Test Metrics is the most significant task. Test Metrics are used to acquire the conclusion for next phase of actions such as, approximation of cost & plan future projects. To achieve the project be aware of the kind of development required. Take decision on procedure or technology to be modified [9].

The efficiency of a process is considered by deriving a set of metrics based on results of the process, such as: inaccuracy exposed before discharge of the software. Imperfection delivered to and reported by the users. Human attempt exhausted. Calendar time finished. Time and exertion to complete each standard activity. To advance the process and eventually the resultant products, process guidelines such as CMM, ISO 9000, SPI and SPICE” were recommended since 90s. For better accepting the attributes of representation and evaluate the quality of the software, Product metrics is very helpful for software engineers [11].

- Product metrics help software engineers to get insight into the plan and creation of the software.
- Focus on exact attributes of software engineering work products consequential from analysis, design, coding, and testing.
- To assess quality based on a set of clearly defined rules provide a systematic way.

Product metrics has two classes:

Dynamic metrics: these metrics are calculated during the implementation of a program.

Static metrics: Static metrics are based on capacity of representations of the system such as the design, program or documentation.

The effectiveness and the consistency of a program Dynamic metrics are related. Complexity, understandability, maintainability, etc are the attributes of software quality which is related to static metrics such as code size

### 4.2 Metrics Life Cycle

Analysis

- Metrics recognition
- Defining the recognized metrics

Communication

- To stakeholders and testing team clarify the necessitate of metrics
- For dealing out the metrics instruct the testing team about the data points required to be captured

Evaluation

- Capture and authenticate the data
- Using the data captured estimate the metrics value

Report

- With successful termination develop the report
- Distribute the report
- Take feedback

### 4.3 Types of Manual Test metrics

Test metrics is mainly divided into two types they are Base Metrics and Calculated Metrics. During the test case improvement and implementation, data collected by the Test Analyst these data is known as Base metrics. Calculated Metrics are consequential from the data congregated in Base Metrics. For Test Reporting intention these Metrics are usually tracked by the test lead [15] [13].

Results and discussions:

Below we have table layout for the data retrieved from the test analyst who is in fact concerned in testing: Table 1 shows testing metrics

**Table1: testing metrics**

S.no	Testing metric	Data retrieved during test case improvement and implementation
1	Number of necessities	10
2	Total number of test cases written for all necessities	100
3	Average number of test cases written for per necessities	25
4	Number of test cases unexecuted	25
5	Number of test cases blocked	12
6	Number of test cases failed	32
7	Number of test cases passed	42
8	Total number of test cases executed	75
9	Low defect counts	8
10	Medium defect counts	9
11	High defect counts	14
12	Critical defect counts	9
13	Total number of defects recognized	40

### Procedures and Definitions for Calculating Metrics:

1) Percentage of Test cases completed:

$$\text{Percentage of Test cases completed} = \frac{\text{Number of Test cases completed}}{\text{Total number of Test cases written}} * 100.$$

2) Percentage of Test cases not executed:

$$\text{Percentage of Test cases not executed} = \frac{\text{Number of Test cases not executed}}{\text{Total number of Test cases written}} * 100.$$

3) Percentage of Test cases accepted:

$$\text{Percentage of Test cases accepted} = \frac{\text{Number of Test cases accepted}}{\text{Total number of Test cases completed}} * 100.$$

4) Percentage of Test cases ineffective:

$$\text{Percentage of Test cases ineffective} = \frac{\text{Number of Test cases ineffective}}{\text{Total number of Test cases completed}} * 100.$$

5) Percentage of Test cases Blocked:

$$\text{Percentage of Test cases Blocked} = \frac{\text{Number of Test cases Blocked}}{\text{Total number of Test cases completed}} * 100.$$

6) Deficiency Density:

$$\text{Deficiency Density} = \frac{\text{Number of Defects identified}}{\text{size}} \text{ (Here "Size" is nothing but necessity)}$$

7) Defect Removal Efficiency (DRE)

$$\text{Defect Removal Efficiency (DRE)} = \frac{\text{Number of Defects found during QA testing}}{(\text{Number of Defects found during QA testing} + \text{Number of Defects found by End user})} * 100.$$

The test efficiency of the system is recognized by Defect Removal Efficiency. During improvement & Quality Analysis testing few defects are recognized. Few faults are recognized during Alpha & Beta testing, end user, which could have been recognized during Quality Analysis testing phase.

8) Defect Leakage: To recognize the effectiveness of the QA testing, Defect Leakage is used

$$\text{Defect Leakage} = \frac{\text{Number of Defects found in UAT}}{\text{Number of Defects found in QA testing.}} * 100.$$

9) Defects by Priority: Number of defects recognized based on the precedence of the defect which is used to make a conclusion, the excellence of the software.

$$\text{Percentage of Critical Defects} = \frac{\text{Number of Critical Defects identified}}{\text{Total number of Defects identified}} * 100$$

10) Percentage of High Defects

$$\text{Percentage of High Defects} = \frac{\text{Number of High Defects identified}}{\text{Total number of Defects identified}} * 100$$

11) Percentage of Medium Defects

$$\text{Percentage of Medium Defects} = \frac{\text{Number of Medium Defects identified}}{\text{Total number of Defects identified}} * 100$$

12) Percentage of Low Defects

$$\text{Percentage of Low Defects} = \frac{\text{Number of Low Defects identified}}{\text{Total number of Defects identified}} * 100$$

### 5. Quality assurance and standards

How the development association knows that the software has the compulsory level of quality and how

software quality can attain defined quality assurance [6]. Selection and definition of standards that are practical to the software development process is the main activity of quality assurance. Product standards and process standards are the two standards of quality. The standards which are applied to software product are product standards. The standards which are followed during software development are process standards. The software standards are based on best performance and they provide a structure for implementing the quality assurance process [6] [7].

## 5.1 ISO (Indian Standard Organization)

To assist association ensure that they meet the needs of customers while meeting constitutional and authoritarian necessities interconnected to a product or program, quality management systems is designed which is the family of ISO 9000[13]. ISO 9000 deals with the basics of quality management systems, with seven quality management principles on which family of standards is based. ISO 9001 deals with the necessities that association desires to meet the standard must accomplish. The ISO 9000 standard is continually being revised by standing technical committees and advisory groups. 1987 version, 1994 version, 2000 version, 2008 version, 2015 version etc are versions revised by STC.

### 5.1.1 How to get ISO 9000 certification

#### 5 Steps to ISO 9001 Certification

##### Step 1: Preparation

- i. Get an Understanding of ISO 9000:2015
- ii. Appoint Management Representative
- iii. The management representative needs to get some training
- iv. Get full support from management and employee is crucial for the success of **ISO 9000** quality management system

##### Step 2: Documentation

Following are the documents required for the ISO 9000 certification

- i. Quality policy
- ii. Procedures
- iii. Scope of the quality management system
- iv. Flowchart
- v. Quality objectives
- vi. Work instructions
- vii. Forms

##### Step 3: Implementation

During this phase new requirements are introduced in customized ISO 9000. To implement successfully we need to understand the benefits of the new ISO 9000 requirements. One step at the time is selected to start with the section on document control. Requirements in a meeting or have the appropriate department leads are explained, so that they can explain to their staff. Preparation and work directions go hand in hand.

##### Step 4: internal audits

Internal audits are the assessment in which that particular association appraises itself to see if all ISO 9000 requirements are followed or not. Internal audit is done by management representative except his work; another employee audits the work of management representative

##### Step 5: Certification

- i. Suitable registrar is selected
- ii. Audit certification
- iii. Reaping the marketing benefits
- iv. Certification is conserved
- v. Decision

## 6. Conclusion

Testing is demanding and significant activity. Software quality assurance problems are recognized and solutions are recommended to deal with the problems and evolve the software quality. In this paper, we reviewed policies, procedures and various testing metrics that are used in software quality assurance. Even though there is computerization in each of the involved activities, the tester's capability remains prerequisites as much as need for fairly accurate solutions under certain situations.

## REFERENCES

- [1] Sametinger, Johannes, “*Software Engineering with Reusable Components*”, SpringerVerlag GmbH, 2006.
- [2] B. W. Boehm, “*Software Engineering Economics*,” Prentice Hall, 1981.
- [3] The Software Crisis.  
Source: <http://www.unt.edu/benchmarks/archives/1999/july99/crisis.htm>
- [4] Balan S, “*A Composite Model for Software Quality Assurance*,” 2003.
- [5] Galin, Daniel, “*Software Quality Assurance – From theory to implementation*,” Pearson – Addison Wesley, England, 2004.
- [6] Chandramouli, pradiba, “*Software Quality Assurance*,” 2003.
- [7] Felipe Ortega, Daniel Izquierdo, Pedro Coca, “*Introduction to software quality evaluation*,” 2004.
- [8] ESA Board for Software Standardization and Control, Guide to software quality assurance
- [9] M. Jorgensen, “*Software quality measurement*,” Vol. 30, Dec. 1999, Pages 907 – 912.
- [10] Norman f. Schneidewind, “*Knowledge Requirements for Software quality measurement*,” Vol. 8, 2001, Pages 201-205.
- [11] Omer Korkmazsystem, “*Simulation for software Quality Assurance (SQA)*,” 2007.
- [12] David Hooker, “*Seven Principle of Software Development*,” 1996.
- [13] Darrel Ince, “*ISO 9001 and Software Quality assurance*”.
- [14] Warden R, “*Software Reuse and Reverse Engineering in Practice*”. 1992, Pages: 283–305.
- [15] NITS (2006), NIST/SEMATECH e-Handbook of Statistical Methods,
- [16] Deepti Mishra .Alok Mishra. “*A Global Software Inspection Process for Distributed Software Development*,” Vol: 18, 2012, Pages: 2731-2746.
- [17] Nirmala, T. Latha Maheswari, “*Automated Test Framework For Software Quality Assurance*,” vol:4, December 2015, Pages.224 – 234 .
- [18] Antonia Bertolino, “*Software Testing Research: Achievements, Challenges, Dreams*” May 2007, Pages 23-25.
- [19] Shivkumar Hasmukhrai Trivedi , “*Software Testing Techniques*,” Vol: 2, October 2012, Pages: 433-439

