# **Tops technology**

# 

# Module–2(Manual Testing)

1. 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

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

1. What is Exploratory Testing?

* Exploratory testing checks the functionality and operations of the software as well as identify the functional and technical faults in it. Exploratory testing aims to optimize and improve the software in every possible way. The exploratory testing technique combines the experience of testers with a structured approach to testing. It is often performed as a black box testing technique.

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. A software process should help you keeping the virtual table up-to-date. Simple technique may be quite valuable (naming convention)

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
* 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.

1. What is Integration testing?

* Integration testingis the process of testing the interface between two software units or modules. It focuses on determining the correctness of the interface. The purpose of integration testing is to expose faults in the interaction between integrated units. Once all the modules have been unit-tested, integration testing is performed.

1. What determines the level of risk?

* ‘A factor that could result in future negative consequences; usually expressed as impact and likelihood
* A Risk could be any future event with a negative consequence .You need to identify the risks associated with your project
* Risks are of two types

1. Project Risks
2. Product Risks
3. What is Alpha testing?

* Alpha Testing is an essential phase in software testingconducted by the developmentor QA teambefore beta testing. It aims to identify and fix bugs in a controlled environmentthat simulates real-world conditions. This helps ensure the software’s functionality , reliability, and stability . Alphatesting combines white-box and black-box testing techniques to explore and evaluate the software.

1. What is beta testing?

* Beta testing is performed by real user
* No tester is required for testing
* It’s also known as external user acceptance testing
* Beta version is released so that user can use the application and provide correct feedback
* Beta testing helps us to test the application in the real time environment

1. What is component testing?

* A minimal software item that can be tested in isolution . It means “A unit is the smallest testable part of software unit testing is a level of the software testing process where individual units

1. What is functional system testing?

* 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 and/or a model
* There is two types of Test Approach:

1. Requirement Based Functional Testing
2. Process Based Testing

* Functional System Testing Functionality As below:

1. Accuracy - Provision of right or agreed results or effects
2. Interoperability - Ability to interact with specified systems
3. Compliance - Adhere to applicable standards, conventions, regulations or laws Adhere to applicable standards, conventions, regulations or laws
4. Auditability - Ability to provide adequate and accurate audit data
5. Suitability - Presence and appropriateness of functions for specified tasks
6. What is Non-Functional Testing?

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

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 a type of software testing that is performed informally and randomly after the formal testing is completed to find any loophole in the system. For this reason, it is also known as Random or Monkey testing. Adhoc testing is not performed in a structured way so it is not based on any methodological approach. That’s why Adhoc testing is a type of Unstructured Software Testing.

1. What is load testing?

* Its 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 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.

1. What is stress Testing?

* Stress Testing is done to make sure that the system would not crash under crunch situations. Stress testing is also known as endurance testing. 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?

* White box testing is a technique , that examine the program structure and derived test data from the program logic .
* The term “ white box “ was used because of the “ see-through box “ concept.
* To test internal logic of the code.
* Create test cases and so that every code line , loops and conditional statement are covered as per the given conditions.
* Types of white box testing
* Unit testing
* Integration testing
* Regression testing

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

* It is always 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 usually conducted for software product.
* There are four specification-based or black-box technique:
* Equivalence partitioning
* Boundary value analysis
* Decision tables
* State transition testing
* Use-case Testing
* Syntax or Pattern Testing

1. Mention what are the categories of defects?

* Data Quality/Database Defect: Deals with improper handling of data in the database.
* Critical Functionality Defects: The Occurrence of these bugs hampers the crucial functionality of the application.
* Functionality Defects: These defects affect the functionality of the application.
* Security Defects: Application security Defects generally involve improper handling of data sent from the user to the application. These defects are the most severe and given highest priority for a fix.
* User Interface Defects: As the name suggests, the bugs deal with problems related to UI are usually considered less severe.

1. Mention what bigbang testing is?

* Big Bang integration testing all components or 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.

1. 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 – i.e. product Go Live
2. End of phase of testing (e.g. hand over from System Test to UAT)
3. When should "Regression Testing" be performed?

* Regression testing is done to verified modified code does not affect the exisiting functionality of the application
* In regression testing only selected or all the test cases are implement to find bugs
* We ensure that old code still works , once the new code change s are done
* Regression testing increase our change of detecting bugs caused b ybchanges to a software and application either enchanement or defect fixes

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

* 7 Key Principle are :

1. Testing shows presence of 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.

1. 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 f ields, each having 5 possible values, then to test all the valid combinations you would need 30 517 578 125 (515) tests.

1. Early Testing :

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

* A small number of modules contain most of the defects discovered during pre-release testing, or are responsible for the most operational failures.

1. 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.

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:

* 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

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

**Quality Assurance**

* Activities which ensure the implementation of 1 2 3 4 5 processes, procedures and standards in context to verification of developed software and intended requirements.
* Focuses on processes and procedures rather than conducting actual testing on the system
* It is a subset of Software Test Life Cycle (STLC).
* Process oriented activities.

**Quality control**

* Activities which ensure the verification of developed software with respect to documented (or not in some cases) requirements.
* Focuses on actual testing by executing Software with intend to identify bug/defect through implementation of procedures and process
* Product oriented activities
* QC can be considered as the subset of Quality Assurance.

**Tester**

* Activities which ensure the identification of bugs/error/defects in the Software.
* Focuses on actual testing.
* Product oriented activities.
* Testing is the subset of Quality Control.

1. Difference between Smoke and Sanity?

**Smoke**

* Smoke Testing is performed to ascertain that the critical functionalities of the program is working fine .
* The objective of this testing is to verified "stability" of the system in order to with more rigorous testing
* This testing is performed by the developers or testers
* Smoke testing is usually documented or scripted is unscripted
* Smoke testing is like General Health Check Up

**Sanity**

* Sanity Testing is done to check the new functionality/ bugs have been fixed
* Sanity testing is usually performed by testers
* Sanity testing is usually performed by tester
* Sanity testing is a subset of Regression testing
* Sanity Testing is like specialized health check up

1. Difference between verification and Validation

**Verification:**

* The process of evaluating work-products (not the actual f inal product) of a development phase to determine whether they meet the specified requirements for that phase**.**
* 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.
* Are we building the product right?
* Plans, Requirement Specs, Design Specs, Code, Test Cases
* Reviews, Walkthroughs, Inspections

**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 specifications were correct in the first place. In other words, to demonstrate that the product fulfils its intended use when placed in its intended environment.
* Are we building the right product?
* The actual product/software
* Testing

1. Explain types of Performance testing.

* Types of Performance testing:

1. Load testing: checks the application’s ability to perform under anticipated user loads the objective is to identify performance b bottlenecks before the software application goes live
2. Stress testing: Involve testing an application under extreme workloads to see how it handle high traffic or data processing. The objective is to identify breaking of an application
3. Endurance testing: Testing is done to make sure the software can handle the excepted load period of time.
4. Spike testing : Tests the software’s reaction to sudden large spikes in the load generation by user.
5. Volume testing : Under volume Testing large no .of.Data is populated in database and the overall software system’s behaviour is monitored .The objective is to check software application’s performed under varying database volumes.
6. Scalability testing: Testing is to determine the software application’s effective in “ scaling up” to support an increase in user load . It helps plan capacity addition to your software system
7. Difference between Priority and Severity

* Priority:

1. The urgency of fixing the defect.
2. Priority defines how soon the defect needs fixing.
3. Affects the project timeline and resource allocation.
4. A minor visual glitch (low severity, but high priority for brand consistency).

* Severity:

1. The impact of the defect on the functionality.
2. Severity defines **how bad** the defect is.
3. Affects how users interact with the system.
4. A system crash (high severity, could be critical)
5. What is Bug Life Cycle?

* A 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. Bugs arise from mistakes and errors, made by people, in either a program’s source code or its design
* 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 .

**Functional testing:**

* Functional testing is performed using the functional specification provided by the client and verifies the system against the functional requirements.
* Functional testing is executed first
* Manual testing or automation tools can be used for functional testing
* Easy 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 Testin

**Non-Functional:**

* Non-Functional testing checks the Performance, reliability, scalability and other non-functional aspects of the software system
* Non-functional testing should be performed after functional testing.
* Using tools will be effective for this testing
* Nonfunctional testing describes how good the product works
* Tough to do manual testing
* Types of Nonfunctional testing are

∙ Performance Testing

∙ Load Testing

∙ Volume Testing

∙Stress Testing

∙ Security Testing

∙ Installation Testing

∙ Penetration testing

. Compatibility Testing

∙ Migration Testing

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

* Software testing life cycle:
* Testing activities to ensure software quality
* Find defects and verify that the software meets requirements
* Requirement Analysis, Test Planning, Test Design, Execution, etc.
* Starts after development, overlaps with the SDLC’s testing phase
* Subset of SDLC, focuses on testing and quality assurance
* Software development life cycle:
* Entire process of developing a software product
* Develop and deliver a functional and maintainable software product
* Requirement Gathering, System Design, Implementation, Testing, etc.
* Covers the entire process from concept to deployment and maintenance
* Full development process from start to finish

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

* Test scenarios:

A test scenario is a high-level description of what needs to be tested. It is a broad outline or concept of the functionality to be tested in the application. Test scenarios focus on the "what" of the testing process.

* Purpose: To define what the system should do.
* Level of Detail: High-level, typically a summary
* Example: Test the login functionality of a web application.
* Test cases:

A test case is a detailed, specific set of actions, conditions, or inputs that are used to verify a particular aspect of the system. A test case is derived from a test scenario and defines the actual steps to be performed, the input data, expected results, and pass/fail criteria.

* Purpose: To describe the step-by-step actions, inputs, and expected outputs to validate the system's behavior.
* Level of Detail: More detailed than a test scenario.
* Example:

Test Case Name: Login with valid credentials.

* Test script:

A test script is a set of instructions written in a specific programming or scripting language to automate the execution of test cases. It is used to automatically perform testing and validate results, typically in automated testing.

* Purpose: To automate the execution of test cases, so the tests can be repeated automatically without manual intervention.
* Level of Detail: Highly technical, containing actual code or instructions.
* Example: A script written in Selenium WebDriver that automates the login process by entering credentials, clicking buttons, and verifying outcomes.

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

* Introduction: Provides an overview of the document and its purpose.
* Objective: Clearly states the goals and objective of the testing efforts
* Scope: Defines the boundaries of the testing, including what will and will not be tested.
* Approach: Describes the overall strategy for testing, including methodologies, techniques, and tools to be used.
* Test Items lists the specific components or features to be tested
* . Features to be tested: Describe the features or functionalities to be tested, often with reference to requirements or specifications.
* Test Environment: specifies the hardware, software, and other resources needed for testing.

1. What is priority?

* Priority refers to how soon a defect should be fixed. It is based on the urgency of addressing the defect, considering the impact it has on the business, user experience, or project timeline.

1. What is severity?

* Severity refers to the impact a defect has on the functionality or operation of the system. It describes the extent to which the defect affects the software and its ability to perform as expected.

1. Bug categories are…

* Functional Bugs
* Performance Bugs
* Usability Bugs
* Compatibility Bugs
* Security Bugs
* Regression Bugs
* Boundary/Edge Case Bugs
* Integration Bugs
* Configuration Bugs
* Data Bugs
* Localization/Internationalization Bug
* Memory Leaks
* Crash Bugs

1. Advantage of Bugzila .

* Open Source and Free
* Scalability
* Email Notifications and Alerts
* Multi-Language Support
* Security and Privacy
* Active Community and Support

1. Difference between priority and severity

* Priority:

1. Describes how urgent it is to fix the defect.
2. Focuses on the business urgency and when it needs to be fixed.
3. Typically assigned by project managers, product owners, or businessanalysts based on the business needs and project deadlines.
4. Priority determines the timing of resolution based on its business or user impact.
5. High priority: Critical bug affecting a release deadline. Low priority: Cosmetic issues that don’t impact functionality.
6. What are the different Methodologies in Agile Development Model?

* In the Agile development model, various methodologies have been developed to implement Agile principles in different ways. While all Agile methodologies are based on the core principles outlined in the Agile Manifesto, they differ in terms of processes, practices, and frameworks. Here are some of the most commonly used Agile methodologies

1. Scrum
2. Kanban
3. Extreme Programming
4. Feature-Driven Development
5. Feature-Driven Development
6. Crystal
7. Dynamic Systems Development Method
8. Agile Unified Process
9. Scrumban
10. Agile Modelling
11. Explain the difference between Authorization and Authentication in Web testing. What are the common problems faced in Web testing?

* **AUTHENTICATION:**
* Authentication is the process of verifying the identity of a user or system. It ensures that the user or system is who they claim to be.
* To validate the credentials provided by the user (like username/password, biometric data, or tokens).
* Examples:

1. Logging in with a username and password.
2. Using multi-factor authentication (MFA), such as entering a code sent to your phone.

* The login mechanism works correctly (e.g., valid credentials allow access, invalid credentials are denied).
* Authentication systems handle edge cases like password resets, account lockouts, and session timeouts.
* Authentication is secure against threats like brute-force attacks, credential stuffing, or session hijacking.

**Authorization:**

* Authorization is the process of granting or denying specific permissions or access to a resource, based on the authenticated identity.
* After a user is authenticated, authorization determines what the user is allowed to do within the system (e.g., access specific pages, perform certain actions).
* Examples:

1. A user might be authenticated but only authorized to view their own profile, not the profiles of other users.
2. A user with an admin role might have access to all resources, whereas a regular user has limited access.

* Users are granted the correct permissions based on their roles.
* Role-based access control (RBAC) works properly, ensuring unauthorized users cannot access restricted resources.
* There are no security flaws like privilege escalation, where a user can gain higher-level permissions

1. Write a scenario of only Whatsapp chat messages
2. Write a Scenario of Pen
3. Write a Scenario of Pen Stand
4. Write a Scenario of Door
5. Write a Scenario of ATM
6. When to used Usability Testing?

* Usability Testing identifies usability errors in the development cycle and can save a product from failure
* Usability testing is a method used to evaluate the userexperience and navigation of websites, apps, and digital products.
* Usability testing involves evaluating the functionality of a website, app, or digital product by observing real users as they navigate through it

1. What is the procedure for GUI Testing?

* GUI testing generally evaluates a design of elements such as layout, colors and also fonts, font sizes, labels, text boxes, text formatting, captions, buttons, lists, icons, links, and content. GUI testing processes may be either manual or automatic and are often performed by third-party companies, rather than developers or end users.

1. Write a scenario of Microwave Owen
2. Write a scenario of Coffee vending Machine
3. Write a scenario of chair
4. To Create Scenario (Positive & Negative) gmail (receiving mail). Online shopping to buy product (Flipkart)
5. Write a Scenario of Wrist Watch
6. Write a Scenario of Lift(Elevator)
7. Write a Scenario of WhatsApp Group (generate group)
8. Write a Scenario of WhatsApp payment .
9. To create HLR & TestCase of 1)(Instagram , Facebook) first page 2) Facebook Login Page :
10. To create HLR & TestCase of WebBased (WhatsApp web) 1. WhatsApp Web : <https://web.whatsapp.com/>
11. Create Test scenario on Compose Mail Functionality. 2. Online shopping to buy product (flipkart)
12. To create HLR and TestCase on this Link. <https://artoftesting.com/>