

Q. Manual Testing:-

- Manual testing means repeatedly testing the software by manually executing test cases.
- The goal is to identify defects and ensure that the software works according to customer requirements.
- This process helps verify the functionality, usability, and overall quality of the application.

Q. Automation Testing:-

Here the test engineer will write the script using tool like selenium or playwright and run the tool against the software. The tool will test the software and give the result as pass or fail.

Q. Black Box Testing

- Testing the functionality of an application against the requirement specification is called as Black Box Testing.
- Black Box Testing is also called as functional testing or behavioral testing.

Q. White Box :- Checking of internal code lines. Done by Developer or white box testers Good at Design & Programming Language. Done after Coding.

Q. Smoke Testing:-

Sanity Testing/Dry run Testing/Health Check-up of the product Testing.

Testing the basic or the critical features of an application is called as Smoke Testing.

Note:- Smoke Testing is also called as positive testing because in Smoke Testing, we test the software with only valid input.

Advantage of Smoke Testing:-

- i. The TE can find all the blocker defect in the initial stage of testing.
- ii. There is no delay in release of the product.
- iii. Developer will get sufficient time to fix the defect.

Q. What is the difference between Smoke Testing & Sanity Testing?

According to me there is no such difference between Smoke Testing & Sanity Testing but I have got to know that there is a difference by referring some website and also some of my friends.

Smoke Testing

1. It is shallow and wide testing
 - i. Here we cover all the features of the software and test it in high level.
2. In smoke testing we only do positive testing.
3. Here we can go for automation.
4. It is done by test engineers and developers.
5. Here we have test cases and test scenarios.

Sanity Testing:-

1. it is deep and narrow testing.
 - i. Here we take one feature and tested in depth.
2. Here we do both positive and negative testing.
3. Here we can't go for automation.
4. It is done by only test engineer.
5. Here we do not have test cases and test scenarios.

Q. Regression Testing:- (Release candidate testing)

Testing the software or an application by testing the unchanged feature to make sure that is not affected or broken because of the change.

Change means adding, modifying, removing a feature or fixing a bug.

- 1. Unit Regression Testing**
- 2. Regional Regression Testing**
- 3. Full Regression Testing**

1. Unit Regression Testing:-

Testing only the changes or the defect fix, is called as Unit Regression Testing (URT).

2. Regional Regression Testing:-

Testing the changes and impacted region is called as Regional Regression Testing (RRT).

3. Full Regression Testing:-

Testing the changes as well as the remain feature is called as Full Regression Testing.

Q. What is the difference between Regression Testing & Retesting?

Regression Testing

- i. Testing the unchanged feature to make sure that it is affected by the changes.
- ii. It is generic
- iii. Regression testing is done on pass test cases.
- iv. Here we go for both automation and manual testing.

Retesting

- i. Testing the defect which is fix to make sure that is it properly fixed or not.
- ii. It is planned.
- iii. Retesting is done on failed test cases.
- iv. Here we can go for only manual testing.

Adhoc Testing:-

Gorilla Testing, Monkey Testing, Negative Testing, out of box Testing.

Testing the software or an application randomly without referring any formal documents such as test cases and test scenarios is called as Adhoc Testing.

Adhoc Scenarios:-

1. In phone pay try to send amount to a phone number which is not register on phone pay.
2. Try to book a cab in ola by keeping the to address and drop address is same.
3. Try to send amount to own number in google pay
4. Try to withdraw amount from ATM without enter ATM card.
5. Try to entry metro gate without swipe metro card.
6. Try to call your own number with your own same number only.

Types of Adhoc Testing:-

1. **Buddy Testing:-** Test Engineer will seat with developers and come up with adhoc scenarios.
2. **Pair Testing:-** Here the TE will seat with another TE and come up with adhoc scenarios.
3. **Monkey Testing:-** Here the TE will test the software without applying any logic.

Q. Usability Testing-:

GUI/Cosmetic/Yellow Box Testing-:

Testing the user friendliness of an application like

- Accessible
- Navigation
- Look and Feel
- Easy language
- Easy to understand
- **Simple -:** This will be done by end users and feedbacks will be implemented and by deriving checklists like
 - Login feature should have forgot password
 - Alt tags for every link and diagrams
 - Link to home page at every page
 - All pages should have Navigation

Q. Performance Testing-:

Testing the response time and stability of an application by applying load is called as performance testing.

Types of Performance Testing-:

1. Load Testing
2. Stress Testing
3. Volume Testing
4. Soak Testing

1. Load Testing-:

Testing the response time and stability of an application by applying load which is lesser than or equal to the designed number of users is called as Load Testing.

2. Stress Testing-:

Testing the response time and stability of an application by applying load which is more than the designed number of users is called as Stress Testing.

3. Volume Testing-:

Testing the response time and stability of an application by transferring huge volume of data is called as Volume Testing.

4. Soak Testing-:

Testing the response time and stability of an application by applying load continuously for particular period of time is called as Soak Testing.

1. Functional Testing-:

Testing each & every component of a software is called as functional testing.

Types of Functional Testing-:

There are 3 types of Functional Testing

1. Donkey Testing
2. Under testing
3. Optimize Testing

1. Donkey Testing-:

Testing the software or feature with same set up scenario or the scenario which does not make any sense is called as Donkey Testing.

2. Under Testing-:

Testing the software or feature with insufficient set of scenarios is called as Under Testing.

3. Optimize Testing-:

Testing the software or feature with sufficient set of scenarios and scenario which makes appropriate sense is called as optimize testing.

2. Integration Testing-:

Testing the dataflow between modules is called as integration testing.

Types of Integration Testing-:

1. Incremental Integration Testing
2. Non-incremental Integration Testing

1. Incremental Integration Testing-:

Incrementally adding the modules and testing the dataflow between them, is known as Incremental Integration Testing.

→ It is 2 type-:

a. Top-Down Incremental Integration Testing

b. Bottom-Up Incremental Integration Testing

a. Top-Down Incremental Integration Testing-:

Incrementally adding the modules and testing the dataflow between them, make sure that the module which is added must be the child of previous module.

b. Bottom-Up Incremental Integration Testing-:

Incrementally adding the modules and testing the dataflow between them, make sure that the module which is added must be the parents of previous module.

2. Non-incremental Integration Testing-:

Here we take the whole software we gather all the module and taste it in one shot this is called as Non-incremental Integration Testing.

System Testing-:

It is an end-to-end testing where in the test environment is similar to production environment.

End To End Testing-: Navigating through all features and testing whether the end feature is working properly or not.

When we do System Testing?

When minimum set up features are ready. When basic feature is ready.

Development Server-: It is a set up containing hardware, software and networks which is used to develop a software.

Testing Server-: It is a set up containing hardware, software and networks which is used to testing a software.

Pre-production Server-: It is a set-up which is similar to production server wherein we partially run the business and also test the software.

3. Accessibility Testing/508 testing/American Disability act Testing (ADA)-:

Testing the user friendliness of an application from physically challenge person point of view is called as accessibility testing.

4. Reliability Testing -:

Testing the capability of a software or the functionality of a software continuously for a longer period of time and testing whether the software is capable of handling it or not is called as reliability testing.

5. Recovery Testing-:

Testing the software how first can its recovery from crass.

6. Exploratory Testing-:

Exploring the software understanding the software based upon understanding identified scenarios prioritise the scenario, document the prioritise scenario, execute the scenario on the software and test whether It Is working properly or not, is called as exploratory testing.

7. Comparison Testing or Parallel Testing:-

Considering a software which is already existing in the market and comparing it to the newly build software wherein we make sure that all the features present in the existing software should also be present in the newly build software. This is called as comparison testing.

Example:- Ola Uber, Swiggy Zomato, Flipkart Amazon.

Yellow Box Testing:-

Testing the warning message in an application is called as yellow box testing. It is a subset of Usability Testing.

8. Acceptance Testing (Red Box Testing):-

It is an end-to-end testing done by IT engineer seating at customer place where in they consider real time end to end business scenario and check whether the software is capable of handling it or not.

Approach No-02 (UAT- User Acceptance Testing):-

It is an end-to-end testing done by end user wherein they use the software for business for particular period of time and check whether the software is capable of handling all kinds of business scenario.

Approach No-03:-

It is an end-to-end testing done by our own engineer seating at the customer place wherein they refer the user scenario given by customer and test whether software is capable of handling it or not.

Approach No-04:-

It is an end-to-end testing done by our own engineer seating in our own place wherein they refer the user scenario given by the customer and check whether the software is capable of handling it or not.

Alpha testing:-

The testing done by the test engineer before we give the product for acceptance testing is called as Alpha Testing. (No Defect)

Beta Testing-:

It is the testing done by end user based on their feedback the product is release to the market.

Q. Globalization Testing-:

Developing the software for multiple language is called as globalization testing.

Testing the software which is developed for multiple language is called as globalization testing.

There are two types of globalization testing,

1. Internationalization (I 18 N)
2. Localization (L 10 N)

Test Case Design Technique-:

It is a technique which is use to write test cases in an order to improve the test case coverage.

Test case design technique types-:

1. Error Guessing Method
2. Equivalence Class Partition
3. Boundary Value Analysis

1. Error Guessing Method -:

Here we guess all possible errors and test the software with those errors.

2. Equivalence Class Partition-:

- a) Pressman Rule
- b) Practice Method

Pressman Rule No-01 -:

If the input is in range of values, then design the test cases for one valid and two invalid inputs.

Pressman Rule No-02 -:

When the input is in set up values, design the test case for all valid inputs and two invalid inputs.

Pressman Rule No-03 -:

When the input is Boolean conditions the design the test case for both

true and false condition.

3. Boundary Value Analysis-:

If the input is in range of values A to B then design the test case for A, A+1, A-1 & B, B+1, B-1.

Q. Severity-:

Severity for a defect is decided based upon the impact of defect on customer business workflow.

There are four types of severity-

1. Blocker Defect/Show Stopper/Fatal
2. Critical Defect
3. Major Defect
4. Minor Defect

1. Blocker Defect-:

Assume any defect which is not let in the test engineer to test the features and also, we are 100% sure that the defect is affecting customer business workflow. This kind of defect is called as Blocker Defect.

2. Critical Defect-:

Assume there is defect in the software and we are 100% sure that the defect is affecting customer business but is not restricting to test engineer to test the features.

3. Major Defect-:

Assume there is a defect in the software and we are not sure how the defect is affecting customer business workflow this kind of defect we called as Major Defect.

4. Minor Defect-:

We are 100% sure that the defect is not affecting customer business these kinds of defects we call as Minor Defect.

Minor defect consists of

- a) Spelling Mistake
- b) Alignment Issue
- c) Object Overlapping
- d) Colour Issue

For Example-: Severity for an ATM machine.

- I. A user inserts ATM card and he can withdraw money without entering the pin. **(Critical Defect)**
- II. If a user enters valid pin code to a valid card and still, he is not able to withdraw money. **(Blocker Defect)**
- III. If a user withdraw money and the ATM machine is not displaying confirmation message. **(Major Defect)**
- IV. If any user finds a spelling mistake in the ATM machine. **(Minor Defect)**

Q. Priority:-

It is the importance given to the defect as how soon the defect should be fixed by the developer.

There are three types of priority

1. High Priority **(P1)**
2. Medium Priority **(P2)**
3. Low Priority **(P3)**

1. High Priority (P1):-

If a defect gets high priority, then the developer should fix the defect immediately.

2. Medium Priority (P2):-

If a defect gets medium priority, then the developer should fix the defect in the next test cycle or upcoming test cycles but in the same release.

3. Low Priority (P3):-

If a defect gets low priority, the developer team should fix the defect in the upcoming release or within 2—3 release.

Example:-

1. If we find a defect in Facebook where we cannot login even with valid credentials. **(Blocker, High Priority (P1))**

2. In WhatsApp mute function is not working. **(Major, Medium Priority(P2))**
3. In YouTube user is not able to view video. **(Blocker, High Priority (P1))**
4. In Flipkart user can order a product but not able to save the credit card details. **(Major, Medium Priority)**
5. In WhatsApp if we sent a message to user A the message is being sent to user C. **(Critical, P1)**
6. In Facebook remember password is not working.

(Major, P2)

V & V Model (Verification & Validation)-:

Verification-:

- I. Verifying or testing CRS, SRS, HLD, LLD and checking whether it is according to the customer requirement specification or not.
- II. It is done by test engineers and it is done before the construction of the software.
- III. Here we check whether we are building product right or software right.
- IV. It is also called as static testing.

Validation-:

- I. Verifying the functionality of an application or software by entering inputs or executing test cases is called as validation.
- II. It is done by test engineer and it is done after the construction of the software (After Coding).
- III. It is also called as dynamic testing.
- IV. Here we verify whether we have built right product or right software.

What is Agile

Agile is model which we use to develop the software in incremental & iterative process.

Here we develop a large software in short cycles called as sprint

Q. Story Point:-

Story point is the estimation of time required to develop and test a story.

Q. Spill Over:-

There are certain feature/stories which we are not able to release in the current sprint, so we release that in the next sprint. This feature/story are called as Spill Over.

Q. What is Retrospective Meeting -:

- i. Here the entire team meets and sometimes customer is also involved.
- ii. Here the team will discuss about all the good activities which they had perform in the project and the mistake that they had committed during the project.
- iii. We document all the good and bad activity that document is called as retrospective document.
- iv. While we start the new project or the next sprint, we refer the retrospective document and we make sure that the good activities are repeat ate and the mistakes are avoided.

Q. What is Sprint Review Meeting:-

- i. Here the whole scrum team meets after the completion of the sprint.
- ii. The engineer will give a demo of what they have built.
- iii. The product owner will tell what is done and what is not done.
- iv. And also plan for the next sprint planning meeting.

Q. What is impact analysis meeting?

TE will conduct a meeting where in the whole testing team, test lead, customer sometime is also included and they discuss about the impact area in software and documented it. This is called as impact analysis meeting.

Defect Report:

Defect ID : Unique ID to identify the defect
Project name : Name of the project
Module name : Name of the project
Release/version : Release name
Status : Status of bug open
Seviority : Critical
Priority : p1/p2/p3
Environment : Testing environment
Test case ID : Test case-executing which the bug is found
Brief description : bug
Detailed description : Steps to reproduce the bug with the details of I / p 's used while testing.
Pre condition :
Test date :
Found by : Tester who found the bug.

Test Case Review :

- Checking whether test case template is w.r.t standards.
- Checking whether test cases covers all the testing techniques.
- Checking the header of test case like naming convention / precondition / seviority.
- The mistakes will be reported using Review Template.
- Spelling and grammar mistake
- Ensure team member captured test data
- Once peer review(others review) done ensure it is incorporated review comments