Module–2(Manual Testing)

• What is Exploratory Testing?

Exploratory testing is a concurrent process where

Testdesign, execution and logging happen simultaneously

* What is traceability matrix?

A traceability matrix is a document that details the technical requirements for a given test scenario and its current state.

* What is Boundary value testing?

Boundary testing is one software testing technique organizations often use, and it involves evaluating data based on its boundary values, or its two opposite ends, such as its minimums and maximum.

* What is Equivalence partitioning testing?

Equivalence Partitioning is a black box technique to identify test cases systematically and is often the first technique to be applied when designing test case.

* What is Integration testing?

Integration testing involves checking individual components or units of a software project to expose defects and problems to verify that they work together as designed.

• What determines the level of risk?

the risk level is determined by two dimensions: probability and impact. Probability: It measures the likelihood of an event occurring, typically expressed as a percentage or qualitative scale

• What is Alpha testing?

Alpha testing is the first phase of formal testing, during which the software is tested internally using white-box techniques. Beta testing is the next phase, in which the software is tested by a larger group of users, typically outside of the organization that developed it.

• What is beta testing?

Beta testing is the next phase, in which the software is tested by a larger group of users, typically outside of the organization that developed it.

• What is component testing?

Component testing, also known as unit testing or module testing, is a level of software testing that focuses on verifying the individual components or units of a system. A component refers to a self-contained module or a group of related functions within the software

• What is functional system testing?

Functional testing is a type of testing that seeks to establish whether each application feature works as per the software requirements.

• What is Non-Functional Testing?

Non-functional testing is a software testing technique that checks the non-functional attributes of the system

• What is GUI Testing?

GUI is the abbreviation of ‘Graphical User Interface’. It contains several visual elements, such as buttons, text boxes, menus, checkboxes, images, etc. GUI testing refers to the validating UI functions or features of an application that are visible to the users, and they should comply with business requirements. GUI testing is also known as [UI testing](https://www.browserstack.com/guide/ui-testing-guide)**.** That means ‘User Interface testing. So, you can use both acronyms alternatively.

• What is Adhoc testing?

Ad hoc testing is an informal testing type with an aim to break the system.

• What is load testing?

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 arrange of loads to determine at what point the system’s response time degradesorfails

• What is stress Testing?

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.

• What is white box testing and list the types of white box testing?

White Box Testing: *Testing based on an analysis of the internal structure of the component or system.*

• What is black box testing? What are the different black box testing techniques?

Black box testing techniques apply to all levels of testing, as well as functional and non-functional testing types. There are four main black box testing techniques: equivalence partitioning, boundary value analysis, decision table testing, and state transition testing

• Mention what are the categories of defects?

Categories of defects: Categories of defects are: Errors of commissions, Errors of omissions, Errors of clarity, and Error of speed and capacity

• Mention what bigbang testing is?

Categories of defects: Categories of defects are: Errors of commissions, Errors of omissions, Errors of clarity, and Error of speed and capacity

• What is the purpose of exit criteria?

Exit criterion is used to determine whether a given test activity has been completed or NOT. Exit criteria can be defined for all of the test activities right from planning, specification and execution.

• When should "Regression Testing" be performed?

Regression testing is performed before each new instance of the product is deployed, guaranteeing that the program works perfectly in each setting. For instance, we need to make sure the product continues to perform properly in a production environment before we release it.

• What is 7 key principles? Explain in detail?

**7 key principles:**

* Testing shows presence of Defects
* Exhaustive Testing is Impossible.
* Early Testing.
* Defect Clustering.
* The Pesticide Paradox.
* Testing is Context Dependent.
* Absence of Errors Fallacy.

##### Testing shows presence of Defects:

* + Testing can show thatdefects are present, but cannotprovethat there are no defects.

4

* + However Testing cannot prove that there are no defects present.

##### 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.
  + For example: In an application in one screen there are 15 input ﬁelds, each having 5 possible values, then to test all the valid combinations you would need 30 517 578 125 (515) tests.

##### Early Testing:

Testing activities should start as early as possible in the software

or system development life cycle, and should be focused on deﬁned objectives.

Testing activities should start as early as possible in the development lifecycle.

These activities should be focused on deﬁned objectives – outlined in the Test Strategy.

5

##### Defect Clustering:

A small number of modules contain most of the defects discovered during pre- release testing, or are responsible for the most operational failures.

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 similarly, most operational failures of a system are usually conﬁned

##### Pesticides Paradox:

If the same tests are repeated overland over again 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.

Testing identiﬁes bugs, and programmers respond to ﬁx them. As bugs are eliminated by the programmers, the software improves

As software improves the eﬀectiveness of previous tests erodes Therefore we must

6

##### Testing is Context Dependent:

Testing is basically context dependent. Testing is done diﬀerently in diﬀerent contexts. Diﬀerent kinds of sites are tested diﬀerently.

For example

Safety– critical software is tested diﬀerently froman E-commerce site.

##### Absence of Errors Fallacy:

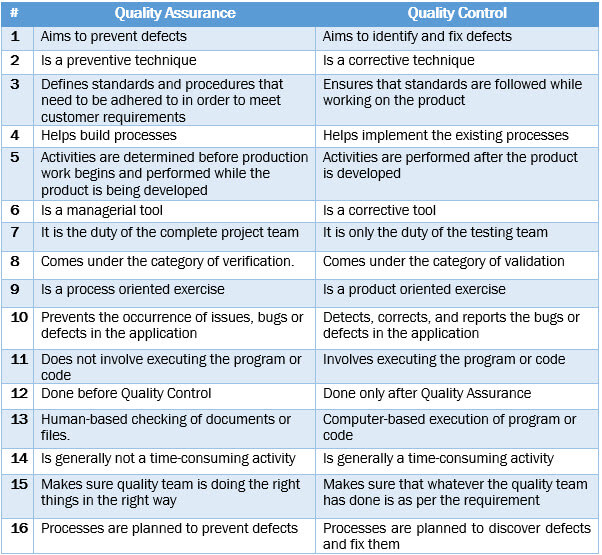
If the system built is unusable and does not fulﬁll the user’s needs and expectations then ﬁnding and ﬁxing defects does not help.

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’ needs and expectations.

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



• Difference between Smoke and Sanity.

Smoke testing is done to assure that the acute functionalities of program is working fine. Sanity testing is done to check the bugs have been fixed after the build. Smoke testing is also called subset of acceptance testing. Sanity testing is also called subset of regression testing.

• Difference between verification and Validation

## Differences Between Verification and Validation

| **S.No.** | **Verification** | **Validation** |
| --- | --- | --- |
| 1. | Verification means checking the documents, languages, designs, and other programming things. | Validation means testing the actual product. |
| 2. | Verification does not involve the execution of the code. | Validation involves the execution. |
| 3. | It is considered static testing. | It is considered dynamic testing. |
| 4. | Verification uses methods such as walkthroughs, reviews, desk-checking, and inspection. | Validation uses a method such as White Box Testing, Black Box Testing, etc. |
| 5. | It has the ability to detect errors quickly | It can only detect errors that could not be determined by the verification method. |
| 6. | It includes checking documents delivered by humans. | It includes the execution of a program executed by a computer. |

• Explain types of Performance testing

* Load Testing. Load testing measures system performance as the workload increases.
* Stress Testing.
* Spike Testing.
* Endurance Testing.
* Scalability Testing.
* Volume Testing.
* Identify the Testing Environment.
* Identify Performance Metrics.

• What is Error, Defect, Bug and failure?

Bug is an error detected in the development environment during testing stage. Defect is a mismatch between the expected and actual result of software development detected by a software developer or end customer in the production environment. Failure is called an error which is founded by the end user.25 Jun 2023

• Difference between Priority and Severity

|  |  |  |
| --- | --- | --- |
| **ammeters** | **Severity in Testing** | **Priority in Testing** |
| Definition | Severity is a term that denotes how severely a defect can affect the functionality of the software. | Priority is a term that defines how fast we need to fix a defect. |
| Parameter | Severity is basically a parameter that denotes the total impact of a given defect on any software. | Priority is basically a parameter that decides the order in which we should fix the defects. |
| Relation | Severity relates to the standards of quality. | Priority relates to the scheduling of defects to resolve them in software. |
| Value | The value of severity is objective. | The value of priority is subjective. |
| Change of Value | The value of Severity changes continually from time to time. | The value of Priority changes from time to time. |
| Who Decides the Defect | The testing engineer basically decides a defect’s severity level. | The product manager basically decides a defect’s priority level. |
| Types | There are 5 types of Severities: Cosmetic, Minor, Moderate, Major, and Critical. | There are 3 types of Priorities: High, Medium, and Low. |

• What is Bug Life Cycle?

The developer first identifies the bug, then moves to the tester for testing, and the tester marks the stages based on the priority of the bug that needs to be fixed. Finally, they fix the bug, develop error-free software, and deliver it to the customer.

• Explain the difference between Functional testing and NonFunctional testing.

Functional testing checks the application's processes against a set of requirements or specifications. Non-functional testing assesses application properties that aren't critical to functionality but contribute to the end-user experience, like performance and reliability under load.

* To create HLR & TestCase of 1)(Instagram , Facebook) only first page

These are some of the test scenarios for the Facebook login page.

* Check the login functionality with a valid email ID and valid password.
* Check the login functionality with a valid email ID and
* Check the login functionality with an invalid email ID and valid password.
* Check the login functionality with an invalid email ID and invalid password.
* Check the login functionality with a blank email ID and password.
* Check the login functionality with a blank email ID and valid password.
* Check the login functionality with a valid email ID and a blank password.
* Check the login functionality with a valid phone number and valid password.
* Check the login functionality with an invalid phone number and valid password.
* Check the login functionality with a valid phone number and invalid password.
* Check the login functionality with a blank phone number and valid password.
* Check the length of the email address and password field.
* Check the error message displayed when any field is left blank.
* Check the Tab key functionality on the Login page.
* Check the Remember Me checkbox functionality.
* Check if the welcome message is displaying after successfully logging into the application.
* Check the Forgotten Password functionality.
* Check whether the password text format is encrypted or not.
* Check if the correct error message is displaying for the invalid inputs.
* What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)

SDLC covers the entire software development process, including planning, analysis, design, coding, testing, deployment, and maintenance. STLC is a part of SDLC and focuses specifically on testing. STLC includes activities like test planning, test case development, test execution, defect tracking, and closure

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

A test scenario is any functionality that a software testing company can examine. It is also called a Test Condition or Test Possibility. A test case is a document that lists the steps a QA engineer needs to execute. A test script is a short program written in a programming language.

• Explain what Test Plan is? What isthe information that should be covered.

A test plan is a document that consists of all future testing-related activities. It is prepared at the project level and in general, it defines work products to be tested, how they will be tested, and test type distribution among the testers.

• What is priority?

The quality or state of coming before another in time or importance. A condition of being given attention before others.

• What is severity?

Definitions of severity. excessive sternness. “Severity of character” synonyms: austerity, hardness, harshness, inclemency, rigor, rigorousness, rigour, rigorousness, severeness, stiffness. type of: sternness, strictness.

• Bug categories are

Software bugs can be classified into multiple categories based on their nature and impact. Broadly speaking, these categories include Functional Bugs, Logical Bugs, Workflow Bugs, Unit Level Bugs, System-Level Integration Bugs, Out of Bound Bugs, and Security Bugs.

• Advantage of Bugzilla .

It improves the quality of the product. It enhances the communication between the developing team and the testing team. It has the capability to adapt to multiple situations.

• Difference between priority and severity.

Severity is basically a parameter that denotes the total impact of a given defect on any software. Priority is basically a parameter that decides the order in which we should fix the defects. Severity relates to the standards of quality. Priority relates to the scheduling of defects to resolve them in software.

• What are the different Methodologies in Agile Development Model?

* Scrum.
* Extreme Programming
* Adaptive Software Development
* Dynamic Software Development Method
* Feature Driven Development
* Kanban.
* Behaviour Driven Development

• Explain the difference between Authorization and Authentication in Web testing. What are the common problems faced in Web testing? To create HLR & Testcase of Web Based (WhatsApp web, Instagram)

Authentication is the process of verifying who someone is, whereas authorization is the process of verifying what specific applications, files, and data a user has access to.

1. WhatsApp Web: <https://web.whatsapp.com>

To create HLR and Test Case on this Link. <https://artoftesting.com>

Write a scenario of only WhatsApp chat messages

• Write a Scenario of Pen.

|  |  |  |
| --- | --- | --- |
| Test Scenario Id | Test Scenario (Positive) | Test Scenario (Negative) |
| 1 | Verify the types of pens whether it is ball pen, gel pen, ink pen etc.. | Verify the pen cap is missing so I cannot put on my pocket |
| 2 | Verify the diameter of the pen whether it is suitable for all or not | Verify the cap is missing so ink should be dry in winter season so I cannot write on time |
| 3 | Verify the material of the pen whether it is fiber, plastic, mentalese. | Check the functioning of a pen at zero gravity. |
| 4 | Verify the height of the pen which is suitable for all | Verify the functioning of the pen by applying extreme pressure. |
| 5 | Verify the weight of the pen which is suitable for all | Verify the strength of the pen’s outer body. It should not be easily breakable. |
| 6 | Verify the ink of the pen | Check if the text written by the pen is waterproof or not |
| 7 | Verify the pen have cap or not | Verify if the pen can support multiple refills or not. |
| 8 | Verify that the pen’s ink should not leak in case it is tilted upside down. |  |
| 9 | Verify if the pen’s ink should not leak at higher altitudes. |  |
| 10 | Verify if the text written by the pen is erasable or not. |  |
| 11 | Check the functioning of the pen by applying normal pressure during writing. |  |
| 12 | Verify the strength of the pen’s outer body. It should not be easily breakable. |  |
| 13 | Verify that text written by pen should not get faded before a certain time as mentioned in the specification. |  |
| 14 | Check if the text written by the pen is waterproof or not. |  |
| 15 | Verify that the user is able to write normally by tilting the pen at a certain angle instead of keeping it straight while writing. |  |
| 16 | Check the grip of the pen, and whether it provides adequate friction for the user to comfortably grip the pen. |  |
| 17 | Verify if the pen can support multiple refills or not. |  |
| 18 | verify that the user is able to refill the pen with all the supported ink types. |  |
| 19 | verify that the mechanism to refill the pen is easy to operate. |  |
| 20 | verify the size of the tip. |  |

• Write a Scenario of Pen Stand

• Write a Scenario of Door

|  |  |  |  |
| --- | --- | --- | --- |
| Test Scenario Id | Test Scenario (Positive) | Test Scenario (Negative) |  |
|  | Verify if the door is single door or bi-folded door. | Check the door condition when you are washing with hot water or cold water |  |
|  | Check if the door opens inwards or outwards. | Check the door condition and different climates like summer, winter, rain, etc. |  |
|  | Verify that the dimension of the doors are as per the specifications. | Check how much amount of force is required to open or close the door |  |
|  | Verify that the material used in the door body and its parts is as per the specifications. | Check whether the big hole door has a peek-hole, and also check the position of that whole is as per the specification document |  |
|  | Verify that colour of the door is as specified. | Check if the door has automatic close functionality in it and if your automatic feature represents the spring mechanism |  |
|  | Verify if the door is sliding door or rotating door. | Check whether the door has a stopper at the correct location, which is mentioned in the specification document. |  |
|  | Check the position and quality | Check what type of locks are present in the door |  |
|  | Check the type of locks in the door. |  |  |
|  | Verify if the door is having peek-hole or not. |  |  |
|  | Verify if the door is having stopper or not. |  |  |
|  | Verify if the door closes automatically or not – spring mechanism. |  |  |
|  | Verify if the door makes noise when opened or closed. |  |  |
|  | Check the door condition when used extensively with water. |  |  |

• Write a Scenario of ATM

|  |  |  |  |
| --- | --- | --- | --- |
| Test Scenario Id | Test Scenario (Positive) | Test Scenario (Negative) |  |
| 1 | Verify that power backup should be present at ATM. | Verify the functionality by entering a wrong pin number for a particular number of times. |  |
| 2 | Verify that card reader should be present. | Verify the card ATM machine functionality by inserting an expired card. |  |
| 3 | Verify that receipt printer should be present and working. | Verify the cash withdrawal functionality by entering invalid amount such as 10, 20, 50. |  |
| 4 | Verify that cash dispenser is working as expected. | Verify the ATM machine functionality by entering amount greater than available balance. |  |
| 5 | Verify that the key pad should be working and covered. | Verify the ATM machine functionality by entering amount greater than per day and per transaction limit. |  |
| 6 | Verify that buttons are displayed on screen of ATM machine. |  |  |
| 7 | Verify the font of text on the screen, it should be clearly visible |  |  |
| 8 | Verify that when card in inserted in ATM, pin should be asked from user. |  |  |
| 9 | Verify that when user enters incorrect pin for a particular number of times, the card is blocked. |  |  |
| 10 | Verify that when user enters correct pin, the user details should be displayed on ATM screen. |  |  |
| 11 | Verify that ATM machine asks to user for the amount to be withdrawn. |  |  |
| 12 | Verify that if use enters amount greater than daily withdraw limit, error message is displayed. |  |  |
| 13 | Verify that if doesn’t enter amount in round off digits, error message is displayed. |  |  |
| 14 | Verify that if user enters valid amount, the exact amount of cash should be dispensed from ATM machine. |  |  |
| 15 | Verify that how much time is taken in a transaction. |  |  |
| 16 | Verify how much time is taken by system to logout user. |  |  |
| 17 | Verify that user is able to use card of other bank on the ATM. |  |  |
| 18 | Verify that message is displayed when the cash in ATM is finished. |  |  |
| 19 | Verify that correct message is displayed after the transaction. |  |  |
| 20 | Verify that user is presented with an option to select language of operation. |  |  |
| 21 | Verify that pin is displayed in masked format. |  |  |
| 22 | Verify that error message is displayed when entered amount is greater than account balance. |  |  |
| 23 | Verify that session is timeout is no activity is performed for a particular time. |  |  |
| 24 | Verify that the user is not allowed to exceed one transaction limit amount. |  |  |
| 25 | Verify that the user is not allowed to exceed the one-day transaction limit amount. |  |  |

• When to used Usability Testing?

1. stability testing ideas. Once you've got an idea, conduct usability testing before putting any design resources to work. ...
2. Usability testing prototypes. ...
3. Usability testing before launch. ...
4. Usability testing after launch.

• What is the procedure for GUI Testing?

1. Check the page label, position, and font.
2. Validate whether the page heading is correct and check the font used.
3. Check the cursor to focus on the default field.
4. Test the mandatory fields by clicking next while the form is blank.
5. Check the position and alignment of the text box.

• Write a scenario of Microwave Owen

* Verify whether the power button of the coffee vending machine is working correctly after pressing the power button.
* Verify whether the coffee vending machine is activated when the user presses the Power ON button.
* Verify whether the coffee vending machine is turned off when the user presses the power OFF button.
* Verify whether the indicator lights display correctly when the coffee vending machine is going to switch off or on.
* Verify whether all the buttons of the coffee vending machine have an image text on them, which indicates what task will be performed if you press the button.
* Verify whether the foamed in the coffee vending machine is working as expected.
* Verify whether the auto cleaner facility is working properly or not.
* Verify whether the half-cup feature works properly or not.
* Verify whether the cup quantity counter should work properly.
* Verify whether the temperature of the coffee served should be the same temperature or not.
* Verify whether the input mechanism for coffee ingredients-milk, water, coffee beans/powder, etc. works as expected.
* Verify whether the quantity of hot water, milk, and coffee powder per serving are correct.
* Verify the effect of suddenly switching off the machine or cutting the power, the machine should stop in that situation and power resumption, the remaining coffee should not come out of the nozzle.
* Verify whether the functioning of all the buttons work properly when pressed
* Verify whether the coffee beans are grinding evenly, check it by picking a test bean and testing how evenly it has been ground.

• Write a scenario of Coffee vending Machine

|  |  |  |
| --- | --- | --- |
| Test Scenario Id | Test Scenario (Positive) | Test Scenario (Negative) |
|  | Verify whether the power button of the coffee vending machine is working correctly after pressing the power button. | Check how the coffee vending machine acts when two/multiple buttons are pressed simultaneously. |
|  | Verify whether the coffee vending machine is activated when the user presses the Power ON button. | Check how the coffee machine functions when the ingredient container’s capacity is exceeded. |
|  | Verify whether the coffee vending machine is turned off when the user presses the power OFF button. | Verify how the coffee machine functions when the power gets cut off. |
|  | Verify whether the indicator lights display correctly when the coffee vending machine is going to switch off or on. | Check how the machine works when the user presses all buttons at once. |
|  | Verify whether the foamed in the coffee vending machine is working as expected. | Check how the machine works when the user places the machine upside down. |
|  | Verify whether the auto cleaner facility is working properly or not. |  |
|  | Verify whether the half-cup feature works properly or not. |  |
|  | Verify whether the cup quantity counter should work properly. |  |
|  | Verify whether the temperature of the coffee served should be the same temperature or not. |  |
|  | Verify whether the input mechanism for coffee ingredients-milk, water, coffee beans/powder, etc. works as expected. |  |
|  | Verify whether the quantity of hot water, milk, and coffee powder per serving are correct. |  |
|  | Verify whether the functioning of all the buttons work properly when pressed |  |
|  | Verify whether the coffee beans are grinding evenly, check it by picking a test bean and testing how evenly it has been ground. |  |

Write a scenario of Chair.

|  |  |  |
| --- | --- | --- |
| Test Scenario Id | Test Scenario (Positive) | Test Scenario (Negative) |
|  | Check the number of legs of a chair. | Check the balance of the chair with one arm. |
|  | Check the chair backrest option. | Check the balance of the chair with three legs. |
|  | Check whether all legs of the chair on a plane surface are equal. | Check the legs are of same size or not. |
|  | Check whether a human can sit comfortably on a chair or not. | Check the material color/ strength. (Check its reaction |
|  | Check if the chair has an adjustment functionality or not. | Check by sitting on it and check whether it’s comfortable or not. |
|  | Check the sitting system. | Check arms of chairs are at proper height or not. |
|  | Check whether the legs of the chair have any wheels or not. | Check how much load it can take. |
|  | Check if the chair is good enough to handle a specified load. | Check it is easily movable or not |
|  | Check what the maximum amount of load the chair is handling is. |  |
|  | Check the date is stable enough to take any human load. |  |

• Write a Scenario of Lift (Elevator)

|  |  |  |
| --- | --- | --- |
| Test Scenario Id | Test Scenario (Positive) | Test Scenario (Negative) |
|  | verify the size of the lift | Verify that the elevator stops at each floor when you click the multi-floor button. |
|  | verify if type of door of the lift is as per the requirements | Verify the system so that it prompts users with audio/visual alerts |
|  | verify the type of material used in the lift interior and exterior | Verify the elevator to prompt the user with audio/visual information about the current floor and the direction it is moving. |
|  | verify the volume of the lift in terms of the total weight | Verify that the system announces the number of floors on each floor. |
|  | Verify the buttons in the lift to close and open the door and the number of buttons as per the number of floors in the tower. | Check the presence of up and down buttons outside the lift. |
|  | Verify if the lift goes to the specific floor upon pressing the floor button. | Check for sensor presence |
|  | Verify that the lift stops when a specific floor’s up/down buttons are clicked. | Verify that you write the maximum number of people and weights on the lift and check the emergency instructions. |
|  | Verify if there is an emergency button to contact associated person in case of any emergency | Verify the presence of the floor number display |
|  | Verify the performance of the lift – the time it takes to reach a floor. |  |
|  | Verify that the lift does not free-fall and gets halted on the specific floor in case of electric failure. |  |

• Write a Scenario of WhatsApp Group (generate group)

• Write a Scenario of WhatsApp payment