QA & Testing

06, Oct,2025
Task-1 Report

MUHAMMAD UMAR HUSSAIN

Types of Testing

Software testing is a process used to ensure that a software or application works correctly, meets its requirements, behaves as expected, and is free of bugs improving the overall quality of the software.

Types of Software Testing:

1. Manual Testing.
2. Automation Testing.

Manual Testing.

Manual Testing is a process in which a human tester ensures that everything in the software works as it is supposed to. All the steps of the Software Testing Life Cycle (STLC) are performed manually by a person without using any automation tools. Although it is time-consuming, it allows testers to perform random and exploratory testing**.** However, it is not efficient for regression testing, where the same tests need to be repeated many times.

Automation Testing.

Automation Testing is a process in which testing is done using special tools or scripts like (selenium, j meter, win runner, load runner) instead of a human performing all steps manually. It helps to execute test cases faster and more accurately, especially when the same tests need to be repeated many times. Although the setup takes time and requires programming knowledge, it saves effort in the long run and is very useful for regression testing and large projects**.**

Levels of Testing

There are 4 levels of Testing.

1. Unit Testing.
2. Integration Testing.
3. System Testing.
4. Acceptance Testing.

Unit Testing.

In Unit Testing, we test a single module, statement, or function by checking it with different inputs and comparing the output with the expected result. It takes place during the development phase and before integration testing**.** Usually, the developers who write the code perform unit testing. **For example**, checking the return value of a function to ensure it gives the correct result.

Integration Testing.

In Integration Testing, we test how two or more modules work together. The goal is to check the interaction between connected modules to ensure they communicate properly. It takes place in Testing phase after unit testing and before system testing. Usually, testers perform integration testing. **For example**, testing if the Login module correctly connects with the Dashboard module after a user successfully logs in.

There are 4 types of integration testing.

1. Big Bang Testing.
2. Top Down Testing.
3. Bottom Up Testing.
4. Mixed Testing.

System Testing.

In System Testing, we test the entire software system as a whole to make sure all modules work together correctly and meet the given requirements. The goal is to check the overall functionality, performance, and behavior of the complete application. It takes place in the testing phase, after integration testing and before acceptance testing. Usually, testers perform system testing.
For example, testing a shopping website by performing all actions like logging in, adding items to the cart, making a payment, and confirming the order to ensure everything works properly together.

There are two types of System Testing.

1. **System Testing (Based on who is testing).**
- Alpha Testing.
- Beta Testing.
- Gamma Testing.
2. **Performance / Non-Functional Testing.**
- Volume Testing.
- Load Testing.
- Stress Testing.
- Security Testing.
- Compatibility Testing.
- Configuration Testing.
- Recovery Testing.
- Installation Testing.

Acceptance Testing.

In Acceptance Testing, we test the entire software systems working to ensure it meets the client's requirements and is ready for release. The goal is to confirm that the software behaves as expected in real-world scenarios and satisfies business needs. It takes place in the final phase of testing, after system testing. Usually, clients, end users, or QA testers perform acceptance testing.
For example, a client tests a food delivery application by placing an actual order to make sure all features from browsing restaurants to making payments work according to their expectations before the app goes live.

Testing Techniques

There are mainly two types of testing techniques in software Testing

1. Black Box Testing.
2. White Box Testing.

Black Box Testing.

In Black Box Testing, we test the functionalities of the system without knowing its internal structure or code. The tester provides the required inputs and checks the output against the expected results. It does not require programming knowledge, as the focus is only on what the software does, not how it does it. Common design techniques for Black Box Testing include Decision Table, All-Pairs Testing, Equivalence Partitioning, Boundary Value Analysis, and Cause-Effect Graphing.
It can be performed at all levels of testing, from unit to acceptance testing.

White Box Testing.

In White Box Testing, we test the internal structure, logic and code of the software. The tester has full knowledge of the program's internal workings and designs test cases to check how the code executes. It requires programming knowledge because the focus is on how the software works rather than what it does. Common techniques for White Box Testing include Statement Coverage, Branch Coverage, Path Coverage, and Condition Coverage.

It is mostly performed by developers during the unit testing phase to ensure that each part of the code works correctly.