Module – 2 (Manual Testing)

1. What is Exploratory Testing?

- The current trend in testing is to push for automation.
- Exploratory testing is a new way of thinking automation has its limit.
- Is not random testing but it is Adhoc testing with purpose of find bugs.
- It is structured and rigorous.
- For example :-

Functionality-1 **(structured testing)**

Functionality-2

Functionality-3

Functionality-4

(Function are checked structured based)

Functionality-1

Functionality-4

(Exploratory testing)

Functionality-3

Functionality-2

(Function are checked in a Adhoc based)

2. What is traceability matrix?

- Traceability matrix is a table which is used to trace the requirement during the software development life cycle.
- A software process should help you keeping the virtual table up to date.
- To protect against changes you should be able to trace back from every system component to the original requirement that caused its presence.

- It can be used for forward tracing **(i.e. from Requirements to Design or Coding)** or backward **(i.e. from Coding to Requirements).**
- Types of Traceability matrix.
    - Forward Traceability.
    - Backward Traceability.

3. What is Boundary value testing?

- Boundary value analysis is a methodology for designing test cases that concentrate software testing effort an access near the limits of valid ranges.
- Boundary value analysis is method which **"refines"** equivalences partitioning.
- Boundary value analysis generated test **"cases that highlight"** error better the equivalence partitioning.

4. What is Equivalence partitioning testing?

- Treat group of inputs as equivalence and to select one representative input to test them all.
- Equivalence Partitioning can be used for all levels of the testing.
- Equivalence Partitioning says that testing just one value we have tested the partition.
    - If one value finds bugs, the other probably will too.
    - If doesn't find a bugs, the other probably won't either.

5. What is Integration testing?

- Integration testing is performed to expose defect in the interface and in the interactive between integrated component and system.
- It is level of software testing process where individual unit are combined and tested group.
- There are two level of integration
    - Component integration
    - System integration

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

- A factor that could result in future negative consequences usually expressed as impact and likelihood.
- There are two level of risk.
    - Project risks
    - Product risks
- Project risks :-
    - Example of project risk is every risk is assigned likelihood.
    - For example, chance of it occurring typically on a scale of 1 to 10.
- Product risks :-
    - Example of product risks would be flight reservation system not installing in test environment.
    - For example, this case would be conducting a smoke or sanity testing will make changes in your scope item to include sanity testing.

7. **What is Alpha testing?**

- Alpha Testing is always performed at the time of Acceptance Testing when developers test the product and project to check whether it meets the user requirements or not.
- It is always performed at the developer's premises in the absence of the users.
- It is considered as the User Acceptance Testing (UAT) which is done at developer's area.
- Alpha Testing is not open to the market and public.

8. **What is beta testing?**

- Beta Testing is always performed at the time when software product and project are marketed.
- It is always performed at the user's premises in the absence of the development team.
- Beta testing can be considered **"pre-release"** testing.

- It is performed in **“Real Time Environment”.**

9. What is component testing?

- Component testing the testing of individual software component.
- Unit/component testing is the first level testing.
- A minimal software item that can be tested to in isolation. It means a unit the smallest testable part of the software.
- It is performed prior to integration testing.

10. What is functional system testing?

- Functional testing based on an analysis of the specification of the functionality of a component or system.
- Specification **(For example: - Requirement specification use cases, functional specification & maybe undocumented)**.
- Functional **“what”** the system does.
- Functional test based on the functions and features applied at all test level.
- Testing mainly involves “black box” testing and it is not concerned about the source code of the application.

11. What is Non-Functional system testing?

- Non-functional testing the attributes of a component or system that do not related to functionality.
- **(For example: - reliability, efficiency, usability, interoperability, maintability, and portability).**
- It is the testing of **“how”** the system works.
- Non-functional testing maybe performed at all test levels.

12. What is GUI Testing?

- GUI (Graphical User Interface) testing is the process of testing the system GUI testing involves checking the screen with the control like menu, buttons, icons and all types of bar such tool bar, navigation bar, etc.

13.What is Adhoc testing?

- Adhoc testing is an informal testing type with to break the system.
- It does not follow any test design technique to create the test cases.
- This testing is primarily performed if the knowledge of tester in the system under test is very high.
- The application as test for without any test cases or any business requirement documents.
- The main purpose of this testing is to find out the defect.

14.What is load testing?

- Load Testing is to test the system behavior under normal workload conditions, and it is just testing or simulating with the actual workload.
- Load testing identifies the bottlenecks in the system under various workloads and checks how the system reacts when the load is gradually increased.
- Load testing does not break the system.

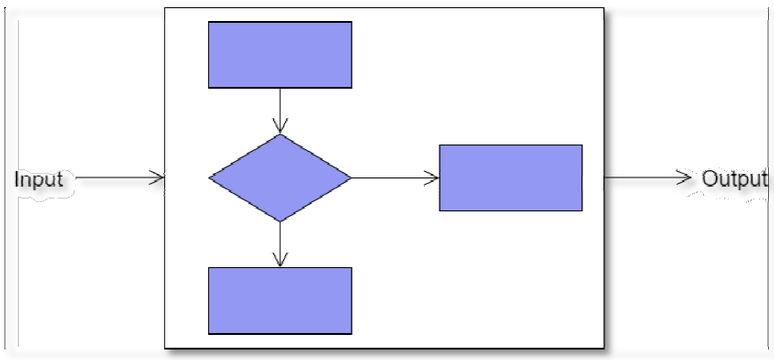
15.What is stress Testing?

- Stress testing is to test the system behavior under extreme conditions and is carried out till the system failure.
- Stress testing determines the breaking point of the system to reveal the maximum point after which it breaks
- Stress testing tries to break the system by testing with overwhelming data or resources.

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

- The white box testing based on an analysis of the internal structure of the component or system.
- Structure based testing technique is also known as “white box” or “glass box” testing.
- The tester requires knowledge of how the software is implemented how it’s works.

- For example, a structure technique may be concerned with exercising loop in the software.



- Technique of white box testing :-
    - Statement coverage
    - Decision coverage
    - Condition coverage

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

- Black box testing either functional or non-functional without reference to the internal structure of the component system.
- Specification based on the system technique is also known as “Black-box” or input/output driven testing.
- The testers have no knowledge of how the system or component is structure inside the box.
- For example :-

- Technique of black box testing
    - Equivalence partitioning
    - Boundary value analysis
    - Decision table
    - State transition testing

18.Mention what big bang testing is?

- Big bang testing has an advantage that everything is finished before integration testing starts.
- Advantages :-
    - It is small convenient for system.
    - Simple and straight forward approach.
    - It can be completed quickly.
    - Does not require a lot of planning or coordination
- Disadvantages :
    - Not good for long project.
    - Fault localization is difficult.
    - The number of interfaces that need to be tested in this approach, some interfaces links to be tested could be missed easily.
    - All modules are tested at once high risk critical modules are not isolated and tested on priority.

19.What is the purpose of exit criteria?

- The purpose of the exit criteria is to define when we stop testing.
- Successful testing of integration application.
- Executed test cases are documented.
- All high prioritized bugs fixed and closed.
- End of all testing **(For example: Product go for live).**
- End of phases of testing **(For example: Hand over from system test to UAT).**

20.When should "Regression Testing" be performed?

- When the system is stable and the system or the environment changes.
- When testing bug-fix releases as part of the maintenance phase.
- It should be applied at all Test Levels.
- It should be considered complete when agreed completion criteria for regression testing have been met.
- Regression test suites evolve over time and given that they are run frequently are ideal candidates for automation.

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

- Principle of software testing :-
    - Testing shows presence of defects
    - Exhaustive testing is impossible
    - Early testing
    - Defect clustering
    - The pesticide paradox
    - Testing is context dependent
    - Absence of error fallacy
- **Testing shows presence of defect :-**
    - Software testing reduces the presence of defects.
    - Software testing talks about the presence of defect and doesn’t talk about the absence of defect.
    - Even multiple testing can never ensure that software is 100% bug free.
- **Exhaustive testing is impossible :-**
    - Exhaustive testing is impossible mean software can never test cases.
    - It can test only some test cases and assume that software is correct and it will produce the correct output in every test case.

- **Early testing :-**
    - The defect detected in early phases of SDLC will very less expensive.
    - For better performance of software, start software testing will stat at initial phase.
- **Defect clustering :-**
    - Defect clustering a small number of the module can contain most of the defects.
    - For example, the software testing state that 80% of software defects comes from 20% of modules.
- **Pesticide paradox :-**
    - Pesticide paradox means the test cases again and again will not find new bugs.
    - It is necessary to review the test cases and add or update test cases to find new bugs.
- **Testing is context dependent :-**
    - Testing context dependent means it is depends on the context of software development.
    - Different types of software needs perform different types of testing.
    - The testing of the e-commerce site is different from the testing of the Android application.
- **Absence of error fallacy :-**
    - Error fallacy is built is 99% of bug free but it does not follow the user requirement then is possible.
    - It is not only necessary that software is 99% bug-free but it also mandatory to fulfill the entire customer requirement.

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

QA	QC	Tester
The implementation of process, producer and standard in context to verification of developed software	The verification of developed software with respect to documented requirement	The identification of bugs, error and defect in the software
Focus on process and producer actual testing on the system	Focus on executing software to identify bugs, defect through implemented of producers and process	Focus on actual testing
Process oriented activities	Product oriented activities	Product oriented activities
Preventive activities	Corrective activities	Preventive activities
Subset of software test life cycle	Subset of Quality Assurance	Subset of Quality control

23. Difference between Smoke and Sanity?

Smoke	Sanity
Smoke testing is performed to as certain that the critical functionality of the programs working.	Sanity testing is done to check the new functionality.
The objective of this testing is to verify “stability” of the system in order to with more rigorous testing.	The objective of this testing is to verify the “rationality” of the system in order performed.
This testing is usually documented or scripted.	This testing is usually documented and unscripted.
Smoke testing is performed by developers or testers.	Sanity testing is performed by testers.
Smoke testing is a subset of the “**Regression**” testing.	Sanity testing is a subset of the “**Acceptance**” testing.
Smoke testing exercise the entire system end to end.	Sanity testing exercise only the particular component of the entire system.
Smoke testing is like **‘General Health Check-up’**	Sanity testing is like **‘Specialized Health Check-up’.**

24.Difference between verification and Validation.

Criteria	Verification	Validation
Definition	The process evaluating work product (not actual final product).	The process of evaluating software during or at end of development.
Objective	Product is being built according to the requirement and design specification.	Product actually meets the user's needs and that the specification were correct in the first phase.
Question	Are we building the product right.	Are we building right product.
Evolution item	Plans, requirement, design, code, test.	The actual product / software.
Activities	Reviews, walk through, inspection.	Testing.

25.Explain types of Performance testing.

- Types of performance testing:
  - Load testing
  - Stress testing
  - Volume testing
  - Scalability testing
- Load testing :-
  - Load testing is to test the system behavior under normal work load condition. It is just testing or simulating with the actual workload.
  - Load testing does not break the system.
- Stress testing :-
  - Stress testing is to the system behavior under extreme conditions and is carried out till the system failure.
  - Stress testing tries to break the system by testing with overwhelming data or sources.

- Volume testing :-
    - Testing which confirms that any values that may become large over time (such as accumulated counts, logs, and data files), can be accommodated by the program and will not cause the program to stop working or degrade its operation in any manner.
- Scalability testing :-
    - Part of the battery of non-functional tests which tests a software application for measuring its capability to scale up - be it the user load supported, the number of transactions, the data volume etc. It is conducted by the performance engineer.

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

- A mistake in coding is called error, error found by tester is called defect, defect accepted by development team then it is called bug, build does not meet the requirement then it is failure.

27. Explain the difference between Functional testing and Nonfunctional testing.

Functional	Non-functional
Functional testing is performed using functional specification provided by the client and verifies the system.	Non-functional testing check the performed reliability, scalability, and non-functional.
Functional testing is executed first.	Non-functional testing should be performed after functional test.
Manual or automation tools can be used	Using tools will be effective for this.
Business requirement are the inputs.	Performance parameters like speed, scalability, are inputs non-functional.
What the product does.	How good the product work.
Easy to manual testing.	Tough to manual testing.
Types of functional unit testing Unit testing Smoke testing Sanity testing Integration testing White box testing Black box testing	Types of functional unit testing Performance Load Volume Stress Security Penetration Compatibility Migration

28.What is difference between the STLC (Software Testing Life Cycle) & SDLC (Software Development Life Cycle)

SDLC (Software Development Life Cycle)	STLC (Software Testing Life Cycle)
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.
SDLC the business analyst gathers the requirements and create Development Plan	STLC, the QA team analyze requirement documents like functional and non-functional documents and create System Test Plan
SDLC, the development team creates the high and low-level design plans	STLC, the test analyst creates the Integration Test Plan
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
SDLC phase also includes post-deployment supports and updates.	Testers, execute regression suits, usually automation scripts to check maintenance code deployed.
SDLC phases : Requirement gathering Analysis Design Implementation Testing Maintenance	STLC phases : Requirement analysis Test case planning Test case development Test environment setup Test execution Test cycle closure

29.What is difference between Test Scenario, Test Cases and Test Script.?

- Test scenario :-
  - A test scenario is any functionality that can be tested. It is call test condition. Test scenario is “what” to be tested. The scenario is derived from use cases.
- Test cases :-
  - The test case involves the set of steps, condition and inputs which can be used while performing the testing task. The test case is “How” to be tested. The cases are derived from test scenario.

- Test script :-
    - Set of sequential instruction that details how to execute a core business function. One script is written to explain how to simulate each business scenario.

30. Explain what Test plan? What is the information that should be covered?

- Test planning is STLC is a phase in which a senior QA manager determines the test plan strategy along with efforts and cost estimates for the project.
- The test plan get prepared and finalized in the same phase.
- This includes defining test objectives, test approach, test tools, test environment, test schedules and team responsibilities and composition.

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

- Various types of Agile methodologies :
    - Customer satisfaction through early and continuous software delivery.
    - Accommodate changing requirement throughout the development process.
    - Frequent delivery of working software.
    - Collaboration between the business stack holder and developers throughout.
    - Support, trust and motivate the people involved.
    - Enable face to face interaction.
    - Working software is the primary measure of progress.
    - Agile process to support a consistent development pace.
    - Attention to technical detail and design enhance agility.
    - Simplicity.
    - Self organizing teams encourage great architecture, requirement and design.
    - Regular reflection on how to become more effective.

32.When to used Usability Testing?

- This testing mainly focuses on the user's-ease to use the application, flexibility in handling controls and ability of the system to meet its objectives.
- This testing is recommended during the initial design phase of SDLC, which gives more visibility on the expectations of the users.
- There are two methods available to do usability testing –
    - Laboratory Usability Testing
    - Remote Usability Testing

33.What is the procedure for GUI Testing?

- GUI elements size, position, width, length and acceptable of characters or number.
- Check the execute the intended functionality of the application using GUI.
- Check error message displayed correctly.
- Check demarcation of different section screen.
- Check the font in application is readable.
- Check the alignment of the text is proper.
- Check the color of the font and warning message is aesthetically pleasing.
- Check the image has good clarity.
- Check the images are properly aligned.
- Check the positioning of GUI elements for different screen resolution.

34.To create HLR & Test case of

a. **(Instagram, Facebook) only first page.**
b. **Facebook login page.**

35.To create HLR & Test case of web based (Whatsapp web, Instagram).
36.Write a scenario of only Whatsapp chat messages.
37.Write a Scenario of Pen.
38.Write a Scenario of Pen stand.
39.Write a Scenario of Door.
40.Write a Scenario of ATM.
41.Write a Scenario of Microwave Owen.
42.Write a Scenario of Coffee vending machine.
43.Write a Scenario of Chair.
44.To Create Scenario (Positive & Negative).

a. **Gmail (Receiving mail)**
b. **Online shopping to buy product (flipkart)**

45.Write a Scenario of Wrist Watch.
46.Write a Scenario of Lift (Elevator).
47.Write a Scenario of whatsapp Group (generate group).
48.Write a Scenario of Whatsapp payment.