**What is Functional Testing?**

**FUNCTIONAL TESTING** is a type of software testing that validates the software system against the functional requirements/specifications. The purpose of Functional tests is to test each function of the software application, by providing appropriate input, verifying the output against the Functional requirements.

Functional testing mainly involves black box testing and it is not concerned about the source code of the application. This testing checks User Interface, APIs, Database, Security, Client/Server communication and other functionality of the Application Under Test. The testing can be done either manually or using automation.

**What do you test in Functional Testing?**

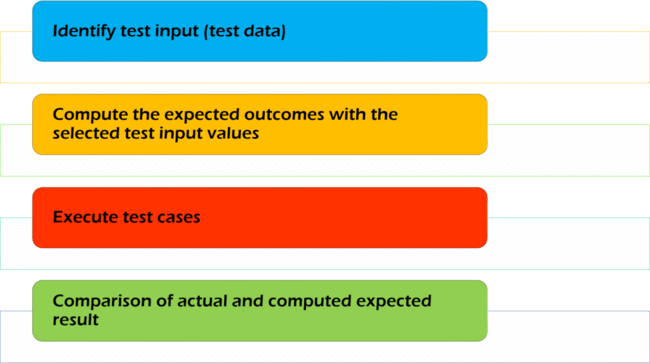
The prime objective of Functional testing is checking the functionalities of the software system. It mainly concentrates on –

* **Mainline functions**:  Testing the main functions of an application
* **Basic Usability**: It involves basic usability testing of the system. It checks whether a user can freely navigate through the screens without any difficulties.
* **Accessibility**:  Checks the accessibility of the system for the user
* **Error Conditions**: Usage of testing techniques to check for error conditions.  It checks whether suitable error messages are displayed.

## How to do Functional Testing

Following is a step by step process on **How to do Functional Testing** :

* Understand the Functional Requirements
* Identify test input or test data based on requirements
* Compute the expected outcomes with selected test input values
* Execute test cases
* Compare actual and computed expected results



## What is Non-Functional Testing?

**NON-FUNCTIONAL TESTING** is defined as a type of Software testing to check non-functional aspects (performance, usability, reliability, etc) of a software application. It is designed to test the readiness of a system as per nonfunctional parameters which are never addressed by functional testing.

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

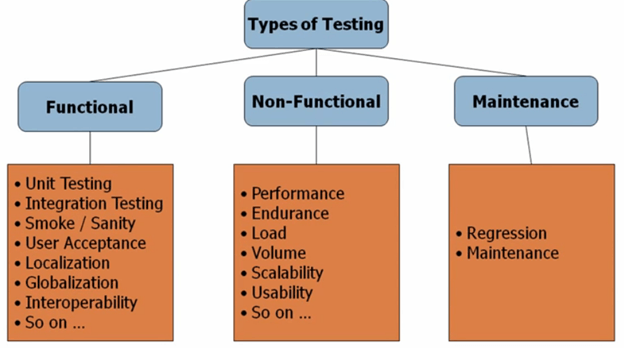
**Objectives of Non-functional testing**

* Non-functional testing should increase usability, efficiency, maintainability, and portability of the product.
* Helps to reduce production risk and cost associated with non-functional aspects of the product.
* Optimize the way product is installed, setup, executes, managed and monitored.
* Collect and produce measurements, and metrics for internal research and development.
* Improve and enhance knowledge of the product behavior and technologies in use.

**Type of Software Testing**

In general, there are three testing types

* **Functional**
* **Non – Functional**
* **Maintenance**



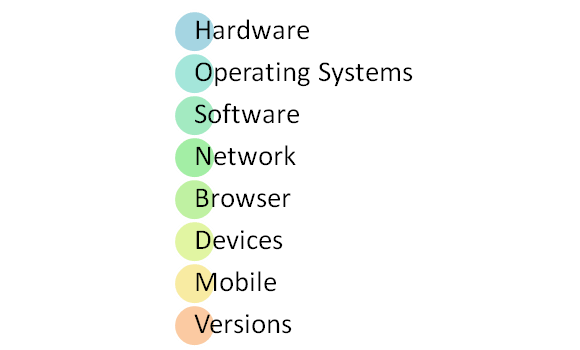
### What is Compatibility?

Compatibility is nothing but the capability of existing or living together. In normal life, Oil is not compatible with water, but milk can be easily combined with water.

### What is Compatibility Testing?

Compatibility Testing is a type of Software testing to check whether your software is capable of running on different hardware, operating systems, applications, network environments or[Mobile](https://www.guru99.com/mobile-testing.html)devices.

**Types of Compatibility Tests**



Let’s look into compatibility testing types

* **Hardware**: It checks software to be compatible with different hardware configurations.
* **Operating Systems**: It checks your software to be compatible with different Operating Systems like Windows, Unix, Mac OS etc.
* **Software**: It checks your developed software to be compatible with other software. For example, MS Word application should be compatible with other software like MS Outlook, MS Excel,[VBA](https://www.guru99.com/vba-tutorial.html)etc.
* **Network:** Evaluation of performance of a system in a network with varying parameters such as Bandwidth, Operating speed, Capacity. It also checks application in different networks with all parameters mentioned earlier.
* **Browser**: It checks the compatibility of your website with different browsers like Firefox, Google Chrome, Internet Explorer etc.
* **Devices**: It checks compatibility of your software with different devices like USB port Devices, Printers and Scanners, Other media devices and Blue tooth.
* **Mobile**: Checking your software is compatible with mobile platforms like Android, iOS etc.
* **Versions of the software:**It is verifying your software application to be compatible with different versions of the software. For instance checking your Microsoft Word to be compatible with Windows 7, Windows 7 SP1, Windows 7 SP2, Windows 7 SP3.

**Performance Testing**

**Performance Testing** is a software testing process used for testing the speed, response time, stability, reliability, scalability and resource usage of a software application under particular workload. The main purpose of performance testing is to identify and eliminate the performance bottlenecks in the software application. It is a subset of performance engineering and also known as “Perf Testing”.

The focus of Performance Testing is checking a software program’s

* Speed – Determines whether the application responds quickly
* Scalability – Determines maximum user load the software application can handle.
* Stability – Determines if the application is stable under varying loads

**Types of Performance Testing**

* **Load testing –** checks the application’s ability to perform under anticipated user loads. The objective is to identify performance bottlenecks before the software application goes live.
* **Stress testing –** involves testing an application under extreme workloads to see how it handles high traffic or data processing. The objective is to identify the breaking point of an application.
* **Endurance testing –** is done to make sure the software can handle the expected load over a long period of time.
* **Spike testing –** tests the software’s reaction to sudden large spikes in the load generated by users.
* **Volume testing** – Under Volume Testing large no. of. Data is populated in a database and the overall software system’s behavior is monitored. The objective is to check software application’s performance under varying database volumes.
* **Scalability testing**– The objective of scalability testing is to determine the software application’s effectiveness in “scaling up” to support an increase in user load. It helps plan capacity addition to your software system.

## Load Testing

**Load Testing** is a non-functional software testing process in which the performance of software application is tested under a specific expected load. It determines how the software application behaves while being accessed by multiple users simultaneously. The goal of Load Testing is to improve performance bottlenecks and to ensure stability and smooth functioning of software application before deployment.

This testing usually identifies –

* The maximum operating capacity of an application
* Determine whether the current infrastructure is sufficient to run the application
* Sustainability of application with respect to peak user load
* Number of concurrent users that an application can support, and scalability to allow more users to access it.

**Goals of Load Testing:**

Loading testing identifies the following problems before moving the application to market or Production:

* Response time for each transaction
* Performance of System components under various loads
* Performance of Database components under different loads
* Network delay between the client and the server
* Software design issues
* Server configuration issues like a Web server, application server, database server etc.
* Hardware limitation issues like CPU maximization, memory limitations, network bottleneck, etc.

## What is Volume Testing?

**Volume Testing** is a type of Software Testing, where the software is subjected to a huge volume of data. It is also referred to as **flood testing.** Volume testing is done to analyze the system performance by increasing the volume of data in the database.

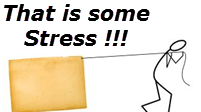
With the help of Volume testing, the impact on response time and system behavior can be studied when exposed to a high volume of data.

**Benefits of Volume Testing**

* By identifying load issues, a lot of money can be saved which otherwise will be spent on application maintenance.
* It helps in a quicker start for scalability plans
* Early identification of bottlenecks
* It assures your system is now capable of real-world usage

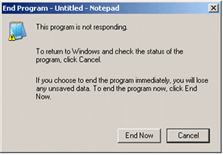
## Stress Testing

**Stress Testing** is a type of software testing that verifies stability & reliability of software application. The goal of Stress testing is measuring software on its robustness and error handling capabilities under extremely heavy load conditions and ensuring that software doesn’t crash under crunch situations. It even tests beyond normal operating points and evaluates how software works under extreme conditions.



In Software Engineering, Stress Testing is also known as Endurance Testing. Under Stress Testing, AUT is be stressed for a short period of time to know its withstanding capacity. A most prominent use **of stress testing is to determine the limit, at which the system or software or hardware breaks**. It also checks whether the system demonstrates effective error management under extreme conditions.

The application under testing will be stressed when 5GB data is copied from the website and pasted in notepad. Notepad is under stress and gives ‘Not Responded’ error message.



## Goals of Stress Testing

The goal of stress testing is to analyze the behavior of the system after a failure. For stress testing to be successful, a system should display an appropriate error message while it is under extreme conditions.

To conduct Stress Testing, sometimes, massive data sets may be used which may get lost during Stress Testing. Testers should not lose this security-related data while doing stress testing.

The main purpose of stress testing is to make sure that the system recovers after failure which is called as **recoverability**.

## Recovery Testing

**Recovery Testing** is software testing technique which verifies software’s ability to recover from failures like software/hardware crashes, network failures etc. The purpose of Recovery Testing is to determine whether software operations can be continued after disaster or integrity loss. Recovery testing involves reverting back software to the point where integrity was known and reprocessing transactions to the failure point

**Recovery Testing Example**

When an application is receiving data from the network, unplug the connecting cable.



* After some time, plug the cable back in and analyze the application’s ability to continue receiving data from the point at which the network connection was broken.
* Restart the system while a browser has a definite number of sessions open and check whether the browser is able to recover all of them or not

In Software Engineering, Recoverability Testing is a type of Non-[Functional Testing](https://www.guru99.com/functional-testing.html). (Non- functional testing refers to aspects of the software that may not be related to a specific function or user action such as scalability or security.)

The time taken to recover depends upon:

* The number of restart points
* A volume of the applications
* Training and skills of people conducting recovery activities and tools available for recovery.

When there are a number of failures then instead of taking care of all failures, the recovery testing should be done in a structured fashion which means recovery testing should be carried out for one segment and then another.

It is done by professional testers. Before recovery testing, adequate backup data is kept in secure locations. This is done to ensure that the operation can be continued even after a disaster.

## What is an Install/Uninstall Testing

**Installation Testing:**It is performed to verify if the software has been installed with all the necessary components and the application is working as expected. This is very important as installation would be the first user interaction with the end users.

**Uninstallation Testing:**Uninstallation testing is performed to verify if all the components of the application is removed during the process or NOT. All the files related to the application along with its folder structure have to be removed upon successful uninstallation. Post Uninstallation System should be able to go back to the stable state.