**What is Software Testing?**

Software testing is an process in the software development lifecycle. It involves verifying and validating that a software application is free of bugs, meets the technical requirements set by its design and development, and satisfies user requirements efficiently and effectively.

This process ensures that the application can handle all exceptional and boundary cases, providing a robust and reliable user experience. By systematically identifying and fixing issues, software testing helps deliver high-quality software that performs as expected in various scenarios.

**What are the dangers of release untested Software?**

1. Poor quality and performance.
2. Stability issues.
3. Poor user experience and unmet customer expectations & standards
4. Unreliable features: Users may become confused and frustrated if features do not work as expected.
5. Security Vulnerabilities.
6. Increased Maintenance & Support Costs.
7. Compromised data integrity: If the software isn’t properly tested, data may be lost or corrupted as a result of inaccuracy in storing, processing, and retrieving data.
8. losing customers trust, market and revenue

**Types Of Software Testing:**

***Functional Testing:***

**What is Functional Testing:**

Functional testing checks an application, website, or system to ensure that it is doing exactly what it is meant to. With focus on performance, security, or usability, functional testing is primarily concerned with the correctness of the application’s functionality.

**Examples of Functional Testing.**

**Sanity testing**, - is a basic test to quickly evaluate whether a claim or the result of a calculation can possibly be true. In computer science, a sanity test is a very brief run-through of the functionality of a computer program, system, calculation, or other analysis, to assure that part of the system or methodology works roughly as expected.

**Smoke tests** are a subset of test cases that cover the most important functionality of a component or system, used to aid assessment of whether main functions of the software appear to work correctly. a smoke test may address basic questions like:

-"does the program run?",

-"does the user interface open?", or

-"does clicking the main button do anything?"

The process of smoke testing aims to determine whether the application is so badly broken as to make further immediate testing unnecessary.

**Usability testing** - a technique used in user-oriented interaction design to evaluate a product by testing it on users. This can be seen as an irreplaceable usability practice, since it gives direct input on how real users use the system.

**Non-Functional Testing:**

**What is Non-Functional Testing?**

Non-functional testing focuses on evaluating the system’s performance, scalability, security, usability, and reliability, rather than its specific functionality. It ensures that the system can handle real-world demands and provides a seamless, high-quality user experience.

**Examples of Non-Functional Testing**

**Security testing** - Security testing is used to find the software application’s weaknesses. The testing is carried out by looking into the design of the system and the mindset of an attacker. Finding the parts of the code where an attack is most likely to occur.

**Documentation testing** - ensures that documentation about how to use the system matches with what the system does, providing proof that system

**Stress Testing -** It checks the stability of a system or software application by applying a load greater than the desired demand to test the operational capacity to a breaking point.