# ***module- 2 and 4 (manual testing )***

1. **What is Exploratory Testing?**

Exploratory testing is a concurrent process where test design, execution and logging happen simultaneously.

1. **What is traceability matrix?**

A traceability matrix is a document that details the technical requirements for a given test scenario and its current state. It helps the testing team understand the level of testing that is a done for a given product. The traceability process itself is used to review the test cases that were defined for any requirement

1. **What is Boundary value testing?**

Boundary value testing is the process of testing between extreme ends or boundaries between partitions of the input values.

1. **What is Equivalence partitioning testing?**

Equivalence partitioning testing is a software testing technique that divides the input data of a software unit into partitions of equivalent data from which test cases can be derived.

1. **What is Integration testing?**

Integration testing: - Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems. Integration Testing is a level of the software testing process where individual units are combined and tested as a group.

1. **What determines the level of risk?**

Risk are two types: -

1. Project risk: -

Example of project risk id senior tram member leaving the project

1. Product risk: -

Product risk would be flight reservation system not installing in test environment mitigation in this case.

1. **What is Alpha testing?**

Alpha testing is a type of software testing performed to identify bugs before releasing the software product to the real users or public. It is a type of acceptance testing. Alpha testing is typically performed by in-house software engineer or QA staff. It is the final testing stage before the software is release into the real world.

1. **What is beta testing?**

Beta testing is performed by “real users” of the software application in “real environment” and it can be considered as a form of external USER ACCEPTANCE TESTING. It is the final test before shipping a product to the customers. Direct feedback from the customers is a major advantage of beta testing. This testing helps to test products in customer’s environment.

1. **What is component testing?**

Component (unit testing): - A minimal software item that can be tested in isolation. It means “A unit is the smallest testable part of software.” Unit testing is the first level of testing and is performed prior to Integration Testing.

1. **What is functional system testing?**

Functional System Testing is a requirement that specifies a function that a system or system component must perform.

1. **What is Non-Functional Testing?**

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

1. **What is GUI Testing?**

Graphical User Interface (GUI) testing is the process of testing the system’s GUI of the System under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons, and all types of bars – tool bar, menu bar, dialog boxes and windows etc.

1. **What is Adhoc testing?**

Adhoc testing is an informal testing type with an aim to break the system. Main aim of this testing is to find the defects by random checking. Adhoc testing can be achieved with the testing technique called Error Guessing.

1. **What is load testing?**

It’s a performance testing to check system behaviour under load. Testing an application under heavy loads, such as testing of a web site under a range of loads to determine at what point the system’s response time degrades or fails. This testing helps determine how the application behaves when multiple users access it simultaneously.

1. **What is 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.

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

White box testing is a software testing technique that involves testing the internal structure and workings of software application. The tester has access to source code and uses this knowledge to design test cases that can verify the correctness the software at the code level.

TECHNIQUES OF WHITE BOX TESTING:

* Branch Condition testing
* Modified Condition Decision testing
* Dataflow testing
* Linear Code Sequence
* Branch Condition Combination testing

1. **What is black box testing? What are the different black box testing techniques?**

BLACK BOX TESTING: Testing, either functional or non-functional, without reference to the internal structure of the component or system. The testers have no knowledge of how the system or component is structured inside the box.

TECHNIQUES OF BLACK BOX TESTING:

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

1. **Mention what are the categories of defects?**

Types of defects: -

* Data quality defects
* Critical functionality defects
* Functionality defects
* Security defects
* UI defects

1. **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. Big Bang testing has the advantage that everything is finished before integration testing starts.

1. **What is the purpose of exit criteria?**

Purpose of exit criteria is to define when we stop testing either at the: End of all testing – example- product Go Live and End of phase of testing.

1. **When should "Regression Testing" be performed?**

Regression testing should be carried out when the system is stable and the system or the environment changes, when testing bug-fix releases as part of the maintenance phase.

1. **What is 7 key principles? Explain in detail?**

**General Testing Principles**

1. **Testing shows presence of Defects:** - Testing can show that defects are present, but cannot prove that there are no defects. Testing reduces the probability of undiscovered defects remaining in the software but, even if no defects are found, it is not a proof of correctness.
2. **Exhaustive Testing is Impossible:** - Testing everything including all combinations of inputs and preconditions is not possible.
3. **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.
4. **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.
5. **The Pesticide Paradox:** - 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.
6. **Testing is Context Dependent:** - Testing is basically context dependent. Testing is done differently in different contexts and Different kinds of sites are tested differently.
7. **Absence of Errors Fallacy:** - Even after defects have been resolved it may still be unusable and/or does not fulfil the users’ needs and expectations
8. **Difference between QA v/s QC v/s Tester**

|  |  |  |
| --- | --- | --- |
| Quality Assurance (QA) | Quality Control (QC) | Tester |
| 1. Focuses on processes and procedures rather than conducting actual testing on the system. | Focuses on actual testing by executing Software with intend to identify bug/defect through  implementation of procedures and process. | Focuses on actual testing. |
| 1. Process oriented activities. | Product oriented activities. | Product oriented activities. |
| 1. Preventive activities | It is a corrective process. | It is a preventive process. |
| 1. It is a subset of Software Test Life Cycle (STLC). | QC can be considered as the subset of Quality Assurance. | Testing is the subset of Quality Control. |

1. **Difference between Smoke and Sanity?**

|  |  |
| --- | --- |
| **Smoke Testing** | **Sanity Testing** |
| 1. Smoke Testing is performed to ascertain that the critical functionalities of the program is working fine | Sanity Testing is done to check the new functionality / bugs have been fixed |
| 1. Smoke testing is usually documented or scripted | Sanity testing is usually not documented and  is unscripted |
| 1. Smoke testing is a subset of Regression testing | Sanity testing is a subset of Acceptance testing |
| 1. Smoke testing performed by developers or testers | Sanity testing performed by testers |
| 1. Smoke testing is may be stable/unstable | Sanity testing is always stable |
| 1. Smoke testing’s main goal is to verify “stability” | Sanity testing’s main goal is to verify “rationality” |
| 1. Smoke testing is like General Health Check Up | Sanity Testing is like specialized health check up |

1. **Difference between verification and Validation**

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

1. **Explain types of Performance testing.**

**Performance testing: -** Software performance testing is a means of quality assurance (QA). It involves testing software applications to ensure they will perform well under their expected workload.

**Types of performance testing:** -

1. **Stress testing: -** System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load.

* **Stress testing tools**

Stress Tester

Neo Load

App Perfect

1. **Load testing: –** 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 tools**

Load runner

Web Load

Astra Load Test

Review’s Web Load

Studio, Rational Site Load

Silk Performer

1. **Endurance testing**
2. **Spike testing**
3. **Volume testing**
4. **Scalability testing**
5. **What is Error, Defect, Bug and failure?**

* ERROR: - A mistake in coding is called error
* DEFECTS: - error found by tester is called defect
* BUG: - defect accepted by development team then it is called bug
* FAILURE: - build does not meet the requirements then it is failure

1. **Difference between Priority and Severity?**

|  |  |  |
| --- | --- | --- |
| **features** | **Severity** | **Priority** |
| **Definitions** | Severity is a parameter to denote the impact of a particular defect on the software. | Priority is a parameter to decide the order in which defects should be fixed. |
| **Value** | Its value is objective. | Its value is subjective. |
| **Association** | It is associated with functionality or standards. | It is associated with scheduling. |
| **Driving Factor** | It is driven by functionality | It is driven by business value. |
| **Based On** | It is based on the technical aspect of the product. | It is based on the customer’s requirements. |
| **Categories** | Severity is divided into 4 categories:   * Critical * Major * Medium * Low | Priority is divided into 3 categories:   * Low * Medium * High |

1. **What is Bug Life Cycle?**

Bug life cycle is the specific set of stages that defects or bugs goes through in its entire life. The purpose of defects life cycle is to easily coordinate or communicate current status of defects which change to various assignee and make the defects fixing process systematic and efficient.

1. **Explain the difference between Functional testing and Non-Functional testing?**

|  |  |
| --- | --- |
| Functional Testing | Non-Functional Testing |
| 1. Functional testing is performed using the functional specification provided by the client and verifies the system against the functional requirements | Non-Functional testing checks the Performance, reliability, scalability and other non-functional aspects of the software system |
| 1. Functional testing is executed first | Non-functional testing should be performed after functional testing |
| 1. Manual testing or automation tools can be used for functional testing | Using tools will be effective for this testing |
| 1. Business requirements are the inputs to functional testing | Performance parameters like speed, scalability are inputs to non-functional testing. |
| 1. Functional testing describes what the product does | Nonfunctional testing describes how good the product works |
| 1. Easy to do manual testing | Tough to do manual testing |
| 1. Types of Functional testing are   ∙ Unit Testing  ∙ Smoke Testing  ∙ Sanity Testing  ∙ Integration Testing  ∙ White box testing  ∙ Black Box testing  ∙ User Acceptance testing  ∙ Regression Testing | Types of Nonfunctional testing are  ∙ Performance Testing  ∙ Load Testing  ∙ Volume Testing  ∙ Stress Testing  ∙ Security Testing  ∙ Installation Testing  ∙ Penetration Testing  ∙ Compatibility Testing  ∙ Migration Testing |

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

* Test Script: - A set of sequential instruction that detail how to execute a core business function
* Test Scenario: - A Scenario is any functionality that can be tested. It is also called Test Condition, or Test Possibility.
* Test Case: - Test cases involve the set of steps, conditions and inputs which can be used while performing the testing tasks.

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

**Difference between SDLC and STLC**: -

|  |  |  |
| --- | --- | --- |
|  | **SDLC** | **STLC** |
| Full form | Software Development Life Cycle | Software Testing Life Cycle |
| Objectives | The main object of SDLC life cycle is to complete successful development of the software including testing and other phases. | The only objective of the STLC phase is testing. |
| Requirement Gathering | In SDLC the business analyst gathers the requirements and create Development Plan | In STLC, the QA team analysis requirement documents like functional and non-functional documents and create System Test Plan |
| Design | In SDLC, the development team creates the high and low-level design plans | In STLC, the test analyst creates the Integration Test Plan |
| Coding | The real code is developed, and actual work takes place as per the design documents. | The testing team prepares the test environment and executes them |
| Maintenance | SDLC phase also includes post-deployment supports and updates. | Testers, execute regression suits, usually automation scripts to check maintenance code deployed |

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

Test plan: - A document describing the scope, approach, resources and schedule of intended test activities it’s called test plan.

1. **What is priority?**

Priority can be defined as how fast or how early the defect should be addressed. The defects having highest priority should be fixed first followed by the defects having lesser priority.

1. **What is severity?**

Severity can be defined as how severe the defect is to the system and how badly it will affect the functionality. For example, an application crash on clicking a button is severe to the system .so its severity will be high. Whereas a spelling/grammatical error will not have much impact on the overall functionality .so its severity will be low.

1. **Advantage of Bugzilla.**

**Key features of Bugzilla included:**

* Advanced search capabilities
* E-mail Notifications
* Modify/file Bugs by e-mail
* Time tracking
* Strong security
* Customization
* Localization
* Bugzilla is an open-source issue/bug tracking system that allows developers effectively to keep track of outstanding problems with their product. It is written in Perl and uses MYSQL database.

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

different Methodologies in Agile Development Model: -

* 1. **Scrum:** - SCRUM is an agile development method which concentrates particularly on how to manage tasks within a team based development environment. Basically, Scrum is derived from activity that occurs during rugby match. Scrum believes in empowering the development team and advocates working in small teams (say- 7 to 9 members).
  2. **Kanban:** - Kanban methodology is about day-to-day workflows and processes. It is a simple, visual means of managing projects that enables teams to see the progress so far and what’s coming up next. Kanban projects are primarily managed through a Kanban board, which segments tasks into three columns: “To Do,” “Doing,” and “Done.”

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

|  |  |
| --- | --- |
| Authentication | Authorization |
| Authentication is the process of identifying a user to provide access to a system. | Authorization is the process of giving permission to access the resources. |
| In this, the user or client and server are verified. | In this, it is verified that if the user is allowed through the defined policies and rules. |
| It is usually performed before the authorization. | It is usually done once the user is successfully authenticated. |
| It requires the login details of the user, such as user name & password, etc. | It requires the user's privilege or security level. |
| Data is provided through token. | Data is provided through the access tokens. |

1. **When to used Usability Testing?**

* Efficiency: - the efficiency parameter explains the end-user who is an expert and can take the minimum amount of time to execute his/her fundamental or, we can say, undeveloped task.
* Memorability: - The Memorability of an application can be beneficial or not beneficial.
* Accuracy: - The next parameter covered by performing the Usability testing is **Accuracy**. The usability testing ensures that no inappropriate/irrelevant data or information exists in the product.
* Learnability: -Another constraint that is encompassed by usability testing is **Learnability.** In this constraint, the end-user takes a minimum amount of time to learn the fundamental task.
* Satisfaction: -The execution of usability testing ensures the **customer's satisfaction** as we know that the satisfied customer can easily or freely use the application.
* Errors: -The last and most important parameter covered by the usability testing is **Errors detection**. At this point, we try to help the end-users fix those errors they made earlier and accomplish their tasks all over again.

1. **What is the procedure for GUI Testing?**

MANUAL BASED TESTING: - Under this approach, graphical screens are checked manually by testers in conformance with the requirements stated in business requirements document.

RECORD AND REPLAY: - GUI testing can be done using automation tools. This is done in 2 parts. During Record, test steps are captured into the automation tool. During playback, the recorded test steps are executed on the Application under Test. Example of such tools - QTP.

MODEL BASED TESTING: - A model is a graphical description of system’s behaviour. It helps us to understand and predict the system behaviour. Models help in a generation of efficient test cases using the system requirements.

1. **Bug categories are…**

* Functional Bugs:
* Performance Bugs:
* Security Bugs:
* Unit-level bugs:
* Usability Bugs:
* Syntax Errors:
* Compatibility Errors:
* Logic Bugs:

1. **To create HLR & Test Case of (login page)**

* **1. Instagram, only first page**

**HLR – Excel (Assignment 2 case) – sheet 1 (Login\_HLR)**

**Testcase – Excel (Assignment 2 case) - sheet 2 (Login\_Testcase)**

* **2. Facebook login page**

**HLR – Excel (Assignment 2 case) – sheet 1 (Login\_HLR)**

**Testcase – Excel (Assignment 2 case) - sheet 2(Login\_Testcase)**

1. **To create HLR & Test Case of Web Based**

* **1. What’s app web**

**HLR – Excel (Assignment 2 case) – sheet 3 (Web\_HLR)**

**Testcase – Excel (Assignment 2 case) - sheet 4 (Web\_Testcase)**

* **2. Instagram web**

**HLR – Excel (Assignment 2 case) – sheet 3 (Web\_HLR)**

**Testcase – Excel (Assignment 2 case) - sheet 4 (Web\_Testcase)**

1. **To create HLR and Test Case on this Link.** [**https://artoftesting.com/**](https://artoftesting.com/)

**HLR – Excel (Assignment 2 case) – sheet 5 (Art of testing HLR)**

**Testcase – Excel (Assignment 2 case ) - Sheet 6( Art of testing Testcase)**

1. **Write a scenario of only What’s app chat messages**

**In excel (Assignment 2 case ) sheet What’s app**

1. **Write a Scenario of Pen**

**In excel (Assignment 2 case ) sheet Scenario**

1. **Write a Scenario of Pen Stand**
2. **Write a Scenario of Door**

**In excel (Assignment 2 case ) sheet Scenario**

1. **Write a Scenario of ATM**

**In excel (Assignment 2 case ) sheet Scenario**

1. **Write a scenario of Microwave Owen**

**In excel (Assignment 2 case ) sheet Scenario**

1. **Write a scenario of Coffee vending Machine**

**In excel (Assignment 2 case ) sheet Scenario**

1. **Write a scenario of chair**

**In excel (Assignment 2 case ) sheet Scenario**

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

**In excel (Assignment 2 case ) sheet Scenario**

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

**In excel (Assignment 2 case ) sheet Scenario**

1. **Write a Scenario of what’s app Group (generate group)**

**In excel (Assignment 2 case ) sheet What’s app**

1. **Write a Scenario of Instagram (video call with chat)**

**In excel (Assignment 2 case ) sheet Instagram**

1. **Write a Scenario of What’s app payment**

**In excel (Assignment 2 case ) sheet What’s app**