

SE - Assignment 2

Q1. Explain equivalence partitioning and boundary value analysis.

Equivalence partitioning & boundary value analysis are two techniques used in software testing to identify test cases and reduce the number of test cases to test a software system.

(i) Equivalence partitioning

It is used to divide a range of input values into smaller and more manageable subsets that are more likely to behave similarly. Input values are divided into two groups or partitions based on their equivalence or similarity. The idea is that if one test case from a partition is successful, all others would be as well.

(ii) Boundary value Analysis

It is a testing technique used to identify errors or defects that occur at or around the boundaries of input values. Input values are chosen from the edges of their ranges under the behaviour of the system may change. The idea is that if a system is functioning correctly at boundaries it will likely function correctly within that range.

Q2. with example, explain OAT

- Operational Acceptance Testing (OAT) is a type of testing that focuses on verifying whether a software application can cooperate in its intended production environment.
- OAT is conducted after completion of a system and user acceptance testing and before software is deployed into production.
- Example: A company has developed a new e-commerce website and they want to ensure that it can handle high traffic loads and transactions without issues.
- They may conduct OAT to verify that the website's servers, network infrastructure and software components can handle the expected load and perform well under stress.
- During OAT, the company may simulate realistic production scenarios such as high traffic loads, simultaneous transactions and data backups.

Q3. Explain version control in SCM

- Version control is a critical aspect of software configuration management (SCM) that helps developers track changes made to source code, documentation and other files over time.
- It is a process that allows multiple developers to work on the same codebase, keep track of changes and collaborate effectively.
- At its core, version control involves creating a repository which is a centralised or distributed system for storing files and recording changes.
- The VCS keeps track of all changes made to them and when they were made.
- There are two main types of VCS
 - (i) Centralised - They have a single central repository where all changes are stored. Developers must keep track of their changes and then check them back in. Examples of CVS include Subversion and Perforce.
 - (ii) Distributed VCS - create a local copy of the entire repository on each developer's machine. This means the developers can work independently without need for checking out each time.