Name : Ayush Jain
SAP ID : 60004200182
Div : B
Computer Science

SE - Assignment 2

Q. 1) Explain equivalence partitioning and boundary value analysis?

→ **Equivalence Partitioning:**

1) It is a software technique that involves dividing the input data of a software applications into groups, or partitions, that are expected to exhibit similar behaviour. The idea behind this technique is to reduce the no. of test cases required to adequately test a system while still ensuring that all possible scenarios are covered.

2) In equivalence partitioning, each partition is tested using a representative test case from that partition. The goal is to ensure that the software is tested thoroughly, without unnecessary duplication of test cases.

3) For eg: if a software application accepts a user's age and input, we can partition the i/p into three groups: ages below 18, ages b/w 18 and 65, and ages above 65. We would then test each partition with an i/p value that is representative of that partition.

Boundary value Analysis:

1) Boundary value analysis is a software testing tech that involves testing the boundaries b/w different i/p data partitions. The goal is to ensure that the software handles boundary conditions correctly, which are often where errors are most likely to occur.

2) For eg: if a software app accepts a users age as i/p, we can test the boundary value for each partition. For ages below 18, we would test with values such as 17 and 18

for ages b/w 18 and 65, we would test with values such as 18, 19, 64, 65. For ages above 65, we would test with values such as 65 and 66.

3) By testing the boundaries, we can ensure that the software handles edge cases correctly and that is robust to unexpected i/p values.

Q. 2> With a suitable example, explain OAT.

→ 1) Operational Acceptance Testing (OAT) is a type of software testing that is carried out to ensure that a software system or application is ready to be used in a production environment. OAT is typically conducted by end-users or business stakeholders who will be using the software in real-world scenarios.

2) An eg of OAT is testing a banking application that has been developed to handle online transactions. Before the application is launched, the bank would conduct OAT to ensure that the app can handle the expected level of traffic, user concurrency, and transaction volumes in real word scenarios.

3) During the OAT process, the bank would simulate various use cases, such as logging in to the system, checking account balances, making transactions, and logging out of the systems. The bank would also test the application's response time, security features, and error handling capabilities.

4) Once the OAT process is complete, the bank can be confident that the application is ready to be used by customers, and that it will provide a reliable and secure platform for online banking transactions.

Q. 3) Explain version control in SCM.

→ 1) Version control is an essential part of software configuration management (SCM). It is the process of tracking changes to source code, documentation, and other files over time.
2) The primary goal of version control is to allow multiple developers to work on the same project simultaneously without interfering with each other's work.
3) VCS provides a way to manage changes to files over time. The VCS maintains a history of all changes made to file, allowing developers to track the evolution of the codebase and roll back to earlier versions if needed.
4) There are two main types of version control system: centralized and distributed. Centralized control system, such as Subversion, store all files and their history on a control server. Distributed version control system, such as GIT allow developers to maintain their own local copies of the entire codebases, including its history.
5) Version control system offer several benifits to software development teams including: (1) collaboration (2) Versioning. (3) Branching and merging. (4) Traceability (5) Continuous integration
6) Overall, version control is a critical component of software development, providing developers with the tools they need to collaborate effectively, manage changes and maintain integrity.