

SQA Techniques:

SQA Techniques include:

- **Auditing:** Auditing is the inspection of the work products and its related information to determine if a set of standard processes were followed or not.
- **Reviewing:** A meeting in which the software product is examined by both internal and external stakeholders to seek their comments and approval.
- **Code Inspection:** It is the most formal kind of review that does static testing to find bugs and avoid defect seepage into the later stages. It is done by a trained mediator/peer and is based on rules, checklists, entry and exit criteria. The reviewer should not be the author of the code.
- **Design Inspection:** Design inspection is done using a checklist that inspects the below areas of software design:
 - General requirements and design
 - Functional and Interface specifications
 - Conventions
 - Requirement traceability
 - Structures and interfaces
 - Logic
 - Performance
 - Error handling and recovery
 - Testability, extensibility
 - Coupling and cohesion
- **Simulation:** A simulation is a tool that models a real-life situation in order to virtually examine the behaviour of the system under study. In cases when the real system cannot be tested directly, simulators are great sandbox system alternatives.
- **Functional Testing:** It is a QA technique that validates what the system does without considering how it does it. Black Box testing mainly focuses on testing the system specifications or features.
- **Standardization:** Standardization plays a crucial role in quality assurance. This decreases ambiguity and guesswork, thus ensuring quality.
- **Static Analysis:** It is a software analysis that is done by an automated tool without executing the program. Software metrics and reverse engineering are some popular forms of static analysis. In newer teams, static code analysis tools such as SonarCube, VeraCode, etc. are used.
- **Walkthroughs:** A software walkthrough or code walkthrough is a peer review where the developer guides the members of the development team to go through the product, raise queries, suggest alternatives, and make comments regarding possible errors, standard violations, or any other issues.
- **Unit Testing:** This is a White Box Testing technique where complete code coverage is ensured by executing each independent path, branch, and condition at least once.
- **Stress Testing:** This type of testing is done to check how robust a system is by testing it under heavy load i.e. beyond normal conditions.

Reviews and Audits:

Audit: An audit is the examination of the work products and related information to assess whether the standard process was followed or not.

Five types of reviews or audits are presented in the IEEE1028-97 standard:

- Management reviews
- Technical reviews
- Inspections
- Walk-throughs
- Audits

Management reviews

The purpose of a management review is to monitor progress, determine the status of plans and schedules, confirm requirements and their system allocation, or evaluate the effectiveness of management approaches used to achieve fitness for purpose. They support decisions about changes and corrective actions that are required during a software project. Management reviews determine the adequacy of plans, schedules, and requirements and monitor their progress or inconsistencies. These reviews may be performed on products such as audit reports, progress reports, V&V reports, and plans of many types, including risk management, project management, software configuration management, software safety, and risk assessment, among others.

Technical reviews

"The purpose of a technical review is to evaluate a software product to determine its suitability for its intended use. The objective is to identify discrepancies from approved specifications and standards. The results should provide management with evidence confirming (or not) that the product meets the specifications and adheres to standards, and that changes are controlled" (IEEE1028-97).

Specific roles must be established in a technical review: a decision-maker, a review leader, a recorder, and technical staff to support the review activities. A technical review requires that mandatory inputs be in place in order to proceed:

- Statement of objectives
- A specific software product
- The specific project management plan
- The issues list associated with this product
- The technical review procedure

Inspections

The purpose of an inspection is to detect and identify software product anomalies. Two important differentiators of inspections as opposed to reviews are as follows:

- An individual holding a management position over any member of the inspection team shall not participate in the inspection.
- An inspection is to be led by an impartial facilitator who is trained in inspection techniques.

Software inspections always involve the author of an intermediate or final product, while other reviews might not. Inspections also include an inspection leader, a recorder, a reader, and a few (2 to 5) inspectors. The members of an inspection team may possess different expertise, such as domain expertise, design method expertise, or language expertise. Inspections are usually conducted on one relatively small section of the product at a time. Each team member must examine the software product and other review inputs prior to the review meeting, perhaps by applying an analytical technique to a small section of the product, or to the entire product with a focus only on one aspect, for example, interfaces. Any anomaly found is documented and sent to the inspection leader. During the inspection, the inspection leader conducts the session and verifies that everyone has prepared for the inspection. A checklist, with anomalies and questions germane to the issues of interest, is a common tool used in inspections. The resulting list often classifies the anomalies and is reviewed for completeness and accuracy by the team. The inspection exit decision must correspond to one of the following three criteria:

- Accept with no or at most minor reworking
- Accept with rework verification
- Reinspect

Walk-throughs

The purpose of a walk-through is to evaluate a software product. A walk-through may be conducted for the purpose of educating an audience regarding a software product. The major objectives are to:

- Find anomalies
- Improve the software product
- Consider alternative implementations
- Evaluate conformance to standards and specifications

The walk-through is similar to an inspection but is typically conducted less formally. The walk-through is primarily organized by the software engineer to give his teammates the opportunity to review his work, as an assurance technique.

Audits

The purpose of a software audit is to provide an independent evaluation of the conformance of software products and processes to applicable regulations, standards, guidelines, plans, and

procedures. The audit is a formally organized activity, with participants having specific roles, such as lead auditor, another auditor, a recorder, or an initiator, and includes a representative of the audited organization. The audit will identify instances of nonconformance and produce a report requiring the team to take corrective action.