What is Exploratory Testing?

Exploratory testing" – as the name suggests, is a simultaneous learning, test design, and test execution process. We can say that in this testing test planning, analysis, design and test execution, are all done together and instantly.

This testing is about exploring the system and encouraging real-time and practical thinking of a tester.

What is a traceability matrix?

A traceability matrix is a document that details the technical requirements for a given test scenario and its current state. It helps the testing team understand the level of testing that is done for a given product.

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.

What is Equivalence partitioning testing?

Equivalence partitioning (EP) is a method for testing software programs. In this technique, the data fed into the software to be tested is divided into partitions of equal sizes. From each partition of data,one test case is needed. - The different test cases must test the classes of the software continuously. This helps in the discovery of errors and bugs which may plague the software.

- Each test case is defined specifically to check a specific type of error. This speeds up the error hunting process as fewer test cases are required.

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.

What determines the level of Risk?

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

What is alpha testing?

Alpha testing is the first end-to-end testing of a product to ensure it meets the business requirements and functions correctly.

It’s the form of the Acceptance Testing.

It’s Usually performed by Developers at the software development site.sometimes it’s also performed by Testing Team.

What is Beta Testing??

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

Beta testing is the final round of testing before releasing a product to a wide audience.

What is component testing?

It is Testing of an individual Software Component.Sometimes known as Unit Testing, Module Testing or Program Testing.

What is 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.

What is 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 is the testing of “how” the system works. Non-functional testing may be performed at all test levels.

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 – toolbar, menu bar, dialog boxes and windows etc.

What is Adhoc Testing?

The Error Guessing is a technique where experienced and good Testers are encouraged to think of situations in which the software may not be able to cope.

Adhoc testing is also called Error Guessing.

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.

What is Stress Testing?

System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large numbers beyond storage capacity, complex database queries, continuous input to system or database load.
Stress testing is also known as endurance testing.

What is Black box testing and the list the types of white box testing?

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

Techniques of Black Box Testing:

Equivalence Testing

Boundary Value Analysis

Decision Tables

State Transition Testing

Use-case Testing

What is White Box Testing?and list the types of white box testing.

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

Statement coverage

Decision coverage

Condition coverage

Branch condition testing

Branch condition combination testing

Modified condition decision testing

Data Flow testing

Mention what are the categories of defects?

Data quality/database Defects

Critical Functionality defects

Functionality defects

Security Defects

User Interface Defects

Mention what Bigbang testing is?

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

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

What is the purpose of exit criteria?

Purpose of exit criteria is to define when we STOP testing either at the:
End of all testing – i.e. product Go Live
End of phase of testing (e.g. hand over from System Test to UAT) .

When should "Regression Testing" be performed?

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

What is the 7 key principal?Explain in detail.

1.Testing shows presence of Defects

2. Exhaustive Testing is Impossible!

3. Early Testing

4. Defect Clustering

5. The Pesticide Paradox

6. Testing is Context Dependent

7. Absence of Errors Fallacy

Testing shows presence of defects:

-Testing can show that defects are present,but can not prove that there are no defects.

-Testing **reduces the probability of undiscovered defects** remaining in the software but, even if no defects are found, it is not a proof of correctness.

Exhaustive Testing is impossible

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.

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 should not start once the code has been written!

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.

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.

Therefore we must learn, create and use new tests based on new techniques to catch new bugs .

Testing is Context Dependent

Testing is done differently in different contexts.

Different kinds of sites are tested differently.

Absence 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.

Even after defects have been solved it may be unusable and/or does not match /fulfill user requirements/needs/expectations.

Difference Between Smoke And Sanity.

SMOKE TESTING	SANITY TESTING
Smoke testing is performed to ascertain that critical functionalities of the program is working fine.	Sanity testing is done to check the new functionality/bugs have been fixed
The Objective of this testing is to verify the “stability”of the system in order to proceed with more rigorous testing	The objective of this testing is to verify the rationality”of the system in order to proceed with more rigorous testing
This testing is performed by the developers or testers	This testing is usually performed by testers.
Smoke testing is usually documented or scripted	Sanity testing is usually not documented or unscripted
It’s a subset of Acceptance testing	It’s a subset of Regression Testing.
Smoke testing exercises the entire system from end to end	Sanity testing exercises only the particular component of the system.
Smoke testing is like General Health check up	Sanity testing is like specialized Health Check up

Difference Between Quality Assurance Quality Control and Testing.

Quality Assurance	Quality Control	Testing
Activities which ensure the implementation of processes, procedures and standards in context to verification of developed software and intended requirements.	Activities which ensure the verification of developed software with respect to documented (or not in some cases) requirements.	Activities which ensure the identification of bugs/errors/defects in the Software.
Focuses on processes and procedures rather than conducting actual testing on the system.	Focuses on actual testing by executing Software with intent to identify bug/defects through implementation of procedures and process.	Focuses on actual testing.
Process oriented activities.	Product oriented activities.	Product oriented activities.
Preventive activities.	Product oriented activities.	Product oriented activities.
It is a subset of the Software Test Life Cycle (STLC).	QC can be considered as the subset of Quality Assurance.	Testing is the subset of Quality Control.

Difference Between Verification and validation.

VERIFICATION	VALIDATION
The process of evaluating work products(not the actual final product)of a development phase to determine whether they meet the specified requirements for the phase	The process of evaluating software during or at the end of development process to determine whether it satisfies specified business requirements
To ensure that the product is being built according to the requirements and design specifications	To ensure that the product actually meets the user's needs and that the specifications were correct in the first place
Are we Building the product Right?	Are we Building the Right Product?
Plans,Requirement specs,Design specs,code,test cases	The actual Product/software
- Reviews - walkthrough - inspections	Testing

Explain Types of Performance Testing.

Load Testing

Stress testing

Endurance Testing

Spike Testing

Volume Testing

Scalability Testing

Load Testing : Confirms that the system can handle the required number of users and still operate at a high level of performance. This ensures that there is no day to day issues in performance.

Volume Testing: Checks that the software can handle and process a large amount of data at once without breaking, slowing down, or losing any information.

Stress Testing: Intentionally tries to break the software by simulating a number of users that greatly exceeds expectations. The launch day of a new iPhone and the sudden spike in user traffic on the Apple website is a good example of a stress test in the real world.

What is Error ,Defect, Bug and failure.

A mistake in coding is called error, error found by tester is called defect, defect accepted by development team then it is called bug, build does not meet the requirements then it is failure

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

Defect: Commonly refers to several troubles with the software products, with its external behavior or with its internal Failures.

Bug: A fault in a program which causes the program to perform in an unintended or unanticipated manner

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

Explain Difference between Functional testing and non-functional testing.

Functional testing	Non-functional testing
Functional testing is performed using the functional specifications provided by the client and verifies the system against the functional requirements.	non -functional testing checks the performance,Reliability,and other non-functional aspects of the software system.
Functional testing is executed first.	Non-functional testing should be performed after functional testing.
Manual testing or Automation tools can be used for functional testing.	Using tools will be effective for this testing.
Business requirements are inputs to functional testing.	Performance parameters like speed,scalability are inputs to non functional testing.
Functional testing describes what the product does.	No-functional testing describes how good the product works.
Easy to do manual testing	Tough 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 testing	Types of non-functional testing are: ● Performance testing ● Load testing ● Volume testing ● Security testing ● Installation testing ● Compatibility testing ● Penetration testing ● Migration Testing

What is the Difference between SDLC(Software Development Life cycle) and STLC(Software testing life cycle).

SDLC	STLC
Origin is development Life cycle	Origin is testing Life Cycle
The main objective of SDLC is to complete successful development of the Software Including testing and other phases.	The main objective of the STLC phase is testing.
In SDLC the Business analyst gathers the Requirements and create development plan	In STLC,the QA team analyzes requirement documents like functional and non-functional documents and creates a system test plan.
In SDLC,the Development team creates the High and Low-level requirements.	In STLC,the test analyst creates the integration Test Plan.
The real code is developed,and actual work takes place as per design documents.	The testing team prepares the test environment and executes them.
SDLC phase also includes post-deployment support and updates.	Testres executes regression suits,usually automation scripts to check maintenance code deployed.

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.

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

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

- The information should be covered as follows:
- The organization’s test policy
- Scope of testing being performed
- Testing Objectives
- Project risks-e.g.Business,Technical,people.
- Constraints-e.g. Business imposed,financial,contractal etc
- Critically(e.g. system/component level)
- Testability
- Availability of Resources

When to use Usability Testing

There are many software applications / websites, which miserably fail, once launched, due to following reasons –

- Where do I click next?
- Which page needs to navigate?
- Which icon or jargon represents what?
- Error messages are not consistent or Displayed properly
- Session time is not sufficient.

  Usability testing Identifies Usability errors in the system early in the development cycle and saves a product from failure.

What is the procedure for GUI Testing.

- **Manual Based testing :** Under this approach, graphical screens are checked manually by testers in conformance with the requirements stated in the business requirements document.
- **Record and Play** : GUI testing can be done using automation tools. This is done in 2 parts. During Record , test steps are captured into the automation tool. During playback, the recorded test steps are executed on the Application under Test.
- **Model based Testing** : A model is a graphical description of a system's behavior. It helps us to understand and predict the system behavior. Models help in a generation of efficient test cases using the system requirements.