1.what is exploratory testing?

Exploratory testing is a type of software testing where test cases are not created in advance but testers checks system on the fly. they may note down ideas about what to test before test execution. the focus of exploratory testing is more on testing as thinking activities

**2.what is tractability matrix?**

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

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

3.what s boundary value testing?

Boundary value analysis is based on the testing the boundary values of the valid and invalid portioning. The behavior at the edge of the equivalence portioning is more likely to be incorrect than the behavior within that portioning, so boundaries are an area when testing is likely to yield defects.

It checks for the input values near the boundary that have higher chance of error. every portioning has its maximum values are the boundary values of portioning

**4. what is equivalence portioning**?

E. p is software testing technique that divides input domain in to class of the data and with help of class and data test can be derived .an ideal test case identifies class of error that might require many arbitrary test cases to be executed before general error is observed.

**5.what is integration testing?**

Integration testing also known as the integration and testing (L&T) is a type of software testing which the different unit’s modules components of a software application are tested a combined entity.

The aim of the integration testing is to test the interface between the modules and expose any defects that may arise when these components are integrated and needed to interact with each other.

**6. what determines level of risk?**

RISK is a factor that could result in future negative consequences.

Types of risk: -

A risk could be future event with a negative consequence

1. Project risk
2. Product risk

**7. what is alpha testing?**

Alpha testing is the initial phase of validating whether a new product will perform as expected.

It is always performed by developers at software testing site

Alpha testing is not open market and public

It is always preformed in virtual environment

It is a form of acceptance testing.

**8. what is beta testing?**

Beta testing is an opportunity for real user s to use product in production environment to uncover any bugs or issues before general release

It usually performed outside of organization

It is a one kind of black box testing

Beta testing can be considered **pre released testing**

**9. what is component testing?**

In this type of testing test objects can be tested independently as component without integrating with other components

For example; - suppose we have one website which includes five different web pages therefore testing each website individually and with the isolation other components

**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 lest documents and/ or a model

**Two types of test approach.**

1.Requirement based functional testing

2.Process based functional Testing

**Functional system testing functionality**

1.**Accuracy**- Provision of right or agreed results or effects

2.**Interaperability**- Ability to interact with specified systems

3.**Compliance**- Adhere to applicable standards convention regulation or laces

4.**Auditability**- Ability to provide adequate and accurate audit data.

5.**Suitablility**- Presence and appropriateness of functions for specified tasks

**11.what is non-functional testing ?**

Non-functional testing checks the performance reliability, scalability and other non- functional aspect of the software system

* Non-functional testing should be this testing after functional testing
* Using tools will be effective for this testing
* Performance parameters like speed, scalability are input to non-functional testing

**12. what is GUI testing?**

Graphic User Interface Testing (GUI) testing is the process of ensuring proper functionality of the graphical user interface (GUI) for a specific application. This involves making sure it behaves in accordance with its requirements and works as expected across the range of supported platforms and devices.

**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

This testing is primarily performed is the knowledge of tester in the system under test is very high

Adhoc testing can be achieved with the testing technique called ERROR GUESSING

Main aim of this testing is find defects by random checking

**14. what is load testing?**

Load testing is a type of performance testing that determines the performance of the system.

Load testing determines the behaviours of the application when multiple users use it at same time.

**15. what is stress testing?**

Stress testing is software testing technique that determines the robustness of the software by testing beyond limits of normal operations.

Stress testing is particularly important for critical software but is used for all types of software.

**16. what is white box testing list type of white box testing?**

White box is a type of software testing that assesses an application's internal working structure and identifies its potential design loopholes.

Loop Testing

Conditional Testing

Unit Testing

Mutation Testing

Integration Testing

Penetration Testing

Testing based on Memory Perspective

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

**black Box Testing** is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths.

Functional testing – This black box testing type is related to the functional requirements of a system; it is done by software testers.

Non-functional testing – This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance, scalability, usability.

Regression testing – [Regression Testing](https://www.guru99.com/regression-testing.html) is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.

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

Different types of defects are mentioned below

1. Logical Defects.
2. Syntax Defects.
3. Multithreading Defects.
4. Interface Defects.
5. Performance Defects.

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

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

* The major disadvantage is that is general it is time consuming and difficult to true the cause of failures because of this late integration
* Here all components are integrated together in once and then tested.

**20. what is purpose of exit criteria?**

The main purpose of the exit crate area is when to stop testing

* Run out of time
* Run out of budget
* This business tells you it went live lost night
* Boss says stop
* All defects have been fixed
* When out exit criteria have been met.

**21.when should regression testing be performed?**

Ideally regression testing should be performed whenever your code base has been modified or altered.

Frequent partial regression testing will help developer fix reported defects on the time

Most common reason to run regression test is the introduction of new functionality.

Regression testing can save developers lot of time with timely detection of bugs that would otherwise cause the projects a lot of pain in long time .

**22. what is 7 key principle? explain in detail.**

According to ISTQB **(international software testing qualification board)** the seven principle of software testing are mentioned below

1. EXHASUSTING TESTING IS NOT POSSIBLE
2. DEFECT CLUSTERING
3. PESTICIDE PARADOX
4. TESTING SHOWS PRESENCE OF DEFECTS
5. ABSENCESE OF ERROR FALLACY
6. EARLY TESING
7. TESTING IS CONTEX DEPENDENT

**1.EXHASUSTING TESTING IS NOT POSSIBLE**

Exhaustive testing is not possible instead we need optimal testing based on risk assessment

Testing everything is not possible

**2.Defect clustering**

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

**3.Pesticide paradox**

If the same test is repeated over and over eventually the same set of test cases will no longer find any new defects

To overcome this problem test case needs to regularly reviewed and revised and new different parts of the software or system to potentially find more defects.

**4.Testing shows presence of defects**

Testing shows that defects are present but cannot improve that there are not defects.

**5. ABSENCESE OF ERROR FALLACY**

If the system built is unusable and does not full fill the user’s need and expectations, then

finding and fixing defects does not help.

**6.Early testing**

Testing activities should start as early as possible in the software or system or system

development cycle and should be focused on defined objective

**7. Testing is context dependent**

Testing is cortex dependent which is basically means that the way you test e- commerce site will be different from the way you test commercial off the application

Exa critical software is tested differently than the e-commerce website

23. Difference between QA v/s QC and Testers

**QA**

-Activities which ensure the implementation of processes, procedures and standards in contest to verification of developed software and intended requirement

- Focuses on processes and procedures rather than conducting actual testing on the system

-process oriented activities

-preventive activities

-It is a subset of software test life cycle (STLC)

**QC**

-Activates which ensure the verification of development software with respect to documented (or not in some cases) requirement

-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

**Testers**

Software testing refers to the activities that are preformed on program after it has been written.

Software testing initiates verification of application of application functionality as per requirements

It is based on product oriented activities .

Testing is subset of quality control.

**24. difference between smoke and sanity**

**Smoke Testing: -**

**-**Check the critical functionality

-It’s done in initial stage

-It checks the stability

-Part of acceptance testing

-General health check up

-Done by tester and developer

-Its checks the system end to end

-20 test cases it should take 30 min to test

**Sanity testing: -**

-Check the new functionality

-It is done after 30 build

-It checks the sanity/rationality

-part of regression testing

-Advance health check up

-Done by tester

-It checks only a particular function of entire system

**25. difference between verification and validation**

**Verification**

It is process of checking if product is developed as per specification

It tests requirement architecture design and code of software product

It does not require executing the code

A few activities involved in verification testing are requirement verification and code

It targets internal aspect such as requirement design software architecture data base and code

**Validation**

It processes of ensuring that the product meets the need and expectation of stake holders

It tests the usability functionalities and reliability of product

It emphasizes executing codes to test the usability and functionality of end product

The commonly used validation activities in software testing are usability, performance, system and security testing

It targets product ready to be deployed

**26.types of performance testing**

There are mainly six types of performance testing is done which are mentioned below

1.load testing

2. stress testing

3.endurance testing

4.spike testing

5.volume testing

6.Scalablity testing

**27.whatis error, bug, defect and failure?**

Error

An error is defining as a human action that produce incorrect result

Defect

A flaw in a component or system that can cause the component or system to fail to perform its required function

Failure

It is defining as the deviation of component or system from its expected delivery service or result

Bug

A fault in program which causes the program to perform in an untitled or unanticipated manner

**28. difference between priority and severity**

Difference Between Severity and Priority in Software Testing

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Severity in Testing** | **Priority in Testing** |
| Definition | 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. | Priority is basically a parameter that decides the order in which we should fix the defects. |
| Relation | Severity relates to the standards of quality. | Priority relates to the scheduling of defects to resolve them in software. |
| Value | The value of severity is objective. | The value of priority is subjective. |
| Change of Value | The value of Severity changes continually from time to time. | The value of Priority changes from time to time. |
| Who Decides the Defect | The testing engineer basically decides a defect’s severity level. | The product manager basically decides a defect’s priority level. |
| Types | There are 5 types of Severities: Cosmetic, Minor, Moderate, Major, and Critical. | There are 3 types of Priorities: High, Medium, and Low. |

**29. what is bug life cycle ?**

A computer bug is an error flaw mistake, failure, or a fault in a computer program that prevents it from working correctly



**30. difference between functional and non-functional testing**

| Functional Testing | Non-functional Testing |
| --- | --- |
| It verifies the operations and actions of an application. | It verifies the behaviour of an application. |
| It is based on requirements of customer. | It is based on expectations of customer. |
| It helps to enhance the behaviour of the application. | It helps to improve the performance of the application. |
| Functional testing is easy to execute manually. | It is hard to execute non-functional testing manually. |
| It tests what the product does. | It describes how the product does. |
| Functional testing is based on the business requirement. | Non-functional testing is based on the performance requirement. |
| **Examples:**  **1.** Unit Testing  **2.** Smoke Testing  **3.** Integration Testing  **4.** Regression Testing | **Examples:**  **1.** Performance Testing  **2.** Load Testing  **3.** Stress Testing  **4.** Scalability Testing |

**31. What is 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. |

**32. what is difference between test cases , test scenarios and test scripts**

**Test cases**

A test case is a high-level document with instructions on the specific functionality of the software product to be tested.

A test case is the software development life cycle’s ‘What to test’ component.

Test cases are written in simple English.

A test case is a document with instructions on testing the specific functionality of an application.

Test scenarios serve as an outline for writing test cases

Test cases are primarily used in Manual Testing.

It ensures end-to-end test coverage with assumed data types.

Test cases take a lot of time and resources to document.

**Test scripts**

Test Script is a step-by-step instruction to test each software product’s functionality (test case)

Test script is the software development life cycle’s ‘How to test’ component.

Test scripts are written in programming languages like VB Script, Python, Java, etc.

Test Script is a program that runs various test data on the functionality of an application.

Test Script is a program that runs various test data on the functionality of an application.

Test Case serves as an outline for writing test scripts.

 Test scripts are widely used in Automation

It can serve as a reusable component. A single script can run tests with various types of data.

The time to execute Test scripts is less than the time to write test cases.

Test scenarios

A test scenario contains high-level documentation which describes an end to end functionality to be tested.

It focuses on more “what to test” **than** “how to test”.

Test scenarios are a one-liner. So, there is always the possibility of ambiguity during the testing.

Test scenarios are derived from test artefacts like BRS, SRS, etc.

It helps in an agile way of testing the end to end functionality

Test scenarios are high-level actions.

Comparatively less time and resources are required for creating & testing using scenarios.

**33. explain what is test plan is? what is the information that should be covered.**

Test plan is determining as a document describing the scope, approach, resources and schedule.

Test plan should include following things

1. Test Strategy and Objectives  
Identify the main purpose of testing (in light of the product requirements) and what a successful completion of a testing cycle looks like. The two important factors in terms of quality assurance are usually coverage and velocity. The test strategy will define which one comes first (usually based on the business objectives).

2. Schedule, Estimation, and Deliverables  
This depends whether your test cycle occurs during or after the cycle, but it is important to be able to determine how long will it take to qualify the tested functionality. And no less important is to understand “what will I get” as a user by the end of the testing cycle — is there any traceability matrix report I can rely on in order to support the decision making process? Predictability is key in this area, as the business would always want to know when we should be ready to ship.

3. Resources Required to Perform Testing  
Given different coverage needs, it is important to understand what will it take — for example — to complete a testing cycle within x amount of days/hours. Resources include people (testing team) and technology (testing platforms/product units, testing tools — frameworks and other orchestration/execution tools).

**34. what is priority?**

Priority is defined as the order in which the defects should be resolved.

The priority status is usually set by the testing team while raising the defects against the developer team mentioning the timeframe to fix defect.

The priority is based on end user requirement.

**35. what is severity?**

The impact of the bug or defect on the application is known as the severity of how blocker defects and what is the impact of the defect on whole system ‘s functionality.

The severity is the parameter set by the tester while opening bug/ defect and essentially in control the tester.

**36. bugs categories are…**

Bugs categories are mention below

1. Database defect
2. Critical defect
3. Functionality defect
4. Security defect
5. User interface defect

Database defect

Deal with improper handling of the data in the database.

Value not inserted in to database properly

Improper values inserted in place of the actual value

Critical defect

The occurrence of the bugs hampers the crucial functionality of the application

Exa exceptions

Functionality defects

These defects affect the functionality of the application

Exa java script

Security defects

Application security defect generally involve import handling of data sent from the user to the improper handling of data sent from the user to the application.

These defects are the most severe given highest priority for a fix.

* Authentication
* Authorisation

User interface defects

As the name suggest the bugs deals with problem related to ui are usually considered less severe

e.x improper error

spelling mistake

alignment problem

**37. Advantages of Bugzilla**

**The Advantages of Bugzilla are:**

* it is an open-source widely used bug tracker;
* it is easy in usage and its user interface is understandable for people without technical knowledge;
* it easily integrates withtest management instruments;
* it integrates with an e-mailing system;
* it automates documentation.

**38. Difference between priority and severity**

## Difference Between Severity and Priority in Testing

|  |  |  |
| --- | --- | --- |
| **Parameters** | **Severity in Testing** | **Priority in Testing** |
| Definition | 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. | Priority is basically a parameter that decides the order in which we should fix the defects. |
| Relation | Severity relates to the standards of quality. | Priority relates to the scheduling of defects to resolve them in software. |
| Value | The value of severity is objective. | The value of priority is subjective. |
| Change of Value | The value of Severity changes continually from time to time. | The value of Priority changes from time to time. |
| Who Decides the Defect | The testing engineer basically decides a defect’s severity level. | The product manager basically decides a defect’s priority level. |
| Types | There are 5 types of Severities: Cosmetic, Minor, Moderate, Major, and Critical. | There are 3 types of Priorities: High, Medium, and Low. |

**39. what are the different methodology in Agile development model**

Agile methodology is a “step by step” dynamic focused on short-term visibility but never losing the long-term product goal.

## Main Agile methodologies:

### **1. Scrum**

### Scrum is the most used of the many frameworks underpinning Agile methodology.

**Scrum is characterised by cycles or stages of development, known as sprints**

It is usually used in the management of the development of software products but can be used successfully in a business-related context.

### **2. Kanban**

The word Kanban is of Japanese origin and its meaning is linked to the concept of “just in time”.

the Kanban method is organised on a board or table (Kanban board), divided into columns, showing every flow within the software production project.

As the development evolves, the information contained in the table changes, and whenever a new task comes into play, a new “card” is created.

This methodology is also useful in individual business departments, such as HR, marketing, etc., bringing the desired visibility over all the team’s tasks.

The Kanban method **requires communication and transparency** so that the members of any team all know exactly what stage development is at and can see the status of a project at any time.

### **3. Extreme Programming**

This is a typical Agile development framework, developed by Kent Beck, and can be adapted to development companies of various dimensions.

**Extreme Programming** methodology is based around the idea of discovering “the simplest thing that will work” without putting too much weight on the long-term product view.

Teamwork is extremely important in XP, since, when there is a problem, it is solved by the whole team of managers, developers or customers, bringing them together to promote conversation and engagement and break down barriers to communication.

### **4. Lean Development**

Lean development is a methodology that comes directly from Lean Manufacturing, created by Toyota, and applied to software development.

This method offers a conceptual framework and follows values, principles and good development practices that can be applied to an Agile development approach.

**Lean development forces the team to ruthlessly remove any activity that does not bring ultimate value to the product.**

### **5. Crystal**

This is a family of Agile methodologies, and **Crystal is one of the most flexible frameworks, giving tremendous freedom to the team to develop their own processes.**

It focuses way more on individuals and how they interact rather than on the process or the tools so communication is an essential key aspect.

Crystal has variants such as **Crystal Clear** (up to an 8-person team), **Crystal Yellow**(up to a 10 to 20-person team), **Crystal Orange** (up to a 20 to 50-person team) and **Crystal Red**(for big teams with 50 to 1000 people).

Crystal focuses on principles such as People, Interactions, Community, Skills, Talent and Communication, aiming to deliver the best possible software development process.

It’s a light methodology in terms of documentation, where teams can find their own ways over preferred work modalities, removing management overheads and creating a “free” process.

**40. explain the difference between authorization and authentication in web testing, what are the common problems faced in web testing?**

**Authentication: - a**ccepting an invalid user name / password

**Authorization: - a**ccessibility to pages through permission not given

In web testing some common problems are mentioned below

1. interoperability
2. security
3. performance
4. responsiveness

**41. when to use usability testing**

usability testing can and should be conducted on the current iteration of a product before beginning any new design work, after you’ve begun the strategy work around a brand new site or app.

This will quickly identify areas for opportunity, and reduce the amount of assumptions your design team will make with regard to what the user wants.

Additionally, after the usability tests analysis, the team should have the ability to pinpoint the steps needed to achieve the project goals with as little disruption as possible.

**42. what is the procedure for G UI testing?**

GUI is a software testing type that checks the graphical user interface of the software

* Check all the GUI element size position, width, length etc.
* Check for error messages are displayed correctly
* Check for cleardemaraction of different section on screen
* Check font used in application is readable
* Check the alignment of text is proper
* Check the colour of the font and warning messages is aesthetically pleasing
* Check all the images have good celerity
* check all the images are properly aligned
* Check the positioning of GUI elements for different screen resolution