SOFTWARE QUALITY ASSURANCE

CHAPTER 05
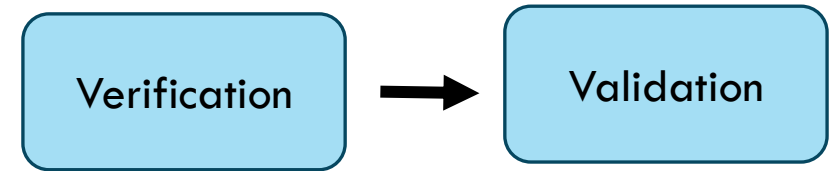
SOFTWARE QUALITY - DEFINITION

Software quality – IEEE definition

Software quality is:

1. The degree to which a system, component, or process meets specified requirements.

2. The degree to which a system, component, or process meets customer or user needs or expectations.

VERIFICATION AND VALIDATION

Verification → Validation

Verification is the process of ensuring that the product is being built correctly. It involves checking that the software meets specified requirements and that it is designed and implemented to deliver all required functionality. Verification activities include reviewing, inspecting, and testing to confirm that the software conforms to its specifications.

Example: Performing code reviews, unit testing, and integration testing to ensure that the software works correctly according to the technical design.

Validation is the process of ensuring that the right product is being built. It involves evaluating the final product to check if it meets the business needs and requirements of the end-users. Validation activities focus on ensuring the software fulfills its intended purpose and delivers the expected outcomes.

Example: Conducting user acceptance testing (UAT) or beta testing to verify that the software provides the desired functionality and meets user expectations.

Verification: Are we building the product right?
Validation: Are we building the right product?

VERIFICATION AND VALIDATION MODEL
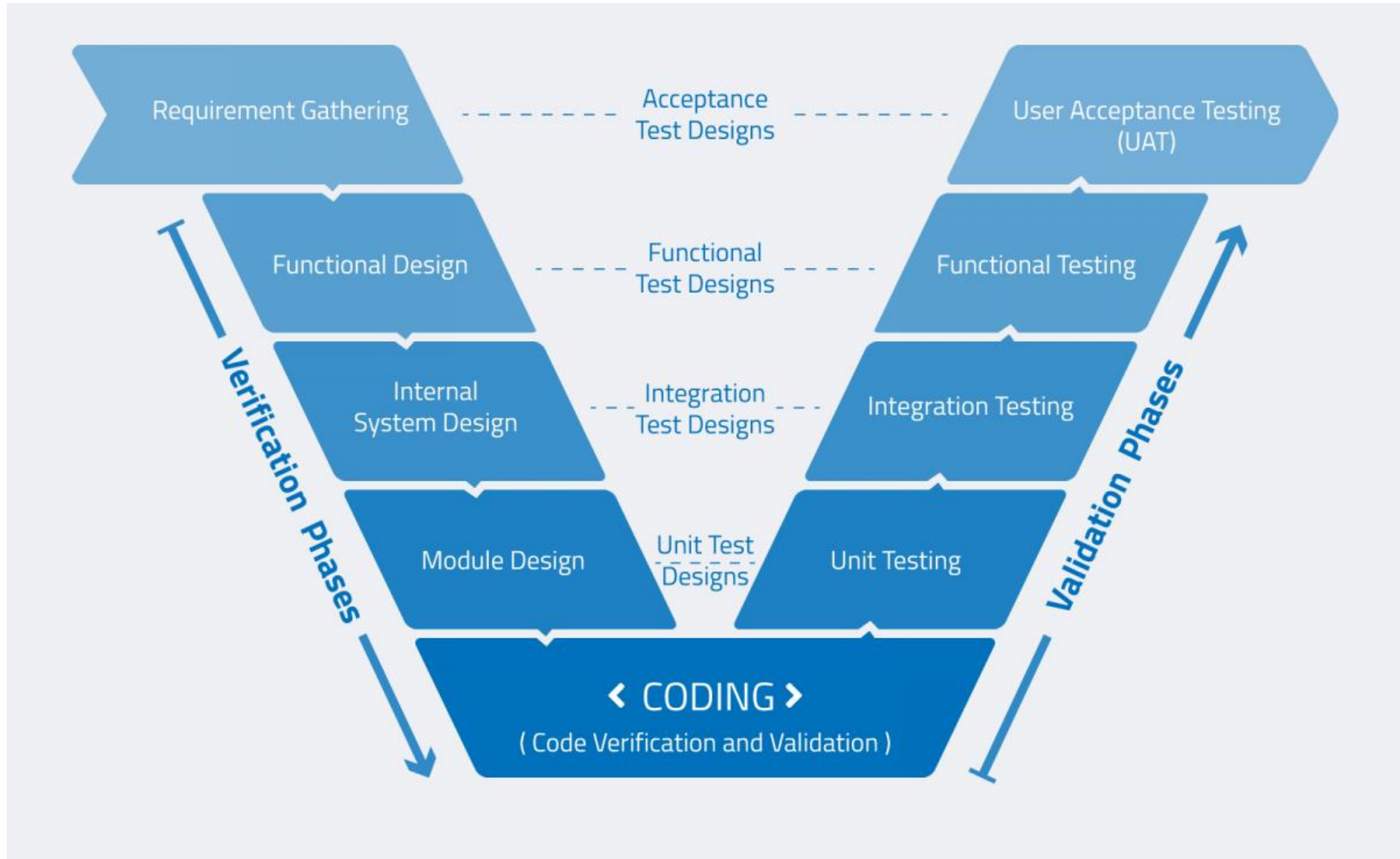
SOFTWARE QUALITY ASSURANCE - DEFINITION

Software quality assurance – The IEEE definition

Software quality assurance is:

1. A planned and systematic pattern of all actions necessary to provide adequate confidence that an item or product conforms to established technical requirements.

2. 2. A set of activities designed to evaluate the process by which the products are developed or manufactured. Contrast with quality control.

SOFTWARE QUALITY ASSURANCE

❑ In the simplest of terms, 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.

❑ The areas of the software that need quality assurance can include the software's performance, speed, maintainability, accuracy etc.

SOFTWARE QUALITY ASSURANCE VS. SOFTWARE QUALITY CONTROL

Two phrases are constantly repeated within the context of software quality: "**Quality control**" and "**Quality assurance**". Are they synonymous? How are they related?

These two terms represent separate and distinct concepts:

❑**Quality control** is defined as "a set of activities designed to evaluate the quality of a developed or manufactured product" in other words, activities whose main objective is the withholding of any product that does not qualify.

❑The main objective of **Quality assurance** is to minimize the cost of guaranteeing quality by a variety of activities performed throughout the development and manufacturing processes/stages

SOFTWARE QUALITY ASSURANCE VS. SOFTWARE QUALITY CONTROL

(1) Quality control and quality assurance activities serve different objectives.

(2) Quality control activities are only a part of the total range of quality assurance activities.

ELEMENTS OF SOFTWARE QUALITY ASSURANCE (SQA)

Testing: Software testing is a quality control function that has one primary goal—to find errors. The job of SQA is to ensure that testing is properly planned and efficiently conducted for primary goal of software.

Standards: The IEEE, ISO, and other standards organizations have produced a broad array of software engineering standards and related documents. The job of SQA is to ensure that standards that have been adopted are followed and that all work products conform to them.

Reviews and audits: Technical reviews are a quality control activity performed by software engineers for software engineers. Their intent is to uncover errors. Audits are a type of review performed by SQA personnel (people employed in an organization) with the intent of ensuring that quality guidelines are being followed for software engineering work.

Error/defect collection and analysis: SQA collects and analyzes error and defect data to better understand how errors are introduced and what software engineering activities are best suited to eliminating them.

ADVANTAGES OF SOFTWARE QUALITY ASSURANCE

1. SQA produces high quality software.

2. High quality application saves time and cost.

3. SQA is beneficial for better reliability.

4. SQA is beneficial in the condition of no maintenance for a long time.

5. High quality commercial software increase market share of company.

6. Improving the process of creating software.

8. It cuts maintenance costs. Get the release right the first time, and your company can forget about it and move on to the next big thing. Release a product with chronic issues, and your business bogs down in a costly, time-consuming, never-ending cycle of repairs.

THE SQA SYSTEM – AN SQA ARCHITECTURE

An SQA system always combines a wide range of SQA components, all of which are employed to challenge the multitude of sources of software errors and to achieve an acceptable level of software quality.

1. Pre-project components. To assure that

(a) the project commitments have been adequately defined considering the resources required, the schedule and budget; and

(b) the development and quality plans have been correctly determined.

2. Components of project life cycle activities assessment. Its components are divided into the following four sub-classes: – Reviews – Expert opinions – Software testing.

THE SQA SYSTEM — AN SQA ARCHITECTURE

3. Components of infrastructure error prevention and improvement. The main objectives of these components, which are applied throughout the entire organization, are to eliminate or at least reduce the rate of errors, based on the organization's accumulated SQA experience.

4. Components of software quality management. This class of components is geared toward several goals, the major ones being the control of development and maintenance activities and the introduction of early managerial support actions that mainly prevent or minimize schedule and budget failures and their outcomes.

5. Components of standardization, certification, and SQA system assessment. These components implement international professional and managerial standards within the organization.

6. Organizing for SQA — the human components. The SQA organizational base includes managers, testing personnel, the SQA unit and practitioners interested in software quality (SQA trustees, SQA committee members and SQA forum members).

WHAT IS AN SQA PLAN?

❑ An SQA plan, or software quality assurance plan, is a document that outlines the process that will be used to guide the QA procedures, measures, and more.

❑ The SQA plan typically includes details such as the scope of the quality assurance, the roles and responsibilities of the QA team, the types of testing that will be performed, and the criteria that will be used to evaluate the software.

❑ An SQA plan will include several components, such as purpose, references, configuration and management, tools, code controls, testing methodology, problem reporting and remedial measures, and more, for easy documentation and referencing.

SQA STANDARDS & CERTIFICATIONS

ISO 9001 - a widely-recognized standard for quality management systems, this is applicable to organizations of all sizes and in all industries. It also provides a framework for developing and implementing a comprehensive quality management system of your own.

ISO/IEC 5055 - this standard was developed to focus on the quality and integrity of the internal construction of a software application. It certifies software based on performance efficiency, reliability, security, and maintenance.

CMMI (Capability Maturity Model Integration) - a framework for improving the maturity and performance of an organization's processes, this can be applied to those related to software development and quality assurance. It provides best practices and guidance for improving the efficiency and effectiveness of these processes.

RESPONSIBILITIES OF THE SQA TEAM

1. The **SQA testing** team plans the tests and works on possible scenarios, procedures, and scripts.

2. The team is responsible for setting benchmark standards, procedures, and methods for **software quality assurance testing** and maintaining product quality.

3. The team is assigned to report any defects in the **software quality assurance activities** using bug-tracking mechanisms.

4. The team also identifies and analyses problems encountered in program functioning or output.

5. It is also responsible for implementing audits and checks and works on follow-up.

6. The team also conducts Training for process implementors.

END