Software Testing

Black box testing – Internal system design is not considered in this type of testing. Tests are based on requirements and functionality.

White box testing – This testing is based on knowledge of the internal logic of an application's code. Also known as Glass box Testing. Internal software and code working should be known for this type of testing. Tests are based on coverage of code statements, branches, paths, conditions.

Unit testing – Testing of individual software components or modules. Typically done by the programmer and not by testers, as it requires detailed knowledge of the internal program design and code. may require developing test driver modules or test harnesses.

Integration testing – Testing of integrated modules to verify combined functionality after integration. Modules are typically code modules, individual applications, client and server applications on a network, etc. This type of testing is especially relevant to client/server and distributed systems.

Functional testing – This type of testing ignores the internal parts and focus on the output is as per requirement or not. Black-box type testing geared to functional requirements of an application.

System testing – Entire system is tested as per the requirements. Black-box type testing that is based on overall requirements specifications, covers all combined parts of a system.

End-to-end testing – Similar to system testing, involves testing of a complete application environment in a situation that mimics real-world use, such as interacting with a database, using network communications, or interacting with other hardware, applications, or systems if appropriate.

Regression testing – Testing the application as a whole for the modification in any module or functionality. Difficult to cover all the system in regression testing so typically automation tools are used for these testing types.

Acceptance testing -Normally this type of testing is done to verify if system meets the customer specified requirements. User or customer does this testing to determine whether to accept application.

Load testing – Its a performance testing to check system behavior under load. Testing an application under heavy loads, such as testing of a web site under a range of loads to determine at what point the system's response time degrades or fails.

Stress testing – System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load.

Performance testing – Term often used interchangeably with 'stress' and 'load' testing. To check whether system meets performance requirements. Used different performance and load tools to do this.

Usability testing – User-friendliness check. Application flow is tested, Can new user understand the application easily, Proper help documented whenever user stuck at any point. Basically system navigation is checked in this testing.

Install/uninstall testing - Tested for full, partial, or upgrade install/uninstall processes on different operating systems under different hardware, software environment.

Recovery testing – Testing how well a system recovers from crashes, hardware failures, or other catastrophic problems.

Security testing – Can system be penetrated by any hacking way. Testing how well the system protects against unauthorized internal or external access. Checked if system, database is safe from external attacks.

Compatibility testing – Testing how well software performs in a particular hardware/software/operating system/network environment and different combination s of above.

Comparison testing – Comparison of product strengths and weaknesses with previous versions or other similar products.

Smoke Testing–, also known as "Build Verification Testing", is a type of software testing that comprises of a non-exhaustive set of tests that aim at ensuring that the most important functions work. The results of this testing is used to decide if a build is stable enough to proceed with further testing.

Alpha testing – In house virtual user environment can be created for this type of testing. Testing is done at the end of development. Still minor design changes may be made as a result of such testing.

Beta testing – Testing typically done by end-users or others. Final testing before releasing application for commercial purpose.

Beta testing: is done at the client side and all end users use the system and see whether the system is working as per their given requirements. This is done after pilot testing is successful and the defects found in case of pilot testing are fixed.

Pilot Testing: is verifying a component of the system or the entire system under a real-time operating condition. It verifies the major functionality of the system before going into production

Pilot testing: is done by group of users who try to test the system prior its full deployment to provide the feedback about the quality.

Software Quality Control (SQC)- is a set of activities for ensuring quality in software products. The activities focus on identifying defects in the actual products produced.

Software Quality Assurance (SQA)- is a set of activities for ensuring quality in software engineering processes. The activities establish and evaluate the processes that produce products

What is Cyclomatic Complexity?

Cyclomatic complexity : It simply measures the amount of decision logic in the program module. Cyclomatic complexity gives the minimum number of paths that can generate all possible paths through the module.

Cyclomatic complexity is defined as

$CC = E - N + P$

Where

E = the number of edges of the graph

N = the number of nodes of the graph

P = the number of connected components

In case of connected graph

$CC = E - N + 2$

In simplified way it can be defined as

$CC = D + 1$

Where

D = the number of decision points in the graph