**Software Testing Assignment**

**Module-2**

**1. What is software testing?**

* Software Testing is a process used to identify the correctness, completeness, and quality of developed computer software.

**2. What is exploratory testing?**

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

**3. What is traceability matrix?**

* Traceability matrix is a table, which is used to trace the requirements during the SDLC.

**4. What is boundary value testing?**

* Boundary testing is the process of testing between extreme ends or boundaries between partitions of the input values. So these extreme ends like Start- End, Lower- Upper, Maximum-Minimum, Just Inside-Just Outside values are called boundary values and the testing is called “boundary testing”.

**5. What is equivalence partitioning testing?**

* Equivalence partitioning (EP) is a specification-based or black box technique. It can be applied at any level of testing and is often a good technique to use first.

**6. What is integration testing?**

* Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems.

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

* The amount of testing planned before release of a system.
* The likelihood of an adverse event and the impact of the event.
* The probability that an adverse event will occur.
* The cost of dealing with an adverse event if it occurs.

**8. What is alpha testing?**

* Alpha Testing is always performed by the developers at the software development site. It is not open to the market and public and always performed in Virtual Environment.

**9. What is beta testing?**

* Beta Testing is always performed by the customers at their own site. It is always open to the market and public and performed in Real Time Environment.

**10. What is component testing?**

* Unit testing is a level of software testing where individual unit/component of software are tested. (Unit – is the smallest testable part of any software.)
* The testing of individual software component

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

* A requirement that specifies a function that a system or system component must perform called functional system testing.

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

* Testing of those requirements that do not relate to functionality called non-functional system testing.

**13. What is GUI Testing?**

* 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.

**14. What is Adhoc testing?**

* Adhoc testing is an informal testing type with an aim to break the system. Testing the application randomly without looking into the requirement.

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

* Testing based on an analysis of the internal structure of the component or system having knowledge of coding.

**White Box Testing Techniques / Types of Coverage**

* Statement / Segment coverage
* Decision / Branch coverage
* Condition coverage

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

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

**Techniques of Black Box Testing:-**

* Equivalence partitioning

Aim is to treat groups of inputs as equivalent and to select one representative input to test them all.

* Boundary value analysis

That concentrates soft. Testing efforts on cases near the limit of valid ranges.

* Decision tables

Is a soft. Testing technique used to test system behaviour for different input combinations.

* State transition testing

A transition between two states of a component or system.

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

* Data quality/ database defects
* Critical functionality defects
* Functionality defects
* Security defects
* User interface defects

**18. Mention what big bang testing is?**

* In Big Bang integration, testing all components or modules is integrated simultaneously.

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

* Purpose of exit criteria is to define when we stop testing either at the:

(1)End of all testing

(2)End of Phase Testing

**20. When should “Regression Testing" be performed?**

* We do regression testing whenever the production code is modified.
* We can perform regression testing in the following scenario, these are:
* When new functionality added to the application.
* When there is a Change Requirement.
* When the defect fixed.
* When there is a performance issue fix.
* When there is an environment change.

**21. What is seven key principles? Explain in detail?**

* Types of 7 key principles:-

1. Testing Shows Presence of Defects
2. Exhaustive testing is Impossible
3. Early Testing
4. Defect Clustering
5. The Pesticide Paradox
6. Testing is Context Dependent
7. Absence of Errors Fallacy

**Testing Shows Presence of Defects:-**

* Even multiple testing can never ensure that software is 100% bug free.
* Software testing reduces the presence of defects.
* We test to find faults.

**Exhaustive testing is Impossible:-**

* Exhaustive testing is impossible means the software can never test at every test cases

**Early Testing**

* The defect detected in early phases of SDLC will very less expensive.
* For better performance of software start software testing will start at initial phase.

**Defect Clustering**

* Defects are not evenly spread in a system.
* In a project, a small number of the module can contain most of the defects.

**The Pesticide Paradox**

* Repeating the same test cases again and again will not find new bug.
* It is necessary to review the test cases and add or update test cases to find new bugs.

**Testing is Context Dependent**

* Testing approach depends on context of software developed.
* Different types of software need to perform different types of testing.
* The testing of the e-commerce site is different from the testing of the android application.

**Absence of Errors Fallacy**

* If a built software is 99% bug-free but it does not follow the user requirement the it is unusable.
* It is not only necessary that software is 99% bug-free but it also mandatory to fulfil all the customer requirements.

**22. Difference between QA v/s QC v/s tester**

|  |  |  |
| --- | --- | --- |
| QA | QC | TESTER |
| QA is a set of activities for ensuring quality in the processes by which products are developed. | QC is a set of activities for ensuring quality in products. The activities focus on identifying defects in the actual product produced. | Activities, which ensure the identification of bug/error/defects in the software. |
| QA is preventing defects. | QC is correcting defects. | It is a preventive process. |
| Process oriented activies. | Product oriented activites. | Product oriented activitie. |
| Focus on processes. | Product as a whole. | Focuses on actual testing. |
| 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. |

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

|  |  |
| --- | --- |
| **Smoke Testing** | **Sanity Testing** |
| Smoke Testing is performed to ensure the critical functionalities/key features of the previous build. | Sanity Testing is done to ensure the new functionality / bugs have been fixed |
| The objective of this testing is to verify the & stability of the system in order to proceed with more rigorous testing | The objective of the testing is to verify the & rationality of the system in order to proceed with more rigorous testing |
| This testing is performed by the developers or testers | Sanity testing is usually performed by testers |
| Smoke testing is usually documented or scripted | Sanity testing is usually not documented and is unscripted |
| Smoke testing is a subset of Regression testing | Sanity testing is a subset of Acceptance testing |
| Smoke testing exercises the entire system from end to end | Sanity testing exercises 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**

|  |  |
| --- | --- |
| **Verification** | **Validation** |
| Verification means are we building the software right. | Verification means are we building the software right? |
| Verification is the static testing. | Verification is the dynamic testing. |
| Done by developers. | Done by testers. |
| Verification is done Before coding | Validation is done after codding |
| Verification phase:   * Requirement analysis * System design * Architecture design * Module design | Validation phase:   * Unit testing * Integration testing * Acceptance testing * System testing |

**25. Explain types of Performance testing.**

* Two Types of Performance Testing
  1. Load testing / Volume Testing / Scalability Testing
  2. Stress testing / Endurance Testing / Spike Testing

**26. What is Error, Bug, Defect and Failure?**

* Error: - A mistake in coding is called error.

Bug: - Defect accepted by development team then it is called bug.

Defect: - Error found by tester is called defect.

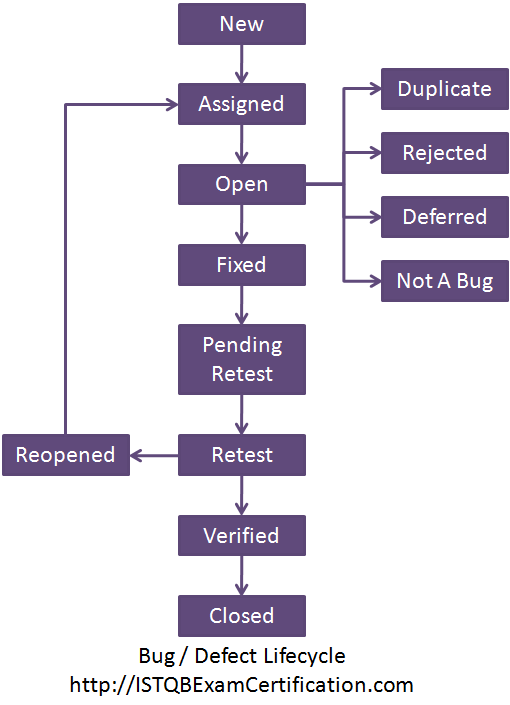
Failure: - Build does not meet the requirements then it is failure.

**27. Difference between Priority and Severity**

|  |  |
| --- | --- |
| **Severity** | **Priority** |
| Severity is associated with functionality | Priority is associated with scheduling |
| QA engineer determine the severity level | Priority of defect is consultation with the client |
| It indicate the seriousness of defect | It indicate how soon the bug should be fixed |
| Severity is driven by functionality | Priority is driven by business level |
| Severity levels are: Critical, major, minor, moderate &amp;  Cosmetic | Priority levels are: Critical, high, medium, low |

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

“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’.”



* **Assigned:** The bug is assigned to the Developer.
* **Tested:** The bug is tested by the Software tester.
* **Verified:** The bug is verified by the QA Lead.

**29. Explain the difference between Functional testing and Non-functional testing.**

|  |  |
| --- | --- |
| **Functional Testing** | **Non-Functional Testing** |
| Testing based on an analysis of the specification of the functionality of a component or system. | Testing the attributes of a component or system that do not relate to functionality. |
| Functional testing is executed first | It should be performed after functional testing |
| Describes what the product does | Describes how good the product works |
| Easy to do manual testing | Tough to do manual testing |
| 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 Non-functional testing are   * Performance Testing * Load Testing * Volume Testing * Stress Testing * Security Testing * Installation Testing * Compatibility Testing * Migration Testing |

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

|  |  |
| --- | --- |
| **SDLC** | **STLC** |
| Focuses on building a product. | Focuses on testing a product. |
| A parent process. | A child of SDLC process. |
| Building a product as user requirement. | Ensuring the product is working as expected. |
| SDLC phases are completed before testing. | STLC phases start after SDLC phases are completed. |
| End goal is to deploy a high quality product to user. | End goal is to finding and fixing the bugs/defects. |

**31. What is the difference between test scenarios, test case, and test script?**

|  |  |  |
| --- | --- | --- |
| **Test Scenarios** | **Test Case** | **Test Script** |
| A scenario is any functionality that can be tested, it is also called test condition or test possibility. | Test cases involve the set of steps, conditions and inputs, which can be used while performing the testing tasks. | A set of sequential instruction that detail how business function. |
| Is more focused on what to test. | Is focused on what to test and how to test. | Is focused on the expected result. |
| Is derived from software Requirements. | Is mostly derived from test scenarios. | Is mostly derived from test cases. |
| Takes less time and fewer resources to create. | Requires more resources and time. | Requires less time for testing but more resources for scripts creating and updating. |
| 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. |

**32. Explain what test plan is? What is the information that should be covered?**

* **Test planning :-**

A document describing the scope, approach, resources and schedule of test Activities.

* **Test plan and strategy :-**

All Projects require a set plans and strategies which define how the testing will be conducted.

There are number of levels at which these are defined:

* Defines how the organization will conduct testing.
* Defines how the project will be conduct testing.
* Defines how each level of testing will be conducted.
* **Test planning Factors:-**

Factors which affect test planning

The Organizations test policy

Scope of the testing being performed

Testing objectives

Project risks – e.g. business, technical, people

Constraints – e.g. Business imposed, financial, contractual etc.

Criticality (e.g. system/component level)

Testability

Availability of Resources

* **Test Planning activities :-**

Making decisions about:

* What to test
* Who do testing? i.e. what roles will perform the test activities
* When and how the test activities should be done and when they should be stopped(exit criteria)
* How the test results will be evaluated
* **Exit Criteria:-**

How do we know when to stop testing?

* Run out of time?
* Run out of budget?
* The business tells you it went live last night!
* Boss says stop?
* All defects have been fixed?
* When out exit criteria have been met.

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

|  |  |
| --- | --- |
| **XP Testing** | **Scrum Testing** |
| Engineering practices like test driven development, refactoring, pair programming, etc… | Managing of requirements or requested  Features |
| Stories are used as single line requirements | User Stories are used as requirements |
| Stories are stored in Parking lot | Stories are stored in Product backlog and  Sprint backlog |
| Iteration is used in XP testing | Sprint is used in Scrum testing |
| Iteration period is for 14 days max | Iteration period is for 30 days |
| Requirement changes are accepted during iteration | Requirements changes are acceptable after the current iteration |
| Mid and end of Iteration review meetings | Daily scrum meeting |

**34. Explain the difference between Authorization and Authentication in Web testing.**

|  |  |
| --- | --- |
| **Authorization** | **Authentication** |
| Usually comes after authentication | Usually the first step of a security access control |
| Grants or denies permissions to the user | Verifies the user’s identify |
| Common methods include:  role-based access control and attribute-based access control | Common methods include:  Username, password, security question answer, code sent via SMS/email |
| Permissions are granted and monitored by the organization | Uses biometric data like fingerprint, face recognition, retinal scan |
| It’s not visible or changeable by the user | It’s visible and changeable by the user |

**35. What are the common problems faced in Web testing?**

* Integration.
* Interoperability.
* Security.
* Performance.
* Usability.
* Quality Testing, Exceptional Services.