Software Testing Assignment

Module.2 - Manual Testing

Q-1. What is Exploratory Testing?

Ans: In exploratory testing tester focuses more on how the software actually works, testers do minimum planning and maximum execution of the software by which they get in depth idea about the software functionality, once the tester starts getting insight into the software he can make decisions to what to test next. Exploratory testing is mostly used if the requirements are incomplete and time to release the software is less.

Exploratory testing is a concurrent process where Test design, execution and logging happen simultaneous. Testing is based on a test charter that may include

* Scope of the testing (in and out)
* The focus of exploratory testing is more on testing as a “thinking” activity.
* A brief description of how tests will be performed
* Expected problems Is carried out in time boxed intervals
* More structured than Error guessing

Q-2. What is traceability matrix?

Ans: Traceability Matrix (also known as Requirement Traceability Matrix - RTM) is a table which is used to trace the requirements during the Software development life Cycle. It can be used for forward tracing (i.e. from Requirements to Design or Coding) or backward (i.e. from Coding to Requirements). There are many user defined templates for RTM.

Types of Traceability Matrix

* Forward Traceability – Mapping of Requirements to Test cases
* Backward Traceability – Mapping of Test Cases to Requirements
* Bi-Directional Traceability - A Good Traceability matrix is the References from test cases to basis documentation and vice versa

Pros

* Make obvious to the client that the software is being developed as per the requirements.
* To make sure that all requirements included in the test cases
* To make sure that developers are not creating features that no one has requested • Easy to identify the missing functionalities.

Cons

* No traceability or Incomplete Traceability Results into:
* Poor or unknown test coverage, more defects found in production

Q-3. What is Boundary value testing?

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

* Boundary value analysis is a methodology for designing test cases that concentrates software testing effort on cases near the limits of valid ranges
* Boundary value analysis is a method which refines equivalence partitioning.
* Boundary value analysis generates test cases that highlight errors better than equivalence partitioning.
* The trick is to concentrate software testing efforts at the extreme ends of the equivalence classes.

This is one of the software testing technique in which the test cases are designed to include values at the boundary. If the input data is used within the boundary value limits, then it is said to be Positive Testing. If the input data is picked outside the boundary value limits, then it is said to be Negative Testing.

Q-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. In this method the input domain data is divided into different equivalence data classes. This method is typically used to reduce the total number of test cases to a finite set of testable test cases, still covering maximum requirements. In short it is the process of taking all possible test cases and placing them into classes. One test value is picked from each class while testing.

* This is a software testing technique which divides the input date into many partitions .
* Values from each partition must be tested at least once. Partitions with valid values are used for Positive Testing.
* While, partitions with invalid values are used for negative testing.

Equivalence partitioning is the process of defining the optimum number of tests by:

* Reviewing documents such as the Functional Design Specification and Detailed Design Specification, and identifying each input condition within a function,
* Selecting input data that is representative of all other data that would likely invoke the same process for that particular condition
* If we want to test the following IF statement: “If value is between 1 and 100 (inclusive) (e.g value >=1 and value <=100) Then...”

Q-5. What is Integration testing?

Ans: Integration testing is associated with the architectural design phase. Integration tests are performed to test the coexistence and communication of the internal modules within the system.

Integration 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.
* The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.
* Integration testing tests integration or interfaces between components, interactions to different parts of the system such as an operating system, file system and hardware or interfaces between systems.
* Integration testing is done by a specific integration tester or test team.
* Components may be code modules, operating systems, hardware and even complete systems
* There are 2 levels of Integration Testing

1. Component Integration Testing
2. System Integration Testing

* Integration Testing is performed after Unit Testing and before System Testing.
* Either Developers themselves or independent Testers perform Integration Testing.

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

Q-6. What determines the level of risk?

Ans: A facor that could result in future negative consequences , usually expressed as impact ans likelihood. Risks should be prioritised according to their level, which is obtained by assessing the likelihood of the event occurring and the impact of that event.

* Then the residual level should be determined by considering the management response to the risk.

Q-7. What is Alpha testing?

Ans: Alpha Testing is performed and carried out at the developing organizations location (virtual environment) by developers (sometimes by independent tester).Alpha Testing is definitely performed and carried out at the developing organization location with the involvement of developers. It is the form of Acceptance Testing.

* Alpha Testing (Also known as intenal acceptance testing) is performed by members of the organization that developed the software but who are not directly involved in the project (Development or Testing).
* Usually, it is the members of Product Management, Sales and/or Customer Support

Q-8 What is beta testing?

Ans: Beta testing (Also known as User Acceptance Testing) is performed by the end users of the software. They can be the customers themselves or the customers’ customers. User Acceptance Testing

* Beta testing can be considered “pre-release” testing.
* Pilot Testing is testing to product on real world as well as collect data on the use of product in the classroom
* It is only a kind of Black Box Testing.
* Beta Testing is always performed at the time when software product and project are marketed.

Q-9. What is component testing?

Ans: component – A minimal software item that can be tested in isolation. It means “A unit is the smallest testable part of software”.

* Component testing- The testing of individual software components.

Unit testing is a level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.

Q-10. What is functional system testing?

Ans: Functional system testing: A requirement that specifies a function that a system or system component must perform.

1. Requirement Based Functional Testing
2. Process Based Testin

A requirement may exist as a text document and/or a model

Q-11. What is Non-functional testing?

Ans: 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.

Testing technique which focuses on testing of a software application for its non-functional requirements. Can be conducted by the performance engineers or by manual testing teams.

May be performed at all Test levels

Measuring the characteristics of the system /software that can be quantified on varying scale – e.g. performance test scaling.

Q-12. What is GUI Testing?

Ans: Graphical user interface testing is process of testing the system’s GUI of the System under testing involves checking the screens with the controls like menus, buttons, icons, and all type of bars – toolbar menu bar, dialog boxes and windows etc.

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

Check all the GUI elements for size position width, length and acceptance of characters or numbers, for instance, you must be able to provide inputs to the fields.

Q-13. What is adhoc testing?

Ans: Adhoc testing is an informal testing type with an aim to break the system. Testers randomly test the application without any test cases or any business requirement document.Adhoc Testing does not follow any structured way of testing and it is randomly done on any part of the application.

Main aim of this testing is to find defects by random checking.

* Adhoc testing can be achieved with the testing technique called Error Guessing.
* Error guessing can be done by the people having enough experience on the system to “guess” the most likely source of errors.

Q-14. What is load testing?

Ans: Load testing – It’s 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 is a kind of performance testing which determines a system’s performance under real -life load conditions. This testing helps determine how the application behaves when multiple users access it simultaneously.

This testing usually identifies :–

* The maximum operating capacity of an application
* Determine whether current infrastructure is sufficient to run the application
* Sustainability of application with respect to peak user load

Q-15. What is stess testing?

Ans: 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, and continuous input to system or database load.

Stess testing is used to test the stability & reliability of the system this test mainly determines the system on its robustness and error handling under extremely.

Q-16) What is white box testing and list the types of white box testing ?

Ans: White Box Testing: 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 technique because here the tester require knowledge of how the software is implemented, how the software is implemented, how it works.

* + White Box Testing is testing based on an analysis of the internal structure of the component or system.
  + It is the detailed investigation of internal logic and structure of the code.
  + The testers require knowledge of how the software is implemented, how it works.
  + TYPES OF WHITE BOX TEST
  + Statement coverage
  + Branch Coverage
  + Decision Coverage

Q-17) What is black box testing? what are different black box testing techniques ?

Ans: Black-box testing is testing of either functional or non-functional, without reference to the internal structure of the component or system.

* TYPES OF BLACK BOX TEST
* Equivalence Partitioning
* Boundary Value Analysis
* Decision Tables
* State Transition Testing
* Use Case Testing

Q-18) Mention what are the categories of defects ?

Ans: Defect is the variance from a desired product attribute (it can be a wrong missing or extra data).

* Data Quality/Database Defects
* Critical Functionality Defects
* Functionality Defects

Q-19) Mention what Bigbang testing is?

Ans: In Big Bang integration testing all components or modules is integrated simultaneously, after which everything is tested as a whole .

* Big-bang integration testing is a type of integration testing that combines all the modules or components of a system into a single unit and tests them as a whole.
* It does not use any intermediate stages or stubs to simulate the behavior of missing or incomplete modules.

Q-20) What is the purpose of exit criteria ?

Ans: Purpose of exit criteria is to define when we STOP testing either at the:

* End of all testing – i.e. product Go live
* End of phase of testing
* 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.
* Exit criterion should be part of test plan and decided in the planning stage.
  + Executed Test Cases are documented.
  + All High prioritized bugs are fixed and closed.
  + Technical documents have been submitted.

Q-21) When should "Regression testing" be performed ?

Ans: Regression Testing: Testing of a previously tested program following modification to ensure that defects have not been introduced or uncovered in unchanged areas of the software, as a result of the changes made. It is performed when the software or its environment is changed.

* Typically, regression testing is applied under these circumstances
* A new requirement is added to an existing feature.
* A new feature or functionality is added.
* The codebase is fixed to solve defects.
* A new version of the software is released.

Q-22) What is 7 key principles? Explain in details

Ans: (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.
* We test to find Faults.
* As we find more defects, the probability of undiscovered defects remaining in a system reduces.
* However Testing cannot prove that there are no defects present.

(2) Exhaustive Testing is Impossible!

* Testing everything including all combinations of inputs and preconditions is not possible.
* Instead of doing the exhaustive testing we can use risks and priorities to focus testing efforts.
* Accessing and managing risk is one of the most important activities and reason for testing in any project.

(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.
* Defects are not evenly spread in a system, they are ‘clustered’

(5) The Pesticide Paradox

* If the same tests are repeated over and 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.

(6) Testing is Context Dependent

* Testing is basically context dependent.
* Testing is done differently in different contexts
* Different kinds of sites are tested differently like Gaming website vs social media vs shopping website.

(7) Absence of Errors Fallacy

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

Q-23) Difference between QA/QC/Tester

|  |  |  |
| --- | --- | --- |
| Quality Assurance (QA) | Quality Control (QC) | Testing |
|  |  |  |
|  |  |
| Process-oriented focuses on making the process of creating software better. | A product-oriented approach is a way to make sure the software meets all its requirements. | Testing the software system is about finding any mistakes or issues. |
| It works with the development process to help stop mistakes and ensure the software is of good quality.This means setting up and keeping standards, processes, procedures, and tools in place to ensure we’re consistently producing high-quality software. | It’s done after the development process and involves running test cases and seeing how the software reacts. | This usually happens after the software has been created, and it’s all about ensuring that the software’s quality is up to standard. |
| The goal is to keep improving our software development process for the best possible results. | The goal is to find any defects or errors in the software and fix them. | It involves running tests and looking at what comes out of them, finding any problems with the software, and ensuring that it does everything it’s supposed to do. |

Q-24) Difference between Smoke and Sanity testing

* + Smoke tesing is performed to varify most critical functionality of software is working fine but Sanity test is performed to check if bugs have been fixed or new functionality added is woring properly or not.
  + Smoke test is part of regression test, Sanity test is part of acceptance test.
  + Smoke test is exercised on entire system from end to end while Sanity test is performed on specific component of entire system.
  + Objective of Smoke test is check stability and Sanity test for rationality.

Q-25) Difference between verification and validation

Ans: Verification include activities like walkthrough, inspection , review while validation include testing activity.

* while The process of evaluating software during or at the end of the development process to determine whether it satisfies specified business requirements.
  + Objective of verification is To ensure that the product is being built according to the requirements, but objective for validation is Product actually meet customer requirements/satisfy needs.

Q-26) Explain type of performance testing

Ans: A software application’s performance like its response time, do matter.

* The goal of performance testing is not to find bugs but to eliminate performance bottlenecks.
* TYPES OF PERFORMANCE TESTING
* Load testing: Load testing is a kind of performance testing which determines a system’s performance under real-life load conditions. This testing helps determine how the application behaves when multiple users access it simultaneously.
* Stress testing: Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load.
* Endurance testing
* Spike testing
* Volume testing
* Scalability testing

Q-27) What is Error, defects, bug, and failure ?

Ans: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 requirements then it is failure.

Q-28) Difference between Priority and Severity

Ans: Priority :-

* Priority is the order in which developer has to fix the bug.
* If high priority is mentioned then the developer has to fix it at the earliest.
* The priority status is set based on the customer requirements.

Severity :-

* Severity is how seriously the bug is affecting the application.
* The severity type is defined by the tester based on the written test cases and functionality.

Q-29) What is Bug life cycle?

Ans: Bug life cycle is nothing but the various phases a bug under goes after it is raised or

reported.

The different phases of Bug life cycle are,

* + New or Opened
  + Assigned
  + Fixed
  + Tested
  + Closed
  + The duration or time span between the first time defects is found and the time that it is closed successfully, rejected, postponed or deferred is called as ‘Defect Life Cycle’.

Q-30) Explain the difference between Functional testing and non-functional testing { No idea }

Ans: Functional Testing:

* Testing based on an analysis of the specification of the functionality of a component or system.
* ‘Specification’ – E.g. Requirements specification, Use Cases, Functional specification or maybe undocumented.
* ‘Function’ – what the system does
* Functional test – based on the Functions and features – may be applied at all Test levels (e.g. Component Test, System Test etc.)
* Considers the external (not internal) behaviour of the software. Black- Box testing. What it does rather than how it does it. More on this later!
* Functional testing verifies that each function of the software application operates in conformance with the requirement specification.

Q-31) What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle) ?

|  |  |  |
| --- | --- | --- |
| **\*** | **SDLC** | **STLC** |
| 1 | SDLC is mainly related to software development | STLC is mainly related to software testing |
| 2 | Besides development other phases like testing also included | It focuses only on testing the software |
| 3 | SDLC involves a total six phases or steps. | STLC involves only five phases or steps. |
| 4 | In SDLC , more number of members (developers) are required for the whole process | In STLC, less number of members (testers) are needed. |
| 5 | In SDLC development team makes the plans and designs based on the requirements | In STLC testing team (Test Lead or Test Architect ) makes the plans and designs. |
| 6 | Goal of SDLC is to complete the successful development of software. | Goal of STLC is to complete the successful testing of software. |
| 7 | It helps in developing good quality software. | It helps in making the software defects-free. |
| 8 | SDLC phases are completed before the STLC phase. | STLC phase are performed after SDLC phase. |
| 9 | Post-deployment support, enhancement and udate are to be included if necessary. | Regression tests are run by QA team to check deployed maintenance code and maintenance code and maintain test cases and automated scripts. |

* + STLC shows step by step process of how the software testing process will be carried out, while SDLC is step by step process of how the softwre product will be developed.
  + STLC start after SDLC.
  + At the end of SDLC, product is handed over to testing team while at the end of STLC product is handed over to customer.

Q-33) What is the difference between test scenarios, test cases, and test script ?

|  |  |  |
| --- | --- | --- |
| Test Scenario | Test Case | Test Script |
| Is any functionality that ca be tested. | Is a set of actions executed to verify particular features or functionality. | Is a set of instructions to test an app automatically |
| Is derived from test artifacts like Business Requirement Specification (BRS) and Software Requirements Specification (SRS). | Is mostly derived from test scenarios. | Is mostly derived from test cases. |
| Helps test the end-to-end functionality in an Agile way. | Help in exhaustive testing of an app. | Helps to test specific things repeatedly. |
| Is more foused on what to test. | Is focused on what to test and how to test | Is focused on the expected result. |
| Take less time and fewer resources to create. | Requires more resources and time. | Require less time for testing but more resources for script creatin and updating. |
| Includes an end-to-end functionality to be tested. | Includes test steps, and data expected results for testing. | Includes different commands to develop a script. |
| The main task is to check the full functionality of a software application. | The main task is to verify compliance with the applicable standards, guidelines and customer requirements. | The main task is to verify that nothing is skipped, and the results are true as the desired testing plan. |
| Allows quickly assessing the testing scope. | Allows detecting errors and defects. | Allows carrying out an automatic execution of test cases. |

Q-34) Explain what Test Plan is ? What is the information that should be covered . { no idea }

Ans: It is a high level document in which how to perform testing is described. The Test Plan document is usually prepared by the Test Lead or Test Manager and the focus of the document is to describe what to test, how to test, when to test and who will do what test.

* Test Planning in STLC is a phase in which a Senior QA manager determines the test plan strategy along with efforts and cost estimates for the project.
* Information of the resources to be used, test environment, test limitations, and the testing schedule are also determined.
* The Test Plan gets prepared and finalized in the same phase.

Q-35) What is priority ?

Ans: Priority is the order in which developer has to fix the bug. If high priority is mentioned then the developer has to fix it at the earliest. The priority status is set based on the customer requirements.

* Priority is Relative and Business-Focused. Priority defines the order in which we should resolve a defect.
* The priority status is set based on the customer requirements.

Q-36) What is severity ?

Ans: Severity is how seriously the bug is affecting the application. The severity type is defined by the tester based on the written test cases and functionality

* Severity is absolute and Customer-Focused.
* It is the extent to which the defect can affect the software.
* In other words it defines the impact that a given defect has on the system.

Q-37) Bug categories are…

* Bug Categories
* Functional bugs
* Compatibility bugs
* Usability bugs
* Unit Level Bugs
* Logical Bugs
* Security

Q-38) Advantage of Bugzila

Ans: Bugzilla is a defect tracking tool, however it can be used as a test management tool as such it can be easily linked with other test case management tools like Quality Center, Testlink etc.

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.

* 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.

Q-39) Difference between priority and severity

Ans: Priority :- Priority is the order in which developer has to fix the bug. If high priority is mentioned then the developer has to fix it at the earliest. The priority status is set based on the customer requirements.

Severity:- Severity is how seriously the bug is affecting the application. The severity type is defined by the tester based on the written test cases and functionality

* 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.

Q-40) What are the different Methodologies in Agile Development Model ?

* Scrum
* Extreme Programming (XP)
* Feature-Driven Development (FDD)
* Adaptive Software Development (ASD)
* Kanban