**1)What is Exploratory Testing?**

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

Testing is based on a test charter that may include

Scope of the testing (in and out)

The focus of exploratory testing is more on testing as a “thinking”

activity.

A brief description of how tests will be performed

Expected problems

Is carried out in time boxed intervals

More structured than Error guessing

Risk Analysis Charter ExploratoryWhat is traceability matrix?

**2) What is traceability matrix?**

A Traceability Matrix is a document that co-relates any two-baseline documents that require a many-to-many relationship to check the completeness of the relationship.

It is used to track the requirements and to check the current project requirements are met.

**3)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.

At those points when input values change from valid to invalid errors are most likely to occur.

Boundary Value Analysis (BVA) uses the same analysis of partitions as EP and is usually used in conjunction with EP in test case design

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

Aim is to treat groups of inputs as equivalent and to select one representative input to test them all

EP can be used for all Levels of Testing

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.

If we want to test the following IF statement: “If value is between 1 and 100 (inclusive) (e.g value >=1 and value <=100) Then...”

We could put a range of numbers as shown in the below ﬁgure.

**5)What is Integration testing?**

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.

The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.

Integration testing tests integration or interfaces between components, interactions to diﬀerent parts of the system such as an operating system, ﬁle system and hardware or interfaces between systems.

Integration testing is done by a speciﬁc integration tester or test team.

Components may be code modules, operating systems, hardware and even complete systems

There are 2 levels of Integration Testing

Component Integration Testing System Integration Testing

**6)What determines the level of risk?**

The likelihood of an adverse event and the impact of the event.

**7) What is Alpha testing?**

* 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. It is always performed in **Virtual Environment**.
* It is always performed within the organization. It is the form of Acceptance Testing.
* Alpha Testing is deﬁnitely performed and carried out at the developing organizations location with the involvement of developers.
* It comes under the category of both White Box Testing and Black Box Testing.

**8) What is beta testing?**

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.

It is performed in Real Time Environment. It is always performed outside the organization. It is also the form of Acceptance Testing.

Beta Testing (ﬁeld testing) is performed and carried out by users or you can say people at their own locations and site using customer data.

It is only a kind of Black Box Testing.

**9) What is component testing?**

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

* Component Testing – The 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. The purpose is to validate that each unit of the software performs as designed.
* Unit testing is the ﬁrst level of testing and is performed prior to Integration Testing.
* Sometimes known as Unit Testing, Module Testing or Program Testing
* Component can be tested in isolation – stubs/drivers may be employed
* Unit testing frameworks, drivers, stubs and mock or fake objects are used to assist in unit testing.
* Functional and Non-Functional testing
* Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended with debugging

**10) 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 Requirement Based Functional Testing Process Based Testing

* **11) 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
* May be performed at all Test levels (not just Non Functional Systems Testing)
* Measuring the characteristics of the system/software that can be quantiﬁed on a varying scale- e.g. performance test scaling
* Non-functional testing includes, but is not limited to, performance testing, load testing, stress testing, usability testing, maintainability testing, reliability testing and portability testing.

**12) 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.

WHAT DO YOU CHECK IN GUI TESTING?

* 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.

**13) What is 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.

In fact is does not create test cases altogether!

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

Testers randomly test the application without any test cases or any business requirement document.

Adhoc Testing does not follow any structured way of testing and it is randomly done on any part of application.

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

Adhoc testing can be achieved with the testing technique called Error Guessing.

Error guessing can be done by the people having enough experience on the system to “guess” the most likely source of errors.

**14) 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.

This testing usually identiﬁes –

The maximum operating capacity of an application

Determine whether current infrastructure is suﬃcient to run the application Sustainability of application with respect to peak user load

Number of concurrent users that an application can support, and scalability to allow more users to access it.

It is a type of non-functional testing. Load testing is commonly used for the Client/Server, Web based applications – both Intranet and Internet.

**15) What is stress Testing?**

In 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 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.

It even tests beyond the normal operating point and evaluates how the system works under those extreme conditions.

Stress Testing is done to make sure that the system would not crash under crunch situations.

Stress testing is also known as endurance testing.

**16) What is white box testing and list the types of white box testing?**

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.

In white-box testing the tester is concentrating on how the software does it.

For example, a structural technique may be concerned with exercising loops in the software.

Diﬀerent test cases may be derived to exercise the loop once, twice, and many times. This may be done regardless of the functionality of the software.

Structure-based techniques are also used in system and acceptance testing, but the structures are diﬀerent.

For example, the coverage of menu options or major business transactions could be the structural element in system or acceptance testing.

**Types of white box testing:**

* **Test code coverage**
* **Decision coverage**
* **Statement coverage**
* **Condition coverage**

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

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

Speciﬁcation-based testing technique is also known as ‘black-box’ or input/output driven testing techniques because they view the software as a black-box with inputs and outputs.

The testers have no knowledge of how the system or component is structured inside the box. In black-box testing the tester is concentrating on what the software does, not how it does it.

Speciﬁcation-based techniques are appropriate at all levels of testing (component testing through to acceptance testing) where a speciﬁcation exists.

For example, when performing system or acceptance testing, the requirements speciﬁcation or functional speciﬁcation may form the basis of the tests.

**Black box testing techniques:**

* Equivalence partitioning
* Boundary value analysis
* Decision table
* State transition testing
* Use-case Testing
* Other Black Box Testing

Syntax or Pattern Testing

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

**- functionality defect**

**-performance defect**

**-UI defect**

**-compatibility defect**

**-security defect**

**-documentation defect**

**-database defect**

**19) Mention what bigbang testing is?**

Big Bang Integration Testing is an integration testing strategy wherein all units are linked at once, resulting in a complete system. When this type of testing strategy is adopted, it is difficult to isolate any errors found, because attention is not paid to verifying the interfaces across individual units.

**20) What is the purpose of exit criteria?**

Exit criterion is used to determine whether a given test activity has been completed or NOT. Exit criteria can be defined for all of the test activities right from planning, specification and execution. Exit criterion should be part of test plan and decided in the planning stage

**21) When should "Regression Testing" be performed?**

Regression testing is necessary after any feature (or application) enhancement, bug fix, or configuration changes. For example, when developers add a new widget to an application. As more regressions are found in software products, companies are moving towards test automation to perform regression tests

**22) What is 7 key principles? Explain in details**

## **1) Testing shows a presence of defects**

Hence, testing principle states that – Testing talks about the presence of defects and don’t talk about the absence of defects. i.e. Software 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.

**2) exaustive testing is imposible**

Testing everything including all combinations of inputs

and preconditions is not possible.

So, instead of doing the exhaustive testing we can use risks and

priorities to focus testing efforts.

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.

**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 These activities should be focused on defined objectives – outlined in the Test Strategy

**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) pestiside 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.

**6) abscence 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.

If we build a system and, in doing so, find and fix defects ....

It doesn’t make it a good system

Even after defects have been resolved it may still be unusable and/or

does not fulfil the users’ needs and expectations

**7) testing is context dependent**

Testing is basically context dependent.

Testing is done differently in different contexts

Different kinds of sites are tested differently.

For example

Safety – critical software is tested differently from an

e-commerce website

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

QA intends to prevent some defects from ever occurring, through collaboration practices to align teams and methods such as code reviews. Through QC, testers identify and prioritize what defects remain. The emphases are different, but the end goal is the same. Who does the work. QA is truly an organization-wide effort

**24) Difference between Smoke and Sanity?**

Smoke testing is executed at the initial stage of SDLC, to check the core functionalities of an application. Whereas Sanity & Regression testing are done at the final stage of SDLC, to check the main functionalities of an application

**25) Difference between verification and Validation**

Validation is the process of checking whether the specification captures the customer's requirements, while verification is the process of checking that the software meets specifications. Verification includes all the activities associated with the producing high quality software.

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

### **Load Testing**

Load testing is a type of performance testing that evaluates an application’s or system’s performance under typical and expected user load. The primary goal of load testing is to understand how the system processes user traffic and transactions, ensuring it remains stable and accessible under these conditions. It is a crucial step in ensuring the reliability and scalability of applications or systems, especially those with high user traffic or those that handle critical business processes.

### **Stress Testing**

Stress testing is a critical performance testing methodology that focuses on determining how an application or system performs under extreme load conditions, often exceeding its maximum capacity. The fundamental purpose of stress testing is two-fold. Firstly, it discerns the system’s absolute limit or its breaking point, post which it can no longer function as expected. Secondly, stress testing aims to understand how the system recovers from these intense stress situations, often called its resilience or robustness.

**Scalability testing** ensures that the application can handle growth in terms of users, data volume, and transaction volume, making it indispensable for forward-looking development.

**27) What is Error, Defect, Bug and failure?**

* Error: A mistake made by humans during coding.
* Defect: An error found during the testing phase.
* Bug: A defect to be resolved by the development team.
* Failure: When a build does not meet its specifications

**28) Difference between Priority and Severity**

Priority is a term that defines how fast we need to fix a defect. Severity is basically a parameter that denotes the total impact of a given defect on any software. Priority is basically a parameter that decides the order in which we should fix the defects. Severity relates to the standards of quality.

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

Defect Life Cycle or Bug Life Cycle in software testing is the specific set of states that defect or bug goes through in its entire life. The purpose of Defect life cycle is to easily coordinate and communicate current status of defect which changes to various assignees and make the defect fixing process systematic and efficient.

**30) Explain the difference between Functional testing and NonFunctional testing**

Functional testing determines whether the software program meets the requirements. Non-functional testing examines the software’s usability, reliability, performance, and other non-functional elements. Functional testing focuses on what the software does, whereas non-functional testing focuses on how well the software performs it.

**31) HLR & TESTCASE of instagram first page**

[**https://docs.google.com/spreadsheets/d/1bz9hL4y8Wq5-bIe-6kXU98pCewOaYC3zSOQaS8tA8Tk/edit?usp=sharing**](https://docs.google.com/spreadsheets/d/1bz9hL4y8Wq5-bIe-6kXU98pCewOaYC3zSOQaS8tA8Tk/edit?usp=sharing)

**32) HLR & TESTCASE of facebook login page**

[**https://docs.google.com/spreadsheets/d/1gJ627RSMytUXgGFcz4dNFlEqNzr17ZfKkgDE7D\_KPvA/edit?usp=sharing**](https://docs.google.com/spreadsheets/d/1gJ627RSMytUXgGFcz4dNFlEqNzr17ZfKkgDE7D_KPvA/edit?usp=sharing)

**33) what is the difference between the STLC and SDLC**

**Difference between SDLC and STLC:**

|  |  |
| --- | --- |
| **SDLC** | **STLC** |
| SDLC is mainly related to software development. | STLC is mainly related to software testing. |
| Besides development other phases like testing is also included. | It focuses only on testing the software. |
| SDLC involves total six phases or steps. | STLC involves only five phases or steps. |
| In SDLC, more number of members (developers) are required for the whole process. | In STLC, less number of members (testers) are needed. |
| In SDLC, development team makes the plans and designs based on the requirements. | In STLC, testing team(Test Lead or Test Architect) makes the plans and designs. |
| Goal of SDLC is to complete successful development of software. | Goal of STLC is to complete successful testing of software. |
| It helps in developing good quality software. | It helps in making the software defects free. |
| SDLC phases are completed before the STLC phases. | STLC phases are performed after SDLC phases. |
| Post deployment support , enhancement , and update are to be included if necessary. | Regression tests are run by QA team to check deployed maintenance code and maintains test cases and automated scripts. |
| Creation of reusable software systems is the end result of SDLC. | A tested software system is the end result of STLC. |

**34) What is the difference between test scenarios, test cases, and test script?**

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.

**35) What is the difference between test scenarios, test cases, and test script?**

A Test Plan is a detailed document that catalogs the test strategies, objectives, schedule, estimations, deadlines, and resources required to complete that project. Think of it as a blueprint for running the tests needed to ensure the software is working correctly – controlled by test managers

**36) What is priority?**

One can define Priority as a parameter for deciding the order in which one can fix the defect. In this, the defect with a higher priority first needs to get fixed. Priority basically defines the order in which one would resolve any given defect.

**37) What is severity?**

Severity is a term that denotes how severely a defect can affect the functionality of the software. Priority is a term that defines how fast we need to fix a defect. Parameter. Severity is basically a parameter that denotes the total impact of a given defect on any software.

**38) Bug categories are…**

Software bugs can be classified into multiple categories based on their nature and impact. Broadly speaking, these categories include Functional Bugs, Logical Bugs, Workflow Bugs, Unit Level Bugs, System-Level Integration Bugs, Out of Bound Bugs, and Security Bugs

**39) Advantage of Bugzila .**

it improves the quality of the product. It enhances the communication between the developing team and the testing team. It has the capability to adapt to multiple situations

**40) What are the different Methodologies in Agile Development Model?**

**Scrum. ...**

**Extreme Programming (XP) ...**

**Lean Product Development. ...**

**Feature-Driven Development (FDD) ...**

**Dynamic Systems Development Method (DSDM) ...**

**Kanban. ...**

**Crystal. ...**

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

authentication is the process of verifying who someone is, whereas authorization is the process of verifying what specific applications, files, and data a user has access to. The situation is like that of an airline that needs to determine which people can come on board.

**42) HLR & TESTCASE on whatsapp web and instagram web**

[**WHATSAPP WEB**](https://docs.google.com/spreadsheets/d/1CMJ5By_7bv9GZosX9Ox6X5QiwLcBntM8f6Fnv9YxHLc/edit?usp=sharing)

[**INSTAGRAM WEB**](https://docs.google.com/spreadsheets/d/1sNhXCLFWX3HVvtmOy_3qxkWvwyFO6BZQEHJPE7C8SdM/edit?usp=sharing)

**43) To create HLR and TestCase on this Link.** [**https://artoftesting.com/**](https://artoftesting.com/)

[**https://docs.google.com/spreadsheets/d/1bS4AH1vysgh-YawOdMmgvdH9iqEKhNVBsRIFUQMLWFE/edit?usp=sharing**](https://docs.google.com/spreadsheets/d/1bS4AH1vysgh-YawOdMmgvdH9iqEKhNVBsRIFUQMLWFE/edit?usp=sharing)

**44) scenarios**

**1)** [**https://docs.google.com/spreadsheets/d/1uVXXRglApBOuf13x\_THMyR57BRANfKZc4r-LJMpRq74/edit?usp=sharing**](https://docs.google.com/spreadsheets/d/1uVXXRglApBOuf13x_THMyR57BRANfKZc4r-LJMpRq74/edit?usp=sharing)

**2)** [**https://docs.google.com/spreadsheets/d/12rDcyu\_R\_HsQHZbOTnmZtH272ruGsxPG9JhQ02spK1o/edit?usp=sharing**](https://docs.google.com/spreadsheets/d/12rDcyu_R_HsQHZbOTnmZtH272ruGsxPG9JhQ02spK1o/edit?usp=sharing)

**45) When to used Usablity Testing?**

Once you've got an idea, conduct usability testing before putting any design resources to work. Identify specific areas where testing and validation can enhance your concept. After you get the results from your initial test, share them with your team. Then, continue testing users as you build a prototype.

**46) What is the procedure for GUI Testing?**

Manual Testing. This approach involves human tester, where each screen is manually checked to validate each functionality by creating and executing test cases. ...

Record and Replay Testing. GUI record and replay tools are used to test the user interface of apps. ...