Testing & Analysis

Unit Testing

Unit testing focuses on verifying individual components or modules of the software. Each function or method is tested in isolation to ensure that it works as expected. Unit testing is typically done early in the development process and often automated using tools like JUnit (for Java), PHP Unit (for PHP), or Jest (for JavaScript).

Integration Testing

Integration testing validates the interaction between different modules or components of the software. After the individual units have been tested, integration testing ensures that these components work together as expected. It helps identify issues related to the interaction, such as data flow problems or interface mismatches, ensuring the system's cohesion.

System Testing

System testing evaluates the complete and fully integrated system to verify that it meets the specified requirements. It tests the entire system in a real-world environment, focusing on end-to-end functionality, performance, security, and usability. System testing includes various subtypes, such as performance and security testing.

Acceptance Testing

Also known as User Acceptance Testing (UAT), acceptance testing is conducted to ensure that the system satisfies business requirements and is ready for deployment. This is typically done by the end users or stakeholders to validate that the software behaves as expected and meets their needs. It’s the final testing phase before release and can include both alpha and beta testing.

Recovery Testing

Recovery testing assesses the system’s ability to recover from failures, such as crashes, hardware malfunctions, or network failures. It checks how well the system can restore itself after encountering an issue and how quickly it returns to normal operation. This type of testing ensures that the software is resilient and fault-tolerant.

Functional Testing

Functional testing focuses on verifying that the software's functionalities operate according to the specified requirements. This type of testing checks whether each feature of the system performs as expected, covering input/output, user commands, data manipulation, and interactions with external systems. It ensures that the system functions in line with user expectations.

Hardware/Software Testing

Hardware/software testing ensures that the software is compatible with the hardware on which it runs. It tests the interactions between hardware components and software systems, verifying that they work together without errors. This type of testing is crucial for systems with specific hardware dependencies, such as embedded systems, IoT devices, or specialized hardware setups.

Security Testing

Security testing identifies vulnerabilities, risks, and potential threats in the system. It ensures that the application is protected from attacks like SQL injections, cross-site scripting (XSS), and unauthorized access. Security testing includes penetration testing, risk assessments, and testing for data encryption, authentication, and access controls to prevent breaches.