ASSIGNMENT
MODULE 2

1. **What is Traceability matrix?**
   A Traceability Matrix is a document that links requirements to their corresponding test cases to ensure all requirements are tested.

2. **What is Boundary value testing?**
   Boundary value testing checks the smallest and largest possible inputs to find errors. It helps make sure the software works correctly at the edges of valid values.

3. **What is Equivalence partitioning testing?**
   Equivalence partitioning testing groups similar inputs together to reduce testing effort. If one input in a group works, the others are likely to work tool.

4. **What is Integration testing?**
   Integration Testing is the process of testing how different modules or components of a system work together after being combined.

5. **What determines the level of risk?**
   Risk level is decided by how likely something is to go wrong and how big the damage would be if it does.

6. **What is Alpha testing?**
   Alpha testing is done by a tester before handing over the customer.

7. **What is Beta testing?**
   Beta testing is done by customer at their own site.

8. **What is component Testing?**
   Component Testing is testing done on individual parts or units of a software.

9. **What is Functional System Testing?**
   Functional system testing is tested against the functional requirement and specifications. It ensures that all functions perform correctly for users.

10. What is Non-Functional Testing?

Non-Functional Testing checks how well a system performs rather than what it does — like speed, security, usability, and reliability.

11. What is GUI Testing?

GUI Testing (Graphical User Interface Testing) checks if the software's visual elements like buttons, menus, and forms work and look as expected.

12. What is Adhoc testing?

Adhoc Testing is an informal testing method where testers explore the application randomly without any planned test cases to find unexpected bugs.

13. What is load testing?

Load testing checks how well a software system handles many users or heavy tasks at the same time. It helps find performance issues before the system goes live.

14. What is stress Testing?

Stress testing checks how a software system performs under extreme conditions, like heavy traffic or high data load. It helps find weaknesses and ensures the system doesn't crash when pushed beyond normal limits.

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

White Box Testing is a testing method where the tester knows the internal code and logic and tests the software's internal workings.

Types of White-box Testing:

(a) Unit Testing
(b) Integration Testing
(c) Regression Testing

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

Black Box Testing is a testing method where the tester checks the software's functionality without knowing the internal code or structure.

Techniques:

(a) Equivalence Partitioning
(b) Boundary Value Analysis
(c) Decision Table Testing
(d) State Transition Testing
(e) Exploratory Testing
(f) Use Case Testing

17. Mention what are the categories of defects?

i. **Functional Defects**
ii. **Performance Defects**
iii. **Usability Defects**
iv. **Security Defects**
v. **Compatibility Defects**
vi. **Interface Defects**
vii. **Documentation Defects**

18. Mention what Big-bang testing is?

Big Bang Testing is an integration testing approach where all components or modules are combined and tested at once, rather than in steps.

19. What is the purpose of exit criteria?

The purpose of exit criteria is to define the conditions that must be met to stop testing, ensuring the product is ready for release or the next phase.

20. When should "Regression Testing" be performed?

Regression Testing is performed after changes in the code to make sure existing features still work properly.

21. What is 7 key principles? Explain in detail?

1. **Testing Show 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 error fallacy**

1. **Testing Shows presence of defects-** Testing reveals bugs but can't prove a system is completely error-free.

2. **Exhaustive Testing is impossible-** It's not feasible to test all inputs and scenarios.

3. **Early Testing-** Detecting defects early reduces cost, efforts and time.

4. **Defect Clustering-** Most bugs are found in a few modules.

5. **The Pestiside Paradox-** Repeating the same tests won't find new bugs.

6. **Testing Is context dependent-** Testing approach varies based on the software type.

7. **Absence of error fallacy-** A bug-free system may still fail if it doesn't meet user needs.

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

Aspect	QA	QC	Tester
Focus	Processes and procedures	Product quality	Finding bugs by testing
Goal	Prevent Defects	Detect defects	Execute test cases
When	Throughout development	After product development	During testing phase
Who	QA team, process owners	QC team, inspectors	Testers
Activities	Audits, process improvements	Inspections, reviews, testing	Writing and running tests
Approach	Proactive	Reactive	Reactive
Example	Defining, Standard Training	Checking Test results	Manual or Automated

23.Difference between Smoke and Sanity?

Smoke Testing	Sanity Testing
Checks if main parts work	Checks if specific fixes work
Done right after a new build	Done after small changes
Tests a little bit of everything	Tests only the changed parts
Makes sure build is good to test	Makes sure changes didn't break anything

24.Difference between verification and Validation

Verification	Validation
Checks if the product is built right (correctly)	Checks if the right product is built (meets needs)
Focuses on process and design	Focuses on actual product
Done during development phases	Done after development (testing)
Static technique (reviews, inspections)	Dynamic technique (testing)
Ensures the product follows specs	Ensures the product meets user requirements

25.Explain types of Performance testing.

1. **Load Testing-** Checks system performance under expected user loads.
2. **Stress Testing-** Tests system behavior under extreme, beyond-limit conditions.
3. **Endurance Testing-** Measures system stability over a long period under normal load.
4. **Spike Testing-** Tests system response to sudden large changes in load
5. **Volume Testing-** Checks system's ability to scale with increasing load
6. **Scalability Testing-** Tests system performance with a large amount of data

26.What is Error, Defect, Bug and failure?

Error- A mistake made by a developer while writing code or design.
Defects- A flaw or issue found in the software during development.
Bug- Another name for a defect, usually found during testing.
Failure- When the software doesn't work as expected during execution.

27. Difference between Priority and Severity

Priority	Severity
Tells how soon the bug should be fixed	Tells how serious the bug is
Set by project manager or client	Set by tester or QA
Based on business needs	Based on technical impact
Example: Logo is missing on homepage (High Priority, Low Severity)	App crashes on rare input (Low Priority, High Severity)

28. What is Bug Life Cycle?

Bug Life Cycle is the step-by-step process a bug goes through—from being found, fixed, tested, and finally closed.

29. Explain the difference between Functional testing and NonFunctional testing.

Functional Testing	Non-Functional Testing
Functional testing is verify the operation of the software	Non-Functional testing is verify the behavior of software
It is based on requirement of client	It is based on expectation of client
It helps to enhance the behavior of the software	It is help to improve the performance
It is easy to execute manually	It is hard to execute manually
It is test what the product does	It is describe how the product does

30. What is Exploratory Testing?

Exploratory Testing is a testing method where testers explore the application freely without predefined test cases to find bugs using their understanding and creativity.

31. What is the difference between SDLC and STLC?

Point	SDLC	STLC
What it means	Steps to develop software	Steps to test software
Main Work	Making the software	Checking the software
Starts When	New software project starts	After requirement is ready
Ends When	Software is delivered to user	Testing is completed
Who works	Developers, designers, testers	Only testers (QA team)
Output	Working software	Tested software
Covers	Planning, designing, coding, testing, deployment	Test planning, writing test cases, executing tests

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

Term	Meaning	Example
Test Scenario	A high-level idea of what to test	Verify login functionality
Test Case	Step-by-step instructions to test one specific condition	1. Open login page2. Enter valid username & password3. Click login4. Check home page is displayed
Test Script	A set of instructions written in code or automation tool to run the test	Selenium script to test login with code

Test Scenario = What to test
Test Case = How to test (with steps, data, and expected result)
Test Script = Automated code to test the case

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

A Test Plan is a document that explains:
"What to test, how to test, who will test, and when to test."

Part	Meaning
Test Plan ID	Unique name or number for the plan
Introduction	Short summary of what the project is
Scope	What will be tested and what will not be tested
Test Objectives	What we want to check or verify
Test Strategy	How testing will be done (manual or automation)
Test Environment	Tools, systems, and setup needed for testing
Test Deliverables	What documents will be given (like test cases, reports)
Roles & Responsibilities	Who will do what in the testing team
Schedule	When testing will start and end
Entry Criteria	When testing can begin
Exit Criteria	When testing is complete
Risks	What problems might happen and how to handle them
Tools Used	Testing tools like Selenium, JIRA, etc.
Approvals	Who will check and approve the plan

34. What is priority?

It shows the importance from a customer point of view.

35. What is Severity?

It shows the how serious the bug is and how much it affects the working of the software.

36. Bug categories are…

Bug Category: Security, Database, Functionality (Critical/General), UI

37. Advantage of Bugzila.

Advantage	Explanation
Free & Open Source	No cost, easy to use and customize
Tracks Bugs Easily	Helps testers report and manage bugs in one place
Email Notifications	Sends updates when a bug is added or fixed
Search & Filter Bugs	You can find bugs quickly using filters
Reports & Charts	Generates reports to see bug status and progress
Multiple Users Support	Many team members can work together
Secure Access	You can control who sees and edits what

38. Difference between priority and severity.

Point	Priority	Severity
Meaning	How **quickly** the bug should be fixed	How **badly** the bug affects the app
Focus	**Urgency** of the bug	**Impact** of the bug
Decided By	Project manager or client	Tester or QA
Fix First?	High priority bug is fixed **first**	High severity bug may not be fixed immediately
Example 1	Spelling mistake → **High priority**, Low severity (client wants it fixed quickly)	App crash → **High severity**, maybe low priority (if not urgent for users)
Example 2	Payment button not working → High priority	App stops working → High severity

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

1. **Scrum**
2. **Kanban**
3. **Extreme Programming (XP)**
4. **Lean**
5. **Crystal**
6. **Dynamic Systems Development Method (DSDM)**
7. **Feature-Driven Development (FDD)**
8. **Agile Unified Process (AUP)**
9. **Disciplined Agile Delivery (DAD)**
10. **Scaled Agile Framework (SAFe)**

40. Explain the difference between Authorization and Authentication in Web testing. what are the common problems faced in Web testing?

Point	Authentication	Authorization
Meaning	Confirms **who you are**	Confirms **what you can do**
Checks	Identity of the user	Permissions or access rights
Happens When?	First – at the time of **login**	After login – when accessing pages/features
Example	Entering username & password to log in	Only admins can delete users
Result	User is **verified**	User is **allowed or denied** access
Tested How?	Test login with valid/invalid credentials	Test access to pages based on roles

Here are some common challenges testers face while testing websites:

Problem	Simple Explanation
1. Browser Compatibility Issues	Website works in Chrome but not in Firefox or Safari
2. Page Load Speed	Website takes too long to open
3. Broken Links	Clicking a link shows “404 Page Not Found”
4. UI Issues	Layout breaks on different screens or devices
5. Mobile Responsiveness	Website does not display properly on mobile phones
6. Security Issues	Login is not secure, or data is not protected properly
7. Form Validation Problems	Forms accept incorrect input or don't show error messages
8. Session Management Errors	User stays logged in even after logging out
9. Integration Issues	Payment gateway, chatbots, or other tools don’t work well
10. Inconsistent Behavior	Same action gives different results at different times
11. Accessibility Problems	Website not usable by people with disabilities
12. Database Errors	Data not saved correctly or not shown properly