Automation Testing

TECHNOLOGY

Introduction to Non-Functional Testing

simplilearn

A Day in the Life of an Automation Test Engineer

Alex is a testing engineer. He is involved in a lot of functional testing projects to check the User-Interface, APIs, database, security, client/ server applications and functionality of the software products. Recently, he received a query to test the performance, usability, and reliability, of the project which is a part of non-functional automation testing.

A lot of different types of testing are on Alex's agenda, including testing and performance of his new project. He decided to do non-functional testing using an automation tool.

To accomplish the above, he must learn a few concepts that can help him to gain knowledge about non-functional automation testing.

Learning Objectives

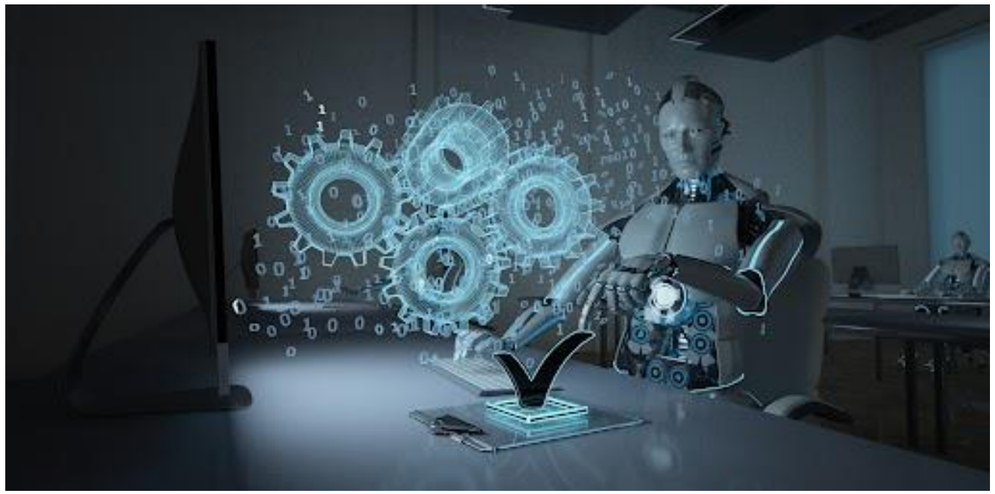
By the end of this lesson, you will be able to:

- Define non-functional testing
- Objectives of non-functional testing
- Classify functional test vs. non-functional test
- Learn types of non-functional testing
- Analyze performance testing and tools

Non-Functional Testing

Non-Functional Testing

Non-functional testing is a type of automated software testing to test the non-functional parameters such as usability, reliability, load test, performance, and accountability of the software.



The goal of functional testing is to ensure that software works according to specifications and in line with user expectations.

Objectives of Non-Functional Testing

Non-functional testing is required to identify potential risks and confirm whether the product will work per the requirements. The main objectives of non-functional testing are:

- Security
- Availability
- Efficiency
- Reliability
- Survivability
- Usability
- Flexibility
- Scalability
- Reusability
- Portability
- Interoperability
- Performance

Testing Parameter Security and Availability

Non-functional testing objectives can be defined as testing parameters.

During security testing, loopholes and vulnerabilities are checked to ensure that the security features are protected and authenticated.

Tests are conducted to ensure that the system is available 24x7 while conducting real-time operations.

Testing Parameter Efficiency and Reliability

Non-functional requirements are vital because they reveal how the system behaves in terms of constraints and prerequisites.

Efficiency

Efficiency verifies the duration of time it takes to execute the program. The period can be determined by dividing the number of test cases by the number of hours.

Reliability

This test is done at the end to ensure there is no failure. The main goal is to ensure the user's ease and utility.

Testing Parameters Survivability and Usability

Non-functional testing parameters survivability and usability are defined as:

Survivability roles come in the event of a failure; it should be able to recover. in extreme conditions, and able to sustain and maintain well.

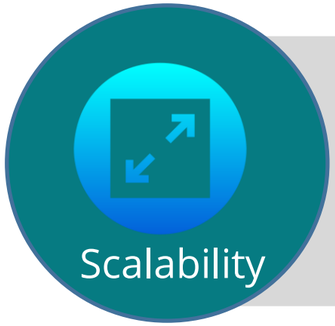
Usability tests examine how easily a user can navigate and accomplish different tasks.

Testing Parameters Flexibility and Scalability

Non-functional testing parameters flexibility and Scalability are defined as:

.

In flexibility, it checks the level to which the user can modify or extend the information system without directly changing the software.



Scalability refers to a software's ability to handle an increase in workload without degradation in performance or its ability to expand quickly.

Non-Functional Testing Reusability and Portability

Non-functional testing parameters flexibility and scalability are defined as:

.

A software system's reusability refers to its ability to be adapted into other systems.

Portability is the ease with which a software system can be transferred from its current hardware or software to another environment.

.

Non-Functional Testing Parameters

Non-functional testing objectives can be defined as testing parameters.

Interoperability is defined as how easily a system can share information and communicate with internal and external applications and systems.

It tests the overall performance of an application or system in accordance with requirements.

Benefits of Non-Functional Testing

Non-Functional testing plays an important role in software testing.

Security & Performance

Improves the overall security and performance of the product

User-Experience

Helps to build an easy and effective user experience

Production Readiness

Checks if the software is ready for production.

Reports

Generates and compares the report as per actual or expected expectations

Functional Testing vs. Non–Functional Testing

Functional Testing vs. Non-Functional Testing

The major difference between functional and non-functional testing is:

Functional Testing

- Functional testing is used to determine if a piece of software behaves per predetermined requirements.

Non-Functional Testing

- Non-functional testing is used to check quality control to ensure that everything works well and know in what circumstances they might fail.

Functional vs. Non-Functional Case Examples

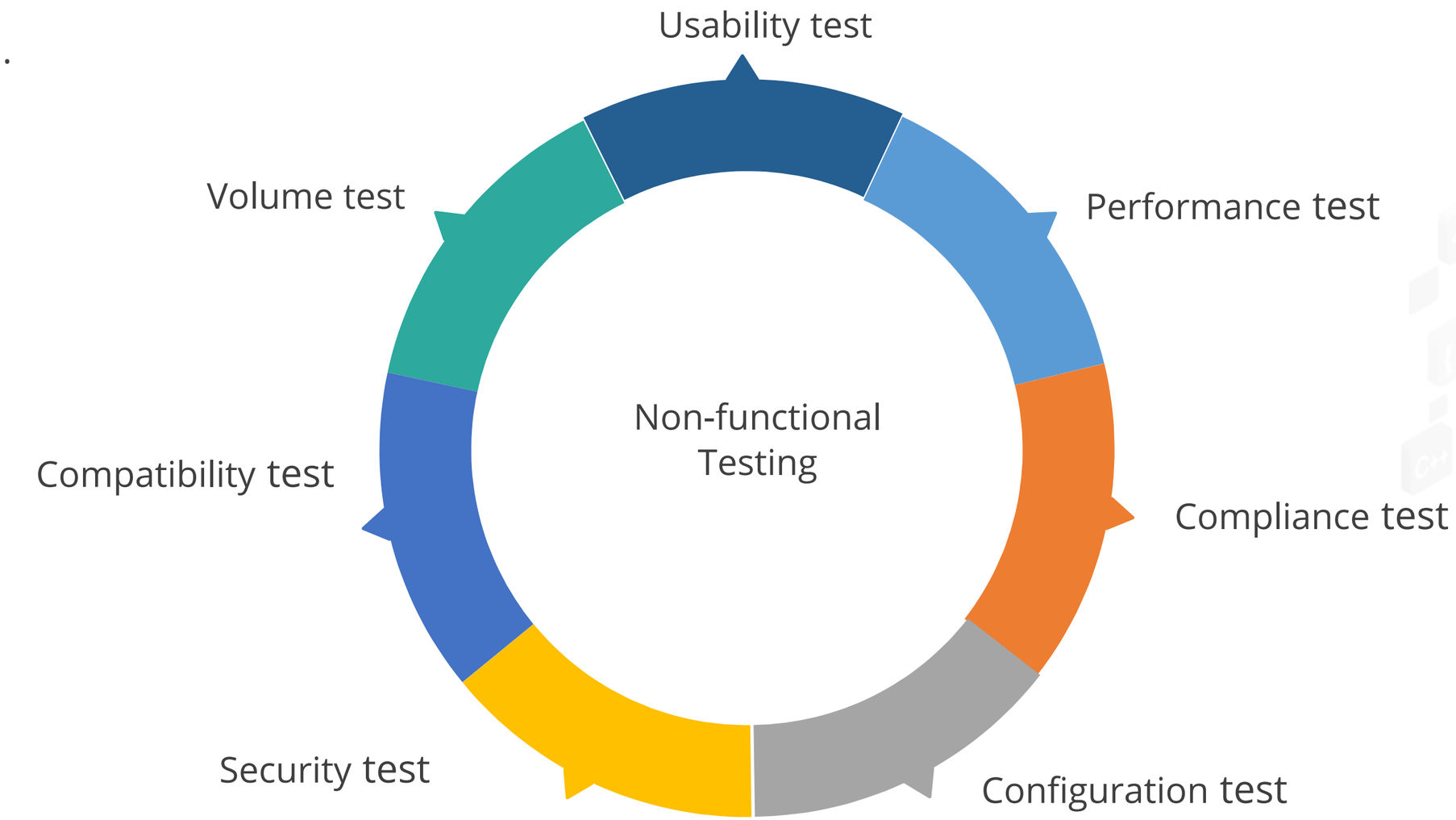
Functional Testing	Non-Functional Testing
Username and password must be validated before performing login functions	Within seconds of logging in, the dashboard should load immediately.
The user will be notified via email whenever email notifications are enabled.	A notification should be sent via email within five minutes.
A settings menu item opens a settings page.	The setting page should match the rest of the GUI in appearance.
When a 1MB JPG file is loaded, the uploader should accept it.	In the case of multiple files being uploaded, they should all be queued up.

Functional Testing vs. Non-functional Testing

Functional Testing	Non-Functional Testing
Tests per customer requirements	Tests per customer expectations
Carries out testing per business needs	Carries out per performance requirements
Uses test cases	Uses automated tools
Defines what to test	Defines how to test
Checks only at component level	Checks the overall system

Types of Non-Functional Testing

Types of Non-Functional Testing



Performance Test

A performance test is used to evaluate a software application's speed, response time, reliability, scalability, stability, and resource used under high workloads.

Performance Testing Checks

Performance test are conducted to prevent future risks.

Speed	Using a speed test, users can determine how responsive an application or system is under load.
Load	JMeter can estimate the maximum number of users that an application can support.
Stress	Performing a load test can help determine the application performance at higher loads.

Performance Testing Checks

Performance test are conducted to prevent future risks.

Soak	It determines if the system can sustain continuous loads for a long time. The soak test is also known as the endurance test.
Volume	During volume testing, a large amount of data is fed into the application to ensure that it is ready for use.
Spike	It examines the behavior of the system in the case of a sudden increase in the number of users.

Performance Testing Tools

Performance Testing Tools

Quality assurance is not possible without a performance test. Several tools are available in the market to do performance tests.

Top-Performing Performance Testing Tools

The important tools for non-functional testing are:

Key Takeaways

- Non-functional testing is a must to ensure the quality of the software or an application. It ensures a good user experience and ease in operations.
- The purpose of non-functional testing is to reduce risk and make it production-ready.
- Performance testing evaluates software applications' speed, response time, reliability, scalability, stability, and resource use under high workloads.
- JMeter collects and produces measurements and metrics for research and development using performance testing tools.