1. What is Black box testing?

**Black Box Testing**: If one performs testing only on the **functional** part of an application without having knowledge of structural part, then it is known as black box testing. During this, external functionality is validated.

Other than Unit Testing, remaining testing types belong to black box testing.

1. What is White box testing?

**White Box Testing**: If one performs testing on the **structural** part of an application, then that method of testing is known as white box testing. This involves verifying the internal structure of the application.

**White Box Testing is coverage of the specification in the code.**

**Code coverage:**

**Segment coverage:**  
Ensure that each code statement is executed once.

**Branch Coverage or Node Testing:**  
Coverage of each code branch in from all possible was.

**Compound Condition Coverage:**  
For multiple condition test each condition with multiple paths and combination of different path to reach that condition.

**Basis Path Testing:**  
Each independent path in the code is taken for testing.

1. What is SDLC?

**Software Development Life Cycle**, or Software Development Process, defines the steps/ stages/ phases in the building of software.

Software Development Life Cycle contains six phases:

* + 1. Initial Or Requirements Phase
    2. Analysis Phase
    3. Design Phase
    4. Coding Phase
    5. Testing Phase
    6. Delivery and Maintenance phase.

1. What is STLC?

The Software Testing Life Cycle (STLC) consists of series of activities carried out methodologically to help certify(test) your software product.

* + - 1. Requirement Analysis
      2. Test Planning
      3. Test Case Development
      4. Test Environment Setup
      5. Test Execution
      6. Test Cycle Closure

1. What is waterfall model?
   1. It is a linear-sequential life cycle model.
   2. In a waterfall model, each phase must be completed fully before the next phase can begin.
   3. In waterfall model phases do not overlap.
   4. At the end of each phase, a review takes place to determine if the project is on the right path and whether or not to continue or discard the project.
   5. Waterfall model works well for smaller projects where requirements are very well understood.



Not a good model for complex and object-oriented projects where requirements are not fully developed at the start of SDLC. Suitable for small projects where requirements get fixed at start.

1. Agile questions????
2. What is Software Configuration Management?

In software engineering, software configuration management (SCM or S/W CM) is the task of tracking and controlling changes in the software,

1. What is Test Plan?

A Software Test Plan is a document describing the testing scope and activities in a particular project. It is the basis for formally testing any softwarein a project.

A test plan can be defined as a document describing the scope, approach, resources and schedule of testing activities

1.0 Introduction

1.1 Objective

1.2 Reference Documents

2.0 Coverage of Testing

2.1 Features to be tested

2.2 Features not to be tested

3.0 Test Strategy

3.1 Levels of Testing

3.2 Types of Testing

3.3 Test Design Techniques

3.4 Configuration Management

3.5 Test Metrics

3.6 Terminology

3.7 Automation Plan

3.8 List of Automated Tools

4.0 Base Criteria

4.1 Acceptance Criteria

4.2 Suspension Criteria

5.0 Test Deliverables

6.0 Test Environment

7.0 Resource Planning

8.0 Scheduling

9.0 Staffing and Training

10. Risks and Contingencies

11. Assumptions

12. Approval Information

1. What is Test strategy? What is the difference between Test plan and Test Strategy?

Test strategy is an organization level plan document which is used for testing all the projects in the organization. This contains approach and strategy used for testing in any project of the enterprise.

Note: Test plan is a project level term which is used for testing a particular project.

1. What is Test Scenario?

Test Scenario - is any functionality that can be tested.

When test requirements are grouped depending on the functionality of a module, Test scenarios are the high level classification that can be done.

1. What is Test Case?

Test case is a set of steps and expected conditions and results.

Before you test any system or test any requirement satisfied in the system, a test case needs to be designed.

A test case has sets of test data, preconditions, post conditions,expected results , developed for a particular test objective in order to verify compliance against a specific requirement.

1. What is Boundary value analysis?

**Boundary Value Analysis (BVA):** The tester tests with the input values at the boundaries with the intent that there is higher probability of finding errors at boundary conditions.

Boundary values include minimum, maximum, minimum-1, maximum+1

Valid boundaries: minimum, maximum

Invalid boundaries: minimum-1, maximum+1

Ex: Name 2 to 30 alphanumeric characters

Values: Ab, AbcdefghijAbcdefghijAbcdefghij

A, AbcdefghijAbcdefghijAbcdefghijA

1. What is Equivalence Class Partitioning?

**Equivalence Class Partitioning(ECP):** In this technique, classes of input values are identified where each member of the class causes same output and results to occur. These classes are called equivalence classes.

For each equivalence partition, test inputs are selected.

--

Driving License Issue – Input data classes

age < 16 --> No license

age between 16 and 18 -- restrictions apply

18 - 50 --> give 15 year license

50 - 80-- give 5 year license

80+ No license

1. What is Negative testing? Positive testing? Main path? Alternate path?

A negative test is when you put in an invalid input and receives errors. While a positive testing, is when you put in a valid input and expect some action to be completed in accordance with the specification.

1. Explain a test case from your project? How did you automate?
2. What reports did you prepare?

Types of Status Reports:

* + Daily Status Report
  + Daily Defect Report
  + Weekly Status Report
  + Retesting Report
  + Build Summary Report
  + Burn down chart

1. Requirements Clarification Note
2. What is bild, release? What is build release note?

**Build** is a Executable file which is handed over to the tester to test the functionality of the developed part of the project. (Ex: Daily Build)

**Release** is that which is finally handed over to the client of the project after the development and testing phases are completed. (Ex: Production Release)

Build Release Note contains:

Requirements implemented in in this build(Req#)

Defects fixed in this build(Defect #) – Second Release and later

URL of the build

Login user and password

DSN(Data Source Name)

Database user name and password

Often, Build Release Note is sent as an email. Get the attached documents from release note. After build is deployed, the email is sent.

1. What is smoke test? What if smoke test fails? What if smoke test passes?
2. What is regression test?
3. How did you do regression test? Duration of regression test. Number of test cases in your project. Number of test cases in your regression. Periodicity of regression. How often regression is is done?
4. How did you do regression automation?
5. What is retest?
6. What is Defect Lift Cycle?

Major activities in defect management:

Defect identification,

Defect reporting,

Defect tracking,

Defect closing

1. What is Priority? What is severity?

Priority: P1, P2,P3, P4 (Importance of the defect with respect to client)

Severity: Critical, Major, Moderate, Minimal (Importance of the defect with respect to code point of view)

Examples of Low priority. Low severity.

High priority High severity

Low priority, High severity

High priority, Low severity

Severity: Critical, Major, Moderate, Minimal (Importance of the defect with respect to functional point of view)

**Critical/Show Stopper** -- A bug that prevents further testing of the product or function. Examples of this include security permission required to access a function under test.

**Major / High**: A defect that does not function as expected/designed or cause other functionality to fail to meet requirements. The workaround can be provided for such bugs. Examples: inaccurate calculations, the wrong field being updated

**Moderate / Medium**: The defects which do not conform to standards and conventions can be classified as Medium Bugs. Easy workarounds exists to achieve functionality objectives. Examples: matching visual and text links which lead to different end points.

**Minor / Low**: User Interface issues. Cosmetic defects which do not affect the functionality of the system can be classified as Minor Bugs.

1. How do you choose automation test cases? Did you do manual testing also?
2. What defect statuses you used?

Status:

New: While reporting for first time

Open: Dev Lead/Developer/DTT Validation

Rejected(Not a bug): Dev Lead/Developer/DTT reject if defect invalid

Duplicate: Two bugs mention the same concept of the bug.

Deferred: Dev Lead/Developer/DTT provide due to time constraint

Fixed: Developer provides after fixing the defect

Fix Verified:

Closed: Tester provides after performing confirmation testing

Reopen: Tester reopens with valid reasons and proof

New🡪 Open 🡪Fixed🡪 Fix Verified 🡪 Closed

New 🡪 Open 🡪Rejected 🡪 Closed

New 🡪 Open 🡪Rejected 🡪 Reopen 🡪 Fixed 🡪 Fix Verified 🡪 Closed

New 🡪 Open 🡪Fixed🡪 Reopen 🡪 Fixed 🡪 Fix Verified 🡪 Closed

New 🡪 Open 🡪Deferred

1. When did you have differences with your team members or project members? How did you resolve?

Triage meeting.

1. What Test Management tool you used? What agile management tool used? What agile tool?

Rally, Jira

1. What us UAT? Who did that?
2. Examples of types of testing. What is the difference between UAT (User Acceptance Testing) and System testing?

Integration testing is a level of software testing process, where individual units of an application are combined and tested. It is usually performed after unit and functional testing.

System Testing: System testing is finding defects when the system under goes testing as a whole, it is also known as end to end testing. In such type of testing, the application undergoes from beginning till the end.

UAT: User Acceptance Testing (UAT) involves running a product through a series of specific tests which determines whether the product will meet the needs of its users.

1. What is functional testing?
2. What is RTM?
3. What is database testing? How did you do?

Data structure, data validation, Data integrity. JDBC, SQL.

1. How did you write test cases?
2. How did you prepare test data?
3. How did you do time estimate?