1 What is Exploratory Testing?

Ans – Exploratory testing is a type of testing where test cases are not created in advance but testers checks system on the fly. In this types of testing the tester is free to select any methodology to test the software.

2 What is traceability matrix?

Ans- Traceability matrix is a table type document that is used in the development of software application to trace requirements. It can be used for both forward (from Requirements to Design or Coding) and backward (from Coding to Requirements) tracing. It is also known as **Requirement Traceability Matrix (RTM) or Cross Reference Matrix (CRM).**

3 What is Boundary value testing?

Ans - Software testing technique in which tests are designed to include representatives of boundary values. It is performed by the QA testing teams.

4 What is Equivalence partitioning testing?

Ans - Software testing technique that divides the input data of a software unit into partitions of data from which test cases can be derived. it is usually performed by the QA teams.

5 What is Integration testing?

Ans - The phase in software testing in which individual software modules are combined and tested as a group. It is usually conducted by testing teams

6 What is functional system testing?

Ans - Type of black box testing that bases its test cases on the specifications of the software component under test. It is performed by testing teams.

7 What is Non-Functional Testing?

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

8 What is GUI Testing?

Ans The process of testing a product that uses a graphical user interface, to ensure it meets its written specifications. This is normally done by the testing teams

9 What is Adhoc testing?

Ans : Testing performed without planning and documentation - the tester tries to 'break' the system by randomly trying the system's functionality. It is performed by the testing teams

10 What is load testing?

Ans -Testing technique that puts demand on a system or device and measures its response. It is usually conducted by the performance engineers.

11 What is stress Testing?

Ans Testing technique which evaluates a system or component at or beyond the limits of its specified requirements. It is usually conducted by the performance engineer.

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

Ans Testing technique based on knowledge of the internal logic of an application’s code and includes tests like coverage of code statements, branches, paths, conditions. It is performed by software developers.

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

Ans : A method of software testing that verifies the functionality of an application without having specific knowledge of the application's code/internal structure. Tests are based on requirements and functionality. It is performed by QA teams.

14 Mention what are the categories of defects?

Ans – There are 4 categories of defect

- blocked defect

- major defect

- minor defect

- critical defect

15 Mention what bigbang testing is?

Ans -: Testing technique which integrates individual program modules only when everything is ready. It is performed by the testing teams.

16 What is the purpose of exit criteria?

Ans – The purpose of exit criteria is to define when we stop testing either at the end of all testing or end of phase of testing

17 When should "Regression Testing" be performed?

Ans - Regression testing is typically performed when there are changes or updates made to a software system. Its purpose is to ensure that the modifications made to the system do not introduce new defects or regressions, meaning that previously working features have not been affected by the changes.

18 What is 7 key principles? Explain in detail?

Ans –

1. Testing shows presense of defect : Testing can show that the defect are presense but can’t prove that there are no defect.Testing reduces the probability of undiscovered defects remaining in the software but even if no defect are found ,it is not a proof of correctness.
2. Exhaustive testing is impossible : Testing everything including all combination of input and preconditions is not possible.so,instead of doing exhaustive testing we can use risk and priorities to focus testing efforts.
3. Early testing : Testing activities should starts as early as possible in the SDLC . It starts from requirement stages.
4. Defect clustering : Most defect found during testing are usually confined to a small number of module.in short defect are not comes in separately it comes in cluster.
5. Pesticide paradox : If we are executing the same set of test cases again and again over a perticuler time,then this kind of test will not be able to find the new bug in software.It is important to review all the test case frequently and the new and different test are necessary to be written for the implementation of multipe parts of the application which helps to find more bugs.
6. Context dependent : Testing is done diffrerently in different context.Different kind of sites test differently
7. Absence of error fallacy : If the system does not fullfill customer’s need and expectations at that time finding defects and fixing does not helps.even after defect have been resolved it may still be unusable does not fullfill the needs of user.

19 Difference between QA v/s QC v/s Tester

Ans -

|  |  |  |
| --- | --- | --- |
| QA | QC | TESTING |
| * QA is a process oriented activities. * It is a subset of STLC * It is preventive activities * Focus on process rather than conducting actual testing on the system * Verifies the quality. | * QC is a product oriented activities * It is a subset of QA * It is a corrective process * Focus on actual testing by executing the software with aim to identify bug through implementation process * Validates the quality | * TESTING is a product oriented ativities * IT is a subset of QC * It is preventive process * Focus on actual testing * Validates the quality |

20 Difference between Smoke and Sanity?

Ans -

|  |  |
| --- | --- |
| Smoke | Sanity |
| * Smoke Testing has a goal to verify “stability” * Smoke Testing is done by both developers or testers * Smoke testing is a subset of acceptance testing * Smoke Testing verifies the critical functionalities of the system * Smoke testing verifies the entire system from end to end | * Sanity Testing has a goal to verify “rationality”. * Sanity Testing is done by testers. * Sanity Testing verifies the new functionality like bug fixes. * Sanity testing is a subset of Regression Testing. * Sanity Testing verifies only a particular component. |

21. Difference between verification and Validation

|  |  |
| --- | --- |
| Verification | Validation |
| The verifying process includes checking documents, design, code, and program | It is a dynamic mechanism of testing and validating the actual product |
| It does not involve executing the code | It always involves executing the code |
| Verification uses methods like reviews, walkthroughs, inspections, and desk- checking etc. | It uses methods like Black Box Testing and non-functional testing |
| Whether the software conforms to specification is checked | It checks whether the software meets the requirements and expectations of a customer |
| It finds bugs early in the development cycle | It can find bugs that the verification process can not catch |
| Target is application and software architecture, specification, complete design, high level, and database design etc. | Target is an actual product |
| QA team does verification and make sure that the software is as per the requirement in the SRS document. | With the involvement of testing team validation is executed on software code. |
| It comes before validation | It comes after verification |

22. Explain types of Performance testing.

Ans - Load testing – checks the application’s ability to perform under anticipated user loads. The objective is to identify performance bottlenecks before the software application goes live.

* Stress testing – involves testing an application under extreme workloads to see how it handles high traffic or data processing. The objective is to identify the breaking point of an application.
* Endurance testing – is done to make sure the software can handle the expected load over a long period of time.
* Spike testing – tests the software’s reaction to sudden large spikes in the load generated by users.
* Volume testing – Under Volume Testing large no. of. Data is populated in a database, and the overall software system’s behavior is monitored. The objective is to check software application’s performance under varying database volumes.
* Scalability testing – The objective of scalability testing is to determine the software application’s effectiveness in “scaling up” to support an increase in user load. It helps plan capacity addition to your software system.

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

Ans - When we have some coding error, it leads a program to its breakdown, which is known as a bug.

When the application is not working as per the requirement is knows as defects.

The Problem in code leads to errors, which means that a mistake can occur due to the developer's coding error as the developer misunderstood the requirement or the requirement was not defined correctly.

Many defects lead to the software's failure, which means that a loss specifies a fatal issue in software/ application or in its module, which makes the system unresponsive or broken.

24 - Difference between Priority and Severity

Severity refers to the level of impact a defect has on the software’s functionality. It measures how severe the defect is and how much it affects the software’s ability to perform its intended function.

Priority is the order in which problems should be fixed based on how much they affect the business. A defect with a high priority needs to be fixed right away, while a defect with a low priority can be fixed later.

25 - What is Bug Life Cycle?

Ans - Defect Life Cycle or Bug Life Cycle in software testing is the specific set of states that defect or bug goes through in its entire life.

## Defect States Workflow

The number of states that a defect goes through varies from project to project. Below lifecycle diagram, covers all possible states

* **New:** When a new defect is logged and posted for the first time. It is assigned a status as NEW.
* **Assigned:** Once the bug is posted by the tester, the lead of the tester approves the bug and assigns the bug to the developer team
* **Open**: The developer starts analyzing and works on the defect fix
* **Fixed**: When a developer makes a necessary code change and verifies the change, he or she can make bug status as “Fixed.”
* **Pending retest**: Once the defect is fixed the developer gives a particular code for retesting the code to the tester. Since the software testing remains pending from the testers end, the status assigned is “pending retest.”
* **Retest**: Tester does the retesting of the code at this stage to check whether the defect is fixed by the developer or not and changes the status to “Re-test.”
* Verified: The tester re-tests the bug after it got fixed by the developer. If there is no bug detected in the software, then the bug is fixed and the status assigned is “verified.”
* Reopen: If the bug persists even after the developer has fixed the bug, the tester changes the status to “reopened”. Once again the bug goes through the life cycle.
* Closed: If the bug is no longer exists then tester assigns the status “Closed.”
* Duplicate: If the defect is repeated twice or the defect corresponds to the same concept of the bug, the status is changed to “duplicate.”
* Rejected: If the developer feels the defect is not a genuine defect then it changes the defect to “rejected.”
* Deferred: If the present bug is not of a prime priority and if it is expected to get fixed in the next release, then status “Deferred” is assigned to such bugs
* Not a bug: If it does not affect the functionality of the application then the status assigned to a bug is “Not a bug”.

25 - Explain the difference between Functional testing and NonFunctional testing

Ans –

|  |  |
| --- | --- |
| Functional testing | NonFuctional testing |
| =) Functional testing verifies each function/ features of the software.  =) Functional testing can be done manually  =)Functional testing is based on customer’s requirement  =)Functional testing has a goal to validate software actions.  =)Functional describes what the product does | =) Nonfunctional testing verifies non functional aspacts like performance,usability etc  =) Nonfunctional testing is hard to perform manually  =)Nonfunctional testing is based on customer’s expectations  =)Non Functional testing has a goal to validate the performance of the software.  =) Non Functional describes how the product works. |

26 -What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)?

Ans -

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

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

Test Scenario: A Test Scenario is defined as any functionality that can be tested. It is also called Test Condition or Test Possibility.

Test Case: A **Test Case** is a set of actions executed to verify a particular feature or functionality of your software application. A Test Case contains test steps, test data, precondition, postcondition developed for specific test scenario to verify any requirement.

Test Script: Test Scripts are a line-by-line description containing the information about the system transactions that should be performed to validate the application or system under test.

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

Test Plan - A **Test Plan** is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product. Test Plan helps us determine the effort needed to validate the quality of the application under test.

29- What is priority?

Priority: Priority is defined as the order in which the defects should be resolved.

30 - What is severity?

Severity: Bug Severity or Defect Severity in testing is a degree of impact a bug or a Defect has on the software application under test.

31- Bug categories are…

Critical, High, Medium and Low

32- Difference between priority and severity

Severity refers to the level of impact a defect has on the software’s functionality. It measures how severe the defect is and how much it affects the software’s ability to perform its intended function.

Priority is the order in which problems should be fixed based on how much they affect the business. A defect with a high priority needs to be fixed right away, while a defect with a low priority can be fixed later.

33- What are the different Methodologies in Agile Development Model?

Kanban, Scrum, Extreme programming, Adaptive project framework