

Module-2 (Manual Testing)

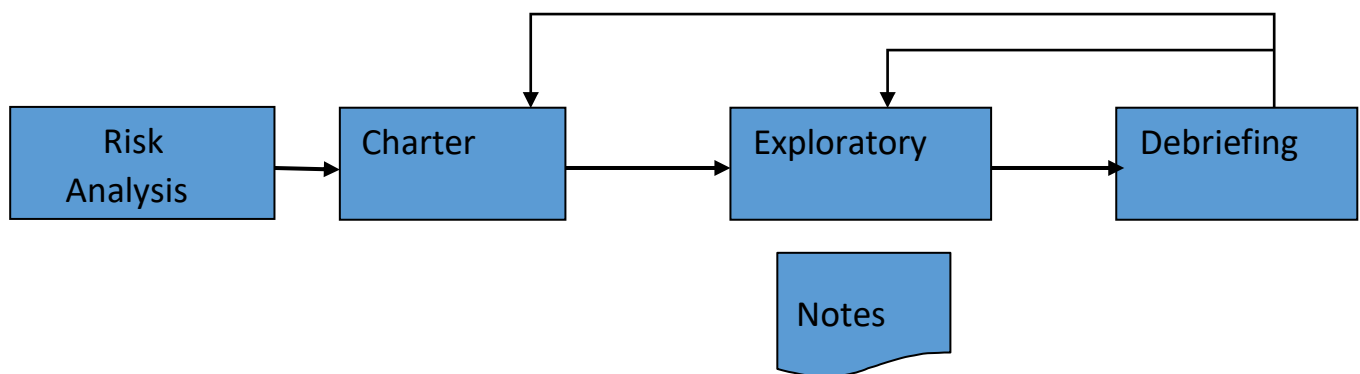
1) What is software testing?

It can be stated as the process of validating and verifying that a software product or application.

2) What is Exploratory Testing?

Exploratory testing is a concurrent process where

- 1) Test design, execution and logging happen simultaneously
- 2) Testing is often not recorded
- 3) Makes use of experience, heuristics and test patterns
- 4) Testing is based on a test charter that may include
 - i. Scope of the testing (in and out)
 - ii. The focus of exploratory testing is more on testing as a “thinking” activity.
 - iii. A brief description of how tests will be performed
 - iv. Expected problems
 - v. Is carried out in time boxed intervals



Though the current trend in testing is to push for automation, exploratory testing is a new way of thinking. Automation has its limits

- Is not random testing but it is Adhoc testing with purpose of find bugs
- Is structured and rigorous
- Is cognitively (thinking) structured as compared to procedural structure of scripted testing. This structure comes from Charter, time boxing etc.
- Is highly teachable and manageable
- Is not a technique but it is an approach. What actions you perform next is governed by what you are doing currently

3) What is traceability matrix?

Traceability Matrix (also known as Requirement Traceability Matrix - RTM) is a table which is used to trace the requirements during the Software development life Cycle. It can be used for forward tracing (i.e., from Requirements to Design or Coding) or backward (i.e., from Coding to Requirements). There are many users defined templates for RTM.

4) What is Boundary value testing?

- a. BVA operates on the basis that experience shows us that errors are most likely to exist at the boundaries between partitions and in doing so incorporates a degree of negative testing into the test design
- b. BVA Test cases are designed to exercise the software on and at either side of boundary values
- c. Find the boundary and then test one value above and below it
- d. Always results in two test cases per boundary for valid inputs and three test cases per boundary for all inputs
- e. inputs should be in the smallest significant values for the boundary (e.g. Boundary of 'a > 10.0' should result in test values of 10.0, 10.1 & 10.2) only applicable for numeric (and date) fields

5) What is Equivalence partitioning testing?

- In this method the input domain data is divided into different equivalence data classes. This method is typically used to reduce the total number of test cases to a finite set of testable test cases, still covering maximum requirements.
- In short it is the process of taking all possible test cases and placing them into classes. One test value is picked from each class while testing.

For example,

Test cases for input box accepting numbers between 1 and 1000 using Equivalence Partitioning.

1. One input data class with all valid inputs. Pick a single value from range 1 to 1000 as a valid test case. If you select other values between 1 and 1000 then result is going to be same. So, one test case for valid input data should be sufficient.
2. Input data class with all values below lower limit. I.e., any value below 1, as an invalid input data test case.

Input data with any value greater than 1000 to represent third invalid input class. Equivalence partitioning uses fewest test cases to cover maximum requirements

6) What determines the level of risk?

The likelihood of an adverse event and the impact of the event.

7) What is Alpha testing?

- It is always performed by the developers at the software development site.
- Alpha Testing is not open to the market and public
- It is always performed in Virtual Environment.
- It is always performed within the organization.
- It is the form of Acceptance Testing.
- Alpha Testing is definitely performed and carried out at the developing organizations location with the involvement of developers.
- It comes under the category of both White Box Testing and Black Box Testing.
- Alpha Testing is always performed at the time of Acceptance Testing when developers test the product and project to check whether it meets the user requirements or not.
- It is always performed at the developer's premises in the absence of the users.
- It is considered as the User Acceptance Testing (UAT) which is done at developer's area.
- Unit testing, integration testing and system testing when combined are known as alpha testing.

8) What is beta testing?

- It is always performed by the customers at their own site.
- It is not performed by Independent Testing Team.
- Beta Testing is always open to the market and public.
- It is performed in Real Time Environment.
- It is always performed outside the organization.
- It is also the form of Acceptance Testing.
- Beta Testing is performed and carried out by users or you can say people at their own locations and site using customer data.
- It is only a kind of Black Box Testing.
- Beta Testing is always performed at the time when software product and project are marketed.
- It is always performed at the user's premises in the absence of the development team.
- Beta testing can be considered "pre-release" testing.
- Pilot Testing is testing to product on real world as well as collect data on the use of product in the classroom.

9) What is component testing?

Component testing is a level of the software testing process where individual units/components of a software/system are tested.

10) What is functional testing?

- Testing based on an analysis of the specification of the functionality of a component or system.
- Functional testing verifies that each function of the software application operates in conformance with the requirement specification.
- This testing mainly involves black box testing and it is not concerned about the source code of the application.
- Each & every functionality of the system is tested by providing appropriate input, verifying the output and comparing the actual results with the expected results.
- This testing involves checking of User Interface, APIs, Database, security, client/ server applications and functionality of the Application under Test. The testing can be done either manually or using automation

11) What is Non-Functional Testing?

- Testing the attributes of a component or system that do not relate to functionality, e.g. reliability, efficiency, usability, interoperability, maintainability and portability

Non-functional testing includes, but is not limited to, performance testing, load testing, stress testing, usability testing, maintainability testing, reliability testing and portability testing.

12) What is 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.

13) What is Adhoc testing?

- Adhoc testing is an informal testing type with an aim to break the system.
- It does not follow any test design techniques to create test cases.
- In fact it does not create test cases altogether!
- This testing is primarily performed if the knowledge of testers in the system under test is very high.
- Testers randomly test the application without any test cases or any business requirement document.
- Adhoc Testing does not follow any structured way of testing and it is randomly done on any part of application.
- Main aim of this testing is to find defects by random checking.
- Adhoc testing can be achieved with the testing technique called Error Guessing.
- Error guessing can be done by the people having enough experience on the system to "guess" the most likely source of errors.

14) 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.

Types of white box testing are,

- 1) test/code coverage
- 2) Statement coverage
- 3) Decision coverage
- 4) Condition coverage

15) What is black box testing? What are the different black box testing techniques?

The technique of testing without having any knowledge of the interior workings of the application is Black Box testing.

Types of black box testing are,

- Equivalence partitioning
- Boundary value analysis
- Decision tables
- State transition testing
- Use-case Testing
- Other Black Box Testing
- Syntax or Pattern Testing

16) Mention what are the categories of defects?

- 1) Data Quality/Database Defects
- 2) Critical Functionality Defects
- 3) Functionality Defects
- 4) Security Defects
- 5) User Interface Defects

17) Mention what big bang testing is?

In Big Bang testing all components or modules are integrated simultaneously, after which everything is tested as a whole.

18) What is the purpose of exit criteria?

Purpose of exit criteria is to define when we STOP testing either at the:

- End of all testing – i.e., product Go Live
- End of phase of testing (e.g., hand over from System Test to UAT)

19) When should "Regression Testing" be performed?

- Change in requirements and code is modified according to the requirement
- New feature is added to the software

20) What are 7 key principles? Explain in detail?

7 key principles are explained in below,

- 1) Testing shows presence of Defects
- 2) Exhaustive Testing is Impossible!
- 3) Early Testing
- 4) Defect Clustering
- 5) The Pesticide Paradox
- 6) Testing is Context Dependent
- 7) Absence of Errors Fallacy

1) Testing shows presence of Defects

- Testing can show that defects are present but cannot prove that there are no defects.
- Testing reduces the probability of undiscovered defects remaining in the software but, even if no defects are found, it is not a proof of correctness.
- As we find more defects, the probability of undiscovered defects remaining in a system reduces.

2) Exhaustive Testing is Impossible!

- Testing everything including all combinations of inputs and preconditions is not possible.
- So, instead of doing the exhaustive testing we can use risks and priorities to focus testing efforts.
- That is we must prioritise our testing effort using a Risk Based Approach.
- We have learned that we cannot test everything (i.e. all combinations of inputs and preconditions).

3) Early Testing

- Testing activities should start as early as possible in the development life cycle.
- Testing activities should start as early as possible in the software or system development life cycle, and should be focused on defined objectives.
- Remember from our Definition of Testing, that Testing doesn't start once the code has been written!

4) Defect clustering

- Defects are not evenly spread in a system
- They are 'clustered'
- In other words, most defects found during testing are usually confined to a small number of modules.
- A small number of modules contain most of the defects discovered during pre-release testing, or are responsible for the most operational failures.

5) Pesticide paradox

- If the same tests are repeated, eventually the same set of test cases will no longer find any new defects.
- To overcome this "pesticide paradox", the test cases need to be regularly reviewed and revised, and new and different tests need to be written to exercise different parts of the software or system to potentially find more defects.
- Therefore, we must learn, create and use new tests based on new techniques to catch new bugs.

6) Testing is Context Dependent

- Different kinds of sites are tested differently.
- Testing is done differently in different contexts.
- For example,
Safety – critical software is tested differently from an e-commerce site.

7) Absence of Errors Fallacy

- If the system built is unusable and does not fulfil the user's needs and expectations, then finding and fixing defects does not help.
- If we build a system and, in doing so, find and fix defects, it doesn't make it a good system.
- Even after defects have been resolved it may still be unusable and/or does not fulfil the users.

20) Difference between QA v/s QC v/s Tester

S. N.	Quality Assurance	Quality Control	Testing
1	Activities which ensure the implementation of processes, procedures and standards in context to verification of developed software and intended requirements.	Activities which ensure the verification of developed software with respect to documented (or not in some cases) requirements.	Activities which ensure the identification of bugs/error/defects in the Software.
2	Focuses on processes and procedures rather than conducting actual testing on the system.	Focuses on actual testing by executing Software with intent to identify bug/defect through implementation of procedures and process.	Focuses on actual testing.
3	Process oriented activities.	Product oriented activities.	Product oriented activities.
4	Preventive activities.	It is a corrective process.	It is a preventive process.
5	It is a subset of Software Test Life Cycle (STLC).	QC can be considered as the subset of Quality Assurance.	Testing is the subset of Quality Control.

21) Difference between Smoke and Sanity?

Smoke Testing	Sanity Testing
Smoke Testing is performed to ascertain that the critical functionalities of the program is 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 testers	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

22) Difference between verification and Validation

Criteria	Verification	Validation
Definition	The process of evaluating work-products (not the actual final product) of a development phase to determine whether they meet the specified requirements for that phase.	The process of evaluating software during or at the end of the development process to determine whether it satisfies specified business requirements.
Objective	To ensure that the product is being built according to the requirements and design specifications. In other words, to ensure that work products meet their specified requirements.	To ensure that the product actually meets the user's needs, and that the specifications were correct in the first place. In other words, to demonstrate that the product fulfills its intended use when placed in its intended environment.
Question	Are we building the product right?	Are we building the right product?
Evaluation Items	Plans, Requirement Specs, Design Specs, Code, Test Cases	The actual product/software.

Activities	<ul style="list-style-type: none"> • Reviews • Walkthroughs • Inspections 	<ul style="list-style-type: none"> • Testing
-------------------	--	---

23) Explain types of Performance testing.

1)load testing

2)stress testing

(1) load testing

- It's 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.
- This testing usually identifies –
 - The maximum operating capacity of an application
 - Determine whether current infrastructure is sufficient to run the application
 - Sustainability of application with respect to peak user load
 - Number of concurrent users that an application can support, and scalability to allow more users to access it.
 - It is a type of non-functional testing. Load testing is commonly used for the Client/Server, Web based applications – both Intranet and Internet.

(2) 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.
- Stress testing is used to test the stability & reliability of the system. This test mainly determines the system on its robustness and error handling under extremely heavy load conditions.
- Stress testing is also known as endurance testing.
- Most prominent use of stress testing is to determine the limit, at which the system or software or hardware breaks.

Examples of stress conditions include:

- The application under testing will be stressed when 5GB data is copied from the website and pasted in notepad.
- Notepad is under stress and gives 'Not Responded' error message

24) What is Error, Defect, Bug, and failure?

- **Error:** A mistake in coding is called error
- **Failure:** Build does not meet the requirements then it is called failure
- **Defect:** Error found by tester is called defect
- **Bug:** Defect accepted by development team then it is called bug

25) Difference between Priority and Severity

Priority	Severity
Priority is a parameter to decide the order in which defects should be fixed.	Severity is a parameter to denote the impact of a particular defect on the software.
Priority means how fast defect has to be fixed.	Severity means how severe defect is affecting the functionality.
It's value changes from time to time.	Its value does not changes from time to time.
Priority is of 3 types: Low, Medium, and High.	Severity is of 5 types: Critical, Major, Moderate, Minor, and Cosmetic.

26) 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 'Bug Life Cycle'.

27) explain the difference between Functional testing and Non-functional testing

Functional Testing	Non-Functional Testing
Functional testing is performed using the functional specification provided by the client and	Non-Functional testing checks the Performance, reliability, scalability and other non-functional aspects
Verifies the system against the functional requirements.	Of the software system.
Functional testing is executed first	Nonfunctional testing should be performed
	after functional testing
Manual testing or automation tools can be used	Using tools will be effective for this testing
for functional testing	
Business requirements are the inputs to functional testing	Performance parameters like speed , scalability are
	Inputs to non-functional testing.
Functional testing describes what the product does	Nonfunctional testing describes how good the
	product works
Easy to do manual testing	Tough to do manual testing
Types of Functional testing are	Types of Nonfunctional testing are
Unit Testing Smoke Testing Sanity Testing Integration Testing White box testing Black Box testing User Acceptance testing Regression Testing	Performance Testing Load Testing Volume Testing Stress Testing Security Testing Installation Testing Penetration Testing Compatibility Testing Migration Testing

28) To create HLR and Test case of (Instagram, Facebook) 1st page

29) What is difference between SDLC and STLC?

SDLC	STLC
SDLC is mainly related to software development.	STLC is mainly related to software testing.
The goal of SDLC is to complete successful development of software.	The goal of STLC is to complete successful testing of software.
It helps in developing good quality software.	It helps in making the software defects free.
SDLC phases are completed before the STLC phases.	STLC phases are performed after the SDLC phases.

30) What is difference between test scenario, test cases, and test script?

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

Test Cases: It is a document that contains the steps that have to be executed; it has been planned earlier.

Test Script: It is written in a programming language and it's a short program used to test part of the functionality of the software system. In other words a written set of steps that should be performed manually.

31) Explain test plan is? What is the information that should be covered?

Test Plan document is usually prepared by the Test Lead or Test Manager and the focus of the document is to describe what to test, how to test, when to test and who will do what test.

Master test plan: A test plan that typically addresses multiple test levels.

Phase test plan: A test plan that typically addresses one test phase.

A test plan will include the following.

- Introduction to the Test Plan document
- Assumptions when testing the application
- List of test cases included in Testing the application
- List of features to be tested

- What sort of Approach to use when testing the software
- List of Deliverables that need to be tested
- The resources allocated for testing the application
- Any Risks involved during the testing process
- A Schedule of tasks and milestones as testing is started

32) What is the different methodologies in agile development model?

Different methodologies of agile are Scrum and Kanban.

33) Explain the difference between authorization and authentication in web testing.

Authentication	Authorization
In the authentication process, the identity of users are checked for providing the access to the system.	While in authorization process, a the person's or user's authorities are checked for accessing the resources.
It is done before the authorization process.	While this process is done after the authentication process.
The authentication credentials can be changed in part as and when required by the user.	The authorization permissions cannot be changed by user as these are granted by the owner of the system and only he/she has the access to change it.
The user authentication is visible at user end.	The user authentication is not visible at user end.
The user authentication is identified with username, password, face recognition, retina scan, fingerprints, etc.	The user authorization is carried out through the access rights to resources by using roles that have been pre-defined.
In the authentication process, users or persons are verified.	While in the process, users or persons are validated.
Authentication determines whether the person is user or not.	While it determines What permission does the user have?

34) What are the common problems faced in web testing?

- Cross browser compability issue
- Application getting slow
- Testing deadline
- Take more time to create test case
- Take more time to create HLR and bug report

35) To create HLR and Test case of web based (WhatsApp web, Instagram)

36)TO create HLR and Test case on this link: <https://artoftesting.com>

37)what is priority?

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

For example: If the company name is misspelled in the home page of the website, then the priority is high and severity is low to fix it.

38)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.

For example: If an application or web page crashes when a remote link is clicked, in this case clicking the remote link by an user is rare but the impact of application crashing is severe. So the severity is high but priority is low.

39) Bug categories are...

- Data Quality/Database Defects
- Critical Functionality Defects
- Functionality Defects
- Security Defects
- User Interface Defects

40)advantages of Bugzilla

- It is an open source widely used bug tracker.
- It is easy in usage and its user interface is understandable for people without technical knowledge.
- It integrates with an emailing system.
- It automates documentation.