Chapter 9

Software Quality Assurance

Software Quality Assurance or SQA refers to a technique to ensure the right quality and standards during the development of software. Similar to other forms of quality testing procedures, Software Quality Assurance is used to keep a check on the functionality as well as the correctness of a particular software. An organization uses Software Quality Assurance to find out if the software being developed for use meets the requirements and quality standards that it seeks.

What is Software Quality Assurance?

The International Organization of Standardization, or ISO, under its ISO/IEC 25010:2011 guidelines, defines Quality Assurance. A simplified version of it can be as follows −

“Quality Assurance relates to the outcome of a product when it will be used in a particular context. It can be done for any system including the computer system, and software products related to other computers.”

“A product quality model relates to static properties of software and dynamic properties of the computer system. The model is applicable to both computer systems and software products”.

Benefits of Software Quality Assurance

There are several benefits of employing Software Quality Assurance in an organization, such as –

- Saving resources

Errors are costly for any organization, be they large or small. Errors require redoing of work as well as carrying an opportunity cost with them. By using Software Quality Assurance to maintain a form of check on the development of software, errors can be avoided. For instance, slow/bad software released to the public will result in disappointed customers and the need for releasing updates.

- Saves time

If an organization conducts simultaneous software checks during the development stage, it can release blemish-free software during the time of release. However, if no checks are maintained, the software will require reworking - that amounts to the expense of man-hours as well as other resources.

- Maintains and upholds goodwill in the market

Good products always sell. Apple Inc is one of the biggest examples of this phenomenon. Apple has been far behind in terms of the release of certain features or software in comparison to Android. There's a simple reason for that. Android chooses to release first and later work on the limitations of its software, whereas Apple follows the opposite approach. They choose to release late in order to release the best version of the same software. It results in a lower frequency of updates as well as happier consumers.

- Increases safety and lowers the risk of contamination

Given the widespread and universal use of computing devices in the 21st century, the possibility of hacking and cyber security issues have gained greater importance than earlier. Bugs and performance issues are an obvious possibility when it comes to the development of software without the usage of Software Quality Assurance practices. However, much graver consequences can ensue if Software Quality Assurance practices and standards are not complied with when considering cyber security issues, and the possible consequences. For instance, all banks and financial institutions in the modern world are run by software due to the vast sizes of funds they manage, the gigantic customer base as well as having branches in different countries and continents. If a financial institution uses software having high penetrability, it is vulnerable to hacking. It can affect millions of people, and could also cause the fall of a nation. Keeping that in mind, it is important for critical infrastructure such as banks, utility companies, government institutions, and so on, to maintain a very high quality of cyber security.

Quality Assurance Criteria:

Below are the Quality assurance criteria against which the software would be evaluated against:

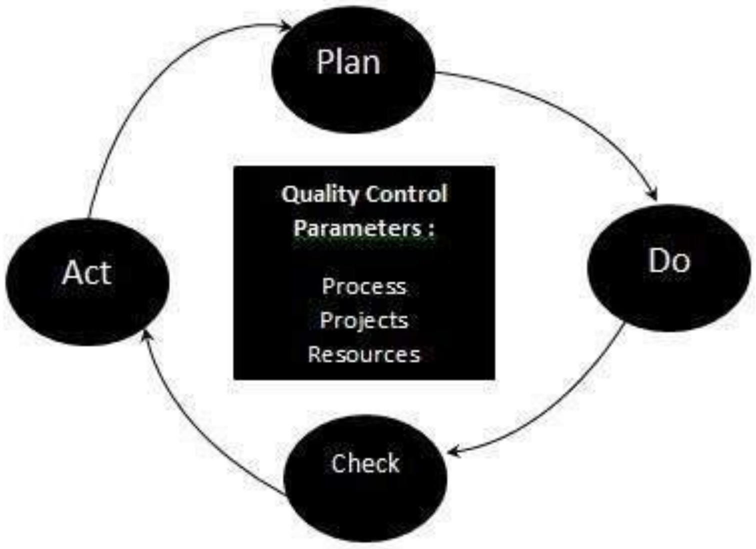
- correctness
- efficiency
- flexibility
- integrity
- interoperability
- maintainability
- portability
- reliability
- reusability
- testability

- usability

What is Quality Control

Quality control is a set of methods used by organizations to achieve quality parameters or quality goals and continually improve the organization's ability to ensure that a software product will meet quality goals.

Quality Control Process:



The three class parameters that control software quality are:

- **Products**
- **Processes**
- **Resources**

The total quality control process consists of:

- **Plan - It is the stage where the Quality control processes are planned**
- **Do - Use a defined parameter to develop the quality**
- **Check - Stage to verify if the quality of the parameters are met**
- **Act - Take corrective action if needed and repeat the work**

Quality Control characteristics:

- **Process adopted to deliver a quality product to the clients at best cost.**
- **Goal is to learn from other organizations so that quality would be better each time.**
- **To avoid making errors by proper planning and execution with correct review process.**

What is software quality management?

Software Quality Management ensures that the required level of quality is achieved by submitting improvements to the product development process. SQA aims to develop a culture within the team and it is seen as everyone's responsibility.

Software Quality management should be independent of project management to ensure independence of cost and schedule adherences. It directly affects the process quality and indirectly affects the product quality.

Activities of Software Quality Management:

- **Quality Assurance - QA aims at developing Organizational procedures and standards for quality at Organizational level.**
- **Quality Planning - Select applicable procedures and standards for a particular project and modify as required to develop a quality plan.**
- **Quality Control - Ensure that best practices and standards are followed by the software development team to produce quality products.**

Software quality metrics can be classified into three categories −

- **Product metrics − Describes the characteristics of the product such as size, complexity, design features, performance, and quality level.**
- **Process metrics − These characteristics can be used to improve the development and maintenance activities of the software.**
- **Project metrics − This metrics describe the project characteristics and execution. Examples include the number of software developers, the staffing pattern over the life cycle of the software, cost, schedule, and productivity.**

Product quality metrics

This metrics include the following −

- **Mean Time to Failure**
- **Defect Density**
- **Customer Problems**
- **Customer Satisfaction**

Mean Time to Failure

It is the time between failures. This metric is mostly used with safety critical systems such as the airline traffic control systems, avionics, and weapons.

Defect Density

It measures the defects relative to the software size expressed as lines of code or function point, etc. i.e., it measures code quality per unit. This metric is used in many commercial software systems.

Customer Problems

It measures the problems that customers encounter when using the product. It contains the customers perspective towards the problem space of the software, which includes the non-defect oriented problems together with the defect problems.

The problems metric is usually expressed in terms of Problems per User-Month (PUM).

Customer Satisfaction

Customer satisfaction is often measured by customer survey data through the five-point scale −

- **Very satisfied**
- **Satisfied**
- **Neutral**
- **Dissatisfied**
- **Very dissatisfied**

In-process Quality Metrics

In-process quality metrics deals with the tracking of defect arrival during formal machine testing for some organizations. This metric includes −

- **Defect density during machine testing**
- **Defect arrival pattern during machine testing**
- **Phase-based defect removal pattern**
- **Defect removal effectiveness**

Maintenance Quality Metrics

Although much cannot be done to alter the quality of the product during this phase, following are the fixes that can be carried out to eliminate the defects as soon as possible with excellent fix quality.

- **Fix backlog and backlog management index**
- **Fix response time and fix responsiveness**
- **Percent delinquent fixes**
- **Fix quality**

Basically, a three-level structure of management exists in software development organizations −

- **Top management**
- **Department management**
- **Project management**

Top management Responsibility in quality assurance

- **Assure the quality of the companys software products and software maintenance services**
- **Communicate the importance of the product and service quality in addition to customer satisfaction to employees at all levels**
- **Assure satisfactory functioning and full compliance with customer requirements**

- **Ensure that quality objectives are established for the organizations SQA system and that its objectives are accomplished**
- **Initiate planning and oversee implementation of changes necessary to adapt the SQA system to major internal as well as external changes related to the organizations clientele, competition, and technology**
- **Intervene directly to support resolution of crisis situations and minimize damages**
- **Ensure the availability of resources required by SQA systems**

The following steps can be taken by the top management to fulfill its responsibilities −

- **Establishing and updating the organizations software quality policy.**
- **Assigning one of the executives such as Vice President for SQA to be in charge of software quality issues**
- **Conducting regular management reviews of performance with respect to software quality issues**

Department Management Responsibilities for SQA

Middle managements quality assurance responsibilities include −

- **Management of the software quality management system (quality system-related tasks)**
- **Management of tasks related to the projects and services performed by units or teams under the specific managers' authority (project-related tasks)**

Quality system-related responsibilities

These include SQA activities to be performed on the department level −

- **Preparation of the departments annual SQA activities program and budget, based on the recommended program prepared by the SQA unit**
- **Preparation of the departments SQA systems development plans, based on the recommended plan prepared by the SQA unit**

- **Control of performance of the departments annual SQA activities program and development projects**
- **Presentation of the departments SQA issues to top management**

Project-related Responsibilities

These vary according to the organizations procedures and distribution of authority; they usually involve −

- **Control of compliance to quality assurance procedures in the departments units, including CAB, SCM and SCCA bodies**
- **Detailed follow-up of contract review results and proposal approvals**
- **Review of unit performance of planned review activities; approval of project documents and project phase completion**
- **Follow-up of software tests and test results; approval of projects software products**
- **Follow-up of progress of software development project schedules and budget deviations**
- **Advice and support to project managers in resolving schedule, budget and customer relations difficulties**
- **Follow-up of quality of maintenance services provision**
- **Detailed follow-up of the project risks and their solutions**
- **Follow-up of projects compliance with customer requirements and customers satisfaction**
- **Approval of large software change orders and significant deviations from project specifications**

Project management responsibilities on software quality

Most project management responsibilities are defined in procedures and work instructions; the project manager is the person in-charge of making sure that all the team members comply with the said procedures and instructions.

His tasks include professional hands-on and managerial tasks, particularly the following −

- **Professional hands-on tasks**
  - **Preparation of project and quality plans and their updates**
  - **Participation in joint customersupplier committee**
  - **Close follow-up of project team staffing, including attending to recruitment, training and instruction**
- **Management tasks**

Project managers address the follow-up issues such as −

- **Performance of review activities and the consequent corrections**
- **Software development and maintenance units performance, integration and system test activities as well as corrections and regression tests**
- **Performance of acceptance tests**
- **Software installation in remote customer sites and the execution of the software system by the customer**
- **SQA training and instruction of project team members**
- **Schedules and resources allocated to project activities**
- **Customer requests and satisfaction**
- **Evolving project development risks, application of solutions and control of results**

Software Process Assessment

A software process assessment is a disciplined examination of the software processes used by an organization, based on a process model. The assessment includes the identification and characterization of current practices, identifying areas of strengths and weaknesses, and the ability of current practices to control or avoid significant causes of poor (software) quality, cost, and schedule.

A software assessment (or audit) can be of three types.

- **A self-assessment (first-party assessment) is performed internally by an organization's own personnel.**
- **A second-party assessment is performed by an external assessment team or the organization is assessed by a customer.**
- **A third-party assessment is performed by an external party or (e.g., a supplier being assessed by a third party to verify its ability to enter contracts with a customer).**

Software Process Assessment Cycle

According to Paulk and colleagues (1995), the CMM-based assessment approach uses a six-step cycle. They are −

- **Select a team - The members of the team should be professionals knowledgeable in software engineering and management.**
- **The representatives of the site to be appraised complete the standard process maturity questionnaire.**
- **The assessment team performs an analysis of the questionnaire responses and identifies the areas that warrant further exploration according to the CMM key process areas.**
- **The assessment team conducts a site visit to gain an understanding of the software process followed by the site.**

- **The assessment team produces a list of findings that identifies the strengths and weakness of the organization's software process.**
- **The assessment team prepares a Key Process Area (KPA) profile analysis and presents the results to the appropriate audience.**

1. What is the primary purpose of measurement in software quality management?

- A. To increase costs
- B. To evaluate and improve software quality
- C. To reduce team size
- D. To delay project timelines

2. Which of the following is NOT a type of measurement mentioned in software quality management?

- A. Product metrics
- B. Process metrics
- C. People metrics
- D. Project metrics

3. What do product metrics generally measure?

- A. Quality of the development team
- B. Characteristics of the software product
- C. Time taken for development
- D. Customer satisfaction

4. Why are process metrics important in software quality management?

- A. They help track team attendance
- B. They assess the efficiency and effectiveness of processes
- C. They focus solely on coding practices
- D. They are used to generate reports only

5. Which metric would help in assessing customer satisfaction?

- A. Defect density
- B. Response time

- C. Customer feedback
- D. Code complexity

6. What is the primary goal of software quality management process assessment?

- A. To improve product features
- B. To evaluate the effectiveness of processes
- C. To increase team size
- D. To reduce costs

7. 2. Which of the following is NOT a key aspect of the software quality management process?

- A. Process evaluation
- B. Quality assurance
- C. User training
- D. Continuous improvement

8. 3. What does the acronym CMMI stand for in the context of software quality?

- A. Capability Maturity Model Integration
- B. Common Management Model Integration
- C. Comprehensive Maturity Model Integration
- D. Current Management Model Integration

9. Which technique is commonly used for performance evaluation in software quality management?

- A. Code reviews
- B. Unit testing
- C. Surveys
- D. All of the above

10. What is a common outcome of a software quality management process assessment?

- A. Identification of process weaknesses

- B. Increased documentation
- C. Reduction in team members
- D. Higher software costs

11. What is the primary role of management in software quality assurance?

- A. To write code
- B. To ensure quality processes are followed
- C. To manage client relations
- D. To design software architecture

12. Which of the following is a key responsibility of management in QA?

- A. Conducting manual testing
- B. Defining quality standards
- C. Fixing bugs
- D. Writing test cases

13. How does management influence team performance in QA?

- A. By providing training and resources
- B. By writing test scripts
- C. By only focusing on deadlines
- D. By avoiding communication

14. What is one of the expected outcomes of effective management in QA?

- A. Increased software defects
- B. Higher customer satisfaction
- C. Decreased team morale
- D. Longer development cycles

15. Why is stakeholder involvement important in QA management?

- **A. It complicates the process**
- **B. It ensures alignment with business goals**
- **C. It creates more documentation**
- **D. It reduces team autonomy**