**Types of Testing- Functional, Non-Functional, and Beyond**

Software testing is broadly categorized into **functional testing** and **non-functional testing**, but there are also other specialized testing types that address specific challenges. Understanding these types helps in ensuring software reliability, performance, security, and usability.

**1. Functional Testing**

Functional testing focuses on verifying that the software behaves according to the specified requirements. It ensures that each feature works as expected.

**Types of Functional Testing**

* **Unit Testing** – Tests individual components or functions of the software.
* **Integration Testing** – Ensures different modules work together correctly.
* **System Testing** – Evaluates the entire system for compliance with requirements.
* **User Acceptance Testing (UAT)** – Validates the software against business needs and user expectations.
* **Regression Testing** – Checks that new changes do not break existing functionality.
* **Smoke Testing** – A quick check to see if the basic functionalities of an application work.
* **Sanity Testing** – Focuses on a particular function or bug fix to verify its correctness.

Sanity Testing is a subset of **Regression Testing**, performed when a **minor change** is made in the codebase to **ensure that the new changes have not broken existing functionality**. It focuses on **quick verification** rather than detailed testing.

**Example Scenario of Sanity Testing**

**Scenario:**

You are testing an **e-commerce website** where users can **log in** and **add items to the cart**. A recent **code update** was made to improve the **"Add to Cart"** functionality.

**Sanity Test Cases:**

1. **Login Functionality**
   * Verify that users can still log in with valid credentials.
   * Ensure that invalid credentials are not accepted.
2. **Add to Cart Functionality (Updated Feature)**
   * Verify that clicking **"Add to Cart"** adds the correct item.
   * Check that the cart reflects the correct **price** and **quantity**.
3. **Checkout Page**
   * Ensure that the checkout button redirects to the payment page.

**2. Non-Functional Testing**

Non-functional testing evaluates aspects like performance, security, usability, and reliability. These tests help improve the software’s overall efficiency and user experience.

**Types of Non-Functional Testing**

* **Performance Testing** – Measures system responsiveness and stability under various conditions.
  + *Load Testing* – Tests system performance under expected loads.
  + *Stress Testing* – Pushes the system beyond normal loads to identify breaking points.
  + *Scalability Testing* – Evaluates how the system handles increased user demand.
* **Security Testing** – Identifies vulnerabilities, threats, and risks in the software.
* **Usability Testing** – Assesses the application’s ease of use and user experience.
* **Compatibility Testing** – Ensures the software works on different browsers, operating systems, and devices.
* **Reliability Testing** – Measures the software’s stability over time under various conditions.

**3. Other Specialized Testing Types ("Beyond")**

Apart from functional and non-functional testing, there are other testing types that address specific software development challenges.

* **Exploratory Testing** – Testers actively explore the application without predefined test cases to find unexpected issues.
* **Ad-hoc Testing** – Informal testing performed without documentation, often to find critical bugs quickly.
* **Mutation Testing** – Modifies code slightly to check if test cases can detect the changes.
* **Chaos Testing** – Introduces random failures to test system resilience, commonly used in cloud environments.
* **A/B Testing** – Compares two versions of a feature to determine which performs better.
* **Accessibility Testing** – Ensures the application is usable for people with disabilities.

**Conclusion**

Testing is a crucial part of software development, covering both **functional** (correctness) and **non-functional** (quality attributes) aspects. Additionally, specialized testing methods address unique challenges like security, usability, and performance under extreme conditions. A well-rounded testing strategy improves software reliability, user satisfaction, and business success.