Software Quality Assurance

GUNJAN SAHNI

Definition

- Software quality assurance (or SQA for short) is the ongoing process that ensures the software product meets and complies with the organization's established and standardized quality specifications. SQA is a set of activities that verifies that everyone involved with the project has correctly implemented all procedures and processes.

- **SQA works parallel to software development**, an ongoing activity applied throughout the software development life cycle. Instead of making quality checks after completion, software quality assurance checks for quality issues in each development phase.
- Quality assurance focuses on **improving the software development process** and making it efficient and effective as per the quality standards defined for software products. Quality Assurance is popularly known as QA Testing
- SQA's ultimate goal is to **catch a product's shortcomings and deficiencies before the general public sees it**. If mistakes get caught in-house, it means fewer headaches for the development team and a lot less angry customers.

What is Quality?

- Quality is extremely hard to define, and it is simply stated: "Fit for use or purpose."
- It is all about meeting the needs and expectations of customers with respect to functionality, design, reliability, durability, & price of the product.

What is Assurance?

- Assurance is nothing but a positive declaration on a product or service, which gives confidence.
- It is certainty of a product or a service, which it will work well.
- It provides a guarantee that the product will work without any problems as per the expectations or requirements.

Quality Assurance cycle

- Plan
- Do
- Check
- Act

The Quality Assurance Cycle

1 Planning

Set up clear, appropriate and measurable goals and objectives in terms of policies, procedures, tasks and human resources

2 Implementation

Establish procedures to ensure the achievement of goals and objectives (e.g. development of partnerships, involvement of stakeholders, allocation of resources, and organisational or operational procedures)

3 Evaluation

Design mechanisms for the evaluation of achievements and outcomes by collecting and processing data in order to make informed assessment

4 Review

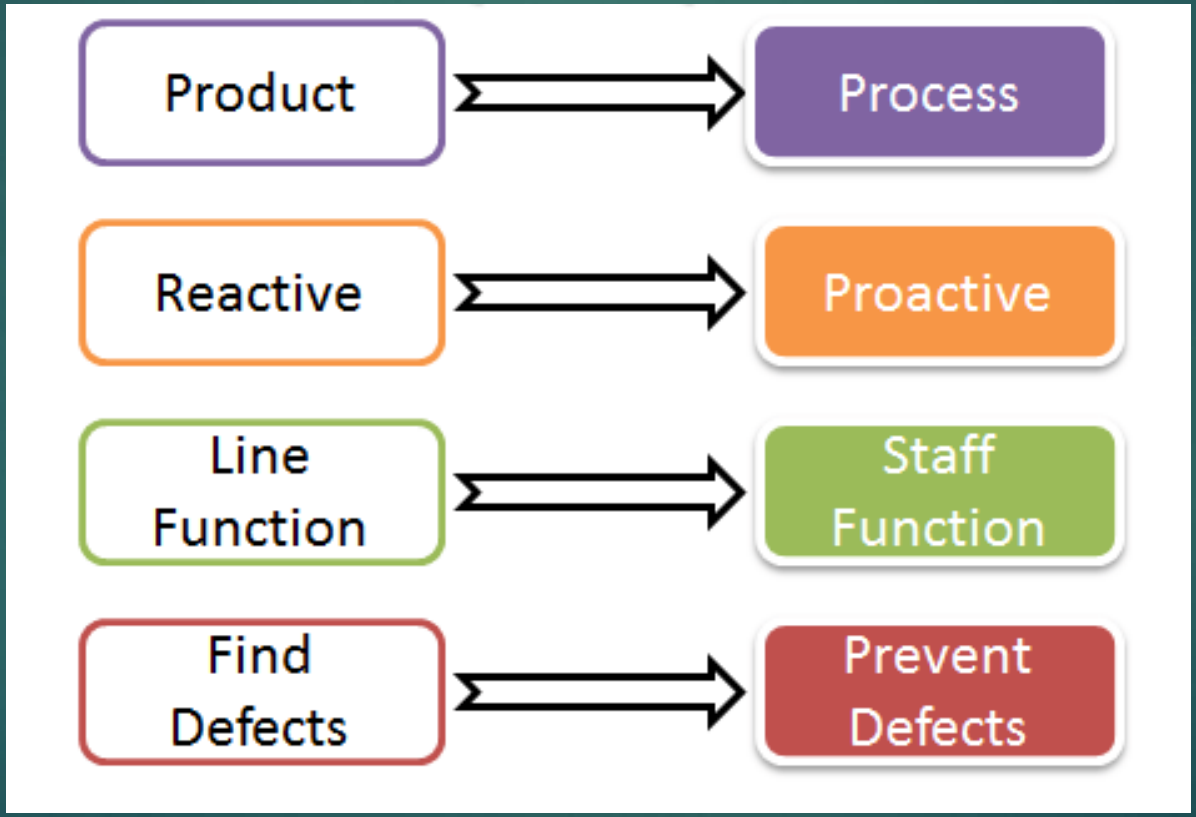
Develop procedures in order to achieve the targeted outcomes and/or new objectives; after processing feedback, key stakeholders conduct discussion and analysis in order to devise procedures for change

- **Plan** – Organization should plan and establish the process related objectives and determine the processes that are required to deliver a high-Quality end product.
- **Do** – Development and testing of Processes and also "do" changes in the processes
- **Check** – Monitoring of processes, modify the processes, and check whether it meets the predetermined objectives
- **Act** – A Quality Assurance tester should implement actions that are necessary to achieve improvements in the processes

Quality Control(QC)

- It is a Software Engineering process used to ensure quality in a product or a service. It does not deal with the processes used to create a product; rather it examines the quality of the "end products" and the final outcome.
- The main aim of Quality control is to check whether the products meet the specifications and requirements of the customer.
- If an issue or problem is identified, it needs to be fixed before delivery to the customer.
- QC also evaluates people on their quality level skill sets and imparts training and certifications.
- This evaluation is required for the service based organization and helps provide "perfect" service to the customers.

Quality Control Vs Quality Assurance



QA BEST PRACTICES

- Work closely with your product development team
- Utilize automation
- Continuously groom the test suite
- Hold peer reviews often
- Integrate agile processes early
- Prioritize bug tickets
- Hold exploratory testing

Capability Maturity Model Integrated

It is a process improvement approach developed specially for software process improvement. It is based on the process maturity framework and used as a general aid in business processes in the Software Industry. This model is highly regarded and widely used in Software Development Organizations. There are five levels of the QA process maturity:

- **Level 1: Initial**. QA processes of this level are very young, poorly managed, and unpredictable. It is typical for newly established QA processes, but it sometimes happens in companies that implement QA for quite some time already. Clearly, this is nothing good.
- **Level 2: Managed**. At this level, the basics of the QA process are established. In particular, there are established test planning, design, execution, and monitoring processes. Test environments are also in place. Yet, the QA process remains reactive.
- **Level 3: Defined**. Processes are well-organized and proactive. The test training program is in place. The QA process is integrated into the development process from its beginning.
- **Level 4: Measured**. At this level, the QA processes are measured and well-controlled. Measurements and reviews are conducted regularly and at each project development stage.
- **Level 5: Optimizing**. This level is all about process improvement. All activities, tools, and methods are assessed and analyzed in order to be optimized later.