

Module 2: Manual Testing

1. What is traceability matrix?

In software development, a traceability matrix (TM) is a document, usually in the form of a table, used to assist in determining the completeness of a relationship by correlating any two baselined documents using a many-to-many relationship comparison.

2. What is integration testing?

Testing performed to expose defects in the interfaces and in the interaction between intergraded components or system.

OR

3. Integration testing is a level of the software testing process where individual units are combined and tested as a group.

Two level of integration testing:

- component integration testing
- system integration testing

4. What is component tasting?

Component testing is performed exposed defects in the interface and interaction between integrated components.

5. What is functional system testing?

Black box testing techniques against the interfacing requirements for the component under test.

6. What is non-functional testing?

Where appropriate, for performance or reliability testing of the component interfaces.

7. What determines the level of risk?

A factor that could results in future negative consequence, usually expressed as impact and likelihood.

Project base risk, product base risk.

8. Mention what bigbang testing is?

Big Bang Integration Testing is an integration testing strategy wherein all units are linked at once, resulting in a complete system. When this type of testing strategy is adopted, it is difficult to isolate any errors found, because attention is not paid to verifying the interfaces across individual units.

9. What is purpose of exit criteria?

To determine whether a given test activity has been completed or NOT.

10. What is 7 key principle? Explain in details.

- I. **Testing shows presence of defects :**
Software testing reduce the presence of defects.
Software testing talk about the presence of defects and don't talk about absence of defects.
- II. **Exhaustive testing is not possible :**
Exhaustive testing is impossible means that software can never test at every test case.
It can test only some test cases and assume that software is correct and it will produce the correct output in every test case.
- III. **Early testing :**
The defect detected in early phases of SDLC will very less expensive.
For better performance of software, start software testing will start at initial phase.
- IV. **Defect clustering :**
In a project, a small number of module can contain most of the defects.
Defects are not evenly spread in a system.
- V. **Pesticide paradox :**
Repeating the same test case again and again will not find new bugs.
It is necessary to review the test cases and add or update test cases to find new bugs.
- VI. **Testing is context depended :**
Testing approach depends on context of software developed.
Different type of software needs to perform different types of testing.
The testing of the e-commerce site is different from the testing of the android application.
- VII. **Absence of error fallacy :**
If a built software is 99% bug free but don't fill the user requirement then it is unusable.
It is not only necessary that software is 99% bug free but it also mandatory to fulfil all the customer requirement.

11. What is error, defect, bug, and failure?

Error: A mistake on coding is called error.

Defect: error found by tester is called defect.

Bug: defect accepted by developer team then it is called Bug.

Failure: when defect reaches the end customer it is called failure.

12. Difference between QA v/s QC v/s Tester.

Quality Assurance (QA)	Quality Control (QC)	Testing
Process-oriented focuses on making the process of creating software better.	A product-oriented approach is a way to make sure the software meets all its requirements.	Testing the software system is about finding any mistakes or issues.
It works with the development process to help stop mistakes and ensure the software is of good quality. This means setting up and keeping standards, processes, procedures, and tools in place to ensure we're consistently producing high-quality software.	It's done after the development process and involves running test cases and seeing how the software reacts.	This usually happens after the software has been created, and it's all about ensuring that the software's quality is up to standard.
The goal is to keep improving our software development process for the best possible results.	The goal is to find any defects or errors in the software and fix them.	It involves running tests and looking at what comes out of them, finding any problems with the software, and ensuring that it does everything it's supposed to do.

13. Difference between verification and validation.

Verification	Validation
Are we implementing the system right?	Are we implementing the right system?
Evaluating products of a development phase	Evaluating products at the closing of the development process
The objective is making sure the product is as per the requirements and design specifications	The objective is making sure that the product meets user's requirements
Activities included: reviews, meetings, and inspections	Activities included: black box testing, white box testing, and grey box testing
Verifies that outputs are according to inputs or not	Validates that the users accept the software or not
Items evaluated: plans, requirement specifications, design specifications, code, and test cases	Items evaluated: actual product or software under test
Manual checking of the documents and files	Checking the developed products using the documents and files

14. What is exploratory technique?

It that can be situation were the tester has no references to actual requirement and never go through software. In such situation we can explore and test application.

15. What is boundary value testing?

In which technique tests are designed to include representative of boundary values in a range. The limits of valid ranges.

16. What is equivalence partitioning testing?

In which input data is divided into same partitions of valid and invalid values, and it is mandatory value is tested of all partitions.

17. What is alpha testing?

It is always performed by the developers at the software development Site. It is always performed in **Virtual Environment**.

18. What is beta testing?

It is always performed by the customers at their own site. It is performed in **Real Time Environment**.

19. What is functional system testing?

A requirement that specifies a function that a system or system component must perform.

There is two types of Test Approach

- Requirement Based Functional Testing
- Process Based Testing

20. What is GUI testing?

Graphical User Interface (GUI) testing is the process of testing the systems GUI of the System under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons, and all types of bars – tool bar, menu bar, dialog boxes and windows etc.

21. What is adhoc testing?

Adhoc testing is an informal testing type with an **aim to break the system**. The testing technique called Error Guessing.

22. What is load testing?

Its a performance testing to check system behaviour 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.

23. What is stress testing?

Stress testing is used to test the stability & reliability of the system. Stress testing is also known as **endurance testing**.

24. What is white box testing and list the types of white box testing?

Testing based on an analysis of the internal structure of the component or system. The tester required knowledge of how the software is implemented, how it work.

List the type of white box testing

- Statement coverage
- Decision coverage
- Condition coverage

25. What is black box testing? What are the different black box testing techniques?

Testing, either functional or non-functional, without reference to the internal structure of the component or system. Tester have no knowledge of how the system or components is structured inside the box.

Black box testing techniques:

- Equivalence partitioning
- Boundary value analysis
- Decision table testing
- State transaction testing

26. When should "Regression Testing" be performed?

Testing of a previously tested program following modification to ensure that defects have not been introduced or uncovered in unchanged areas of the software, as a result of the changes made. It is performed when the software or its environment is changed.

27. Explain types of Performance testing.

Load testing

Stress testing

Spike testing

Volume testing

Scalability testing

28. Explain the difference between Functional testing and Non-Functional testing.

Functional Testing	Non-Functional Testing
It verifies the operations and actions of an application.	It verifies the behavior of an application.
It is based on requirements of customer.	It is based on expectations of customer.
It helps to enhance the behavior of the application.	It helps to improve the performance of the application.
Functional testing is easy to execute manually.	It is hard to execute non-functional testing manually.
It tests what the product does.	It describes how the product does.

Functional testing is based on the business requirement.	Non-functional testing is based on the performance requirement.
Examples: <ol style="list-style-type: none"> 1. Unit Testing 2. Smoke Testing 3. Integration Testing 4. Regression Testing 	Examples: <ol style="list-style-type: none"> 1. Performance Testing 2. Load Testing 3. Stress Testing 4. Scalability Testing

29. What is bug life cycle?

The duration or time span between the first time defects is found and the time that it is closed successfully, rejected, postponed or deferred is called as 'Defect Life Cycle'.

30. Difference between priority and severity.

Severity is absolute and Customer-Focused. It is the extent to which the defect can affect the software. In other words it defines the impact that a given defect has on the system.

Priority is Relative and Business-Focused. Priority defines the order in which we should resolve a defect.

31. What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)?

STLC	SDLC
STLC is related to software testing.	The SDLC is primarily concerned with software development.
STLC fewer people are involved.	SDLC a more people involved in all processes, (number of developers).
STLC ensures that anything we produce meets customer needs and that the products are of high quality.	The SDLC ensures that we are building the correct thing in the correct manner.
STLC is a testing life cycle.	SDLC is a development life cycle.
STLC is concerned with both the development and testing processes, but it is primarily concerned with the testing process.	Software development life cycle assures that we deliver high-quality software which is as per client needs.
In STLC QA team analysis all the requirement from the requirement document and create a system test plan.	In SDLC business analysts gather all project-related requirements from a stockholder and create a development plan.
In the software testing life cycle, the Test Architect or test manager creates a strategy to test a software application.	In software development, the life cycle development team create the high-level and low-level design of project based on clients' requirement.

32. What is the difference between test scenarios, test cases, and test script?

Test script: A set of sequential instruction that detail how to execute a core business function.

Test scenario: A Scenario is any functionality that can be tested. It is also called test Condition, or test Possibility.

Test case: Test cases involve the set of steps, conditions and inputs which can be used while performing the testing tasks.

33. Explain what Test Plan is? What is the information that should be covered?

A Test Plan is a detailed document that catalogues the test strategies, objectives, schedule, estimations, deadlines, and resources required to complete that project. Think of it as a blueprint for running the tests needed to ensure the software is working correctly – controlled by test managers.

34. What is severity?

Severity is absolute and Customer-Focused. It is the extent to which the defect can affect the software. In other words it defines the impact that a given defect has on the system.

35. What is priority?

Priority is Relative and Business-Focused. Priority defines the order in which we should resolve a defect.

36. Difference between smoke testing and sanity testing.

Smoke testing	Sanity testing
Smoke Testing is performed to ascertain that the critical functionalities of the program are working fine.	Sanity Testing is done to check the new functionality / bugs have been fixed.
The objective of this testing is to verify the "stability" of the system in order to proceed with more rigorous testing.	The objective of the testing is to verify the "rationality" of the system in order to proceed with more rigorous testing.
This testing is performed by the developers or tester.	Sanity testing is usually performed by testers.
Smoke testing is usually documented or scripted.	Sanity testing is usually not documented and is unscripted.
Smoke testing is a subset of Regression testing.	Sanity testing is a subset of Acceptance testing.
Smoke testing exercises the entire system from end to end.	Sanity testing exercises only the particular component of the entire system.
Smoke testing is like General Health Check-up.	Sanity Testing is like specialized health check-up.

37. Mention what are the categories of defects?

Data Quality/Database Defects: Deals with improper handling of data in the database.

Critical Functionality Defects: The occurrence of these bugs hampers

the crucial functionality of the application. Examples: - Exceptions

Functionality Defects: These defects affect the functionality of the application.

Security Defects: Application security defects generally involve improper handling of data sent from the user to the application. These defects are the most severe and given highest priority for a fix.

User Interface Defects: As the name suggests, the bugs deal with problems related to UI are usually considered less severe.

38. Advantage of Bugzilla.

- Open source, free bug tracking tool.
- Automatic Duplicate Bug Detection.
- Search option with advanced features.
- File/Modify Bugs By Email.
- Move Bugs Between Installs.
- Multiple Authentication Methods (LDAP, Apache server).
- Time Tracking.
- Automated bug reporting; has an API to interact with system.

39. What are the different Methodologies in Agile Development Model?

- Scrum: SCRUM is an agile development method which concentrates particularly on how to manage tasks within a team based development environment.
- Kanban: Kanban is a very popular framework for development in the agile software development methodology.

40. When to use Usability Testing?

If possible, usability testing can and should be conducted on the current iteration of a product before beginning any new design work, after you've begun the strategy work around a brand new site or app.

41. What is the procedure for GUI Testing?

Graphical User Interface (GUI) testing is the process of testing the system's GUI of the System under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons, and all types of bars – tool bar, menu bar, dialog boxes and windows etc.

- Check all the GUI elements for size, position, width, length and acceptance of characters or numbers. For instance, you must be able to provide inputs to the input fields.
- Check you can execute the intended functionality of the application using the GUI

- Check Error Messages are displayed correctly
- Check for Clear demarcation of different sections on screen
- Check Font used in application is readable
- Check the alignment of the text is proper
- Check the Color of the font and warning messages is aesthetically pleasing
- Check that the images have good clarity
- Check that the images are properly aligned
- Check the positioning of GUI elements for different screen resolution.