**Module–2(Manual Testing)**

**1. What is Exploratory Testing?**  
Exploratory Testing is a type of testing where the tester actively explores the application without predefined test cases. It focuses on learning the system, designing test cases, and executing them simultaneously.

**2. What is Traceability Matrix?**  
Traceability Matrix is a document that maps and traces user requirements with test cases. It ensures that all requirements are covered by test cases and helps in tracking test coverage.

**3. What is Boundary Value Testing?**  
Boundary Value Testing is a black box testing technique that focuses on testing the boundary values of input fields. For example, if an input field accepts values between 1 to 100, test cases are created for 0, 1, 100, and 101.

**4. What is Equivalence Partitioning Testing?**  
Equivalence Partitioning divides input data into valid and invalid partitions and tests only one value from each partition. This helps in reducing the number of test cases while maintaining coverage.

**5. What is Integration Testing?**  
Integration Testing checks the data flow and interaction between integrated modules of the software. It ensures that modules work together as expected.

**6. What Determines the Level of Risk?**  
Risk level is determined by the probability of failure and the impact of that failure. Factors include complexity, business impact, and past defect history.

**7. What is Alpha Testing?**  
Alpha Testing is performed by the internal team (developers or testers) at the end of development to identify bugs before releasing the product to external users.

**8. What is Beta Testing?**  
Beta Testing is conducted by real users in a real environment before the final release. It helps in gathering feedback and identifying issues not caught during alpha testing.

**9. What is Component Testing?**  
Component Testing, also known as Unit Testing, focuses on testing individual components or modules of the software in isolation.

**10. What is Functional System Testing?**  
Functional System Testing checks the system's functionality as a whole based on the specified requirements. It is a black box testing technique.

**11. What is Non-Functional Testing?**  
Non-Functional Testing evaluates attributes like performance, usability, reliability, and scalability of the software rather than specific behaviors.

**12. What is GUI Testing?**  
GUI Testing involves testing the graphical user interface of the application to ensure that it meets design specifications and is user-friendly.

**13. What is Adhoc Testing?**  
Adhoc Testing is an informal testing technique where testers randomly test the application without any test cases or documentation.

**14. What is Load Testing?**  
Load Testing evaluates the application’s performance under expected user loads to identify performance bottlenecks.

**15. What is Stress Testing?**  
Stress Testing determines the stability and robustness of the system under extreme conditions or load beyond normal expectations.

**16. What is White Box Testing and List the Types of White Box Testing?**  
White Box Testing involves testing internal structures or workings of an application. Types include:

* Unit Testing
* Integration Testing
* Code Coverage Testing
* Path Testing

**17. What is Black Box Testing? What are the Different Black Box Testing Techniques?**  
Black Box Testing tests the software without knowledge of internal code. Techniques include:

* Equivalence Partitioning
* Boundary Value Analysis
* Decision Table Testing
* State Transition Testing

**18. Mention What are the Categories of Defects?**  
Categories of defects include:

* Functional Defects
* Performance Defects
* Usability Defects
* Compatibility Defects
* Security Defects

**19. Mention What Big Bang Testing is?**  
Big Bang Testing is an integration testing approach where all modules are integrated and tested at once, rather than one by one.

**20. What is the Purpose of Exit Criteria?**  
Exit Criteria define the conditions that must be met to conclude testing. It includes completion of test cases, no critical bugs, and successful regression testing.

**21. When Should Regression Testing be Performed?**  
Regression Testing is performed after code changes to ensure that the existing functionalities are not broken.

**22. What is 7 Key Principles? Explain in Detail?**  
The 7 key principles of testing are:

1. Testing shows presence of defects
2. Exhaustive testing is impossible
3. Early testing saves time and money
4. Defect clustering
5. Pesticide paradox
6. Testing is context dependent
7. Absence of errors fallacy

**23. Difference Between QA vs QC vs Tester**

* QA (Quality Assurance): Focuses on process improvement and defect prevention.
* QC (Quality Control): Focuses on product quality and defect detection.
* Tester: Performs the testing tasks to identify defects in the application.

**24. Difference Between Smoke and Sanity Testing**

* Smoke Testing: Basic tests to check if the major functionalities are working.
* Sanity Testing: Narrow and deep testing to verify specific functionality after changes.

**25. Difference Between Verification and Validation**

* Verification: Checks whether the product is built correctly (process-oriented).
* Validation: Checks whether the right product is built (product-oriented).

**26. Explain Types of Performance Testing**

* Load Testing
* Stress Testing
* Endurance Testing
* Spike Testing
* Volume Testing

**27. What is Error, Defect, Bug, and Failure?**

* Error: Human mistake in coding.
* Defect: Deviation from requirements.
* Bug: Synonym for defect.
* Failure: When the system does not perform as expected.

**28. Difference Between Priority and Severity**

* Priority: How soon the defect should be fixed.
* Severity: The impact of the defect on the system.

**29. What is Bug Life Cycle?**  
Bug Life Cycle includes:

* New
* Assigned
* Open
* Fixed
* Retest
* Verified
* Closed
* Reopen (if failed again)

**30. Explain the Difference Between Functional Testing and Non-Functional Testing**

**Functional Testing:**

* It verifies that the application works as expected based on functional requirements.
* Focuses on *what* the system does.
* Examples: Login, Signup, Add to Cart, Payment.
* It checks user interactions, data processing, and output.
* Tools: Selenium, QTP, Test Complete.
* Performed before non-functional testing.

**Non-Functional Testing:**

* It verifies non-functional aspects like performance, usability, load, and security.
* Focuses on *how* the system performs.
* Examples: Load Testing, Stress Testing, Security Testing.
* It checks the speed, scalability, and reliability of the application.
* Tools: JMeter, LoadRunner, Apache Benchmark.
* Performed after functional testing.

**26. To create HLR & Testcase of 1) (Instagram, Facebook) first page and chat functionality**

<https://docs.google.com/spreadsheets/d/1m3z8qzYpRZMRXVQArNPvdTorSRqPZOvRXsz6I2JpOE0/edit?usp=sharing>.

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

* **STLC** focuses on the *testing phase* of software, starting from test planning to test closure.
* **SDLC** covers the *entire development life cycle*, from requirement gathering, design, coding, testing, deployment, and maintenance.

**28. What is the difference between Test Scenarios, Test Cases, and Test Scripts?**

* **Test Scenario**: High-level functionality to be tested (e.g., "Verify Login Functionality").
* **Test Case**: Detailed steps, input, expected result for testing a scenario.
* **Test Script**: Automated script/code used to perform the test cases.

**29. Explain what Test Plan is? What is the information that should be covered.**  
A **Test Plan** is a document that outlines the strategy, scope, objectives, resources, and schedule of testing activities.  
It includes:

* Test objectives
* Test scope
* Test items
* Test deliverables
* Environment requirements
* Roles and responsibilities
* Entry and exit criteria
* Schedule
* Risk and mitigation

**30. What is Priority?**  
Priority defines **how urgently a defect should be fixed**.

* High Priority: Business-critical defects.
* Low Priority: Minor issues that can be fixed later.

**31. What is Severity?**  
Severity defines the **impact of the defect on the system’s functionality**.

* High Severity: Application crash or data loss.
* Low Severity: UI misalignment, spelling error.

**32. Bug Categories are...**

* Functional Bugs
* Performance Bugs
* UI/UX Bugs
* Compatibility Bugs
* Security Bugs
* Logical Bugs
* Boundary Value Bugs
* Integration Bugs

**33. Advantage of Bugzilla:**

* Open-source bug tracking tool
* Email notifications
* User-friendly interface
* Customizable workflows
* Powerful search and reporting
* Integration with other tools (like Git, Jenkins)

**34. Difference between Priority and Severity:**

**Priority** refers to how urgently a defect should be fixed, while **Severity** refers to the impact of the defect on the functionality. Priority is decided by the Project Manager or Business Analyst based on business needs, whereas Severity is decided by the tester based on how badly the system is affected. For example, a login page not working is both high priority and high severity. A minor spelling mistake on the homepage may be low severity but high priority if it's on a client-facing page.

**35. What are the different Methodologies in Agile Development Model?**

* **Scrum**
* **Kanban**
* **Extreme Programming (XP)**
* **Lean Software Development**
* **Crystal Methodology**
* **Feature-Driven Development (FDD)**  
  Each has its own way of handling iteration, communication, and feedback.

**36. Explain the difference between Authorization and Authentication in Web Testing. What are the common problems faced in Web Testing?**

* **Authentication**: Verifying the identity of the user (e.g., login username/password).
* **Authorization**: Checking what actions the authenticated user is allowed to perform (e.g., access admin page).

**Common problems in Web Testing:**

* Broken links
* Cross-browser compatibility issues
* UI alignment problems
* Session management bugs
* Insecure data transfer
* Performance issues on slow networks
* Responsive design errors

**37. To create HLR & Testcase of Web Based (WhatsApp web)**

**38. Write a Scenario of Only WhatsApp Chat Messages**

**Scenario Title: Verify functionality of sending and receiving messages in WhatsApp**

* **Positive Scenarios:**
  + **User can send a text message to a contact.**
  + **User receives a message from a contact in real-time.**
  + **User can send multimedia messages (images, videos, audio).**
  + **User can see "delivered" and "read" status for messages.**
  + **User can send messages to a group and all members receive the messages.**
* **Negative Scenarios:**
  + **Message fails to send due to no internet connectivity.**
  + **Message delivery fails when the recipient is offline.**
  + **User cannot send a message to a contact that is blocked.**
  + **Multimedia message fails to upload.**
  + **App crashes when trying to send a message with large attachments.**

**39. Write a Scenario of Pen**

**Scenario Title: Verify the functionality of a ballpoint pen**

* **Positive Scenarios:**
  + **Pen writes smoothly without skipping or smudging.**
  + **Ink flow is consistent.**
  + **Pen writes on all standard paper types.**
  + **Pen cap securely fits to prevent ink drying.**
  + **Pen is comfortable to hold for long periods.**
* **Negative Scenarios:**
  + **Ink leaks from the pen.**
  + **Pen runs out of ink too quickly.**
  + **Ink doesn’t dry, causing smudging.**
  + **Cap does not securely close.**
  + **Pen tip gets clogged and stops writing.**

**40. Write a Scenario of Pen Stand**

**Scenario Title: Verify the functionality of a pen stand**

* **Positive Scenarios:**
  + **Pen stand holds pens securely without them falling over.**
  + **Pen stand is stable and does not tip over when used.**
  + **Pen stand accommodates multiple pens of varying sizes.**
  + **Pen stand is made of durable material (e.g., plastic, wood).**
  + **Pen stand is easy to clean.**
* **Negative Scenarios:**
  + **Pen stand is too small to hold a variety of pens.**
  + **Pen stand topples over when a pen is placed in it.**
  + **Pen stand material breaks easily.**
  + **Pen stand is difficult to clean due to complex design.**
  + **Pen stand holds pens too tightly, making them hard to remove.**

**41. Write a Scenario of Door**

**Scenario Title: Verify functionality of a door**

* **Positive Scenarios:**
  + **Door opens and closes smoothly.**
  + **Door locks and unlocks properly using the key or handle.**
  + **Door is secure, preventing unauthorized access.**
  + **Door functions properly in both manual and automatic modes.**
  + **Door’s hinges and components are well-maintained and do not squeak.**
* **Negative Scenarios:**
  + **Door gets stuck or jams when opening/closing.**
  + **Door does not lock or unlock due to mechanical failure.**
  + **Door doesn’t fit well into its frame and leaves gaps.**
  + **Automatic door fails to open or close.**
  + **Door handle or knob falls off.**

**42. Write a Scenario of ATM**

**Scenario Title: Verify the functionality of an ATM**

* **Positive Scenarios:**
  + **User can insert ATM card and access the account.**
  + **User can check balance and withdraw money without issues.**
  + **ATM displays correct account information.**
  + **ATM provides an option to change the PIN.**
  + **ATM gives an accurate receipt after every transaction.**
* **Negative Scenarios:**
  + **ATM does not accept the card due to reader error.**
  + **ATM displays an incorrect balance or error messages.**
  + **User cannot withdraw money due to system failure.**
  + **ATM malfunctions and eats the card.**
  + **ATM network goes down, causing transaction failures.**

**43. When to Use Usability Testing?**

**Scenario Title: Verify the use of usability testing**

* **Usability testing should be used:**
  + **During the design phase to evaluate user interaction and interface usability.**
  + **After functional testing to verify ease of use and intuitive interface.**
  + **When introducing new features to ensure users can easily understand and use them.**
  + **To identify potential pain points for users before final release.**
  + **When testing new versions of software to ensure they are more user-friendly than previous versions.**

**44. What is the Procedure for GUI Testing?**

**Scenario Title: Verify the procedure for GUI testing**

* **Procedure:**
  + **Identify and verify all graphical elements such as buttons, text fields, and labels.**
  + **Ensure that the layout is consistent across different screen sizes and resolutions.**
  + **Check the alignment of text, buttons, and images.**
  + **Verify that all GUI elements are functional (e.g., clickable buttons, input fields).**
  + **Ensure the color scheme, fonts, and contrast are visually appealing and accessible.**
  + **Test user interactions with GUI elements to ensure no unexpected behavior occurs.**

**45. Write a Scenario of Microwave Oven**

**Scenario Title: Verify the functionality of a microwave oven**

* **Positive Scenarios:**
  + **Microwave turns on and heats food to the correct temperature.**
  + **Microwave settings are easy to navigate.**
  + **Microwave door opens and closes properly.**
  + **Microwave beeps when cooking is finished.**
  + **Timer works accurately to track cooking duration.**
* **Negative Scenarios:**
  + **Microwave does not heat food evenly.**
  + **Microwave door fails to latch properly.**
  + **Microwave settings reset after a power cut.**
  + **Microwave timer does not stop cooking when time is up.**
  + **Microwave emits strange noises or sparks during operation.**

**46. Write a Scenario of Coffee Vending Machine**

**Scenario Title: Verify the functionality of a coffee vending machine**

* **Positive Scenarios:**
  + **Machine dispenses coffee when the user selects the option.**
  + **Machine accepts payment and gives change.**
  + **Coffee machine provides the correct amount of milk and sugar as per the selection.**
  + **Coffee temperature is appropriate and consistent.**
  + **Machine provides a clean cup every time.**
* **Negative Scenarios:**
  + **Machine fails to dispense coffee after payment.**
  + **Machine does not accept coins or bills properly.**
  + **Coffee machine runs out of supplies (milk, sugar, coffee beans).**
  + **Machine malfunctions and produces incorrect coffee (e.g., too strong, too weak).**
  + **Vending machine doesn’t recognize user input from the touchscreen.**

**47. Write a Scenario of Chair**

**Scenario Title: Verify the functionality of a chair**

* **Positive Scenarios:**
  + **Chair is comfortable to sit on for extended periods.**
  + **Chair has proper back support and ergonomic design.**
  + **Chair can be adjusted to different heights.**
  + **Chair legs are stable and do not wobble.**
  + **Chair material is durable and easy to clean.**
* **Negative Scenarios:**
  + **Chair is uncomfortable or causes back pain after sitting for a while.**
  + **Chair height adjustment mechanism does not work.**
  + **Chair legs are unstable and cause the chair to tip over.**
  + **Chair material wears out quickly or stains easily.**
  + **Chair armrests are too short or positioned poorly.**

**38. Write a Scenario of Only WhatsApp Chat Messages**

**Scenario Title: Verify functionality of sending and receiving messages in WhatsApp**

* **Positive Scenarios:**
  + **User can send a text message to a contact.**
  + **User receives a message from a contact in real-time.**
  + **User can send multimedia messages (images, videos, audio).**
  + **User can see "delivered" and "read" status for messages.**
  + **User can send messages to a group and all members receive the messages.**
* **Negative Scenarios:**
  + **Message fails to send due to no internet connectivity.**
  + **Message delivery fails when the recipient is offline.**
  + **User cannot send a message to a contact that is blocked.**
  + **Multimedia message fails to upload.**
  + **App crashes when trying to send a message with large attachments.**

**39. Write a Scenario of Pen**

**Scenario Title: Verify the functionality of a ballpoint pen**

* **Positive Scenarios:**
  + **Pen writes smoothly without skipping or smudging.**
  + **Ink flow is consistent.**
  + **Pen writes on all standard paper types.**
  + **Pen cap securely fits to prevent ink drying.**
  + **Pen is comfortable to hold for long periods.**
* **Negative Scenarios:**
  + **Ink leaks from the pen.**
  + **Pen runs out of ink too quickly.**
  + **Ink doesn’t dry, causing smudging.**
  + **Cap does not securely close.**
  + **Pen tip gets clogged and stops writing.**

**40. Write a Scenario of Pen Stand**

**Scenario Title: Verify the functionality of a pen stand**

* **Positive Scenarios:**
  + **Pen stand holds pens securely without them falling over.**
  + **Pen stand is stable and does not tip over when used.**
  + **Pen stand accommodates multiple pens of varying sizes.**
  + **Pen stand is made of durable material (e.g., plastic, wood).**
  + **Pen stand is easy to clean.**
* **Negative Scenarios:**
  + **Pen stand is too small to hold a variety of pens.**
  + **Pen stand topples over when a pen is placed in it.**
  + **Pen stand material breaks easily.**
  + **Pen stand is difficult to clean due to complex design.**
  + **Pen stand holds pens too tightly, making them hard to remove.**

**41. Write a Scenario of Door**

**Scenario Title: Verify functionality of a door**

* **Positive Scenarios:**
  + **Door opens and closes smoothly.**
  + **Door locks and unlocks properly using the key or handle.**
  + **Door is secure, preventing unauthorized access.**
  + **Door functions properly in both manual and automatic modes.**
  + **Door’s hinges and components are well-maintained and do not squeak.**
* **Negative Scenarios:**
  + **Door gets stuck or jams when opening/closing.**
  + **Door does not lock or unlock due to mechanical failure.**
  + **Door doesn’t fit well into its frame and leaves gaps.**
  + **Automatic door fails to open or close.**
  + **Door handle or knob falls off.**

**42. Write a Scenario of ATM**

**Scenario Title: Verify the functionality of an ATM**

* **Positive Scenarios:**
  + **User can insert ATM card and access the account.**
  + **User can check balance and withdraw money without issues.**
  + **ATM displays correct account information.**
  + **ATM provides an option to change the PIN.**
  + **ATM gives an accurate receipt after every transaction.**
* **Negative Scenarios:**
  + **ATM does not accept the card due to reader error.**
  + **ATM displays an incorrect balance or error messages.**
  + **User cannot withdraw money due to system failure.**
  + **ATM malfunctions and eats the card.**
  + **ATM network goes down, causing transaction failures.**

**43. When to Use Usability Testing?**

**Scenario Title: Verify the use of usability testing**

* **Usability testing should be used:**
  + **During the design phase to evaluate user interaction and interface usability.**
  + **After functional testing to verify ease of use and intuitive interface.**
  + **When introducing new features to ensure users can easily understand and use them.**
  + **To identify potential pain points for users before final release.**
  + **When testing new versions of software to ensure they are more user-friendly than previous versions.**

**44. What is the Procedure for GUI Testing?**

**Scenario Title: Verify the procedure for GUI testing**

* **Procedure:**
  + **Identify and verify all graphical elements such as buttons, text fields, and labels.**
  + **Ensure that the layout is consistent across different screen sizes and resolutions.**
  + **Check the alignment of text, buttons, and images.**
  + **Verify that all GUI elements are functional (e.g., clickable buttons, input fields).**
  + **Ensure the color scheme, fonts, and contrast are visually appealing and accessible.**
  + **Test user interactions with GUI elements to ensure no unexpected behavior occurs.**

**45. Write a Scenario of Microwave Oven**

**Scenario Title: Verify the functionality of a microwave oven**

* **Positive Scenarios:**
  + **Microwave turns on and heats food to the correct temperature.**
  + **Microwave settings are easy to navigate.**
  + **Microwave door opens and closes properly.**
  + **Microwave beeps when cooking is finished.**
  + **Timer works accurately to track cooking duration.**
* **Negative Scenarios:**
  + **Microwave does not heat food evenly.**
  + **Microwave door fails to latch properly.**
  + **Microwave settings reset after a power cut.**
  + **Microwave timer does not stop cooking when time is up.**
  + **Microwave emits strange noises or sparks during operation.**

**46. Write a Scenario of Coffee Vending Machine**

**Scenario Title: Verify the functionality of a coffee vending machine**

* **Positive Scenarios:**
  + **Machine dispenses coffee when the user selects the option.**
  + **Machine accepts payment and gives change.**
  + **Coffee machine provides the correct amount of milk and sugar as per the selection.**
  + **Coffee temperature is appropriate and consistent.**
  + **Machine provides a clean cup every time.**
* **Negative Scenarios:**
  + **Machine fails to dispense coffee after payment.**
  + **Machine does not accept coins or bills properly.**
  + **Coffee machine runs out of supplies (milk, sugar, coffee beans).**
  + **Machine malfunctions and produces incorrect coffee (e.g., too strong, too weak).**
  + **Vending machine doesn’t recognize user input from the touchscreen.**

**47. Write a Scenario of Chair**

**Scenario Title: Verify the functionality of a chair**

* **Positive Scenarios:**
  + **Chair is comfortable to sit on for extended periods.**
  + **Chair has proper back support and ergonomic design.**
  + **Chair can be adjusted to different heights.**
  + **Chair legs are stable and do not wobble.**
  + **Chair material is durable and easy to clean.**
* **Negative Scenarios:**
  + **Chair is uncomfortable or causes back pain after sitting for a while.**
  + **Chair height adjustment mechanism does not work.**
  + **Chair legs are unstable and cause the chair to tip over.**
  + **Chair material wears out quickly or stains easily.**
  + **Chair armrests are too short or positioned poorly.**

**48. To Create Scenario (Positive & Negative) • Create Test Cases on Compose Mail Functionality.**

https://docs.google.com/spreadsheets/d/14VFp\_EFHBd94egBaIuVdmvL2VG8xs-VBjJlzM0yAefw/edit?usp=sharing

**49. Write a Scenario of Wrist Watch**

**Scenario Title:** Verify functionality of digital wristwatch

* **Positive Scenarios:**
  + Watch displays correct time and date.
  + User can set an alarm successfully.
  + Stopwatch starts, pauses, and resets properly.
  + Battery status is displayed correctly.
  + Watch is water-resistant as per specifications.
* **Negative Scenarios:**
  + Time cannot be changed when buttons are non-functional.
  + Incorrect date/time displayed after battery replacement.
  + Alarm does not ring at the set time.
  + Buttons get stuck during usage.

**50. Write a Scenario of Lift (Elevator)**

**Scenario Title:** Verify Lift (Elevator) operations

* **Positive Scenarios:**
  + Lift moves to the selected floor when a floor button is pressed.
  + Door opens and closes automatically at each stop.
  + Overload indicator alerts when capacity exceeds the limit.
  + Emergency button connects to support.
* **Negative Scenarios:**
  + Lift does not move when door is open.
  + Floor button doesn't respond due to control failure.
  + Door doesn't open on reaching the floor.
  + Lift stops between floors (simulate power failure).

**51 Write a Scenario of WhatsApp Payment**

**Scenario Title:** Verify WhatsApp UPI Payment Functionality

* **Positive Scenarios:**
  + User can link their bank account with UPI ID.
  + Payment is sent successfully to a verified contact.
  + Payment confirmation message is displayed.
  + Transaction history is shown properly.
* **Negative Scenarios:**
  + Payment fails due to incorrect UPI PIN.
  + Transaction fails due to no internet connectivity.
  + User enters invalid UPI ID.
  + Bank server down leads to payment timeout.