**Software Testing Assignment**

**Module 2**

1. **What is Exploratory Testing?**

Ans. Exploratory testing” is the practice of allowing testers to enter a product and find bugs and errors without the help of a script or test cases. It encourages testers to think creatively and simulate real-world usage, mimicking the actions of end users. In the case of Global App Testing, testers are incentivized to find bugs which the QA team considers high or critical priority.

**2).What is traceability matrix?**

Ans. A Traceability Matrix is a document that maps or rather traces the relationship between two baseline documents. Here, one of the documents has the requirement specifications, whereas the other one has test cases. The Traceability Matrix is an essential document used during the [**software** **development lifecycle**](https://www.browserstack.com/guide/learn-software-development-process) of a product, and it ensures completeness and transparency of the underlying product. It is also called Cross Reference Matrix (CRM) or Requirement Traceability Matrix (RTM).

**3). What is Boundary value testing?**

Ans. Boundary Value Testing is one of the popular software testing mechanism, where testing of data is done based on boundary values or between two opposite ends where the ends may be like from start to end, or lower to upper or from maximum to minimum. This testing process was introduced to select boundary values that came from the boundary based on the inputs at different ends of testing values. This black box testing strategy was introduced after equivalence class partitioning where the partition of classes takes place first followed by a partition at the boundaries.

**4).** **What is Equivalence partitioning testing?**

Ans. Equivalence partitioning is a black-box testing technique that allows testers to group input data into sets or classes, making it possible to reduce the number of test cases while still achieving comprehensive coverage. This technique is particularly useful when dealing with a large range of input values

**5).What is Integration testing?**

Ans. integration testing is a type of software testing where components of the software are gradually integrated and then tested as a unified group. Usually these components are already working well individually, but they may break when integrated with other components. With integration testing, testers want to find defects that surface due to code conflicts between software modules when they are integrated with each other.

**6). What determines the level of Risk?**

Ans. Risk should be evaluated at the Business Level, Technological Level, Project Level and Testing Level.

Risk also used to decide where to start and where more testing is needed

**7.) What is Alpha Testing?**

Alpha testing is definitely performed and carried out at the developing organizations location with the involvement of developers.

It is always performed by the developers at the software development site.

Sometimes it is also performed by independent testing team.

It is always performed in virtual Environment.

It comes under the category of both White box testing and Black box testing

**8). What is Beta Testing?**

**Ans.** Beta testing is performed and carried out by users or you can say people at their own location and site using customer data

It is always performed by the customer at their own site

It is not performed by independent testing team

Beta testing is always open to the market and public.

It is performed in Real Time Environment

It is only kind of Black box testing

**9). What is Component Testing?**

**Ans.** Component testing- A minimal software item that can be tested in isolation. It means “A component is the smallest testable part of software.”

Component testing – The testing of individual software components.

Component testing is a level of 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

Component testing is the first level of testing and is performed to integration testing.

Sometimes also known as unit testing, module testing or program testing

**10). What is Functional System Testing?**

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

A requirement may exist as a text document or a model.

**11). What is Non-Functional Testing?**

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

Non-functional testing includes, but is not limited to, performance testing, load testing, stress testing, usability testing, maintainability testing, reliability testing and portability testing.

It is the testing of “how” the system works. Non – Functional testing may be performed at all test levels.

**12).What Is GUI Testing?**

**Ans**. Graphical User Interface (GUI) testing is the process of testing the system’s GUI of the system under test. GUI testing involves checking the screen with the controls like menus, buttons, icons and all type of bars - tool bar , menu bar , dialog boxes and windows etc.

**13). What is Adhoc testing?**

**Ans.** Main aim of Adhoc testing is to find defects by random checking.

Adhoc testing is informal testing type with an aim to break the system.

It does not follow any test design techniques to create test case.

In fact is does not create test case altogether.

This testing is primarily performed if the knowledge of testers in the system under test is very high.

Adhoc testing can be achieved with the testing technique called error guessing.

**14.) What is load Testing?**

**Ans.** Load testing – It’s a performance testing to check system behavior under load Testing an application under heavy loads, such as testing of a website under a range of loads to determine at what point the system’s response time degrades or fails.

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

**15).What is Stress 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 , continuous input to system or database load.

Stress testing is done to make sure that the system would not crash under crunch situation.

Stress testing is also known as endurance testing.

**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 testers require knowledge of how the software is implemented , how it works.

White box testing is also called glass testing or open box testing. In order to perform white box on an application, the tester needs to possess knowledge of the internal working of the code.

Types of coverage

* Statement coverage
* Decision coverage
* Condition coverage

**17). What is black box Testing? What are the different black box testing.?**

**Ans**.The technique of testing without having any knowledge of the interior workings of the application is Black box testing.

The tester is oblivious to the system architecture and does not have access to the source code. Typically, when performing a black box test, a tester interact with the system’s user interface by providing inputs and examining outputs without knowing how and where the inputs are worked.

Techniques of Black box testing

* Equivalence partitioning
* Boundary value analysis
* Decision Tables
* State transition testing
* Use case testing
* Other black box testing

**18). Mention what are the categories of defects?**

**Ans**. There are some categories which are as below

Data quality/Database Defects

Critical Functionality Defects

Functionality Defects

Security Defects

User Interface Defects

**19).Mention what big bang testing is?**

**Ans. In** big bang integration testing all components and modules is integrated simultaneously, after which everything is tested as a whole.

Big bang testing has the advantages that everything is finished before integration testing starts.

The major disadvantage is that in general it is time consuming and difficult to trace the cause of failures because of this late integration.

**20). What is the purpose of exist 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 (handover from system test to UAT)

**21). When should “Regression Testing “be performed?**

**Ans**. Change in requirements and code Is modified according to the requirements

New feature is added to the software

Defect fixing

Performance issue fix

**22). What is 7 key principles? Explain in detail?**

**Ans. Testing** shows presence of Defects

Exhaustive Testing is Impossible

Early testing

Defect Clustering

The pesticide Paradox

Testing is context dependent

Absence of Errors Fallacy

**1. Testing shows presence of Defects:**

Testing can show that defects are present, but cannot prove that there are no defects.

Testing reduce the probability of undiscovered defects remaining in the software but. Even if no defects are found, it is not a proof of correctness.

As we find more defects. The probability of undiscovered defects remaining in a system reduces

**2. Exhaustive Testing is Impossible:**

Testing everything including all combinations of inputs and preconditions is not possible

For example, in an application in one screen there are 15 input fields. Each having 5 possible values then to test all the valid combination you would need

That is, we must Priorities our testing efforts using a Risk Based Approach

**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

Testing activities should start as early as possible in the development life cycle.

**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

In other words, most defects found during testing are usually confined to small number of modules

**5. The pesticide Paradox:**

If the same tests are repeated over and over again, eventually the same set of test case will no longer find any new defects

To overcome this “pesticides 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 context

Different kinds of sites are tested differently

**7. Absence of Errors Fallacy:**

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

Even after defects have been resolved it may still be unusable and/or does not fulfill the users need and expectations

**23). Difference between QA v/s QC v/s Tester**

**Ans.** QA (Quality Assurance)

It is a subset of Software Test Life Cycle (STLC)

Preventive activities

Process oriented activities

Focus on processes and procedures rather than conducting actual testing on the system

QC (Quality Control)

QC can be considered as the subset of Quality Assurance

It is a corrective process

Product oriented activities

Focuses on actual testing by executing software with intend to identify bug/defect through implementation of procedures and process

Tester: Testing is the subset of Quality Control

It is a preventive process

Product oriented activities

Focuses on actual testing

**24). Difference between smoke and sanity?**

**Ans . Smoke Testing**

Smoke testing is performed to ascertain that the critical functionalities of the program is working fine

This testing is performed by the developers or testers

Smoke testing is usually documented or scripted

Smoke testing is a subset of Regression testing

**Sanity Testing**

Sanity testing is done to check the new functionality / bugs have been fixed

Sanity testing is usually performed by testers

Sanity testing is usually not documented and is unscripted

Sanity testing is a subset of Acceptance Testing

**25). Difference between verification and validation**

**Ans . Verification:** 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 phaseTo ensure that the product is being built according to the requirements and design specification. In other words, to ensure that work products meet their specified requirement

Are we building the product right?

**Validation:**

The process of evaluating software during or at the end of the development process. To determine whether it satisfies specified business requirements

To ensure that the product actually meets the user’s needs and that the specification were correct in the first place in other word, to demonstrate that the product fulfills its intended use when placed in its intended environment Are we building the right product

**26). Explain types of performance testing.**

**Ans**. Load testing

Stress testing

Endurance testing

Spike testing

Volume testing

Scalability testing

**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

**2. Stress testing:** 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 capacity queries, continuous input to system or database load.

**27**). **What are 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”

**28). Difference between priority and severity**

**Priority:** Priority is relative and business focused. Priority defines the order in which we should resolve a defect, should we fix it now or can it wait? This priority status is set by the tester to the developer mentioning the time frame to fix the defect. If high priority is mention then the developer has to fix it at the earliest. The priority status is set based on the customer requirements.

**Severity:** 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 given defect has on the system

**29). What is Bug Life Cycle?**

**Ans .“A** computer bug is an error, flow, mistake, failure or fault in a computer program that prevents it from working correctly or produces an incorrect result. Bug arise from mistake and errors made by people, in either a program’s source code or its design.”

When bug is discovered, it goes through several states and eventually reaches one of the terminal states where it becomes inactive and closed

**30). explain the difference between Functional testing and Non-functional testing**

**Ans .**Functional testing based on an analysis of the specification of the functionality of a component or systemFunctional testing is based on the Functions and features – may be applied at all test levelsFunctional testing verifies that each function of the software application operates in conformance with the requirement specification

**Non – Functional testing:**

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

Measuring the characteristics of the system/software that can be quantified on varying scale

Non – functional testing includes, but is not limited to, performance testing, load testing, stress testing , usability testing , maintainability testing , reliability testing and portability testing.

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

**Ans. Software Testing Life Cycle:** STLC is mainly related to software testing

It focuses only on testing the software

STLC involves only five phases or steps

In STLC, less number of members are needed

Goal of STLC is to complete successful testing of software

**Software Development Life Cycle:**

SDLC is mainly related to software development

Beside development other phases like testing is also included

SDLC involves total six phases or steps

In SDLC, more number of developers are required for the whole process

**32). What is the difference between test scenario, test cases and test script?**

**Ans .Test Scenario:** Is any functionality that can be tested

Is derived from test artifacts like Business Requirements Specification (BRS) and Software Requirements Specification (SRS)

Helps test the end to end functionality in Agile way

Is more focused on what to test

Test less time and fewer resources to create

**Test Cases:** Is a set of actions executed to verify particular features or functionality

Is mostly derived from test scenario

Helps in exhaustive testing of an app

Is focused on what to test and how to test

Requires more resources and time

**Test Script :** Is a set of instructions to test an app automatically

Is mostly derived from test cases

Helps to test specific things repeatedly

Is focused on the expected result

Requires less time for testing but more resources for scripts creating and updating

**33). What is priority?**

**Ans. Priority** is Relative and Business focused. Priority defines the order in which we should resolve a defect. Should we fix it now or Can it wait? This priority status is set by the tester to the developer mentioning the same frame to fix the defect. If the high priority is mentioned then the developer has to fix it at earliest. The priority status to set based on the customer requirements.

**For example** -- >> If the company name is misspelled in the home page of the website then the priority is high and severity is high and severity is low to fix.

**34). What is severity?**

**Ans. Severity** is absolute and customer focused. It is the extent to which the defect can affect the software. In other word it defines the impact that a given defect has on the system.

**For example** 🡪 if an application of the web page crashes when a remote link is clicked in this case clicking the remote link by an user is rare but the impact of application crashing severe. So the severity is high but priority is low.

**35). Bugs categories are…**

**Ans**. There are some categories

Security

Database

Functionality (Critical/General)

UI

**36). Advantages of Bugzilla**

**Ans.** Open source, free bug tracking

* Automatic Duplicate bug detection
* Search option with advanced features
* File/Modify bugs by mail
* Move bugs between installs
* Multiple authentication methods
* Time tracking
* Automated bug reporting, has an API to interact with system

**37). What are the different Methodology in Agile Development Model?**

**Ans 1. Scrum methodology:** Scrum is a lightweight framework of Agile Project Management, it can be adopted to conduct iterative and all types of incremental projects.

Due to specific characteristics like simplicity, sustained productivity and strength for blending several underlying approaches adapted by other agile methods, Scrum has obtained popularity over the years.

2**. Kanban:**

Kanban is an eminently visual workflow management approach, that can be employed for visualizing and thoroughly maintaining the making of products, it focuses on continual delivery of the product , but is not making stress to the entire software development life cycle.

Similar to scrum, kanban is the process developed for supporting collaborative teamwork more effectively.

**38). Explain the difference between Authorization and Authentication in web testing. What are the common problems faced in web testing?**

|  |  |
| --- | --- |
| **Authentication** | **Authorization** |
| Authentication verifies who the user is. | Authorization determines what resources a user can access. |
| Authentication works through passwords, one-time pins, biometric information, and other information provided or entered by the user. | Authorization works through settings that are implemented and maintained by the organization. |
| Authentication is the first step of a good identity and access management process. | Authorization always takes place after authentication. |
| Authentication is visible to and partially changeable by the user. | Authorization isn’t visible to or changeable by the user. |
| Example: By verifying their identity, employees can gain access to an HR application that includes their personal pay information, vacation time, and 401K data. | Example: Once their level of access is authorized, employees and HR managers can access different levels of data based on the permissions set by the organization. |

**39). When to use Usability Testing?**

**Ans. Usability** testing is a method of testing the functionality of a website, app or other digital product by observing real users as they attempt to complete tasks on it. The users are usually observed by researchers working for a business.

Usability testing can and should be conducted on the current iteration of a product before beginning any new design work, after you have begun the strategy work around a brand new site or app.

**40). What is the procedure for GUI Testing?**

**Ans**. 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 input fields.

Check you can execute the intended functionality of the application using the GUI.

* Check Error Messages are displayed correctly
* Check for Clear demarcation of different sections on screen
* Check Font used in application is readable
* Check the alignment of the text is proper
* Check the Color of the font and warning messages is aesthetically pleasing
* Check that the images have good clarity
* Check that the images are properly aligned
* Check the positioning of GUI elements for different screen resolution.