

What is Software Testing?

- ▶ Several definitions:

“Testing is the process of establishing confidence that a program or system does what it is supposed to.” by Hetzel 1973

“Testing is the process of executing a program or system with the intent of finding errors.” by Myers 1979

“Testing is any activity aimed at evaluating an attribute or capability of a program or system and determining that it meets its required results” by Hetzel 1983

What is Software Testing?

- ▶ One of very important software development phases
A software process based on well-defined software quality control and testing standards, testing methods, strategy, test criteria, and tools.
Engineers perform all types of software testing activities to perform a software test process.
The last quality checking point for software on its production line
- ▶ Software testing is about testing a feature by applying test data to get a result and then comparing the actual result with expected result.
Expected Result = Its from client side
Actual Result = Its obtain after a test execution.

It is not only finding the defect in the s/w; it is the completely dedicated discipline of evaluating the **quality** of s/w

Why Testing is Important

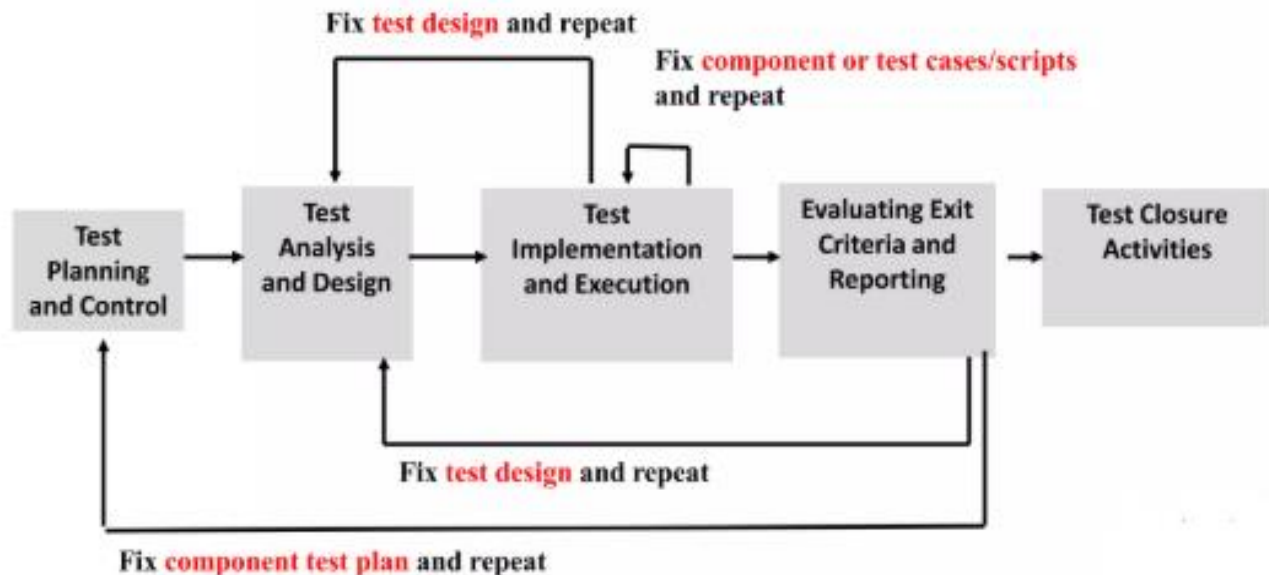
- ▶ Software testing is really required to point out the defects and errors that were made during the development phases.
- ▶ It's essential since it makes sure of the Customer's reliability and their satisfaction in the application.
- ▶ It is very important to ensure the Quality of the product. Quality product delivered to the customers helps in gaining their confidence.
- ▶ Testing is necessary in order to provide the facilities to the customers like the delivery of high quality product or software application which requires lower maintenance cost and hence results into more accurate, consistent and reliable results.
- ▶ Testing is required for an effective performance of software application or product.
- ▶ It's important to ensure that the application should not result into any failures because it can be very expensive in the future or in the later stages of the development.
- ▶ It's required to stay in the business.

Seven Principle of Software Testing

- 1) **Testing shows presence of defects:** Testing can show the defects are present, but cannot prove that there are no defects. Even after testing the application or product thoroughly we cannot say that the product is 100% defect free.
- 2) **Exhaustive testing is impossible:** Testing everything including all combinations of inputs and preconditions is not possible. So, instead of doing the exhaustive testing we can use risks and priorities to focus testing efforts
- 3) **Early testing:** In the software development life cycle testing activities should start as early as possible and should be focused on defined objectives.
- 4) **Defect clustering:** A small number of modules contains most of the defects discovered during pre-release testing or shows the most operational failures.
- 5) **Pesticide paradox:** If the same kinds of tests are repeated again and again, eventually the same set of test cases will no longer be able to find any new bugs. To overcome this "Pesticide Paradox".
- 6) **Testing is context depending:** Testing is basically context dependent. Different kinds of sites are tested differently. For example, safety - critical software is tested differently from an e-commerce site.
- 7) **Absence - of - errors fallacy:** If the system built is unusable and does not fulfil the user's needs and expectations then finding and fixing defects does not

Fundamental Test Process

5 Phases of the Fundamental Test Process



QA, QC AND TESTING

Quality Assurance	Quality Control	Testing
QA includes activities that ensure the implementation of processes, procedures and standards in context to verification of developed software and intended requirements.	It includes activities that ensure the verification of a developed software with respect to documented (or not in some cases) requirements.	It includes activities that ensure the identification of bugs/error/defects in a software.
Focuses on processes and procedures rather than conducting actual testing on the system.	Focuses on actual testing by executing the software with an aim to identify bug/defect through implementation of procedures and process.	Focuses on actual testing.
Process-oriented activities. Preventive activities.	Product-oriented activities. It is a corrective process.	Product-oriented activities. It is a preventive process.
It is a subset of Software Test Life Cycle (STLC).	QC can be considered as the subset of Quality Assurance.	Testing is the sub Control.

Types of Testing

- ▶ This section describes the different types of testing that may be used to test a software during SDLC.

- ▶ **Manual Testing:-** Manual testing includes testing a software manually, i.e., without using any automated tool or any script. In this type, the tester takes over the role of an end-user and tests the software to identify any unexpected behaviour or bug. There are different stages for manual testing such as unit testing, integration testing, system testing, and user acceptance testing.

Testers use test plans, test cases, or test scenarios to test a software to ensure the completeness of testing. Manual testing also includes exploratory testing, as testers explore the software to identify errors in it.

- ▶ **Automation Testing:-** Automation testing, which is also known as Test Automation, is when the tester writes scripts and uses another software to test the product. This process involves automation of a manual process. Automation Testing is used to re-run the test scenarios that were performed manually, quickly, and repeatedly.

Testing Method

- ▶ There are three type of testing:-

- 1) **White Box Testing:-**

- Unit Testing
 - Structural Testing

- 2) **Black Box Testing:-**

- Functional Testing
 - Integration testing
 - System Testing
 - Usability Testing
 - Compatibility Testing
 - Configuration Testing
 - Concurrency Testing

- 3) **Non Functional Testing:-**

- Security Testing
 - Automation Testing
 - Performance Testing
 - Load Testing
 - Stress Testing
 - Volume Testing
 - Recovery Testing

Testing and Quality Assurance

The software system needs to be checked for its intended behavior and direction of progress at each development stage to avoid duplication of efforts, time and cost overruns, and to assure completion of the system within stipulated time. The software system needs to be checked for its intended behavior and direction of progress at each development stage to avoid duplication of efforts, time and cost overruns, and to assure completion of the system within stipulated time.

System testing and quality assurance come to aid for checking the system. It includes –

- Product level quality (Testing)
- Process level quality.

Let us go through them briefly –

Testing

Testing is the process or activity that checks the functionality and correctness of software according to specified user requirements in order to improve the quality and reliability of system. It is an expensive, time consuming, and critical approach in system development which requires proper planning of overall testing process.

A successful test is one that finds the errors. It executes the program with explicit intention of finding error, i.e., making the program fail. It is a process of evaluating system with an intention of creating a strong system and mainly focuses on the weak areas of the system or software.

Characteristics of System Testing

System testing begins at the module level and proceeds towards the integration of the entire software system. Different testing techniques are used at different times while testing the system. It is conducted by the developer for small projects and by independent testing groups for large projects.

Stages of System Testing

The following stages are involved in testing –

Test Strategy

It is a statement that provides information about the various levels, methods, tools, and techniques used for testing the system. It should satisfy all the needs of an organization.

Test Plan

It provides a plan for testing the system and verifies that the system under testing fulfils all the design and functional specifications. The test plan provides the following information –

- Objectives of each test phase
- Approaches and tools used for testing
- Responsibilities and time required for each testing activity
- Availability of tools, facilities, and test libraries
- Procedures and standards required for planning and conducting the tests
- Factors responsible for successful completion of testing process

Test Case Design

- A number of test cases are identified for each module of the system to be tested.
- Each test case will specify how the implementation of a particular requirement or design decision is to be tested and the criteria for the success of the test.
- The test cases along with the test plan are documented as a part of a system specification document or in a separate document called **test specification** or **test description**.

Test Procedures

It consists of the steps that should be followed to execute each of the test cases. These procedures are specified in a separate document called test procedure specification. This document also specifies any special requirements and formats for reporting the result of testing.

Test Result Documentation

Test result file contains brief information about the total number of test cases executed, the number of errors, and nature of errors. These results are then assessed against criteria in the test specification to determine the overall outcome of the test.

Types of Testing

Testing can be of various types and different types of tests are conducted depending on the kind of bugs one seeks to discover –

Unit Testing

Also known as Program Testing, it is a type of testing where the analyst tests or focuses on each program or module independently. It is carried out with the intention of executing each statement of the module at least once.

- In unit testing, accuracy of program cannot be assured and it is difficult to conduct testing of various input combination in detail.
- It identifies maximum errors in a program as compared to other testing techniques.

Integration Testing

In Integration Testing, the analyst tests multiple module working together. It is used to find discrepancies between the system and its original objective, current specifications, and systems documentation.

- Here the analysts are try to find areas where modules have been designed with different specifications for data length, type, and data element name.
- It verifies that file sizes are adequate and that indices have been built properly.

Functional Testing

Function testing determines whether the system is functioning correctly according to its specifications and relevant standards documentation. Functional testing typically starts with the implementation of the system, which is very critical for the success of the system.

Functional testing is divided into two categories –

- **Positive Functional Testing** – It involves testing the system with valid inputs to verify that the outputs produced are correct.
- **Negative Functional Testing** – It involves testing the software with invalid inputs and undesired operating conditions.

Explore our **latest online courses** and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

Rules for System Testing

To carry out system testing successfully, you need to follow the given rules –

- Testing should be based on the requirements of user.

- Before writing testing scripts, understand the business logic should be understood thoroughly.
- Test plan should be done as soon as possible.
- Testing should be done by the third party.
- It should be performed on static software.
- Testing should be done for valid and invalid input conditions.
- Testing should be reviewed and examined to reduce the costs.
- Both static and dynamic testing should be conducted on the software.
- Documentation of test cases and test results should be done.

Quality Assurance

It is the review of system or software products and its documentation for assurance that system meets the requirements and specifications.

- Purpose of QA is to provide confidence to the customers by constant delivery of product according to specification.
- Software quality Assurance (SQA) is a techniques that includes procedures and tools applied by the software professionals to ensure that software meet the specified standard for its intended use and performance.
- The main aim of SQA is to provide proper and accurate visibility of software project and its developed product to the administration.
- It reviews and audits the software product and its activities throughout the life cycle of system development.

Objectives of Quality Assurance

The objectives of conducting quality assurance are as follows –

- To monitor the software development process and the final software developed.
- To ensure whether the software project is implementing the standards and procedures set by the management.
- To notify groups and individuals about the SQA activities and results of these activities.
- To ensure that the issues, which are not solved within the software are addressed by the upper management.
- To identify deficiencies in the product, process, or the standards, and fix them.

Levels of Quality Assurance

There are several levels of QA and testing that need to be performed in order to certify a software product.

Level 1 – Code Walk-through

At this level, offline software is examined or checked for any violations of the official coding rules. In general, the emphasis is placed on examination of the documentation and level of in-code comments.

Level 2 – Compilation and Linking

At this level, it is checked that the software can compile and link all official platforms and operating systems.

Level 3 – Routine Running

At this level, it is checked that the software can run properly under a variety of conditions such as certain number of events and small and large event sizes etc.

Level 4 – Performance test

At this final level, it is checked that the performance of the software satisfies the previously specified performance level.

Software Testing Tutorial



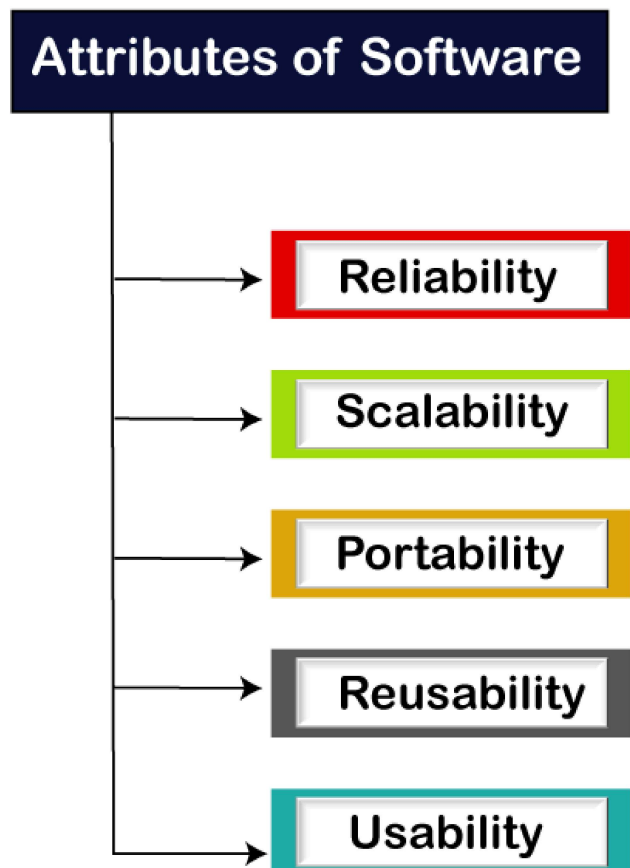
Software testing tutorial provides basic and advanced concepts of software testing. Our software testing tutorial is designed for beginners and professionals.

Software testing is widely used technology because it is compulsory to test each and every software before deployment.

Our Software testing tutorial includes all topics of Software testing such as Methods such as Black Box Testing, White Box Testing, Visual Box Testing and Gray Box Testing. Levels such as Unit Testing, Integration Testing, Regression Testing, Functional Testing. System Testing, Acceptance Testing, Alpha Testing, Beta Testing, Non-Functional testing, Security Testing, Portability Testing.

What is Software Testing

Software testing is a process of identifying the correctness of software by considering its all attributes (Reliability, Scalability, Portability, Re-usability, Usability) and evaluating the execution of software components to find the software bugs or errors or defects.



Software testing provides an independent view and objective of the software and gives surety of fitness of the software. It involves testing of all components under the required services to confirm that whether it is satisfying the specified requirements or not. The process is also providing the client with information about the quality of the software.

Testing is mandatory because it will be a dangerous situation if the software fails any of time due to lack of testing. So, without testing software cannot be deployed to the end user.

What is Testing

Testing is a group of techniques to determine the correctness of the application under the predefined script but, testing cannot find all the defect of application. The main intent of testing is to detect failures of the application so that failures can be discovered and corrected. It does not demonstrate that a product functions properly under all conditions but only that it is not working in some specific conditions.

Testing furnishes comparison that compares the behavior and state of software against mechanisms because the problem can be recognized by the mechanism. The mechanism may include past versions of the same specified product, comparable products, and interfaces of expected purpose, relevant standards, or other criteria but not limited up to these.

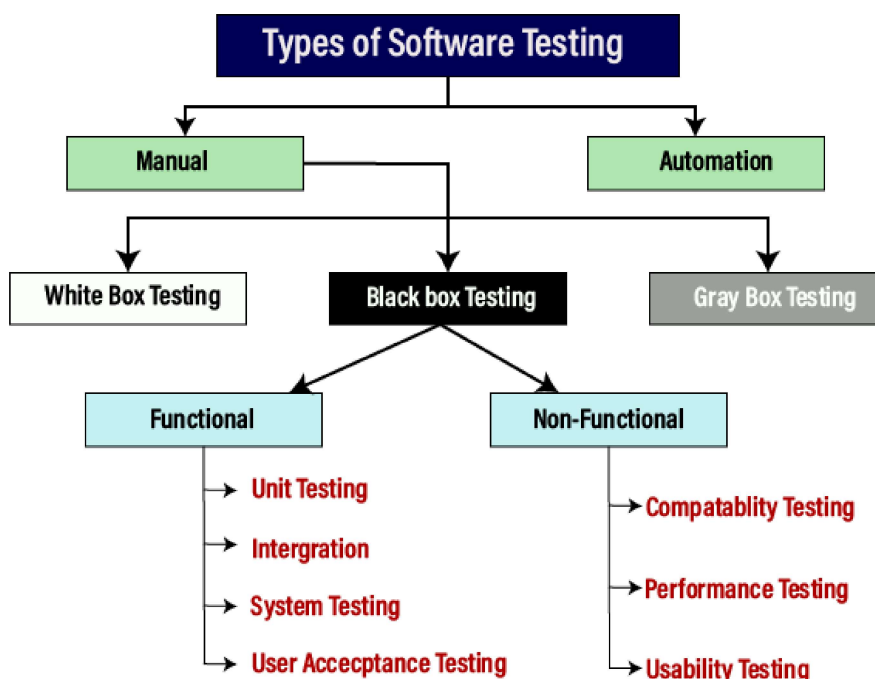
Testing includes an examination of code and also the execution of code in various environments, conditions as well as all the examining aspects of the code. In the current scenario of software development, a testing team may be separate from the development team so that Information derived from testing can be used to correct the process of software development.

The success of software depends upon acceptance of its targeted audience, easy graphical user interface, strong functionality load test, etc. For example, the audience of banking is totally different from the audience of a video game. Therefore, when an organization develops a software product, it can assess whether the software product will be beneficial to its purchasers and other audience.

Type of Software testing

We have various types of testing available in the market, which are used to test the application or the software.

With the help of below image, we can easily understand the type of software testing:



Manual testing

The process of checking the functionality of an application as per the customer needs without taking any help of automation tools is known as manual testing. While performing the manual testing on any application, we do not need any specific knowledge of any testing tool, rather than have a proper understanding of the product so we can easily prepare the test document.

Manual testing can be further divided into three types of testing, which are as follows:

- **White box testing**
- **Black box testing**
- **Gray box testing**

For more information about manual testing, refers to the below link:

<https://www.javatpoint.com/manual-testing>

Automation testing

Automation testing is a process of converting any manual test cases into the test scripts with the help of automation tools, or any programming language is known as automation testing. With the help of automation testing, we can enhance the speed of our test execution because here, we do not require any human efforts. We need to write a test script and execute those scripts.

For more information about manual testing, refers to the below link:

<https://www.javatpoint.com/automation-testing>

Prerequisite

Before learning software testing, you should have basic knowledge of basic computer functionality, basic mathematics, computer language, and logical operators.

Audience

Our software testing tutorial is designed for beginners and professionals.

Problems

We assure that you will not find any problem in this Software Testing Tutorial. But if there is any mistake, please post the problem in contact form.

[Software Testing Principles](#)