

Name: AIAN JAE S. GARCIA
Section: BSIT - 3A

1. Acceptance Testing

A type of testing performed to determine whether the system meets the business requirements and is ready for delivery to the end users or customers. Usually done by clients or end-users.

2. Ad Hoc Testing

An informal testing approach without test cases or documentation, where testers try to find defects through random, unstructured testing based on intuition.

3. Black Box Testing

A testing method where the tester examines functionality without knowing the internal code, structure, or implementation. Focuses on input and output behavior.

4. Build Validation Testing (BVT)

A quick test performed after a new build is released to verify that the build is stable and major functionalities are working before deeper testing begins. Similar to smoke testing.

5. Exploratory Testing

A simultaneous process of learning, test design, and execution where testers explore the application to discover defects using creativity and experience.

6. Functional Testing

Testing that verifies a system's features works according to functional requirements. Focuses on *what* the system does, not how it does it.

7. Integration Testing

Testing individual modules or components combined together to ensure they work correctly as a group.

8. Load Testing

A type of performance testing that evaluates how a system behaves under expected user loads to ensure it performs efficiently.

9. Penetration Testing

A security testing technique where testers simulate cyberattacks to find vulnerabilities that attackers could exploit.

10. Performance Testing

Testing conducted to evaluate speed, responsiveness, scalability, and stability under various workloads.

11. Regression Testing

Re-running previously executed test cases after changes (updates, bug fixes) to ensure existing functionality still works correctly.

12. Sanity Testing

A brief testing process to confirm that specific, newly added or fixed functionalities work properly before proceeding to more detailed testing.

13. Smoke Testing

A high-level testing process to check basic functionality and verify that the software build is stable enough for further testing.

14. Soak Testing

A type of performance testing that checks system performance over an extended period to detect memory leaks, slowdowns, or reliability issues.

15. Stress Testing

Testing that evaluates how the system performs under extreme or beyond-normal load conditions to determine breaking points and stability.

16. Unit Testing

Testing individual units or components (e.g., functions, methods, classes) in isolation to ensure they work correctly. Usually done by developers.

17. Usability Testing

Testing that evaluates how easy and user-friendly a system is by observing real users performing tasks.

18. White Box Testing

A testing method where the tester knows the internal structure, code, and logic. Focuses on validating internal paths, conditions, loops, and code quality.