Functional and Non-Functional Testing

Functional Testing

Definition:
Functional testing verifies that each function of the software application operates in conformance with the requirement specifications. It involves:

- Providing appropriate input
- Verifying the output
- Comparing actual results with expected results

Types of Functional Testing:

Unit Testing:

- Tests individual subprograms, subroutines, classes, or procedures.
- Eases debugging tasks.

Smoke Testing:

- Tests core functionality quickly.
- Example: Logging in and accessing main modules.

Regression Testing:

- Ensures changes or additions haven't introduced new errors.
- Involves re-running previously scripted tests, often with automated tools.

Sanity Testing:

- Performed after receiving a software build with minor changes.
- Verifies specific functionalities work as expected.

System Testing:

- Compares the entire system to its original objectives.
- Ensures the system meets overall requirements.

User Acceptance Testing (UAT):

- Conducted by end users to verify the system meets their needs.
- Typically occurs before the final product delivery.

Non-Functional Testing

Definition:
Non-functional testing checks the non-functional aspects of a software application, such as performance, usability, and reliability.

Types of Non-Functional Testing:

Performance Testing:

- Measures the performance of components to identify bottlenecks.
  Types include:
  - **Load Testing:** Measures system performance under varying load.
  - **Stress Testing:** Tests behavior under extreme conditions.
  - **Spike Testing:** Sudden increase in user load.
  - **Endurance Testing:** Long-term performance under expected load.
  - **Scalability Testing:** Ability to scale up in terms of non-functional requirements.
  - **Volume Testing:** Performance with a large amount of data.

Compatibility Testing:

- Ensures software compatibility across different environments (OS, browsers, hardware).
  Types:
  - **Backward Compatibility Testing:** Checks compatibility with older versions.
  - **Forward Compatibility Testing:** Checks compatibility with upcoming versions.

Security Testing:

- Tests whether the application meets security requirements.
- Important for web applications due to their accessibility.

Usability Testing:

- Assesses the user-friendliness of an application.
- Focuses on ease of use, flexibility, and meeting objectives.

Localization Testing:

- Ensures software behaves according to local cultures and settings.
- Verifies linguistic and cultural appropriateness for target regions.

Top Tools for Performance Testing:

- LoadRunner
- Apache JMeter
- NeoLoad
- Rational Performance Tester
- Loadster
- QEngine (ManageEngine)
- Testing Anywhere
- Loadstorm

Key Points for Exam Preparation:

- Understand the definitions and purposes of both functional and non-functional testing.
- Familiarize yourself with the various types of functional and non-functional tests and their objectives.
- Know the examples and scenarios where each type of testing is applied.
- Review tools used for performance testing and their applications.

By focusing on these key areas, you can effectively prepare for questions related to software testing in your exam.