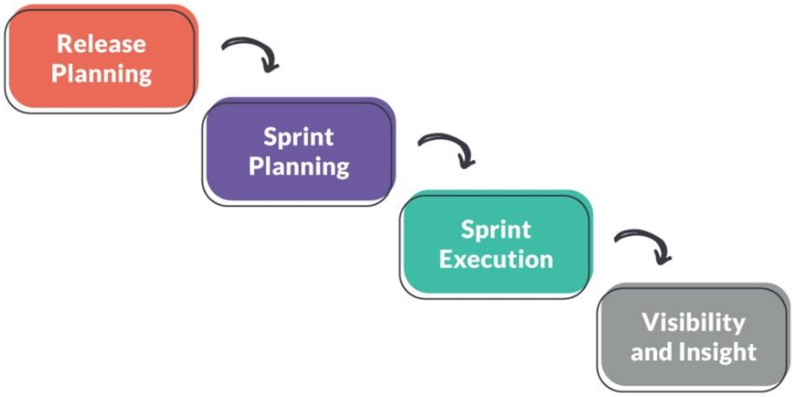
Software Testing Assignment

Module: -2

(1) What is Exploratory Testing?

- Exploratory testing is an unscripted testing approach where testers simultaneously design and execute tests.
- It is like flying visit or thinking kind of activity to explore the application.
- It is a concurrent process for test design or execution simultaneously.
- Is not a technique but it is an approach, what actions you perform next is governed by what you are doing currently?



(2) What is traceability matrix?

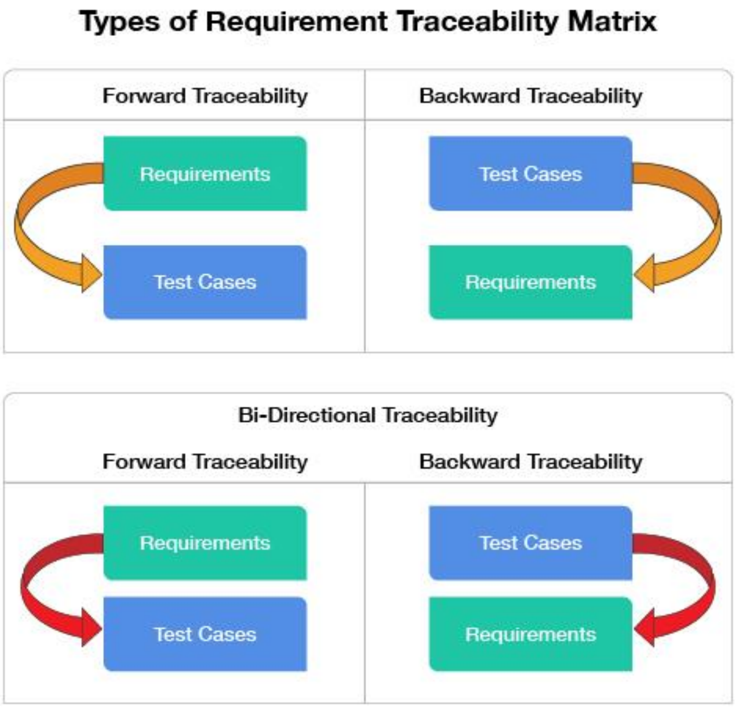
- Test conditions should be able to be linked back to their sources in the test basis, this is known as traceability.
- Matrix means combination of row & Column in table format.
- There are three types of traceability Matrix:

1) Forward Traceability – Mapping of Requirements to test cases

2) Backward Traceability – Mapping of Test cases to Requirements

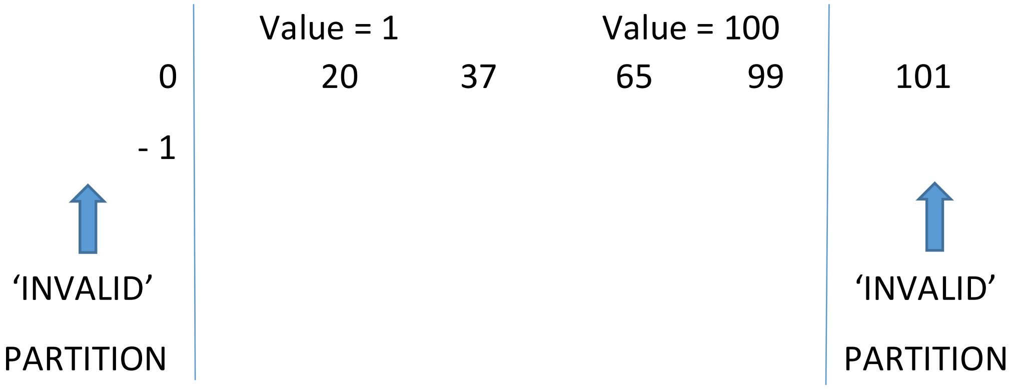
3) Bi-Directional Traceability - A Good Traceability matrix is the

References from test cases to basis documentation and vice versa.



(3) What is Boundary value testing?

- It is better version of equivalent partitioning to validate the boundaries like upper & lower.
- If one value finds a bug, the others probably will too.
- If one doesn't find a bug, the others probably won't either.
- Example:



Value = 1 Value = 100

0 20 37 65 99 101

- 1

'INVALID'

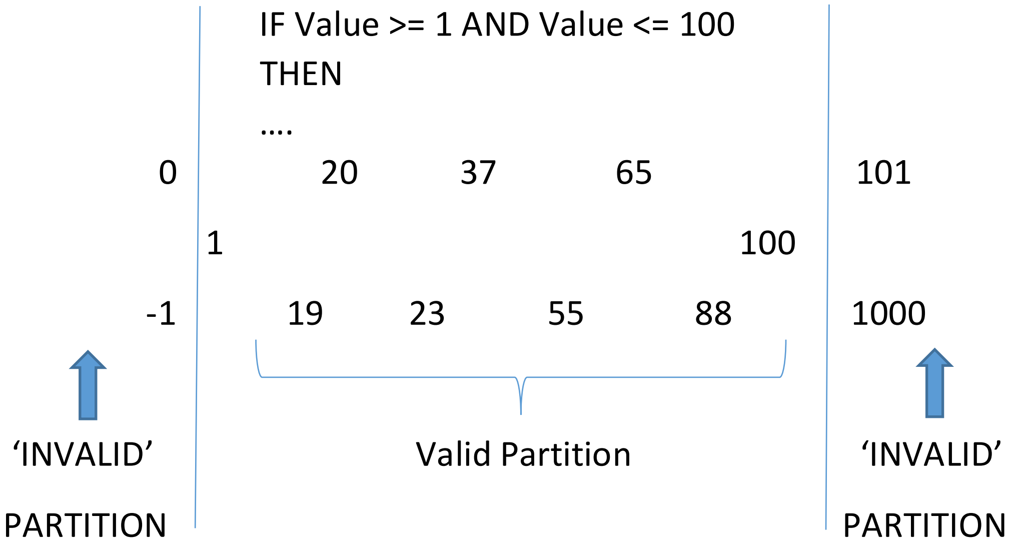
PARTITION

'INVALID'

PARTITION

(4) What is Equivalence partitioning testing?

- Equivalence partitioning is a methodology to design the test cases like to divide range into equivalent partitions and select representative input value to test each partition if representative value will be passes the whole partition will be passed, if the representative value will be fail then the whole partition will be failed.
- For Example: If value is between 1 & 100 then value >= 1 and value <=100.



(5) What is Integration testing?

- Integration testing is a level of the software testing where individual units are combined and tested as a group.
- Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems.
- Integration testing is done by a specific integration tester or test team. Two level of integration testing:
    1. Component integration testing (CIT)
    2. System integration testing (SIT)

(6) What determines the level of risk?

- Risk: A factor that could result in future negative consequences, usually expressed as impact and likelihood.
- When testing does find defects, the Quality of the software system increases when those defects are fixed.
- Process Improvement can prevent those defects reoccurring.
- The Quality of systems can be improved through Lessons learned from previous projects.
- Risks are of two types:
  1) Project Risks
  2) Product Risk

(7) What is Alpha testing?

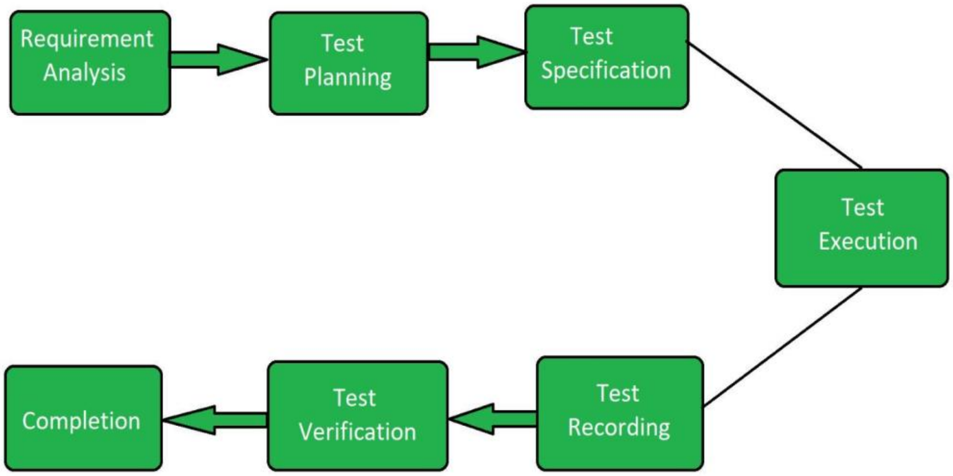
- Alpha testing is definitely performed and carried out at the developing organizations location with the involvement of developers.
- It is always performed by the developers at the software development site.
- Alpha Testing is not open to the market and public.
- It is always performed in Virtual Environment.
- It is the form of Acceptance Testing.

(8) What is beta 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 always performed by the customers at their own site.
- It is performed in Real Time Environment.
- It is also the form of Acceptance Testing.
- Beta Testing is always performed at the time when software product
- and project are marketed.

(9) <u>What is component testing?</u>

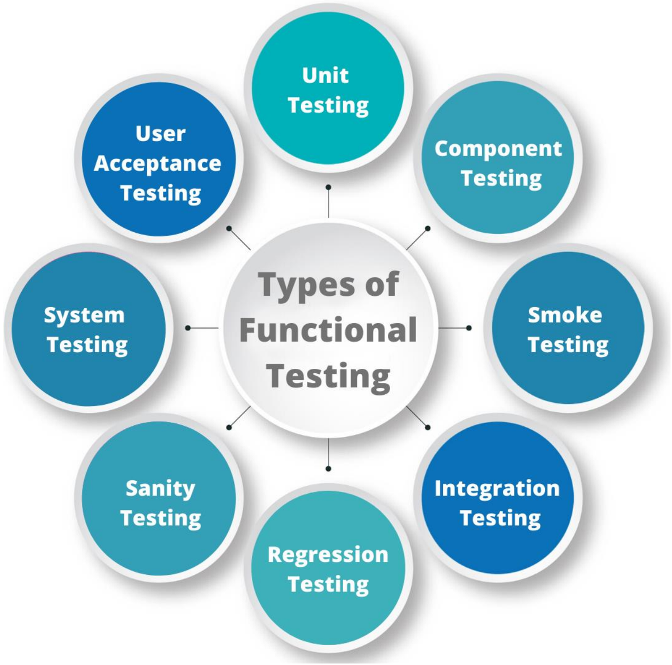
- Component(Unit) testing is a level of software testing process where individual units/components of a software/system are tested.
- Component testing will be performed by the developer only.
- Component testing <u>frameworks, drivers, stubs and mock or fake objects</u> are used to assist in component testing.
- Component testing is the first level of testing and is performed prior to Integration Testing.
- It is also known as unit testing.



(10) <u>What is functional system testing?</u>

- Functional Requirements are very important system requirements in the system design process.
- Functional testing can be possible by manual testing as well as automation tools.
- Easy to do manual testing.
- Like**,** Unit testing, Sanity testing, Integration testing, White box, Black box, User acceptance testing, Regression testing.
- Testing the attributes(features) of the system directly related to the functionality.

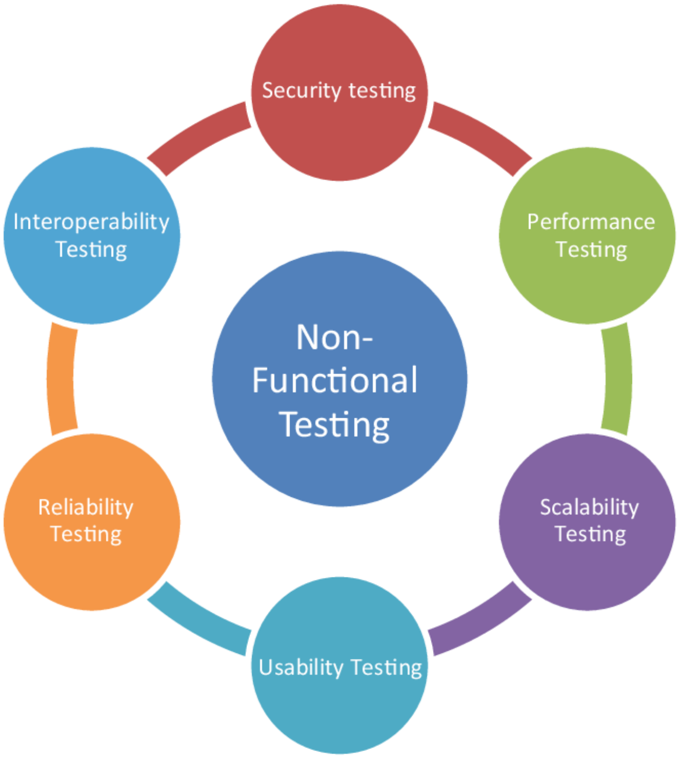
  For example: Login Form: - Login will be always possible by entering username or password otherwise with any warning massage.



(11) What is Non-Functional Testing?

- Non-functional requirements are requirements that specify criteria that can be used to judge the operation of a system, rather than specific behaviors.
- Non-functional testing can be always performed using tools.
- Tough to do manual testing.
- Like, Performance testing, Load testing, Stress testing, Volume testing, Security testing.
- Testing the attributes(features) of the system not directly related to the functionality.

  For example: The application will handle limited number of users so we need to check performance of the applications by applying load.



(12) What is GUI Testing?

- 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.
- Approach of GUI testing:
    1. Manual based testing
    2. Record and reply
    3. Model based testing

(13) What is Adhoc testing?

- Adhoc testing is an informal testing type with an aim to break the system.
- Types of adhoc testing:
    1. Buddy testing
    2. Pair testing
    3. Monkey testing
- 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 archived with the testing technique called error guessing.
- It requires as a tester a lot of experience for this techniques.

(14) What is load testing?

- To check the stability of the application by applying load with response time and check the stability of application.
- It's a performance testing to check system behavior 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.
- Load testing is a kind of performance testing which determines a system's performance under real-life load conditions. This testing helps determine how the application behaves when multiple users access it simultaneously.
- For example: Application will handle 1000 users at every 5 second you have to check

  1000 or >=1000 users with your application.

(15) What is stress Testing?

- It even tests beyond the normal operating point and evaluates how the system works under those extreme conditions.
- Stress testing is to test the system behavior under extreme conditions and is carried out till the system failure.
- Stress testing tries to break the system.
- Stress testing also known as endurance testing.

e.g. Application will handle 1000 users at every 5 seconds.

You have to check 1000 or >= 1000 users with your application.

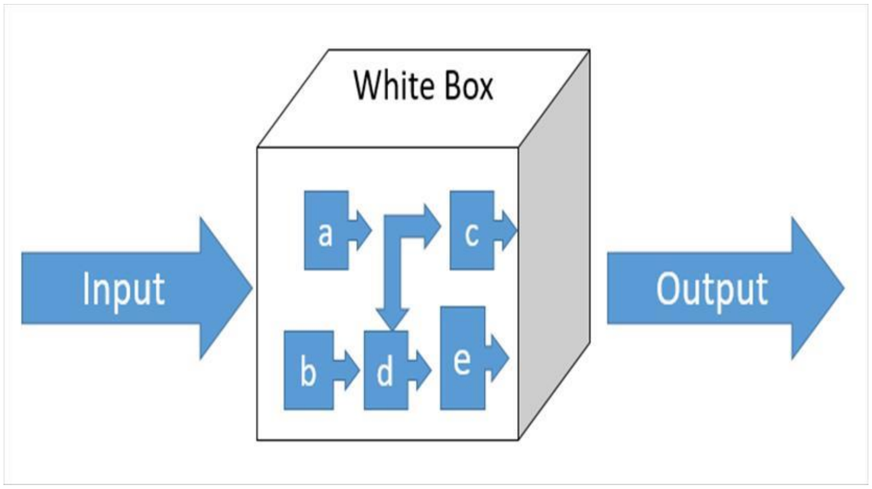
(16) What is white box testing and list the types of white box testing?

- White box testing is also called glass testing or open box testing. In order to perform white box testing on an application, the tester needs to possess knowledge of the internal working of the code.
- White Box Testing based on an analysis of the internal structure of the component or system.
- White box testing is the detailed investigation of internal logic and structure of the code.
- Three types of white box testing coverage:

1. State/Segment coverage

2. Decision/Branch coverage

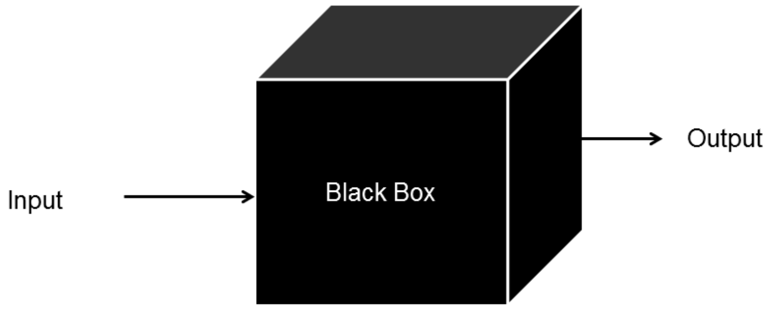
3. Condition coverage



(17) What is black box testing? What are the different black box testing techniques?

- **Black box testing**: A testing can be performed without knowing the internal structure of the application or system.
- The testers have no knowledge of how the system or component is structured inside the box.
- In black-box testing the tester is concentrating on what the software does, not how it does it.
- There are four different black box testing techniques:

1. Equivalence partitioning
2. Boundary value analysis
3. Decision tables
4. State transition testing



Internal behavior of the code is unknown

(18) Mention what are the categories of defects?

- Defect: Commonly refers to several troubles with the software products with its external behaviour or with its internal features.
- Defects categories are:
  1. Performance defect
  2. Design defects
  3. Compatibility defects
  4. Functional defects
  5. Security defects
  6. Arithmetic defects

(19) Mention what big bang testing is?

- In Big Bang integration testing all components or modules is integrated simultaneously, after which everything is tested as a whole.
- The major disadvantage is that in general it is time consuming and difficult to trace the cause of failures because of this late integration.
- Here all component is integrated together at **once**, and then tested.
- Advantages:
  - Convenient for small systems.

- Disadvantages:
    - Fault Localization is difficult.
    - Big Bang testing has the advantage that everything is finished before integration testing starts.

(20) What is the purpose of exit criteria?

- Run out of time.
- Run out of budget.
- The business tells you it went live last night.
- Boss says stop.
- All defects have been fixed.
- When out exit criteria have been met.

(21) When should "Regression Testing" be performed?

- Regression testing should be performed whenever there are changes made to the software, such as new features, bug fixes, or updates.
- 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.
- Its purpose to ensure that the recent modifications haven’t adversely affected the existing functionalities of the system.

(22) Difference between QA v/s QC v/s Tester.

QA(Quality Assurance)	QC(Quality Control)	TESTING
Focuses on processes rather than conducting the actual testing on the software.	Focuses on actual testing by executing software.	Focuses on actual testing
Process oriented activities.	Product oriented activities.	Product oriented activities.
Preventive activities.	It is a corrective process.	It is a preventive process.
It is a subset of Software Test Life Cycle (STLC).	QC is a subset of Quality assurance.	Testing is a subset of Quality control.

(23) Difference between Smoke and Sanity?

Smoke	Sanity
Smoke testing is a 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 object of this testing is to verify that the "**stability**" of the system.	The object of this testing is to verify that the "**rationality**" of the system.
Smoke testing is performed by the developer and tester.	Sanity testing is performed by testers.
Smoke testing is usually documented or scripted.	Sanity testing is usually not documented or unscripted.
Smoke testing is like "General health check-up".	Smoke testing is like "Specialized health check-up".

(24) Explain types of Performance testing.

- performance testing means quality assurance (QA). It involves testing software applications to ensure they will perform well under their expected workload.
- To check the stability of the application by applying load (Designed number of users).
- Stability + response time + applying load = Performance.
- Types of performance testing:

1) Load testing:

- Stability + response time + applying load (Application will with stand with designed num. of users.)
- For example: App will handle 1000 users at every 5 second you have to check
  1000 or <=1000 users with your application.

2) Stress testing:

- Stability + response time + applying load (Application will with stand with designed num. of users.)
- For example: App will handle 1000 users at every 5 second you have to check
  1000 or >=1000 users with your application.

3) Scalability testing:

- Stability + response time + applying load (Application will with stand with designed num. of users.)
- You are checking the performance of the application continue with load until your system will be crashed.
- For example: App will handle 1000 users at every 5 second,
  1500 users: 10 second
  2000 users: 20 second
  ……
  1,00,000 users…. crashed.

4) Volume testing:

- Stability + response time + applying load (Application will with stand with designed num. of users.)
- To check the capacity or volume of database.

5) Endurance(Soak) testing:

- Stability + response time + applying load (Application will with stand with designed num. of users.)
- For example, to check how the system will run continuously.

6) Spike testing:

- Stability + response time + applying load (Application will with stand with designed num. of users.)
- For example, to check extreme increment of decrement of load according to the response time.

(25) What is Error, Defect, Bug and failure?

- **Error**: A mistake in coding is called Error.
- **Defect**: Any error found by tester is called Defect.
- **Bug**: Any defect accepted by developer team is called Bug.
- **Failure**: Build does not meet requirements is called Failure.

(26) 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 do note the impact of a particular defect on the software.
Its value is subjective.	Its value is objective.
Its value change from time to time.	Its value does not change from time to time.
It is driven by business value.	It is driven by functionality.

(27) What is Bug Life Cycle?

- A computer bug is an error, flaw, mistake, failure or fault in a computer program that prevents it from working correctly or produces an incorrect result. Bugs arise from mistakes.
- When a bug is discovered, it goes through several states and eventually reaches one of the terminal states, where it becomes inactive and closed.
- Error made by people in either a program's source code or its design.
- The process by which the defect moves through the life cycle is depicted next slide.

(28) Explain the difference between Functional testing and Non Functional testing.

Functional Testing	**Non-Functional Testing**
Executed first.	Executed after functional testing.
To check the features that are directly related to the functionality. Ex. Click on "add to cart" button to add the items to the cart.	To check the features those are not directly related to the functionality but will mandatory (important) to check.
Functional testing can be possible by manual testing as well as automation tools.	Non-functional testing can be always performed using tools.
Business requirement are inputs for the functional testing.	Performance parameter like speed, load are inputs for the non-functional testing.
Easy to do manual testing. **Ex.** Unit testing, Sanity testing, Integration testing, White box, Black box, User acceptance testing, Regression testing.	Tough to do manual testing. **Ex.** Performance testing, Load testing, Stress testing, Volume testing, Security testing.

(29) To create HLR & Test Case of (Instagram, Facebook) only first page.

HLR_Instagram	Click here
Testcase for Instagram	Click here

HLR_Facebook_First page	Click here
Testcase for Facebook First page	Click here

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

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

(31) What is the difference between test scenarios, test cases, and test script?

Test Scenario	Test Cases	Test Script
Any functionality that can be tested.	Test cases are set of input and output given to system.	A **set of sequential** instruction to execute business function.
It provides an overall view of what needs to be tested without going into detailed steps.	It outlines the specific steps to be taken and expected outcomes for a particular test scenario.	It provides the detailed steps, including input data and expected results, that need to be executed during automated or manual testing.
It focuses on more what to test than how to test.	A complete an phases on what to test and how to test.	Test script is an automatic approach of software testing.

Test scenario are derived from test artefacts like SRS.	Test cases are low level action.	Test script is done by scripting format.

(32) Explain what Test Plan is? What is the information that should be covered?

- A document describing the scope, approach, resources and test activities. It serves as a plan for the testing process and provides a comprehensive overview of how testing will be conducted to ensure the quality and reliability of the software.
- Key components of a test plan templates:

1. Test planning process:

A document describing the scope, approach, resources and test activities.

2. Test plan & strategy:

All project requires a set of plans and strategy which defines how to testing. Test policy, master test plan, functional test plan, system integration plan, UAT test plan.

3. Test planning factors:

- Scope of the testing

- Testing objectives

- Project risk (e.g. Business, technical)

- Availability or resources.

4. Test planning activities:

- Decide approach: integrating and cording the testing activities into the SDLC activities.

- Decide: Who will be test, what will be tested, how it will be tested, when will be tested.

5. Exit criteria:

- Run out of time.

- Run out of budget.

- Boss says stop.

- All defects have been fixed.

(33) What is priority?

- Priority is term that defines how fast we need to fix a defect.
- Priority is a parameter to decide the order in which defects should be fixed.
- Priority relates to the scheduling of defects to resolve them in software.
- The value of priority is subjective.
- The value of priority changes from time to time.
- The product manager basically decides a defects priority level.

(34) What is severity?

- Severity is a term that denotes how severely a defect can affect the functionality of the software.
- Severity is basically a parameter that denotes the total impact of a given defect on any software.
- Severity relates to the standards of quality.
- The value of severity is objective.
- The value of severity changes continually from time to time.

(35) Bug categories are…

- Bug: defect accepted by development team is called bug.
- Bug categories are:
    - Functional bugs
    - Performance bugs
    - Usability bugs
    - Compatibility bugs
    - Unit-level bugs

- Security bugs
- Syntax bugs

(36) Difference between priority and severity.

Priority	Severity
Priority is term that defines how fast we need to fix a defect.	Severity is a term that denotes how severely a defect can affect the functionality of the software.
The value of priority is subjective.	The value of priority is objective.
The product manager basically decides a defects priority level.	The testing engineer basically decides a defects severity level.
Priority is basically a parameter that decides the order in which we should fix the defects.	Priority is basically a parameter that decides the total impact of a given defect on any software.
There are three types of priorities: High, medium and low.	There are five types of severity: Cosmetic, minor, moderate and critical.

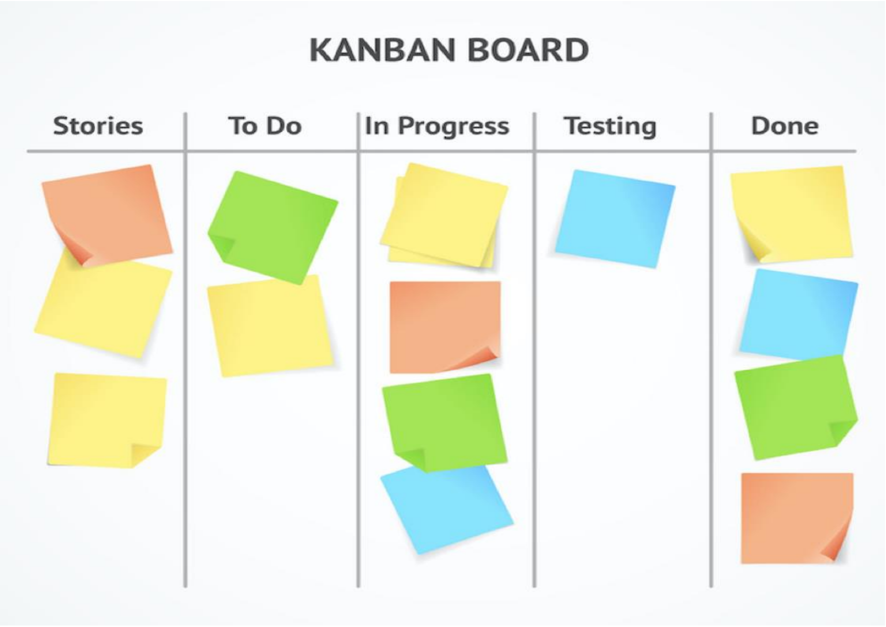
(37) What are the different Methodologies in Agile Development Model?

- Agile is based on Iterative – Incremental approach for project management and software development that helps a team to deliver faster values to their customers. Requirements, plans, and results are evaluated continuously so teams have a natural mechanism for responding to change quickly.

- Manifesto for Agile Software Development:
  Individuals and interactions over processes and tools
  Working software over comprehensive documentation
  Customer collaboration over contract negotiation
  Responding to change over following a plan

- Different methodologies in agile development model is:
  1. Scrum
  2. Kanban
  3. XP
- Scrum:
- Scrum is the most popular Agile framework, defined by the Scrum Guide.
- Scrum is a framework through which we build software product by following Agile principles.
- Scrum includes group of people called a scrum team. Normally contains 5 to 9 members.
- Roles of scrum:
  1. Product Owner
  2.Scrum Master
  3.Scrum Team
- Artifacts of scrum:
  1. Product Backlog
  2. Sprint Backlog
  3. Burn-down Charts
- Ceremonies of scrum:
  1. Spint Planning
  2. Sprint Review
  3. Sprint Retrospective
  4.Daily Scrum Meeting

- Kanban:
- Kanban is a very popular framework for development in the agile software development methodology.
- It provides a transparent way of visualizing the tasks and work capacity of a team.
- The Kanban board has columns and story cards.



(38) Explain the difference between Authorization and Authentication in Web testing. What are the common problems faced in Web testing?

Authorization	Authentication
While in authorization process the users authorization are checked for accessing the resources.	In the authentication process the identify of users are checked for providing the access to the system.
The user authorization is not visible at the user end.	The user authentication is visible at user end.
While in this process users are validated.	In the authentication process users are verified.
While it needs the users security levels.	It needs usually the users login details.
While this process is done after the authentication process.	It is done before the authorization process.

(39) To create HLR & TestCase of WebBased (WhatsApp web, Instagram) 1. WhatsApp Web : https://web.whatsapp.com/.

HLR_Whatsapp web	Click here
Testcase for Whatsapp web	Click here

HLR_Instagram_Web	Click here
Testcase for Instagram Web	Click here

(40) To create HLR and testCase on this link. https://artoftesting.com/

HLR_Artoftesting	Click here
Testcase for Artoftesting	Click here

(41) Write a scenario of only whatsapp chat messages.

Scenario_Whatsapp chat message	Click here

(42) Write a Scenario of Pen.

Scenario_Pen	Click here

(43) Write a Scenario of Pen Stand.

Scenario_Pen stand	Click here

(44) Write a Scenario of Door.

Scenario_Door	Click here

(45) Write a Scenario of ATM.

Scenario_ATM	Click here

(46) When to used Usablity Testing?

- To check the user friendliness of the application.
- For example: Radio button, text box, drop-down list, all the controls should be alien properly. If drop-down list is available, then user can select the particular item without typing.

(47) What is the procedure for GUI Testing?

- GUI (Graphical User Interface)
- GUI testing involves checking the screens with the controls like menus, buttons, icons and all types of bars, tool-bar, menu-bar, dialogue boxes and windows etc.
- Approach of GUI testing type:
  1. Manual based testing
  2. Record and Reply
  3. Model based testing

(48) Write a scenario of Microwave Oven.

Scenario_Microwave oven	Click here

(49) Write a scenario of Coffee vending Machine.

Scenario_Coffee machine	Click here

(50) Write a scenario of chair.

Scenario_Chair	Click here

(51) To Create Scenario (Positive & Negative) gmail.

Scenario_Gmail	Click here

(52) To create Scenario (Positive & Negative) Online shopping to buy product (flipkart).

Scenario_Flipcart	Click here

(53) Write a Scenario of Wrist Watch.

Scenario_Watch	Click here

(54) Write a Scenario of Lift(Elevator).

Scenario_Lift	Click here

(55) Write a Scenario of whatsapp Group (generate group).

Scenario_Whatsapp Group	Click here

(56) Write a Scenario of Whatsapp payment.

Scenario_Whatsapp Payment	Click here