

Functional Testing and Non Functional Testing

Functional Testing: is a type of testing which verifies that each functions of the software application operate in regard with the requirement specification.

Non Function Testing: It is used to check non functional aspect [performance, usability, reliability]. A good example of non functional parameters test is to check how many people can simultaneously login into software.

Functional Testing	Non Functional Testing
It is based on customer's requirement	It is based on customer's expectation
It describes what the product does	It describes how the product work
Functional testing is easy to execute by manual testing	It is hard to perform non functional testing manually
Ex: unit testing, integration test, system testing, sanity testing, smoke testing, interface testing, regression testing etc.	Ex: performance testing, load testing, volume testing, security testing, usability testing etc.

Types of software testing:

- **Alpha testing:** Alpha testing is conducted at the developer's site. The objective of this alpha testing is to identify all possible issues. It is carried out at the end of software development phase but, before the beta testing.
- **Acceptance testing:** This is also called user acceptance testing. Acceptance testing, testing techniques to determine whether or not software system has met the requirement specification.
- **Ad-hoc testing:** Testing is done on ad-hoc basis, with no reference to test case, without plan or documentary.

- **Accessible testing:** The aim of accessible testing is to determine whether the software or application is accessible to disabled people or not.
- **Beta testing:** Beta testing is carried out by customer. It is performed in real environment.
- **Boundary value testing:** Boundary value testing is performed for checking if defects exist at boundary values. There is upper and lower boundary for each range and testing is performed on this boundary value.
- **Branch testing:** It is type of white box testing and is carried out during unit testing.
- **Comparison testing:** Comparison of product's strength and weakness with its previous version or similar product termed as comparison testing.
- **Compatibility testing:** Compatibility testing ensures that software can run on different configuration, different database, different browser, and different version.
- **Component testing:** Mostly performed by developers after completion of unit testing. Component testing involves testing of multiple functionalities as a single code and its objective is to identify if any defects exists after connecting those multiple functions.
- **End to End testing:** End to end testing, testing of a complete application environment in a situation that mimics real world use.
- **Example testing:** It means real time testing, it includes real scenarios.
- **Backend testing:** Whenever an input or data is entered on front end application, it stores in the database and testing of such database is known as database testing or backend testing.
- **Browser compatibility testing:** It is performed for web application and it ensures that the software can run with the combination of different browser and operating system.
- **Backward compatibility testing:** It is type of testing which validates whether the newly developed software or updated software works with the older version of environment or not.
- **Black box testing:** It is a software testing method in which functionality of software application are testing without having the knowledge of internal code.

- **Equivalence testing:** It is a type of black box testing, during this equivalence testing a set of group is selected and a few values or numbers are picked for testing.
- **Exploratory testing:** Exploratory testing is an approach to software testing that is concisely described as simultaneous learning, test design, and test execution.
- **GUI testing:** The objective of graphical user interface testing is to validate the GUI as per the requirement.
- **Gorilla testing:** is a testing technique performed by tester and a developer as well. In gorilla testing one module or functionality of module is tested heavily.
- **Happy path testing:** The objective of happy path testing is to test an application successfully on positive flow.
- **Install/Uninstall testing:** is done on full/partial/upgrade on different operating system under different hardware and software.
- **Incremental integration testing:** is bottom up approach, continuous testing of application when new functionalities is added.
- **Integration testing:** Testing of all integrated module to verify the combined functionalities after integration.
- **Load testing:** This is to check how much load or maximum workload a system can handle without performance degradation.
- **Monkey testing:** is carried out by the tester assuming that if monkey using the application, how random input values will be created by monkey without having the knowledge or understanding the system
- **Mutation testing:** is a type of white box testing in which source code of one of the program is changed, and then verified that test cases are able to find the error or not.
- **Negative testing:** is performed using incorrect data, invalid data and inputs.
- **Performance testing:** is non functional testing techniques performed to determine the system parameters in terms of responsiveness, and stability under various workload.

- **Recovery testing:** It is type of testing which validated how well the application or system recovers from the crash or disaster.
- **Regression testing:** is done to confirm that recent program or code change has no adverse effect on existing application.
- **Risk based testing (RBT):** In these functionalities are tested based on their priority.
- **Sanity testing:** is a kind of software testing performed after receiving a software build with minor changes in the code to ascertain that the bugs have been fixed and no further bug have been reported due to this change in code.
- **Smoke testing:** is a type of testing that comprises of non exhaustive set of tests that aims at ensuring that most important functions work.
- **Security testing:** is done to check how the software or application is secure from internal and external threats.
- **Static testing:** is a testing which is executed without code. It involves review, walkthrough, and inspection. It checks code syntax, naming convention.
- **Dynamic testing:** This checks an application by executing the code.
- **Stress testing:** This testing is done when a system is stressed beyond its specification in order to check how and when it fails.
- **Volume testing:** It checks the system behavior and response time of application when system comes across such a high volume of data.
- **System testing:** Entire system is tested as per the requirement after having an integration testing.
- **Unit testing:** Testing of an individual module or software component is known as unit testing.
- **Usability testing:** Here user friendliness is checked that is ease to access.
- **Vulnerability testing:** Vulnerability testing involves identify weakness in software, hardware and networking. It can be any vulnerability that is virus, bug and worms.

- **White box testing:** is based on the knowledge about the internal logic of the application, here we check the code of the software.
- **Grey box testing:** is software testing method which is a combination of black box and white box testing. Internal structure is partially known in the grey box testing.