**What is Functional Testing?**

Functional testing is a type of testing which verifies that each **function** of the software application operates in conformance with the requirement specification. This testing mainly involves black box testing, and it is not concerned about the source code of the application.

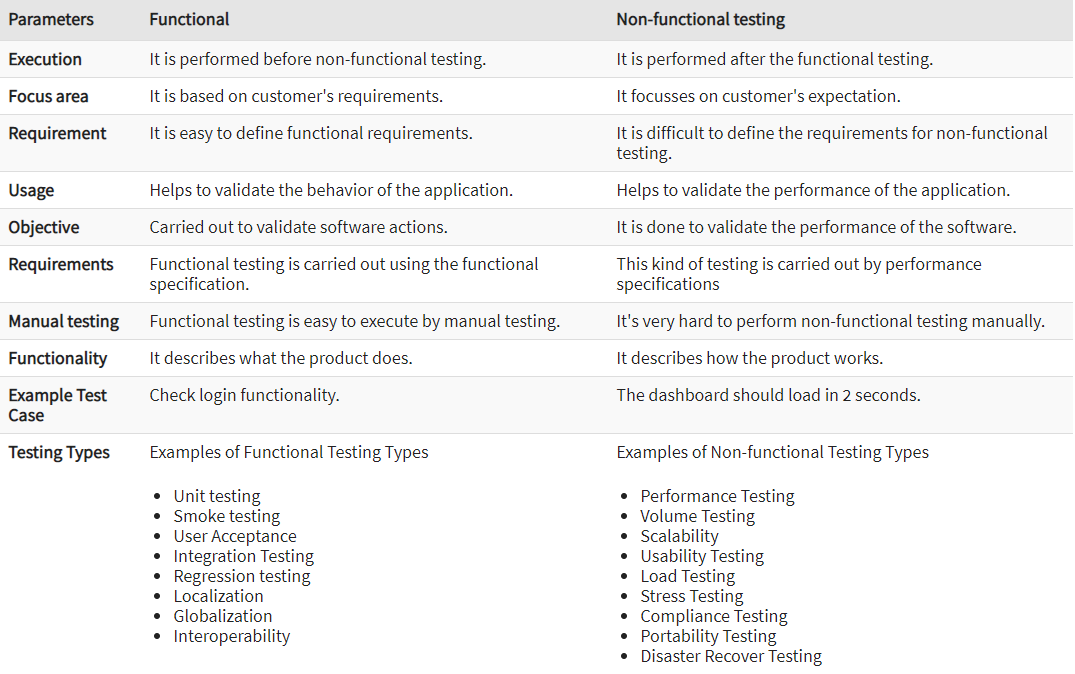
Every functionality of the system is tested by providing appropriate input, verifying the output and comparing the actual results with the expected results. This testing involves checking of User Interface, APIs, Database, security, client/ server applications and functionality of the Application Under Test. The testing can be done either manually or using automation

**What is Non-Functional Testing?**

Non-functional testing is a type of testing to check non-functional aspects (performance, usability, reliability, etc.) of a software application. It is explicitly designed to test the readiness of a system as per non-functional parameters which are never addressed by functional testing.

A good example of non-functional test would be to check how many people can simultaneously login into a software.

Non-functional testing is equally important as functional testing and affects client satisfaction.



**Types of Functional Testing:**

1. **Unit testing:**

Testing of an individual software component or module is termed as [Unit Testing](https://www.softwaretestinghelp.com/unit-testing/). It is typically done by the programmer and not by testers, as it requires a detailed knowledge of the internal program design and code. It may also require developing test driver modules or test harnesses.

1. **Smoke testing:**

Whenever a new build is provided by the development team then the software testing team validates the build and ensures that no major issue exists.

The testing team ensures that the build is stable, and a detailed level of testing is carried out further. [Smoke Testing](https://www.softwaretestinghelp.com/smoke-testing-and-sanity-testing-difference/) checks that no showstopper defect exists in the build which will prevent the testing team to test the application in detail.

If testers find that the major critical functionality is broken down at the initial stage itself then testing team can reject the build and inform accordingly to the development team. Smoke Testing is carried out to a detailed level of any functional or regression testing.

1. **System testing:**

Under [System Testing technique](https://www.softwaretestinghelp.com/system-testing/), the entire system is tested as per the requirements. It is a Black-box type testing that is based on overall requirement specifications and covers all the combined parts of a system.

1. **User Acceptance:**

An [acceptance test](https://www.softwaretestinghelp.com/what-is-acceptance-testing/) is performed by the client and verifies whether the end to end the flow of the system is as per the business requirements or not and if it is as per the needs of the end user. Client accepts the software only when all the features and functionalities work as expected.

It is the last phase of the testing, after which the software goes into production. This is also called User Acceptance Testing (UAT).

1. **Integration Testing:**

Testing of all integrated modules to verify the combined functionality after integration is [termed as Integration Testing](https://www.softwaretestinghelp.com/what-is-integration-testing/). Modules are typically code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant to client/server and distributed systems.

1. **Regression testing:**

Testing an application as a whole for the modification in any module or functionality is termed as Regression Testing. It is difficult to cover all the system in [Regression Testing](https://www.softwaretestinghelp.com/regression-testing-tools-and-methods/), so typically [automation testing tools](https://www.softwaretestinghelp.com/automation-testing-tutorial-1/) are used for these types of testing.

1. **Localization:**

Localization testing is the software testing process for checking the localized version of a product for that culture or locale settings.  The areas affected by localization testing are UI and content.

1. **Globalization:**

Globalization testing is to ensure that application can function in any culture or locale (language, territory and code page) It is also called as Internationalization Testing.

1. **Interoperability:**

Interoperability testing is defined as a software testing type, that checks whether software can inter-operate with other software component, software's or systems.

In other words, interoperability testing means to prove that end-to-end functionality between two communicating systems is as required by the standard on which those systems are based.

For example, interoperability testing is done between smartphone and tablet to check data transfer via Bluetooth.

**Types of Non-Functional Testing:**

1. **Performance Testing:**

This term is often used interchangeably with ‘stress' and ‘load' testing. [Performance Testing](https://www.softwaretestinghelp.com/introduction-to-performance-testing-loadrunner-training-tutorial-part-1/) is done to check whether the system meets the performance requirements. Different performance and load tools are used to do this testing.

1. **Compatibility Testing:**

It is a testing type in which it validates how software behaves and runs in a different environment, web servers, hardware, and network environment. [Compatibility testing](https://www.softwaretestinghelp.com/software-compatibility-testing/) ensures that software can run on a different configuration, different database, different browsers, and their versions. Compatibility testing is performed by the testing team.

1. **Volume Testing:**

[Volume testing](https://www.softwaretestinghelp.com/what-is-volume-testing/) is a type of non-functional testing performed by the performance testing team.

The software or application undergoes a huge amount of data and Volume Testing checks the system behavior and response time of the application when the system came across such a high volume of data. This high volume of data may impact the system’s performance and speed of the processing time.

1. **Scalability:**

Scalability Testing is defined as the ability of a network, system or a process to continue to function well when changes are done in the size or volume of the system to meet a growing need. It is a type of non-functional testing.

Scalability testing ensures that an application can handle the projected increase in user traffic, data volume, transaction counts frequency, etc. It tests the system, processes, and databases ability to meet a growing need.

It is also referred to as [performance testing](https://www.guru99.com/performance-testing.html), as such, it is focused on the behavior of the application when deployed to a larger system or tested under excess load.

1. **Usability Testing:**

Under [Usability Testing](https://www.softwaretestinghelp.com/usability-testing-guide/), User-friendliness check is done. Application flow is tested to know if a new user can understand the application easily or not, Proper help documented if a user gets stuck at any point. Basically, system navigation is checked in this testing.

1. **Load Testing:**

It is a type of non-functional testing and the objective of Load testing is to check how much of load or maximum workload a system can handle without any performance degradation.

[Load testing helps](https://www.softwaretestinghelp.com/introduction-to-performance-testing-loadrunner-training-tutorial-part-1/) to find the maximum capacity of the system under specific load and any issues that cause the software performance degradation. Load testing is performed using tools like[JMeter](https://www.softwaretestinghelp.com/jmeter-tutorials/), LoadRunner, Web Load, Silk performer etc.

1. **Stress Testing:**

This testing is done when a system is stressed beyond its specifications in order to check how and when it fails. This is performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to the system or database load.

1. **Compliance Testing:**

Compliance Testing is defined as a software testing type that determines whether the system complies with requirements of specifications and conditions, regulations and standards, etc. along with its documentation.

1. **Portability Testing:**

Portability testing is a process of testing with ease with which the software or product can be moved from one environment to another. It is measured in terms of maximum amount of effort required to transfer from one system to another system.

The portability testing is performed regularly throughout the software development life cycle in an iterative and incremental manner.

1. **Disaster Recover Testing**

It is a type of testing which validates that how well the application or system recovers from crashes or disasters.

Recovery testing determines if the system is able to continue the operation after a disaster. Assume that application is receiving data through the network cable and suddenly that network cable has been unplugged.

Sometime later, plug the network cable; then the system should start receiving data from where it lost the connection due to network cable unplugged.