UNIT-8
Software Testing

⊛ Introduction:

Software testing is intended to show that a program does what it is intended to do and to discover program defects before it is put into use. When we test software, we execute a program using artificial data. We check the results of the test run for errors, anomalies or information about the programs non-functional attributes.

Program testing goals:

↳ To demonstrate to the developer and the customer that the software meets its requirements.

↳ To discover situations in which the behaviour of the software is incorrect.

⊛ Verification and Validation testing:

Verification specifies that "Are we building the system right". The software should conform to its specification. Validation specifies that "Are we building the right system". The software should do what the user really requires. Verification and validation is the name given to checking and analyzing processes that ensure that software confirms to its specification and meets the need of customer who are paying for that software.

V & V are independent procedure that is used together for checking that a product service or system meets requirement specifications and that it fulfills its intended purpose.

Differences between Verification and Validation:

Verification	Validation
i) It is the process of evaluating product of development phase to find out whether they meet the requirement and design specification.	i) It is the process of evaluating software at the end to determine whether it meets the customer expections and requirements, or works or not as it was intended.
ii) Activities involved are: review, meetings and inspections.	ii) Activities involved are; black box testing, white box testing.
iii) Verification is carried out by QA team.	iii) Validation is carried out by testing team.
iv) Execution of code does not come under verification.	iv) Execution of code comes under validation.
v) It is basically manual checking of documents like: requirement specification, design specification etc.	v) It is basically checking of developed software by executing source code.
vi) It specifies "Are we building the system right"	vi) It specifies "Are we building the right system"

⊛ Software Inspection:

Software inspection is a control technique for ensuring that the documentation produced during a given phase remains consistent with the documentation of the previous phases and respects reestablished rules and standards. These involve people examining the source representation with the aim of discovering anomalies and defects. The aim of inspection is to locate faults and process should be driven by fault check list.

Inspection Roles:

i) **Author:** An author is the person who created the work product being inspected, responsible for producing documents, fixing defects.

ii) **Moderator:** Leader of the inspection plans the inspection and coordinates it.

iii) **Chief Moderator:** Responsible for inspection process improvement, checklist updating etc.

iv) **Reader:** Reader is the person reading through the documents one item at a time presents the code or documents at inspection meeting.

v) **Inspector:** Person that examines the work product to identify possible defects, find errors etc.

Inspection Process:

i) **Planning:** The inspection is planned by moderator which involves selecting an inspection team, organizing a meeting room and materials.

ii) **Overview:** In this step, the software and documents to be inspected are presented to the inspection team.

iii) **Individual Prepration:** In this step, each inspection team member studies the specification and the program and looks for defects.

iv) **Inspection Meeting:** During this meeting, the readers read through the work product part by part and inspectors point out defects for every part.

v) **Rework:** The author makes changes to work product according to the action plan from the inspection meeting in this step.

vi) **Follow up:** In this step, the changes made by the author are checked to make sure that everything correct. The moderator should decide whether reinspection of the code and document is required. If not the software is then approved by moderator for release.

⊛ Software Testing Process:

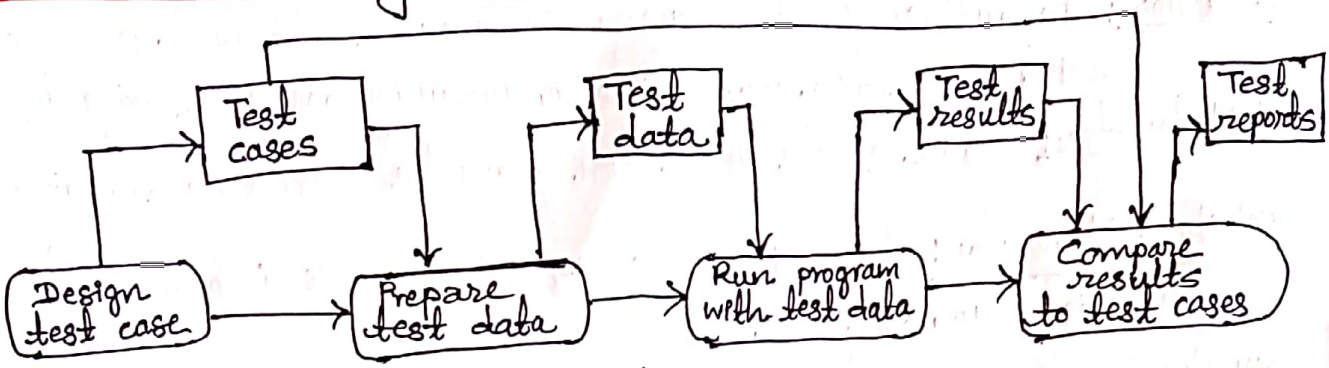

Fig: Software testing process.

Software testing process starts with prepration of test cases. Test cases are specification of the inputs to the test and the expected output from the system together with what is being tested. From the design of test cases we get test case list. After that we prepare the test data. Test data are the inputs which have been designed to test the system. Test data can be generated automatically or manually. From this step we get test data set. Then we run the program with test data for all test cases and generate the result. Finally we compare the test result to test cases and generate the test report.

Types of software testing/Software Testing Levels:

1) **Unit Testing:** It focuses on the smallest unit of software design. In this, we test an individual unit or group of interrelated units. It is often done by the programmer by using sample input and observing its corresponding outputs.

2) **Integration Testing:** The objective is to take unit tested components and build a program structure, then the components are integrated to produce output.

3) **System Testing:** A complete and integrated software is tested to verify that it meets specified requirements.

iv) Acceptance testing: Acceptance testing is done to verify if system meets the customer specified requirements. It has two types:

Alpha testing: It is a testing performed to identify bugs before releasing product to real users. It is typically done by QA people.

Beta testing: In this testing version is released for a limited number of users for testing in a real-time environment.

v) Regression Testing: Every time a new module is added leads to changes in the program. This testing makes sure that the whole component works properly even after adding components to system.

Testing Methods: Mainly there are two testing methods black-box testing and white-box testing.

Black box testing	White box testing
i) The internal workings of an application are not required to be known.	i) Tester has full knowledge of the internal working of the application.
ii) Performed by end users, testers and developers.	ii) Normally done by testers and developers.
iii) This is least time consuming and exhaustive.	iii) This is most exhaustive and time consuming.
iv) Not suited to algorithm testing.	iv) Suited for algorithm testing.
v) It is performed by testing team.	v) It is performed by developers themselves.
vi) Basis of test case is requirement specification.	vi) Basis of test case is detail design of the system.

⊛ Development Testing:

It is a method of applying testing practices consistently throughout the software development life cycle process. The testing ensures the detection of bugs or errors at the right time which further ensures delay of any kind of risk in terms of time and cost. It aims to establish a framework to verify whether the requirements of a given project are met in accordance with the rules of the mission to be accomplished. This testing is performed by the software developers or other engineers during the construction phase of the software development life cycle.

⊛. Test-driven development:

Test-driven development (TDD) is an approach to program development in which we inter-leave testing and code development. Tests are written before code and 'passing' the tests is the critical driver of development. We develop code incrementally, along with a test for that increment. We don't move on to the next increment until the code that we have developed passes its test.

⊛. Release testing:

Release testing refers to coding practices and test strategies that give teams confidence that a software release candidate is ready for users. Release testing aims to find and eliminate errors and bugs from a software releases so that it can be released to users. The primary goal of the release testing process is to convince the customer of the system that it is good enough for use.

⑩. User testing:

User testing is a stage in the testing process in which users or customers provide input and advice on system testing. User testing is essential even when comprehensive system and release testing have been carried out. The reason for this is that influences from user's working environment have a major effect on the reliability, performance, usability and robustness of a system. These cannot be replicated in a testing environment.

Q11. What is software quality assurance? Explain with example. [Model set]

Ans: Software Quality Assurance (SQA) is simply a way to assure quality in the software. It is the set of activities which ensure processes, procedures as well as standards for the project and implemented correctly. It is a process which works parallel to development of software. It focuses on improving the process of development of software so that problems can be prevented before they become a major issue.

Example: Library Management System:

To build this software system, first of all we have to create an software quality assurance (SQA) management plan that includes how SQA will be carried out in our project. Then, we setup the checkpoints or making schedule by dividing work so that particular work can be finished on time. Then we apply software engineering techniques (specification development testing) and then executing it for formal review to assure its quality. In this way we can assure SQA in our project.