**Module–2**

1. What is Exploratory Testing?

**Ans. :** Exploratory testing is a concurrent process where Test design, execution and logging happen simultaneously Testing is often not recorded Makes use of experience, heuristics and test patterns.

1. What is traceability matrix?

**Ans. :** Test conditions should be able to be linked back to their sources in the test basis, this is known as traceability. Traceability can be horizontal through all the test documentation for a given test level or it can be vertical through the layers of development documentation.

1. What is Boundary value testing?

**Ans. :** 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.

1. What is Equivalence partitioning testing?

**Ans. :** 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.

1. What is Integration testing?

**Ans. :** Integration Testing is 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.

1. What determines the level of risk?

**Ans. :** A properly designed test that passes, reduces the overall level of Risk in a system.

1. What is Alpha testing?

**Ans. :** It is always performed by the developers at the software development site. Sometimes it is also performed by Independent Testing Team. Alpha Testing is not open to the market and public It is conducted for the software application and project.

1. What is beta testing?

**Ans. :** Beta Testing is always performed at the time when software product and project are marketed. It is always performed at the user’s premises in the absence of the development team. It is also considered as the User Acceptance Testing (UAT) which is done at customers or users area. Beta testing can be considered “pre-release” testing.

1. What is component testing?

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

1. What is functional system testing?

**Ans. :** Functional System Testing is a requirement that specifies a function that a system or system component must perform.

1. 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.

1. What is GUI Testing?

**Ans. :** GUI Testing is Graphical User Interface testing. This 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?

**Ans. :** Adhoc testing is an informal testing type with an aim to break the system. The Error guessing is a technique where the experienced and good testers are encouraged to think of situations in which the software may not be able to cope.

1. What is load testing?

**Ans. :** Load testing is a kind of performance testing which determines a system’s performance under real-life load conditions. – It is a performance testing to check system behaviour under load.

1. What is stress Testing?

**Ans. :** Stress Testing System is stressed beyond its specifications to check how and when it fails. It even tests beyond the normal operating point and evaluates how the system works under those extreme conditions.

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

**Ans. :** White Box is based on an analysis of the internal structure of the component or system.

(1) Statement Coverage (2) Branch Coverage (3) Condition Coverage (4) Branch Condition testing (5) Branch Condition Combination testing (6) Modified Condition Decision testing (7) Dataflow testing (8) Linear Code Sequence And Jump (LCSAJ) testing.

1. What is black box testing? What are the different black box testing techniques?

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

1. Equivalence partitioning (2) Boundary value analysis (3) Decision tables (4) State transition testing (5) Use-case Testing (6) Other Black Box Testing Syntax or Pattern Testing.
2. Mention what are the categories of defects?

**Ans. :** Data Quality/Database Defects, Critical Functionality Defects, Functionality Defects, Security Defects, User Interface Defects.

1. Mention what big bang testing is?

**Ans. :** Big bang integration testing is a testing approach where all components or modules are integrated and tested as a single unit. This is done after all modules have been completed and before any system-level testing is performed.

1. What is the purpose of exit criteria?

**Ans. :** The Purpose of the exit criteria isSuccessful Testing of Integrated Application, executed Test Cases are documented, all High prioritized bugs fixed and closed, technical documents to be submitted followed by release Notes.

1. When should "Regression Testing" be performed?

**Ans. :** The Regression testing is performed when the system is stable and the system or the environment changes and when testing bug-fix releases as part of the maintenance phase.

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

**Ans. : (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

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. 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. For example: In an application in one screen there are 15 input fields, each having 5 possible values, then to test all the valid combinations you would need 30 517 578 125 (515) tests. This is very unlikely that the project timescales would allow for this number of tests. So, accessing and managing risk is one of the most important activities and reason for testing in any project. We have learned that we cannot test everything (i.e. all combinations of inputs and pre-conditions).
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 They are ‘clustered’ In other words, most defects found during testing are usually confined to a small number of modules.
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. N.B It's called the "pesticide paradox" after the agricultural phenomenon, where bugs such as the boll weevil build up tolerance to pesticides, leaving you with the choice of ever-more powerful pesticides followed by ever-more powerful bugs or an altogether different approach.’ – Beizer 1995.
6. Testing is Context Dependent : Testing is basically context dependent. Testing is done differently in different contexts. Whilst, Testing can be 50% of development costs, in NASA's Apollo program it was 80% testing 3 to 10 failures per thousand lines of code (KLOC) typical for commercial software. Also different industries impose different testing standards.
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 fulfil the users’ needs and expectations.
8. Difference between QA v/s QC v/s Tester

**Ans. :**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.N.** | **Quality Assurance** | **Quality Control** | **Testing** |
| 1 | Activities which ensure the implementation of processes, procedures and standards in context to verification of developed software and intended requirements. | Activities which ensure the verification of developed software with respect to documented (or not in some cases) requirements. | Activities which ensure the identification of bugs/error/defects in the Software. |
| 2 | Focuses on processes and procedures rather than conducting actual testing on the system. | Focuses on actual testing by executing Software with intend to identify bug/defect through implementation of procedures and process. | Focuses on actual testing. |
| 3 | Process oriented activities. | Product oriented activities. | Product oriented activities. |
| 4 | 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. |

1. Difference between Smoke and Sanity?

**Ans. :**

|  |  |
| --- | --- |
| **Smoke Testing** | **Sanity Testing** |
| Smoke Testing is performed to ascertain that the critical functionalities of the program is working fine. | Sanity Testing is done to check 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 not documented and is unscripted. |
| 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. | |

1. Difference between verification and Validation.

**Ans. :**

|  |  |  |
| --- | --- | --- |
| **Criteria** | **Verification** | **Validation** |
| Definition | 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 phase. | The process of evaluating software during or at the end of the development process to determine whether it satisfies specified business requirements. |
| Objective | To ensure that the product is being built according to the requirements and design specifications. In other words, to ensure that work products meet their specified requirements. | To ensure that the product actually meets the user’s needs, and that the specifications were correct in the first place. In other words, to demonstrate that the product fulfilled its intended use when placed in its intended environment. |
| Question | Are we building the product right? | Are we building the right product? |
| Evaluation Items | Plans, Requirement Specs, Design Specs, Code, Test Cases | The actual product/software. |
| Activities | ∙ Reviews  ∙ Walkthroughs  ∙ Inspections | ∙ Testing |

1. Explain types of Performance testing ?

**Ans. :** Load testing, Stress testing, Endurance testing, Spike testing, Volume testing, Scalability testing.

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

**Ans. :** Error **:** Error is a discrepancy between a computed, observed, or measured value or condition and the true, specified, or theoretically correct value or condition.

Defect**:** Commonly refers to several troubles with the software products, with its external behaviour or with its internal features.

Bug: A fault in a program which causes the program to perform in an unintended or unanticipated manner. Bug is terminology of Tester.

Failure: The inability of a system or component to perform its required functions within specified performance requirements.

1. Difference between Priority and Severity?

**Ans. :**

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Priority** | **Severity** |
| Definition | Priority is a term that defines how fast we need to fix a defect. | Severity is a term that denotes how severely a defect can affect the functionality of the software. |
| Parameter | Priority is basically a parameter that decides the order in which we should fix the defects. | Severity is a basically a parameter that denotes the total impact of given defect on any software. |
| Relation | Priority relates to the scheduling of defects to resolve them in software. | Severity relates to the standards of quality. |
| Value | The value of priority is subjective. | The value of severity is objective. |
| Change of value | The value of priority changes from time to time. | The value of severity changes continually from time to time. |

1. What is Bug Life Cycle?

**Ans. :** “Bug Life Cycle is a process of computer bug is an error, flaw, mistake, failure, or fault in a computer program that prevents it from working correctly or produces an incorrect result. 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’.

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

**Ans. :**

|  |  |
| --- | --- |
| **Functional Testing** | **Non-Functional Testing** |
| Functional testing is performed using the functional specification provided by the client and verifies the system against the functional requirements. | Non-Functional testing checks the Performance, reliability, scalability and other non-functional aspects of the software system. |
| Functional testing is executed first. | Non-functional testing should be performed after functional testing. |
| Manual testing or automation tools can be used for functional testing Using tools will be effective for this testing | Using tools will be effective for this testing |
| Business requirements are the inputs to functional testing | Performance parameters like speed , scalability are inputs to non-functional 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  • Penetration Testing  • Compatibility Testing  • Migration Testing |

1. To create HLR & Test Case of 1)(Instagram , Facebook) only first page

**Ans. : Excel File Uploaded.**

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

**Ans. :** SDLC covers the entire software development process, including planning, analysis, design, coding, testing, deployment, and maintenance. STLC is a part of SDLC and focuses specifically on testing. STLC includes activities like test planning, test case development, test execution, defect tracking, and closure.

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

**Ans. :** A test case is a document with instructions on testing the specific functionality of an application. Test Script is a program that runs various test data on the functionality of an application. Test scenarios serve as an outline for writing test cases.

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

**Ans. :** A test plan is essentially a project management plan, which should always include the major components of a normal plan for project management purposes, including an overview, scope, methodology, resources, schedule, risks, and communication plan.

1. What is priority?

**Ans. :** 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 mentioned then the developer has to fix it at the earliest.

1. What is severity?

**Ans. :** 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.

1. Bug categories are…

**Ans. :** Functional Bugs, Logical Bugs, Workflow Bugs, Unit Level Bugs, System-Level Integration Bugs, Out of Bound Bugs, Security Bugs.

1. Advantage of Bugzila .

**Ans. :** Key features of Bugzilla includes, Advanced search capabilities, E-mail Notifications, Modify/file Bugs by e-mail, Time tracking, Strong security, Customization, Localization.

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

**Ans. :** Kanban, Scrum, Feature-driven development (FDD), Behavior-driven development (BDD), Lean development, Adaptive software development (ASD), Crystal, Extreme programming (XP)

1. Explain the difference between Authorization and Authentication in Web testing. What are the common problems faced in Web testing?

**Ans. :**

|  |  |
| --- | --- |
| **Authentication** | **Authorization** |
| In the authentication process, the identity of users are checked for providing the access to the system. | While in authorization process, a the person’s or user’s authorities are checked for accessing the resources. |
| In the authentication process, users or persons are verified. | While this process is done after the authentication process. |
| It needs usually the user’s login details. | While it needs the user’s privilege or security levels. |
| Authentication determines whether the person is user or not. | While it determines **What permission does the user have?** |
| **Example**: Employees in a company are required to authenticate through the network before accessing their company email. | **Example:** After an employee successfully authenticates, the system determines what information the employees are allowed to access. |

1. To create HLR & TestCase of WebBased (WhatsApp web , Instagram)

**Ans. : Excel File Uploaded.**

1. Write a scenario of only Whatsapp chat messages?

**Ans. : Excel File Uploaded.**

1. Write a Scenario of Pen?

**Ans. : Excel File Uploaded.**

1. Write a Scenario of Pen Stand?

**Ans. : Excel File Uploaded.**

1. Write a Scenario of Door?

**Ans. : Excel File Uploaded.**

1. Write a Scenario of ATM?

**Ans. : Excel File Uploaded.**

1. When to used Usablity Testing?

**Ans. :** Usability testing is implemented in the early stage of the Software Development Life Cycle (SDLC). The execution of usability testing offers more visibility on the prospects of the end-users

1. What is the procedure for GUI Testing?

**Ans. :** We can perform the Gui testing with help of automation tools, which can be completed two types. Throughout the record part, the test steps are encapsulated by the automation tool. And the playback, these recorded test steps are implemented on the application under test.

1. Write a scenario of Microwave Owen?

**Ans. : Excel File Uploaded.**

1. Write a scenario of Coffee vending Machine?

**Ans. :**  **Excel File Uploaded.**

1. Write a scenario of chair?

**Ans. :** **Excel File Uploaded.**

1. To Create Scenario (Positive & Negative) Of Gmail and . Online shopping to buy product (flipkart)

**Ans. : Excel File Uploaded.**

1. Write a Scenario of Wrist Watch

**Ans. :** **Excel File Uploaded.**

1. Write a Scenario of Lift(Elevator)

**Ans. : Excel File Uploaded.**

1. Write a Scenario of Whatsapp Group (generate group)

**Ans. : Excel File Uploaded.**

1. Write a Scenario of Whatsapp payment

**Ans. :** **Excel File Uploaded.**