**Software Testing Assignment**

**Module-II (Manual Testing)**

1. **What is Exploratory Testing?**

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

1. **What is traceability matrix?**

To protect against changes you should be able to trace back from every system component to the original requirement that caused its presence.

1. **What is Boundary value testing?**

Boundary value analysis is a methodology for designing test cases that concentrates software testing effort on cases near the limits of valid ranges.

1. **What is Equivalence partitioning testing?**

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

1. **What is integration testing?**

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

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

Risk – ‘A factor that could result in future negative consequences; usually expressed as impact and likelihood’.

1. **What is Alpha testing?**

Alpha testing is the first end-to-end testing of a product to ensure it meets the business requirements and functions correctly.

* Alpha testing is performed by the developers at the software development site.
* It is performed by Independent Testing Team.
* Alpha Testing is not open to the market and public.
* It is conducted for the software application and project.
* It is always performed in Virtual Environment.
* It is always performed within the organization.
* It comes under the category of both White Box Testing and Black Box Testing.

1. **What is beta testing?**

Beta testing is the process of **testing a software product or service in a real-world environment** before its official release.

* Beta testing is performed by the customers at their own site.
* It is not performed by Independent Testing Team.
* Beta Testing is always open to the market and public.
* It is conducted for software product.
* It is performed in Real Time Environment.
* It is always performed outside the organization.
* It comes under the Black Box Testing.

1. **What is component testing?**

Component testing is a level of the software testing process where individual units/components of a software/system are tested.

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

Functional system testing is based on analysis of the specification of the functionality of a component or system.

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

Non-Functioanal testing is testing the attributes of the component or system that do not relate to the functionality, e.g. reliability, efficiency, usability, maintainability etc.

1. **What is GUI Testing?**

Graphical User Interface (GUI) testing is the process of testing the system’s GUI of the System under Test. 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.

1. **What is Adhoc testing?**

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

1. **What is load testing?**

Load testing is a kind of performance testing which determines a system’s performance under real-life load conditions.

1. **What is stress Testing?**

Stress 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 heavy load conditions.

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

White box testing is testing based on an analysis of the internal structure of component or system.

* **Types of white box testing**

1. Statement coverage
2. Decision coverage
3. Condition coverage
4. **What is black box testing? What are the different black box testing techniques?**

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

* **Types of technique in Black box testing:** 
  1. Equivalence Partitioning
  2. Boundary Value Analysis
  3. Decision Table
  4. State transition

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

* Database Defects
* Critical Functionality Defects
* Functionality Defects
* Security Defects
* User interface Defects

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

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

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

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 (hand over from System Test to UAT)

1. **When should "Regression Testing" be performed?**

* Change in requirements and code is modified according to the requirement.
* New feature is added to the software.
* Defect fixing.
* Performance issue fix.

1. **What is 7 key principles? Explain in detail?**

**7 keys 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 Error Fallacy
8. **Testing shows presence of Defects**

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

Testing reduces the probability of undiscovered defects remaining in the software.

1. **Exhaustive Testing is Impossible!**

Testing everything including all combination of inputs and preconditions is not possible.

1. **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.

1. **Defect Clustering**

Defects are not evenly spreadin asystem they are clustered.

1. **The Pesticide Paradox**

If the same test are repeated ,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.

1. **Testing is Context Dependent**

Testing is basically context dependent.

Different kinds of sites are tested differently.

For example: Safety-critical software is tested differently from an E-commerce site.

1. **Absence of Error 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.

It doesn’t make it a good system.

1. **Difference between QA v/s QC v/s Tester**

|  |  |  |  |
| --- | --- | --- | --- |
| SR. NO | QA | QC | Tester |
| 1 | QA stands for **Quality Assurance** | QC stands for **Quality Control**. |  |
| 2 | Focuses on **preventive activities** to avoid defects. | |  | | --- | | Focuses on **corrective processes** to identify and fix defects. |  |  | | --- | |  | | Focus on **Preventing Process** to identify and report bugs or issues. |
| 3 | It is a **process-oriented.** | It is a **product-oriented**. | It is a **product-oriented**. |
| 4 | Quality assurance is subset of **software testing life cycle.** | Quality Control is sub set of **QA.focusing on testing activities.** | Tester is sunset of the **Quality Control.** |
| 5 | Focuses on **processes and procedures.** | Focuses on **actual testing by executing software**. | Executes test cases and validates the software. |
| 6 | Performed by QA engineers or managers who design processes. | Performed by QC teams or testers focusing on the product. | Testers are responsible for test execution and defect reporting. |

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

|  |  |  |
| --- | --- | --- |
| SR. NO | Smoke testing | Sanity testing |
| 1 | Smoke testing performs after receiving new software build to ensure that critical functionalities of the software are working correctly. | Sanity testing performs after receiving new software build with minor changes in functionality or code to ensure that the bugs have fixed and no new problems were caused by these changes. |
| 2 | Smoke testing is performed by the tester and developer. | Sanity testing is performed by the tester. |
| 3 | Smoke testing is usually scripted and documented. | Sanity testing is usually unscripted and not documented. |
| 4 | |  | | --- | | Checks the overall system's functionality. | | |  | | --- | | Focuses on specific components or functionalities. | |
| 5 | Smoke testing is like general health check up. | Sanity testing is like specialized heath check up. |

1. **Difference between Verification and Validation**

|  |  |  |
| --- | --- | --- |
| SR. NO | Verification | Validation |
| 1 | Verification is a static testing. | Validation is dynamic testing. |
| 2 | Activities – reviews, walkthrough, inspection. | Activities – Testing. |
| 3 | It comes before validation. | It comes after verification. |
| 4 | Verification is for prevention of errors. | Validation is for detection of errors. |
| 5 | Left side of v represents development level activities. | Right side of v represents testing level activities. |

1. **Explain types of Performance testing.**

* Performance testing is Quality Assurance process that involves testing software applications to ensure they perform well under their expected workload.
* There are following types of performance testing:

1. **Stress testing**: Stress testing perform to test the robustness, stability and reliability of the system or software application under the extreme heavy load.
2. **Load testing**: Load testing is a type of performance testing that test the systembehaviour under their real life work load conditions.
3. **Spike testing**: Spike testing involves evaluating a system's ability to handle sudden, extreme surges in traffic
4. **Endurance testing**: Verifies the system's reliability over extended periods of use.
5. **Volume Testing**: Focuses on the system's ability to handle large amounts of data effectively.
6. **Scalability Testing**: Examines the system's ability to grow as user demands increase.
7. **What is Error, Defect, Bug and failure?**

* Error - A mistake in coding is called error.
* Defect - Error found by tester is called defect.
* Bug - Defect accepted by development team then it is called bug.
* Failur - Build does not meet the requirements then it is called failure.

1. **Difference between Priority and Severity**

|  |  |  |
| --- | --- | --- |
| SR. NO | Severity | Priority |
| 1 | Severity is absolute and Customer-Focused. | Priority is Relative and Business-Focused. |
| 2 | Severity means how severe the defect is affecting the functionality. | Priority means how fast the defect has to be fixed. |
| 3 | It is driven by functionality | It is driven by business value. |
| 4 | It is associated with functionality or standards. | It is associated with scheduling. |
| 5 | Severity is related to the quality standard. | Priority is related to scheduling to resolve the problem. |
| 6 | |  | | --- | | Ex. A crash = High Severity | | Ex. A minor defect in a critical feature = High Priority. |

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

The duration between the first time defect is found and the time that it Is closed successfully rejected , postponed and deferred is called “Bug life cycle”.

1. **Explain the difference between Functional testing and Non-Functional testing**

|  |  |  |
| --- | --- | --- |
| SR. NO | Functional testing | Non Functional testing |
| 1 | Functional testing is a type of software testing in which the system is tested against the functional requirement of specification. | Non-Functional testing thattest the attributes of a component or system that do not relate to functionality, e.g. reliability, efficiency. |
| 2 | It is based on requirement of customer. | It is based on expectation of customer. |
| 3 | Easy to execute manually. | Tough to execute manually. |
| 4 | It tests what the product does. | It describes how the product does. |
| 5 | Based on business requirements. | Based on performance requirement. |
| 6 | Types of functional testing  Smoke testing  Sanity testing  Unit testing | Types of functional testing  Load testing  Stress testing  Security testing |

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

|  |  |  |
| --- | --- | --- |
| SR. NO | STLC | SDLC |
| 1 | STLC stands for software testing life cycle. | SDLC stands for software development life cycle. |
| 2 | STLC is mainly related to software testing. | SDLC is mainly related to software development. |
| 3 | Goal of SDLC is to complete successful development of software | Goal of STLC is to complete successful testing of software. |
| 4 | In STLC, less number of members (testers) are needed. | In SDLC, more number of members (developers) are required for the whole process. |
| 5 | It helps in developing good quality software. | It helps in making the software defects free. |

1. **What is the difference between test scenarios, test cases, and test script?**

|  |  |  |  |
| --- | --- | --- | --- |
| SR. NO | Test Scenarios | Test Cases | Test Script |
| 1 | Test Scenario is any functionality that can be tested. | Test case is a set of actions executed to verify particular features or functionality. | Test script is a set of instructions to test an app automatically. |
| 2 | Helps test the end-to-end functionality in an Agile way | Helps in exhaustive testing of an app. | Helps to test specific things repeatedly. |
| 3 | Is more focused on what to test | Is focused on what to test and how to test. | Is focused on the expected result. |
| 4 | Includes an end-to-end functionality to be tested. | Includes test steps, data, expected results for testing. | Includes different commands to develop a script. |
| 5 | Allows quickly assessing the testing scope. | Allows detecting errors and defects. | Allows carrying out an automatic execution of test cases. |

1. **Explain what Test Plan is? What is the information that should be covered.**

* Test plan is a document that describing the scope , approach , resources and schedule of intended test activities.
* Test plan determining the scope and risks and identifying the objectives of testing.
* Test plan also covered level of detailed structure and templets for test documentation.

1. **What is priority?**

Priority is Relative and Business-Focused. Priority is how quickly the defect needs to be fixed.

1. **What is severity?**

Severity is absolute and Customer-Focused. Severity is how badly the defect affects the system.

1. **Bug categories are…**

There following bug categories :

1. Database defect

2.Critical Functionality defect

3.Functionality defect

4. Security defect

5. User interface defect

1. **Advantage of Bugzilla .**

* **Free and Open-Source** – No licensing cost, making it a budget-friendly choice for organizations.
* **Easy to Use** – Simple web-based interface that allows testers and developers to report and manage bugs efficiently.
* **Customizable Workflow** – Users can define their own workflow, statuses, and rules according to project needs.
* **Email Notifications** – Sends automatic updates to team members when a bug is created, updated, or resolved.
* **Multi-Language Support** – Available in various languages, making it suitable for global teams.

1. **Difference between priority and severity**

|  |  |  |
| --- | --- | --- |
| SR. NO | Severity | Priority |
| 1 | Severity is absolute and Customer-Focused. | Priority is Relative and Business-Focused. |
| 2 | Severity means how severe the defect is affecting the functionality. | Priority means how fast the defect has to be fixed. |
| 3 | It is driven by functionality | It is driven by business value. |
| 4 | It is associated with functionality or standards. | It is associated with scheduling. |
| 5 | Severity is related to the quality standard. | Priority is related to scheduling to resolve the problem. |
| 6 | |  | | --- | | Ex. A crash = High Severity | | Ex. A minor defect in a critical feature = High Priority. |

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

* There are the different Methodologies in Agile Development Model :

1. Kanban
2. Scrum
3. **Explain the difference between Authorization and Authentication in Web testing.What are the common problems faced in Web testing?**

* In web testing,

"Authentication" refers to the process of verifying a user's identity by checking their credentials (like username and password), essentially confirming "who they are".

"Authorization" determines what actions a verified user is allowed to perform within the application, deciding "what they can do" based on their access level.

* Common problems encountered in web testing include: cross-browser compatibility issues, performance concerns, security vulnerabilities, user experience (UX) challenges, testing across different devices, data privacy issues, load testing, and ensuring proper functionality across various browser versions and operating systems.

1. **When to used Usablity Testing?**

* There are many software applications / websites, which miserably fail, once launched, due to following reasons –
* Where do I click next?
* Which page needs to be navigated?
* Which Icon or Jargon represents what?
* Error messages are not consistent or effectively displayed
* Session time not sufficient.

1. **What is the procedure for GUI Testing?**

* Graphical User interface (GUI) testing involves a series of steps to ensure that a user interface is functional, usable, and visually consistent. The steps include:
* **Planning**: Define the scope of testing and identify key areas of the UI
* **Preparation**: Set up the testing environment with the necessary tools and resources
* **Test case development**: Create detailed test cases that cover different aspects of the UI
* **Test execution**: Perform the tests by interacting with the UI as a user would
* **Issue reporting**: Document any issues or inconsistencies encountered
* **Fix review**: Review and validate fixes to ensure that issues have been resolved
* **Continuous testing**: Continue to test and improve the UI