1. What is Exploratory Testing?

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

1. What is Traceability Matrix?

**Ans**.A Traceability Matrix is a document that maps and traces user requirements to test cases and vice versa to ensure that all requirements are covered by test cases.

1. What is Boundary Value Testing?

**Ans.**Boundary Value Testing is a Technique of Black Box Testing.

In Boundary Value Analysis, It’s 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?

**Ans.**Equivalence Partitioning Testing is a Technique of Black Box Testing.

IN Equivalence Partitioning the aim is to treat groups of inputs as equivalent and to select one representative input to test them all.

E.P. can be used for all levels of testing.

1. What is Integration Testing?

**Ans.**Integration Testing: Testing 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 Risk: A factor that could result in future negative consequences, usually express as impact and likelihood(possibility).

To identify risk is not enough, you need to identify the mitigation.

Two types of Risk: Project Risk

Product Risk

1. What is Alpha Testing?

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

1. What is Beta Testing?

* 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 UAT (User Acceptance Testing) which is done at customers or users area.
* Beta Testing can be considered “prerelease “testing.

1. What is component Testing?

**Ans.**Unit/ Component Testing is a level of the Software process where individual units/ components of a software/ system are tested.

Component(Unit): A minimal software item that can be tested in isolation. Means “A unit is the smallest testable part of software.”

Run by software developers.

Unit testing is performed by using the White Box Testing method.

1. What is Functional System Testing?

**Ans.**Functional Testing: Testing based on the analysis of the specification of the functionality of a component or system.

‘Specification’: E.g. Requirements specification, use cases, Functional specification or may be undocumented.

‘Function’: What the system do

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.

It May be performed at all test levels.

It is the testing of ‘how’ the system works.

1. What is GUI Testing?

**Ans.**GUI Testing is a process of testing the system’s GUI of the system under test.

GUI (Graphical User Interface) 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.

Approach of GUI Testing: Manual based GUI

Record & Replay GUI

Model based GUI

1. What is Adhoc Testing?

**Ans.**Adhoc Testing: Adhoc testing is an informal testing type with an aim to break the system

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

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

Main aim of this testing is to find defects by random checking.

Adhoc Testing, Error Guessing, Random Checking, Monkey Testing

Types of Adhoc Testing: Buddy Testing

Pair Testing

Monkey Testing

1. What is load Testing?

**Ans.**Load Testing: It is 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.

Stability + response time + applying load

Performance of System components under various loads.

1. What is Stress Testing?

**Ans.**Stress Testing/ Endurance Testing/ Spike 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 database queries, continuous input to system or database load.

1. What is White Box Testing? and List the Types of white box testing.

**Ans.**White box/ Glass box/ Open box/Structure based/

White Box Testing: Testing based on an analysis of the internal structure of the component or system.

Types of White box testing:

* Statement Coverage: Covers only true conditions
* Decision Coverage: Covers both the True & False conditions
* Condition Coverage: Full condition coverage doesn’t guarantee full decision coverage

1. What is Black Box Testing? What are the different Black Box Testing Techniques?

**Ans.**Black box testing /Specification Based testing/ Input – output driven Testing

Black box Testing: Testing, either functional or non- functional, without reference to the internal structure of the component or system.

Tester have no knowledge of the source code/ structure inside the box.

In Black Box Testing the tester is focusing on what system does not how it does it.

Techniques of Black Box Testing:

* Equivalence partitioning
* Boundary Value Analysis
* Decision Tables
* State Transition Testing

1. Mention what are the categories of defects?

**Ans.**[Defect is a variance from a desired product attribute (it can be a wrong, missing or extra data).]

Defect Categories:

* Data Quality/ Database Defects
* Critical Functionality Defects
* Functionality Defects
* Security Defects
* User Interface Defects

1. Mention what Big Bang Testing is?

**Ans.**Big bang is a method if Integration Testing.

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

Big Bang testing has the advantage that everything finished before Integration testing starts.

The major disadvantage is, it is time consuming and bug tracing is difficult as all modules are tested together.

1. What is the purpose of Exit criteria?

**Ans.**Exit criteria is used to determine when testing at any stage is complete, the set of generic and specific conditions, agreed upon with the stakeholders, for permitting a process to be officially completed.

Exit criteria may be defined in terms of:

Thoroughness, Cost or time constraints, Percentage of tests run with incident number of faults remaining.

1. When should “Regression Testing “be performed?

**Ans.** Regression Testing should be carried out:

When the system is stable and the system or the environment changes when testing bug – fix releases as part of the maintenance phase,

It should be applied at all Test levels.

It should be considered complete when agreed completion criteria for regression testing have been met.

1. What is 7 key principles? Explain in detail

**Ans.**7 Key Principles:

* + 1. Testing shows presence of Defects

Testing can show that defects are present.

It reduces the probability of undiscovered defects.

However, Testing cannot prove that there are no defects present.

* + 1. Exhaustive Testing is Impossible

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

* + 1. Early Testing

Testing activities should start as early as possible in the software or the Software Development Life Cycle, and should be focused on defined objectives (Outlined in the Test Strategy).

Testing doesn’t start once the code has been written.

* + 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.

Defects are not evenly spread in a system; they are ‘clustered’.

* + 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 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 Context Dependent.

Testing is done differently in different context, as Different kinds of sites are tested differently.

* + 1. 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.

1. Difference between QA vs QC vs Tester:

|  |  |  |  |
| --- | --- | --- | --- |
|  | QA | QC | Tester |
| Approach | Preventive | Corrective | Corrective |
| Focus | Process Oriented | Product Oriented | Focuses on Finding error/bugs |
| Technique | Implementation of processes & standards in context to verification of intended requirements | Verifying the Quality of the software using testing and with respect to documented requirement | Conducting all types of tests through manual/ automation tools for finding bugs/ errors/ defects |
| Timeframe | Throughout the development process | Before the product release | Along with the development process |
|  | Subset of STLC | Subset of QA | Subset of QC |

1. Difference between Smoke & Sanity:

|  |  |
| --- | --- |
| Smoke | Sanity |
| Smoke Testing is performed to check the application’s critical functionalities working properly or not. | Sanity testing is performed to check the newly added functionalities/ bug fixes. |
| Smoke Testing tests the entire system from end to end. | Sanity testing tests only the critical and new features/component added to the system. |
| Smoke Testing is performed on Initial Builds until the build is stable. | Sanity Testing is performed on stable build. |
| Smoke Testing is subset of acceptance testing. | Sanity Testing is subset of regression testing. |
| The majority of Smoke Testing is documented or scripted. | The majority of Sanity Testing is not documented or scripted. |

1. Difference between Verification & Validation:

|  |  |
| --- | --- |
| Verification | Validation |
| The process of evaluating work products of a development phase to determine whether they meet the specified requirement for that phase. | The process of evaluating software during /at the end of the development process to determine whether it satisfies specified business requirements. |
| To ensure that work products meet their specified requirements. | To demonstrate that the product fulfills its intended use when placed in its intended environment. |
| Are we building the product in right way? | Are we building the right product? |
| Evaluations of Plans, Requirements specifications, Design specifications, Code, Test Cases. | Evaluation of the Actual product/ software. |
| Activities: Reviews, Walkthroughs, Inspections | Testing |

1. Explain Types of Performance Testing

**Ans.**Load Testing (Volume Testing, Scalability Testing)

Stress Testing (Endurance Testing, Spike 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 is called Bug.
* Failure: If the software build does not meet the requirements then it is Failure.

1. Difference between Priority and Severity:

|  |  |
| --- | --- |
| Priority | Severity |
| Priority determines the order in which defects should be fixed. | Severity measures the impact of a defect on the software. |
| Priority is related to scheduling and fixing issues. | Severity is related to quality standards. |
| It is detected by the business value. | It is detected by the software’s functionality. |
| The Product manager sets the priority of bugs. | The test engineer determines the severity level. |
| It can be of High, medium or low types. | It can be Critical, major, moderate, minor types. |
| Priority = ‘How important it is’ | Severity = ‘How bad is it’ |

1. What is Bug Life Cycle?

**Ans.**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 / time span between the first time defects is found and the time that it is closed successfully, rejected, postponed or deferred is called as ‘Bug Life Cycle’.



1. Explain difference between Functional and Non Functional Testing:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | Functional Testing | Non Functional Testing | | Functional Testing is based on an analysis of the specification of the functionality of a component / system. | Nonfunctional testing is the testing of attributes of a component/ system that do not relate to functionality, e.g. reliability, efficiency, usability, interoperability, maintainability and portability. | | Functional Testing should be executed first. | Non Functional Testing should be performed after Functional Testing. | | Easy to do through Manual Testing | Tough to do through Manual Testing | | Functional Testing describes what the product does. | Non Functional Testing describes how good the product works. | | 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 (Instagram, Facebook) only first login page
2. What is the Difference between STLC (Software Testing Life Cycle) & SDLC (Software Development Life Cycle)?

|  |  |
| --- | --- |
| STLC | SDLC |
| Software Testing Life Cycle | Software Development Life Cycle |
| It is for ensuring the developed software is Tested thoroughly. | It is the complete development process of a new Software product. |
| Consists of Test Planning, Test Case Development, Test Strategies’ and Environment setup. | Consists of Planning, Designing, Building/ Coding, Testing, and Maintenance. |
| Ensures the software is defect free and meets the quality standard. | Develops functional and feature complete software. |
| STLC includes Activities like:  Test Planning  Test Planning Strategy  Test Planning Factor  Test Planning Activity  Exit criteria | SDLC includes phases like:  Req. Collection/ Gathering  Analysis  Design  Implementation / coding  Testing  Maintenance |

1. What is the difference between Test Scenarios, Test cases and test Scripts?

|  |  |  |
| --- | --- | --- |
| Test Scenario | Test Cases | Test scripts |
| A test Scenario is any functionality that can be tested. | 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 to execute a core business function. |
| It is ‘what to be tested’. | It is ‘how to be tested’. | It is written to explain how to stimulate each business scenario. |
| It is derived from test artifacts like Business Requirement Specification (BRS) and Software Requirement Specification(SRS) | It is mostly derived from test scenarios. | It is mostly derived from Test cases. |
| Helps to test the End To End functionality. | Helps in exhaustive testing of an app. | Helps to test specific things repeatedly. |
| 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. |
| Includes an End to End Functionality to be tested. | Includes test steps, data, expected results for testing. | Includes different commands to develop a script. |
| The main task is to check the full functionality of a software application. | The main task is to verify the applicable standards and customer requirements. | The main task is to verify nothing is skipped and the results are as per the testing plan. |
| 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?

**Ans.**A document describing the scope, approach, resources and schedule of intended test activities.

Determining the scope and risks, and identifying the objectives of testing.

The covered activities that should be covered:

Determining the scope and risks and identifying the objectives of the testing.

Defining the overall approach of testing including test entry and exit criteria.

Integrating and coordinate the testing activities in to software life cycle.

Scheduling test analysis, design, implementation, execution and evaluation activities.

1. What is Priority?

**Ans.**Priority defines the order in which we should resolve a defect.

Priority is Relative and Business Focused.

Priority is customer’s point of view, but it can be set by QA tester.

The importance given to the bug to fix it is that is priority.

The high priority indicates that the bug to fix it first.

Priority can be of following types:

Low, Medium, High, Critical

1. What is Severity?

**Ans.**Severity is the extent to which the defect can affect the software.

Severity is Absolute and Customer Focused.

If that impact is more then, there is high severity.

If the impact is less then, there is low severity.

Severity can be of following type:

Critical, Major, Moderate, Minor, Cosmetic

1. Bug Categories are?

**Ans.**Security, Database, Functionality (critical/ general), UI

1. Advantage of Bugzilla

**Ans.**Bugzilla is an open source issue / bug tracking system.

Advantages are like: Advanced search capabilities, Email notifications, Modify/ file Bugs by e-mail, Time Tracking, Strong Security, Customized Localization

1. Difference between Priority and Severity?

Same As Ans. 28

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

* Agile SDLC model is a combination of Iterative and Incremental process models with focus on process adaptability and customer satisfaction by rapid delivery of working software product.
* Agile methods break the product in to small incremental builds.
* These builds are provided in Iterations.
* Each Iteration typically last from about one to three weeks.
* Every Iteration involves cross functional teams working simultaneously on various areas like planning, requirements, analysis, design, coding, unit testing and acceptance testing.

1. At the end of Iteration, a working product is displayed to the customers and important stakeholders When to use Usability Testing?

Usability testing is a process that helps to understand how users interact with your software, It’s used throughout the development process to identify problems early and improve user experience.

When to use Usability Testing:

* Early in Development: to Identify problems at early stage which can save time and money
* Before Launch: Ensure a smooth user experience from the start
* After updates: Verify new features are user-friendly and don’t break existing functionality.

1. What is the procedure for GUI Testing?

Procedure for GUI Testing is as following:

* Checking the screens with controls like menus, buttons, icons and all types of bars tool bar, menu bar, dialog boxes and windows etc.
* Check to the execute intended functionality of the appliocation using GUI.
* Check all the GUI elements for size, position, width, length and acceptance of characters of numbers.
* You must be able to provide inputs to the input fields.
* Check for the Error messages, if they are displayed correctly.
* Check for the demarcation of different sections on screen.
* Check for Various elements like:
  + - * + Font used are readable
        + Alignment of text is proper
        + Color of the fonts
        + Warning messages are displayed properly
        + Images are of good clarity
        + Images are properly aligned
        + The positioning of GUI elements of different screen resolution

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

In Web Testing

Authentication: Authentication verifies a user’s identity.

Authorization: Authorization determines what actions a user can perform.

Both are security processes that helps to prevent unauthorized users from accessing resources.

|  |  |
| --- | --- |
| Authentication | Authorization |
| Verifies a user’s identity by asking for credentials like a username and password | Determines what actions a user can perform |
| Usually the first step of a security access control | Usually comes after authentication |
| Verifies the user identity | Grants or denies permission to the user |
| Its visible and changeable by the user | Its not visible or changeable by the user |
| Occurs before authorization | Occurs after authentication |
| Example: A student can authenticate himself before accessing the learning management system of an institute | Example: Student can access lecture videos and other learning material of the courses based on the permission given to him |

Common problems faced in web testing include:

* Security: Web applications are susceptible to vulnerabilities and cyber-attacks.
* Performance: Poorly written code, un-optimized databases and traffic spikes can cause performance issues.
* Scalability: As web applications grow they need to handle more users and data without slowing down or crashing.
* User experience: Poor design can frustrate users and make it difficult to interact with the application.
* Accessibility: Web applications should be easily accessible by all.
* Testing/Test Automation: If testing is not done properly, it can lead to low reliability, high maintenance and poor performance.
* Communication: Poor communication can lead to misunderstandings and delays.

1. To Create HLR & Test Case of
   * + Instagram (only first page)
     + Facebook (only first page)
     + Facebook Login Page
2. To Create HLR & Test Case of Web Based
   * + WhatsApp Web
     + Instagram Web
3. To Create HLR & Test Case on the link <https://artoftesting.com/>
4. Write Scenario of only WhatsApp chat messages
5. Write Scenario of Pen
6. Write Scenario of Pen Stand
7. Write Scenario of Door
8. Write Scenario of ATM
9. Write Scenario of Microwave Oven
10. Write Scenario of Coffee Vending machine
11. Write Scenario of Chair
12. To create Scenario of Gmail
13. To create Scenario of online shopping to buy product (flip cart)
14. Write Scenario of Wrist Watch
15. Write Scenario of Lift(Elevator)
16. Write Scenario of WhatsApp Group (generate group)
17. Write Scenario of WhatsApp payment