**What is exploratory testing?**

Exploratory testing first name is ad-hoc testing. This is the type of software testing in which it is free to select any possible methodology to test the software.

Currently many users should use this Exploratory testing…..because…

* **Random and unstructured testing**:- In this testing test all modules and functionality are unstructured. Thus can help to find bug that was would of undiscovered during structured phases testing.
* **Tester can play around with user stories:-** tester can annotate defects, add ascertain, and user story is converted to a test case.
* **Facilitate agile workflow:**- Everyone can participate in exploratory testing with the help of visual feedback that’s why team can easily adapt the changes quickly and facilitating agile flow.
* **Do minimal planning test maximum:-** In this testing tester can do test with few planning about the software, and do maximum testing of the software.

This testing mostly do for experience person. This testing apply when requirements are missing and tester understand the requirements and make documented flow of the application.

In this testing application understand by tester that’s why easy to find out defects, bugs, and error. For example \_\_ your application is ready but you have no idea about the application at that time tester take that application, explore that software, and identify what are requirements of this application. Finding the all possible scenario and write down into the document.

**Drawbacks:-**

When you do exploratory testing then sometimes extra features stay missing because generally when test any application at that time not requirement of that feature is needed that’s why tester not put that and test simple according to basic requirement.

**More time consuming: -** When tester explore any application then **so** many time consuming because in this case tester have not any requirements that’s tester must be understand the requirements, and getting knowledge about this product having knowledge of another person.

**What is Traceability of Matrix?**

Traceability matrix also known as Cross Reference Matrix.

Requirements traceability of matrix maps all the requirements with the test cases. By using this document cover all functionality of the application as per the requirements of the customer.

**Traceability: ability to trace the tests.**

**Matrix: The data which can be stored in rows and columns forms.**

* The main purpose of the traceability matrix is to verify that all requirements of the clients are covered in the test cases designed by the tester.
* When any changes do into the software then must be change in traceability matrix.
* Requirements can be traced to determine a particular group or person that wanted requirement, and it can be used to it is prioritize the requirements.
* It is help to keep a check between requirements and other development artifacts like technical and other requirements.

Some parameters are following :-

* Requirement ID
* Requirement Description
* Requirement type
* Test cases Id

**Three types of Traceability Matrix**

1. Forward Traceability Matrix:
2. Backward Traceability Matrix
3. Bi-directional Traceability Matrix

* **Forward Traceability Matrix**

Forward Traceability Matrix we mapped the requirements with the test cases. Here we can verify the requirements are covered in the test cases and no functionality is missing in the test cases.

It is use for the work in right direction.

|  |
| --- |
| Test Cases |

Mapping

|  |
| --- |
| Requirements |

**What is Boundary value Analysis (BVA)?**

Boundary value is a technique of Black box testing and it’s also method of equivalence partitioning.

It is a methodology of design test cases that makes project better efforts near the suitable range.

Boundary value analysis check the all fields that user inputs any data in any application are valid or not.

This methodology using for entering valid input or output. If any client entering wrong data then generate proper validation, if not generate then it is a bug. This type of scenario check in Boundary Value Analysis.

Boundary value check validation of input data and design test cases and it is very useful method to check bugs.

**For E.G**

If I take variable 6 to 35.

Value = 5 Value = 6 Value=15 Value=31 Value=35 Value=50

(Invalid) (Valid) (Valid) (Valid) (Valid) (Invalid)

**What is Equivalence Partitioning?**

* Equivalence partitioning is technique of black box. It is used for all inputs are dividing in proper partition.
* EP can test all levels of testing. Developers developed any application. In that application lots of inputs are available then this technique using for testing are dividing.
* Top modules check first and other modules are check second priorities.
* Developers put specific range of each and every input and all functionality. Tester checks the all functionality and inputs with smart work.
* Tester makes the group of testing check any one specific input and checks all inputs.
* When selecting any one input data and the specific input represent all data that invoke the same process of that specific condition.

**For E.G**

If I put condition value >=9 and value <=50 then

1 5 20 35 25 40 45 50 55 60 70 100

Invalid Invalid Range

Range Valid Range

* This all numbers make good partition then tester can do partitions and find bugs in one group then it is fixed that others have also defects.

**What is Integration Testing?**

Integration testing is a level of the software testing process where individual units are combined and tested in a group.

This testing aim is all units do combined and test as a group and test interaction between all units.

When we test interaction between all units then some time fixed bugs. In this situation once again check all components and solve that.

In integration testing do interaction between operating system, hardware, file system, interfaces between systems.

Integration testing performed by integration tester or tester team.

**Integration testing is required because when all components are together and make lots of units in that situation we need to check that all units are work together properly or not.**

**Two types of Integration Testing**

* Component Integration Testing
* System Integration Testing

Interfaces of the software modules with database could be errors.

Hardware interfaces, if any then could be some time issues.

**There are two types of Integration Testing methods**

* Bing Bang Integration Testing
* Incremental Integration Testing

Bottom Up Approach

**Top down Approach**

* Integration testing is performed after unit testing and before system testing

**What is determines levels of risk?**

* Reduces the all level of risk and design the proper project or product.
* The quality of the software tests, and improve the performance of the system and reduce the risk.
* When any project or product test, then evaluate which kind of risks are generated after finalize the project and product.
* Without evaluate the risks of the product or project, we can’t make product or project better as per client requirement.
* Risk is identifying the factor to make system slow or bad performance reasons.
* After identifying risk tester and developers can make system better as per client requirements.
* Risk could be finding future any negative consequence
* There are two types of Risks.
* Project Risk
* Product Risk
* **Project Risk**
* Project risk is when any developers leave project between continue developing period at that time

Project stay in risk because that developer know everything about specific modules or task.

After leaving the project it is difficult to handle this project to other developers.

* **Product Risk**
* Product Risk is when any employee less experience and pause a higher risk to the existing area.
* When product is lunch then after you will know that this product is not more proper because know one can purchase or use. So that’s why lost of money wasted and also time consuming too.
* Company invest the time and money on specific product. All employee test risk about this product, and then lunch product.
* We need also mitigation on product risk and make better.

**What is Alpha Testing?**

* Alpha testing is always testing privately. It will happen when developing is over of particular module task. Then only check that particular portion or task to work properly or not.
* Alpha testing is performed in virtual testing.
* Alpha testing always performed within the organization.
* When any functionality wants to do update then developers works on that task. After completion that task we performed alpha testing.
* It is not for public testing.
* Alpha testing is considering into two both techniques like White box and Black Box.
* It is performed always by independent team.

**What is Beta Testing?**

* Beta testing is performed by users.
* Beta testing always open for publicly
* It is test in real time environment.
* It is directly tested by users and users give the reviews directly to the employee.
* It is only performed in black box.
* Beta testing is performed on customers own sites.
* This testing is form of acceptance testing.
* This testing giving to users to fetching their locations and other data and sending new update of the specific application.
* If users like that update then review will get positive otherwise negative.
* If review is negative then organization need to more improvement in that product.

**What is Component Testing?**

* Component testing is test individual part of the system and fixing error.
* In this testing, each and every components need to independent state and also should be in controllable state.
* Before performed integration testing component testing performed.
* In this testing check the all usability of components.
* Component testing is also module testing, and this will test individual parts of the components.
* Component testing through easy to finding bugs and errors and units are developed as per client requirements.
* Tester tests all components by component testing make lots of units.
* Component unit take more time for testing because a component made up of multiple code.
* Component testing is test small-small part of the system.

**What is Functional System Testing?**

* Functional testing is performed for when any change the functionality of any module then developers developed that functionality, tester test that functionality as per user requirement, if it is work properly then that called functionality testing.
* Functional testing is only test when anyone wants to update any functionality. This will test only that particular function.
* Functional system testing check that particular function according to requirement of that function.
* Generally functional system testing function as below:
* **Accuracy:** Check this type because that particular function work properly or not according to expected.
* **Interoperability:** Check ability to when developed functionality interact with system properly or not.
* **Compliance:** Check functions regulations, or lows.
* **Auditability:** Check accurate data, when we entered data according to input and get result as per requirement.
* **Suitability :** When developed any module or task then check that presence and appropriate functionality work or not.

**What is Non-Functional System Testing?**

* Non-functional testing performs to verifies the non-functional requirements of the application. It is verify to whether the behavior of the system as per requirement or not.
* It will improve system performance and check the timing to very little consumption. This will perform after execution of the software.
* Non-functional testing verifies the environment of the system because when software installed properly or not. If take more load at that time this testing is used.
* Using this testing reduce the time or load consumption. It is also very secure. It makes product more effective and make real time system.
* Non-functionality attributes not concern with functionality, reliability, usability, maintainability, and portability.
* Non-functional testing works on how to work system. It will check the all features and characteristics of the system. It will increase the system performance.

**What is GUI Testing?**

* GUI stands for Graphical User Interface. In this testing test all interface of the system like screen, icon, buttons, links, menu.
* In GUI testing the process of testing that test system GUI.
* GUI testing is much important because of when any client or user start the system or any application in available in system.
* In GUI testing check following types of.
  + Check images open properly or not
  + Check font of the system or application that make better appearance of the system.
  + Check alignment of the system or application.
  + Check position of the elements, and look better or not.
  + Check validation and error message display properly or not.
  + Check Grammar or Spelling.

**Way of GUI testing**

* Manual Based Testing
* Record and Replay testing
* Model based testing
* Manual Based Testing

In manual based testing tester test the Graphical User Interface of the software and system, and as per requirement tester apply knowledge.

* Record and Replay Testing

In this testing applying automation tool. During test apply record tool and during playback time execute the recorded and execute the application.

* Model Based Testing

This testing performs system based behavior. In this type of testing test the model to check all functionality perform properly or not.

In this testing test the actual result with expected result. Some time apply techniques based on Charts, Decision Table.

**Following are the example of GUI.**

**Mobile Based Testing**:-

* Mobile based technique is when any user or client use phones then showing images, display are proper or not.
* Each and every application work properly or not.
* What are the extra features in mobile?
* Camera work properly or not.

**WEB Based Testing**

* Web based testing is when any developed website at that time check that site icon, toolbar, header, font etc.
* If any text area available then check their functionality to expand or not if multi-line available.

**What is Adhoc Testing?**

* Adhoc testing perform when tester have no any planning or requirements.
* In adhoc testing not follow any techniques or structure, and check randomly done any part of the system or software.
* In this testing main aim is finding defects by random checking.
* Randomly check the software or application and guessing the errors.
* Error guessing is mostly done by experienced person.
* In this testing knowledge is much required under the system.
* In testing not follow any design or structure.
* Adhoc testing is also known as ERROR GUESSING.
* Error guessing is the technique of where any experienced tester test software and think about which kind of errors will be coming that should not be covered.

**Adhoc testing types are following.**

* Buddy testing
* Pair Testing
* Monkey Testing

**Buddy Testing**:

In buddy testing two person are their and first person comes from developers team and second person comes from tester team. This testing help to tester to make better test case and developer team can also make changes in design.

**Pair Testing**

In pair testing Two testers are work together. Assigned same task to both person. Both person are work mutually on same machine.

One person make test case and second person make notes on findings defects.

**Monkey Testing**

In this testing test randomly of the product or application without make any test cases for break the system.

**What is load testing?**

In this testing check the behavior of the system. When lots of users are logged in application at that time how system work that will check in load testing.

Load testing is check the performance of the system. When many users use the same application and work together at that time system behavior is proper or not.

For example: Lots of users use facebook. At same time 600 users login into facebook then facebook work properly, after few minits 700 users login into facebook at that time server go in slow. After few minit 1000 users login at that time application go down and crash.

According to below example we understand at specific level of any application take load properly and application work smoothly, but at the end of the limit system will be crashed.

So, this testing we will apply in load testing. Using load testing we check the system performance.

Why we should apply LOAD testing because when load testing is done of the any particular website then company have confidence to continue with that software or system.

If system performance is slow then employees are know that what they do for better performance.

**AIM of the LOAD testing**

* Make response time better, when any user input any data then give proper response in time.
* If any server configuration issue then solved with this testing.
* System Interface issue.
* Network issue when transaction performed.

When we test the load testing then first one check the basic things like…

Operating System

Memory

Processor

Storage

Configuration

**Load testing advantages are….**

* Decrease risk related to project
* Reduce cost of the failure.
* Customer support and satisfaction fulfillment.
* System makes in performance better.

**What is Stress testing?**

Stress testing perform under system testing. It will check how much take load to system can handle. This testing happens when high load putting under system performance.

Stress testing check the system storage or handle threads of the application or software.

It is evaluate if give such a load on to the system and at that situation system will perform which types of. If system don’t take much load and that’s why system will be crashed. So this is stress testing.

Stress testing emphasizes error handling, availability, and robustness under a heavy load rather than what is correct behavior under normal situation.

Following are the type of stress testing…

* Application Stress testing
* Transactional Stress testing
* Systematic Stress Testing
* Exploratory Stress Testing

**Application Stress Testin :-**

Application stress testing is also known as product stress testing is also focus on identifying the performance bottleneck, and network issues in a software product.

**Transactional Stress Testing:-**

Transactional stress testing performed more and more transaction between two or more application. It is carried out for fine tuning and optimizing system.

**Systematic Stress Testing:-**

Systematic stress testing integrated testing that is used to perform tests across multiple system running on same server. It is used to discover defects where one application data blocks another application.

**Exploratory Stress Testing:-**

Exploratory testing is testing the system abnormal parameters or conditions that are unlikely to happen in real scenario.

It is carried out that find defects in unusual scenario like large number of users logged in same time or database going offline when it is accessed from a website.

**Stress testing tools are following**

* Stress
* Tester
* Neo Load
* App Perfect

**Matrix for Stress Testing**

* Measuring Scalability & Performance
* Application Response
* Failures

Measuring Scalability & Performance:-

* Page par Second:

Measures how many pages have requested per second and number of pages loaded per second.

* Throughput:

Measures data size per second

* Rounds

Number of scenario are tested has been tested number of times users executed.

Application Response:-

* Hit Time:

When I request for any page or images then operation performed in accurate time.

* Time to the first byte:

Check how many time taken to return any first byte data or response, and information.

* Page Time:

Represent the time taken between user request of any page and retrieve that page in proper in time.

Failures:-

* Failed Connection:-

Many times failed connection refused by the users.

* Failed Rounds:-

Lots of time rounds get it failed

* Failed Hits

Number of failed generated by the system.

**What is white Box testing?**

White box testing is depends on analysis of any software or application’s internal structure of the component.

In white box testing testing internal structure and outer structure too. Tester known all about software and applications.

Tester can access source code and use the knowledge of design the test cases that can verify the correctness of the software at the code level.

White box testing is also known as structural testing or code based testing, and it is used for test internal logic.

White box testing is investigate internal logic and structural code.

Tester needs to look inside to source code and find out in which part not work properly.

Application Code



**Test case Output**

**Test case Input**

**T**

**Following are the Technique of Black Box:-**

**Test/Code Coverage**

**Decision Coverage**

**Condition Coverage**

**Test/Code Coverage”-**

* Test Code Coverage is also known as Statement or Line coverage.
* Test code coverage cover the maximum line of the code and finally get the output how many percent are cover the source code of the line.
* Statement coverage cover the source code based on code that how many statements are provide into the code.
* This coverage cover the true condition and also according to related statements are executed.
* Following formula is use for evaluate the statement coverage
* Statement Coverage = (Number of Executed Formula/ Total number of statement in source code)\*100
* This formula check the whether the source code expected to perform is valid or not.
* Test internal coding and infrastructure.

Drawback Of Test/code Coverage:-

* Cannot Check the false condition
* Different input values of all conditions.
* More than one test case required to cover all path with a coverage 100%.

**Decision/Branch Coverage**

* Branch Coverage cover the true and false condition.
* Decision coverage check both condition to if first condition is false then this coverage go in second condition to check true or false, in this case check all condition and getting the output.
* In this coverage evaluate the all conditions at least once.
* Decision coverage can be calculated by following
* Decision coverage = (Number of decision outcomes exercised/ Total number of decision outcomes)\*100%
* A Decision may be in If statement, a loop Control statement or switch case statement.
* Where the outcomes are multiples.

**Advantages**:-

* To check all conditions are validate or not.
* Check all condition are work abnormally

Disadvantages

* When technique is ignore branches within a Boolean expression which occurs due to shot-circuit operation.

**Condition Coverage**

* This technique is of white box technique.
* This coverage is also known as predicate coverage.
* In this coverage check all condition if that condition is true or false.
* If condition is true then also check that condition and that condition is false then also check that condition.
* In this condition check condition independently.

Other Techniques of white box testing..

* Data flow testing
* Branch condition Testing
* Modified Condition Combination Testing
* Liner Code sequence and Jump testing.

**What is Black Box testing?**

Black box testing is test either functional or non-functional, within reference of internal structure of the application or system or component.

In this testing tester have no knowledge about internal structure, designing, or logic.

In this testing tester shows only input and output. Internal process can not show by the tester.

For Example:

|  |
| --- |
| Any employee give input --🡪 Username - Anjali  - 🡪 Password – anjali123  Message generate -- Successfully login  Input-Anjali/anjali123  pass  Output |

This all process done, after that tester shows input like Anjali and anjali123, but tester can’t show what is the internal process of getting successfully message for login..

Tester doesn’t know about which type of logic is work behind this process. So, that’s called black box testing.

Black box techniques are following:-

* Equivalence Partitioning
* Boundary Value Analysis
* Decision tables
* State Transaction Testing
* Use case Testing
* Other Black box testing

**Equivalence Partitioning**

This testing is make a group of module testing and divide into same equivalence partitioning, and then take one by one and test that modules.

This all modules do partition as per priority, if any module is very important then that module test first, and second, and third.

In this technique we giving value 10 to 50 then using this technique selecting data of the application according to this value.

**For Example**

Equivalence Partitioning

**1 2 3 10 20 30 40 45 50 60**

10>= >=50

This testing is using all level of testing.

**Boundary Value Analysis**

Boundary value analysis testing is based on valid or invalid partition. The behavior at the edge of the equivalence partition is more likely to be incorrect than the behavior within the partition, So boundary is area where testing is likely find defect.

* A boundary value for a valid partition is a valid boundary value.
* For each variable we check
  + Minimum Value
  + Just above the minimum
  + Nominal Value
  + Just below Max value
  + Max value

**For Example**

* Valid Test cases:- for the above can be any value entered than 17 and less than 57.
* Enter value – 18
* Enter value -19
* Enter value – 38
* Enter Value – 54
* Enter Value – 56

Invalid Test cases:-When any value less than 18 and greater than 56 is entered.

* Enter value -17
* Enter value – 57

**Decision Table:-**

Decision table is based on take decision on particular condition or inputs and corresponding outputs.

Using decision table we test the all condition or inputs like

* Suppose we have a login page and take two inputs.
* First one is USERNAME: \_\_\_\_\_and
* Second one is Password:\_\_\_
* Take first input username if I entered wrong username than that is make 1 decision, second one I entered nothing than generate validation that is second decision, and last one I entered right username. So we make three different inputs for test one input.
* If I entered right password than that is first condition, after that I entered second value wrong password, and than I entered nothing than that make third condition. So this technique is used for taking decision for how many possible way to test and execute the code according to expected result.

**State Transaction Testing**

* State transaction techniques is a type of software testing which is performed to check the change in the state of the application under varying input.
* When any tester test the different types of functionality of software than tester input different-different input and show which functionality generate which type of output and check the application, so this technique OBSERVED the behavior of the system or application passed in sequence.
* Basically This testing use following terms:-
  + State Diagram:-S
    - This diagram shows the application or software events or circumstances that cause and shows the result will be change one state to another.

State Table:-

This table shows the all transactions for each state combined with each possible event, and also showing that all transaction is valid or not.

State Transition:-

This state showing the transition between two state or components or system.

State Transition Testing:-

This testing use for test execute transition to valid or not valid, and design this type of technique.

Event

Action

State 1

Output

Input

State 2

**Mention what are the categories of defect?**

* Defect is some kind of error, flow or some kind of mistake from development team which prevent the software from smooth working.
* This type of defect directly prevent the software quality.

Following are the defect categories:-

* **Data Quality/ Database Defects:-**
  + When data having improper and improper handling data.
* **Critical functionality Defects:-**
  + When any functionality is major and in that functionality create defects. For example- exceptions.
* **Functionality Defects:-**
  + This defects effect on functionality.
  + Like Insert, Delete, Cancel these type of operations are not performed this type of functions.
* **Security Defects:-**
  + Application security defects affect when data is improper handling of data improper sent from the user to application.
* **User Interface Defect:-**
  + When user go an any option at that time getting the suggestions. This called User interface defect. User could not reliable proper which type of icon or image or tool working functionality.

**Mention What Big bang testing is?**

* Big band testing is method invoke in Integration testing technique.
* Big bang testing is a testing approach where all individual components or modules of a software application are tested together.
* This is done after all modules or component development finished.
* When System is to complex than big bang methodology is only one option available which tested all modules and components are tested together.
* Advantages

It is simplest as integration testing as well all modules are integrated at once and tested as whole.

All the errors and bugs identify at once as all the modules are tested together.

Types of big band testing:-

* Top-Down Integration:-
  + In this integration test all modules and component of software or application is top to down.
  + In this integration integrated higher level modules first and lower modules level second.
  + In this integration lower level modules are not yet ready for dependencies between modules or components.
  + Check all functionality, icons, or menus are in sequence of top to down.

Top-Down

Module 1

Module 3

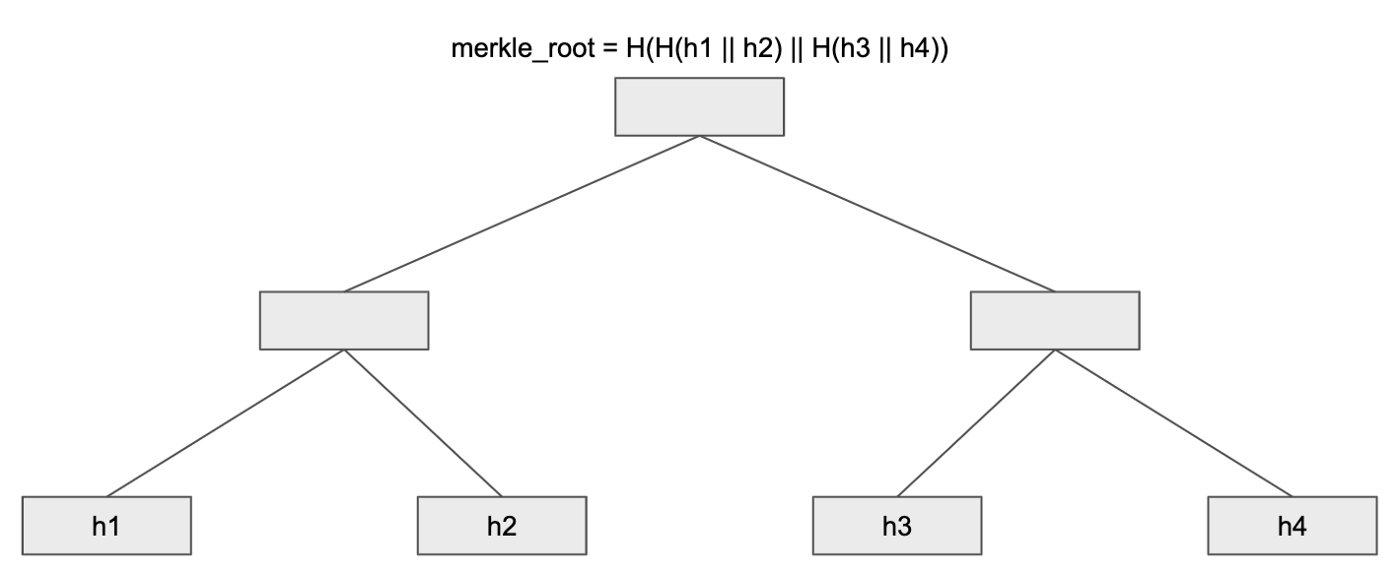
Module 2

Module 4

Module 5

Module 6

* Bottom-Up Integration:-
  + In this Integration integrated first lower level modules or component together and second level put high level of modules and components.
  + In this integration higher level modules are not yet ready for dependencies between components and modules.



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

Exit criteria of the design phase include the development, review, and approval of the test cases. Each phases across a thorough review process to identify and address to any defects before approval.

The test design is finalize then after we finalize the test then design documentation containing test cases. In this criteria all high prioritize bugs fixed and closed.

**Some factor is matter for when we apply exit criteria**

* When time is out then tester must be stop testing and exit of the project.
* When budget is over then also tester must be leave the testing
* When tester fixed all defects then after tester exit into the project
* When project handover to UAT then than tester go outside into the testing.
* When product/Project go for live then tester can exit

**When Should Regression Testing Perform?**

* Regression testing is performed when any new functionality add in old software.
* It is testing is making sure that the product works fine with new functionality, bug fixes, or any change in the exiting features.
* Regression testing is a type testing of software. Test cases are re execute for check previous functionality in application to work fine, and changes have not produced any bugs.
* Regression testing is performed when one a new build when there is a significant change in the original functionality.

Main reason is matter to performed Regression Testing:-

* We do regression testing whenever the production code is modified.

**What is 7 key principles? Explain in detail**

This are general principle of testing .

* Testing shows presence of defects
* Exhausting testing is not possible
* Early Testing
* Defect Clustering
* The pesticide Paradox
* Testing is context dependent
* Absence of error fallacy

**Testing shows presence of defects**

With testing we can find the defects in software and improve the quality of software as per client requirements. When developed any software by developers then it is not possible developed any software without error. That’s why testing is very necessary that can improve quality and satisfied client.

Lots of testing are done by the tester and find the presence of defects. Developed software as per client requirements that is very necessary

**Exhausting testing is not possible**

In testing exhausting testing is not possible because in one page lots of inputs and validations are available then it is not possible to performed all testing of inputs.

If any tester tests all inputs and validation then lots of time is consuming. Lots of software are pending for testing then all inputs have specific time for testing.

**Early Testing**

Early testing focused on portfolio of software or objective of software. Testing start as soon as possible in the software or development and mainly focused on objective of software.

**Defect Clustering**

Defect clustering principle define to us when developed any software then small amount of defects are finding in modules.

Mainly focused on when in small module find defect then this defect does not spread in other modules.

**Pesticide Paradox**

In this principle informed to us that same testing do again and again then tester cannot find more defects, in that case test case revised regularly and reviewed. When regularly revised test cases then find more defects and make actual development of software.

Testing finds bugs, developer except for that and solves that bugs.

This all factors are necessary for create and use new test techniques.

**Testing is context dependent**

Basically all testing are based on sites.

If site is based on e-commerce then testing is done based on e-commerce type.

Different sites are tested differently.

**Absence of fallacy**

If system is unusable and environment is not proper then finding defects and error it does not help because system are gone totally in wrong direction.

We find defects and bugs but that testing is not requirements of client then still are unusable.

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

|  |  |  |
| --- | --- | --- |
| **QA** | **QC** | **TESTER** |
| QA stands for Quality Assurance. | QC stands for Quality Control. | Tester work which sure that finding bugs, errors, defects in the software. |
| QA focus on improving the software development process. | QC focus on an examination of the quality of the product outcomes rather than on the process used for create a product. | Tester Focus on finding defects. |
| QA work on product design and development to ensure it match with requirements from the start. | QC inspects and test what’s been made to ensure it’s right. | Tester conduct software after development process finished. |
| QA objective is improve the software system quality | Quality Control is all about finding and fixing mistakes. | Tester objective is to find the defects and ensure to meets requirements. |

**Difference between Smoke and Sanity:-**

|  |  |
| --- | --- |
| **Smoke Testing** | **Sanity Testing** |
| Smoke testing is documented | Sanity testing is isn’t documented |
| Smoke testing is performed either developer or tester | Sanity testing is generally performed by tester. |
| Smoke testing is may be stable or unstable | Sanity testing is stable |
| Smoke testing is scripted | Sanity testing is usually unscripted. |
| Smoke testing is test the all over functions of the system | Sanity testing is only performed on modified or defect functions of system. |
| Smoke testing can performed either manually or by using automation tool | Sanity testing is commonly executed manually, not by automated tool. |
| Smoke testing first perform initial build. Smoke testing is done first | Sanity testing is done on stable build. For the introduce new features in the software. |
| There is end-to-end system verification one in smoke testing | A specific component gets verified in sanity testing. |
| Smoke testing is also subset of acceptance testing | Sanity testing is also subset of regression testing. |

**Difference between verification and validation:-**

|  |  |
| --- | --- |
| Verification | Validation |
| Verification is set of activities that ensure software correctly implements the specific function | Validation is set of activities that ensure that software has been built traceable or customer requirements. |
| It includes checking codes, documents, designs, and programs | It includes testing and validating the actual product. |
| Verification is the static testing | Validation is dynamic testing |
| It does not includes the execution of code | It includes the execution of the code |
| It check the whether the software confirms to specification or not | It checks the whether the software meets the requirements and expectations of a customer or not. |
| It can find bugs in the early stage of the development. | It can only find the bugs that could not be found by the verification process. |
| It comes before validation | It comes after verification. |
| It is based on the opinion of reviewer and may be change person to person | Validation is based on the fact and is often stable. |

**Explain types of performance Testing:-**

Performance testing is a type of software testing that ensures software applications to perform properly under their expected work load.

Performance testing types are following:-

* Load Testing
* Stress Testing
* Endurance Testing
* Spike Testing
* Volume Testing
* Scalability Testing

***Load Testing***

Load testing is performed when test application or software in real world to see how much the system take load at maximum level.

Tester will start to check at low level to at highest level and test system which type of behave system.

Load testing is very important testing because when we test before any software or application put in market. That’s why we handle the risk will come in future.

Lots of users are do transaction by system so we can handle this all type of load by load testing.

***Stress Testing***

* Stress testing is a type of load testing that tests the system’s ability to handle high load above normal usage of usage levels.
* It can help to identify break point of the system and any potential issues that may occur under heavy load condition.
* At extreme stress level can break the system and this side effect put on all users So before we put any software and application in real world we must test this testing.

***Spike Testing:-***

* Spike testing is a type of load testing that tests the system’s ability to handle sudden spike in traffic.
* It helps to identify any issues that may occur when the system is suddenly hit with a high number of requests.

***Endurance Testing:-***

* This testing is focus on long term behavior of a system under constant load.
* It is performed to ensure the software can handle the expected load over a long period.

***Scalability Testing.***

* In scalability testing is performed software effectiveness is determined by scaling up to support an increase in user load.
* It helps in planning capacity addition to your software system.

***Volume Testing:-***

* In volume testing, a large number of data is saved in a database and over all system’s behavior is observed.
* This testing to check product’s performance under varying database volumes.

**What are error, defect, bug and failure?**

Software testing defines a set of procedures and methods that check whether the actual requirements are match with requirements. The main purpose of software testing is finding the errors, defect, bug, and failure.

|  |
| --- |
| Errors/Mistakes |

|  |
| --- |
| Defects/Bugs |

|  |
| --- |
| Failure |

Result in

Result in

**Errors:**

Error is the situation that when developers developed any product or software then they cannot understand and fails to developed task as per requirements. This situation consider as errors because its defined by the developers.

It is calculated by differentiating between the expected results and actual results.

Errors raised due to several reasons like coding issue, logic issue, system specific issue.

**Defects:**

Defects refer to a situation when the application not working as per expected result and requirements of the application or software are not sync with each other.

Defects are arise when developers make mistake and found by tester.

Defects define that efficiency and inability of the application to meet actual result and performing desire work.

**Bugs:**

Defects refer to a situation when application not works as per requirements or expected result in this case this defect except by developers that’s call bugs. Developers work on bugs.

Tester makes the template for defining defects and reporting to developers and developers’ works on bugs.

**Failure:**

Failure defines several mistakes that ultimately lead to software failure and loose of information in difficult modules. This situation is rarely arise to failed any software developing.

Failure detected by the end-users once they face a particular issue in the software.

**Difference between priority and severity:-**

|  |  |
| --- | --- |
| **Priority** | **Severity** |
| Priority is a parameter to decide the order to in which defect should be fixed. | Severity is a parameter to denote the impact of a particular defect on the software. |
| Priority meance how fast the defect has to fixed. | Severity means how severe the defect is affecting the functionality. |
| Priority is related to scheduling to resolve the problem | Severity is related to quality standard. |
| Priority is divided into three categories  High  Medium  Low | Severity divide into 4 categories  Critical  Major  Medium  Low |
| The product manager to decide the priority of defect. | The testing engineer decides the severity level of the defect. |
| Its value change to time to time | Its values doesn’t change time to time |
| It is associate with scheduling | It is associated with standard. |
| It indicate the how soon the bug should be fixed. | It indicates the seriousness of the bug in the product functionality |
| It based on customer’s requirements | It is based on technical aspect of the product. |

**What is traceability matrix?**

A traceability matrix is a document that traces and map the relationship between requirements and test cases.

Traceability is a one type of document that can map and trace the requirements by the clients and make test cases by the tester.

Traceability can map and trace the all requirements are fulfill or not. It can store all the requirements given by clients. Write the test cases according to requirements.

Traceability test the proper requirements because it is compulsory to fulfill all requirements given by the client, when software is ready to survey without traceability then clients are not satisfied with that software because software is not developed as per requirements. That’s why traceability is very important document that can handle proper tast cases and requirements.

**Types of traceability matrix**

* Forward Traceability
* Backward Traceability
* Bi-directional Traceability

**Forward Traceability**

Forward traceability works on mapping of requirements to test cases. Its works first check the requirements and then after go on test cases and make the test cases.

**Backward Traceability**

Backward traceability works on mapping of test cases to requirements. Its works first make the test cases and then after go on requirements and make the traceability.

**What is Integration Testing?**

Integration testing implemented when unit testing completed then after start integration testing. When integrated components or system interact to each other at that time perform integration testing.

Integration testing test the interfaces and integration between components and test the all parts are works together proper or not.

For example Any one developed software then that software is installed into the system. At that time apply the integration testing because this testing test software worked with operating system or other drivers work properly or not.

**There are two types of integration testing**

* Components Integration Testing
* System Integration Testing

COMPONENTS INTEGRATION TESTING

This testing is a software testing process where software components, modules, or units are tested or evaluate system compliance concerning functional requirements. Components testing are the process of components of the software are gradually integrated and tested by unified group.

Component integration testing -> test the components and interfaces between integrated components.

SYSTEM INTEGRATION TESTING

System integration testing is a process of hardware of software both are integrated each other and performed all operation. This will check the full functionality of the system or software.

System integration testing process performed first because first one check the environment of the system and then after considered all requirements as per our system.

If system not supported to clients requirements then developing is not possible. Its control the flow of system. It can also minimize the time consumption for testing process.

Its check the proper execution of the software and check the integration of the components are proper or not.

**What is Functional System Testing?**

Functional testing test the software against the functional requirements or specification. Functional testing ensures that the requirements or specification are properly satisfied or not.

This type of testing is particularly concerned with the result of the processing.

Functional system testing is basically defined as a type of testing that verifies that each functions of the software works in conformance with requirements and specifications.

Each functions compared to the corresponding requirement to ascertain whether its output consistent with the end user’s expectations. As per requirements specifies functions that perform by system components.

In functional testing included black box testing and this is not concern with the source code of the software. Each and every function have own functionality and that all are performed with particular functionality. User should input as per functionality.

Functional testing perform many types of and that all are following:-

**Unit testing, Integration testing, Black box testing, Component testing, System testing, White box testing, Interface testing.**

**What is Bug Life Cycle?**

A defect is an error or bug in an application that is created during the building or designing of software and due to which software starts to show abnormal behaviors during its use.

It is one of the responsibility of the tester to find defect as much as possible to ensure the quality of the products not affected and end of the product is fulfilling all requirements.

New

Duplicated

Rejected

Differed

Not a Bug

Assign

Open

Fixed

Reopend

Pending Retest

Retest

Verified

Closed

**New:-**

* When any new defect found by the tester, it falls into the new state. The tester provide all documents related to according bugs to the development team.
* So that development team refer to defect document and can fix bug accordingly.

Assigned:-

* Defects that are status as new defects will be approved and that newly identified defect assigned to the development team for working on defect and to resolve that.
* When defect is assigned to developer team than states will be changed will be ‘Assigned’ stage.

Open:-

* In this open state the defect is being addressed by the developer team and the developer team works on that defect for fixing the bug.
* Based on the some specific reason if the developer feels the that defect is not appropriate than it is transfer to either the ‘Rejected’ or ‘Differed’ stage.

Fixed:-

* After needed changes into the codes or after fixing identified bugs developer team marks the state as ‘FIXED’.

Pending Request:-

* During the fixing of the defect is completed, the developer team passes the new code to the testing team for retesting. And the code/application is pending for retesting on the Tester side so the status is assigned as ‘Pending Retest’.

**6. Retest:**

* At this stage, the tester starts work of retesting the defect to check whether the defect is fixed by the developer or not, and the status is marked as ‘Retesting’.

**7. Reopen:**

* After ‘Retesting’ if the tester team found that the bug continues like previously even after the developer team has fixed the bug, then the status of the bug is again changed to ‘Reopened’.
* Once again bug goes to the ‘Open’ state and goes through the life cycle again. This means it goes for Re-fixing by the developer team.

**8. Verified:**

* The tester re-tests the bug after it got fixed by the developer team and if the tester does not find any kind of defect/bug then the bug is fixed and the status assigned is ‘Verified’.

**Explain the difference between functional testing and non-functional testing:-**

|  |  |
| --- | --- |
| FUNCTIONAL TESTING | NON-FUNCTIONAL TESTING |

|  |  |
| --- | --- |
| Functional testing test first | Non-functional testing test after execute the function |
| Functional testing verifies the operations and actions of an application. | Non-functional testing verifies the behavior of the software. |
| It is based on the requirements of the clients | It is based on expectation of the clients. |
| It is validate to software actions | It is validate software performance |
| It is carried out of the functional specifications | It is carried out of the performance specifications. |
| It is describe that what product does. | It is described how the product works. |
| Manual testing or automation testing is used in functional testing | Using different types of tool will be effective. |

**What is the difference between the STLC and SDLC?**

|  |  |
| --- | --- |
| **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. |
| 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. | |  | |
| Creation of reusable software systems is the end result of SDLC. | A tested software system is the end result of STLC. |
| It helps in developing good quality software. | It helps in making the software defects free. |
| 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. | |  | |

**Write a difference between Test scenarios, test cases, and test scripts…**

|  |  |  |
| --- | --- | --- |
| **Test Scenario** | **Test Cases** | **Test Scripts** |
| Is any functionality that can be tested | Is a set of actions executed to verify particular features or functionality | Is a set of instructions to test an application functionality |
| Helps to end- to-end functionality in an Agile way | Helps in exhaustive testing an application | Helps to test specific things repeatedly |
| Is more focused on what to test. | Is focus on what to test and how to test. | Is focused on expected result |
| Takes less time and fewer resources to create. | Require more resources and more time | Require less time for testing but, more resources for scripts creating and updating. |
| The main task is to check the full functionality of a software application | The main focus on to check compliance with the application standards, guidelines, and customer requirements. | The main task is to check nothing is skipped, and the results are true as the desire testing plan. |
| Allows the quickly assessing the testing scope. | Allows detecting error and defects. | Allows carrying out an automatic execution of test cases. |

**Explain What test plan is? What is the information that should be covered?**

A test plan is a formal document that serves as a comprehensive and structured description of the testing activities and strategies that will be employed to check the quality of the software system under test. This document provides detailed information on the approach, methodology, scope, objectives, resources, timelines, and risks associated with the project.

The information should be covered, all team member have cleared about plan of the project or major components of the project.

It is also include the test environment, including hardware, software, and network configuration, as well as any external dependencies or services required for testing.

* Scheduling test implimantation, execution, and evalution.
* Making decision about what to do test
* Scope of the testing being performed.
* Criticality
* Project Risk
* Testability
* Availability of Resources
* Approach:-
  + Describe overall approach of testing, include the definition of the test levels and entry and exit criteria.
* Process:-

Set the all level of information for test procedures in order to getting full information to help reproduced test preparation and execution.

**What is priority?**

* Priority defines in order to which defects resolved defects.
* Priority set by the tester.
* If priority is higher than developers need that resolved first.
* Priority have following types:
  + Low
    - This menace when any defect is not should be required for earlier fixed, it can be fixed after few time, that’s called low priority.
  + Medium:-
    - The defect should be solved in normal cases and of development process.
  + High:
    - In this case defect solved as earlier as possible just because of this defect is effect on software or application. If this defect is not solved fastly than application or software not work properly. So, this called high priority.
* Critical:-

It is very urgent requirement and solved it immediately.

**What is Severity?**

* Severity focused on customer focused and absolute.
* It is the describe the which defect affect on application or software.
* It is impact that the given defect has on the system.
* Severity following steps:-
* **Critical**:-
  + It will find the what is the reason to terminal of the whole system or more and more component of the system and causes extensive corruption of data.
* **Major**:-
  + This is the significant flow to causes the system to fail.
  + However certain parts of the system remain functional.
* **Medium:**
* This flaw results in unfavorable behavior but the system remains functioning.
* **Low:**
* This type of flaw won’t cause any major breakdown in the system.

**Bug categories are…**

Different types of software bug categories are following:-

1. Functional Bugs
2. Logical Bugs
3. Workflow Bugs
4. Unit Level Bugs
5. System level Integration Bugs
6. Out of Bound Bugs
7. Security Bugs

Functional Bugs:-

Functional bugs are associated with the functionality of a specific software component. Like if submit button is not allowed for submit.

**Logical Bugs:-**

A logical bugs interrupts workflow of software and causes it to behave incorrectly. These kind of bugs can result in unexpected software behavior and even sudden crashes.

**Workflow Bugs:-**

Workflow bugs are according to user journey of a software or application. For example When any user enter detail about their profile then after filling all details than user want to save and exit, but user only do exit without saving information, that’s called workflow bugs.

**Unit Level Bugs:-**

Unit level of bugs are very easy to fixed. This is the small level of code are functioning as expected. For example… When any developer fill signup page and enter valid inputs and valida buttons for functionality. In case inputs are not accepting appropriate input than it is unit level bugs.

**System Level Integration Bugs:-**

When performed two or more units of code written by different developers fails to interact with each other. This bug occurs when two or more components execute together.

**Out of Bound Bugs:-**

These bugs occur when an end-user enters a value or a parameter outside the limits of unintended use.

For example, entering a significantly larger or a smaller number or entering an input value of an undefined data type. These bugs often pop up in form validations during functional testing of web or mobile apps.

**Security Bugs:-**

Security is a major concern for software development. Security Bugs are a major risk for users and should be taken very seriously and resolved.

These bugs might not hinder the operation but can compromise the whole system. These should be checked thoroughly at regular intervals.

**Advantages 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.
* It is open Source widely used bug tracker.
* It is easy in usage and its user interface is understandable for people without technical knowledge.
* It is easy integrates with test management instruments.
* It integrates with an emailing system.
* It automates documentation.

**What are the different methodologies in Agile Development Model?**

Scrum:-

It focuses primarily on how to handle tasks in a team-based development setting, and it is an agile development methodology

Scrum promotes operating in small teams and thinks that the development team should be empowered

* **Scrum Lead:**The scrum master is in charge of organizing the team, the sprint meeting, and removing roadblocks.
* **Product creator:**The product owner builds the product backlog, organizes it by priority, and is in charge of delivering features at each iteration.
* **Agile Team:**Team coordinates and oversees its own work to finish the sprint or cycle.

#### Extreme Programming (XP):-

The Agile framework for software development processes most closely resembles XP. It strives to build high-quality software while also simplifying the entire process for the development team.

It works best when,

* The criteria are always shifting.
* Team deadlines are constrained.
* Stakeholders desire to lower risk while meeting timelines.
* Unit and functional testing can be automated by teams.

#### Adaptive Software Development (ASD)

Learn, collaborate, and speculate is the term of the dynamic growth process that is utilized in ASD. Because the business setting is continuously changing, this process is concentrated on close customer and development engagement and ongoing learning

#### Dynamic Software Development Method (DSDM):-

With the Dynamic Software Development Method , a roadmap of continuous and early deliveries can be created for the project. This allows for the implementation of an incremental solution, adaptation in response to feedback received along the way, and verification that the anticipated benefits are being realized.

#### Feature Driven Development (FDD):-

The key component of this methodology is “designing & creating” features. Domain walkthrough, design review, promotion to build, code review, and design are all included

#### Kanban:-

Without adding to the stress of the software development lifecycle, Kanban is a highly visual workflow management technique that enables teams to actively supervise product creation, with a focus on continuous delivery (SDLC). It has gained popularity among groups that use Lean software development techniques.

The three fundamental tenets of Kanban are to visualize the workflow, reduce the amount of work that is in process, and enhance the flow of work

#### Behavior Driven Development (BDD)

A behavior-focused agilesystem development methodology is called behavior driven development

The BDD development process is built on the writing of test cases and features.

It defines what is necessary for the functionality to start, what will happen next, and what the results will be after it has been completed

**Explain difference between Authorization and Authentication in web testing. What are the common problem face in web testing?**

* Simply put, 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.
* Authentication and authorization accomplish these same goals. Authentication is used to verify that users really are who they represent themselves to be. Once this has been confirmed, authorization is then used to grant the user permission to access different levels of information and perform specific functions, depending on the rules established for different types of users.

|  |  |
| --- | --- |
| **Authentication** | **Authorization** |
| Authentication verifies who the user is. | Authorization determines what resources a user can access. |
| Authentication works through password, one-time pins, biometric information, and other information provided or entered by the user. | Authorization works through settings that are implemented and maintained by the organization. |
| Authentication is the first step of a good identity and access management process. | Authorization always takes place after authentication. |
| Authentication is visible to and partially changeable by the user | Authorization isn’t visible to or changeable by the user. |
| Example: By confirming their identity, employees can gain access to a human resources (HR) application that includes their personal pay information, vacation time. | Example: Once their level of access is authorized, employees and HR managers can access different levels of data based on the permissions set by the organization. |

**Write the scenario of PEN:-**

1. Verify the type of pen, whether it is a ballpoint pen, ink pen, or gel pen.
2. Verify that the user is able to write clearly over different types of papers.
3. Check the weight of the pen. It should be as per the specifications. In case not mentioned in the specifications, the weight should not be too heavy to impact its smooth operation.
4. Verify if the pen is with a cap or without a cap.
5. Verify the color of the ink on the pen.
6. Check the odor of the pen’s ink on writing over a surface.
7. Verify the surfaces over which the pen is able to write smoothly apart from paper e.g. cardboard, rubber surface, etc.
8. Verify that the text written by the pen should have consistent ink flow without leaving any blob.
9. Check that the pen’s ink should not leak in case it is tilted upside down.

10.Verify if the pen’s ink should not leak at higher altitudes.

11.Verify if the text written by the pen is erasable or not.

12.Check the functioning of the pen by applying normal pressure during writing.

**Write a scenario of Pen Stand :-**

1. Verify that the length and the diameter of the pen are as per the specifications.
2. Verify the outer body material of the pen. Check if it is metallic, plastic, or any other material specified in the requirement specifications.
3. Check the color of the outer body of the pen. It should be as per the specifications.
4. Verify that the brand name and/or logo of the company creating the pen should be clearly visible.
5. Verify that any information displayed on the pen should be legible and clearly visible.

**Write a scenario of Door:-**

1. Verify if the door is single door or bi-folded door.
2. Check if the door opens inwards or outwards.
3. Verify that color of the door is as specified.
4. Verify if the door is sliding door or rotating door.
5. Check the position, quality and strength of hinges.
6. Check the type of locks in the door.
7. Verify if the door makes noise when opened or closed.
8. Check the door condition when used extensively with water.
9. Check the door condition in different climatic conditions- temperature, humidity etc.

10.Check the amount of force- pull or push required to open or close the door.

**Write a scenario of ATM:-**

1. Verify the type of ATM machine, if it has a touch screen, both keypad buttons only, or both.
2. Verify that on properly inserting a valid card different banking options appear on the screen.
3. Check that no option to continue and enter credentials is displayed to the user when the card is inserted incorrectly.
4. Verify that the touch of the ATM screen is smooth and operational.
5. Verify that the user is presented with the option to choose a language for further operations.
6. Check that the pin is displayed in masked form when entered.
7. Verify that the user is presented with different account type options like- saving, current, etc.
8. Verify that the user is allowed to get account details like available balance.
9. Check that the correct amount of money gets withdrawn as entered by the user for cash withdrawal.
10. Verify that the user is only allowed to enter the amount in multiple denominations as per the specifications.
11. Verify that the user is prompted to enter the amount again in case the amount entered is less than the minimum amount configured.
12. Check that the user is not allowed to proceed with the expired ATM card and that a proper error message gets displayed.

**When to used Usability testing?**

Usability testing is more important because of any software or application how looks like usually determine how well it work.

Using usability testing we find errors early in the system in development.

For example:-

* When any buttons are not clickable.
* What is use of icon.
* Error messages are not generate in consistent way.

Usability testing do with following tool:=

* **Web based testing**
* **Desktop Based Testing**
* **Mobile Based Testing**
* **Game Based Testing**

**Goal Of usability testing:-**

* Developed product or software with Accuracy.
* Developed product or software with Efficiency.
* Software or product designing must stay user friendly.
* Using this testing product or software developed strong features and effectiveness.

**What is the procedure for GUI testing?**

Currently, all website or software are beyond the desktop based or cloud based.

All user want to need more friendly more about customer requirements.

A GUI testing team always pays close attention to each detail in visual dynamics to ensure end-user satisfaction and ease.

* It provides a customizable test report.
* It is run tests in parallel or distributed on a Selenium Grid with built-in Selenium Webdriver.
* It allows you to test the functionality from a user’s perspective.
* Sometimes the internal functions of the system work correctly but the user interface doesn’t then GUI testing is good to have in addition to the other types.
* It provides reliable object identification, even for web elements with dynamic IDs.

**Types of Graphical User Interface Testing (GUI) Testing:**

There are two types of GUI testing which are given below: Analog Recording, and Object based recording. These are explained as following below.

1. **Analog Recording:**   
   This is always what people associate with GUI testing tools by analog recording, the testing tool basically captures specific mouse clicks, keyboard presses, and other user actions and then simply stores them in a file for playback.
2. **Object based Recording:**   
   In object based recording, the testing tool is able to connect programmatically to the application being tested and “see” each of the individual user interface components as separate entities and is able to perform operations click, enter text and read the state reliably regardless of where that object is on the screen.

**Write a scenario for Microwave oven:-**

* Verify that the dimensions of the oven are as per the specification provided.
* Verify that the oven’s material is optimal for its use as an oven and as per the specification.
* Verify that the oven heats the food at the desired temperature properly.
* Verify that the oven heats food at the desired temperature within a specified time duration.
* Verify the ovens functioning with the maximum attainable temperature.
* Verify the ovens functioning with minimum attainable temperature.
* Verify that the text written over the oven’s body is clearly readable.
* Verify that the digital display is clearly visible and functions correctly.
* Verify that the temperature regulator is smooth to operate.
* Verify that the temperature regulator works correctly.
* Check the maximum capacity of the oven and test its functioning with that volume of food.
* Check the oven’s functionality with different kinds of food – solid, and liquid.

.

**Write a scenario for Cofee vending machine:-**

* Verify that outer body, as well as inner part’s material, is as per the specification.
* Verify that the machine’s body color as well brand is correctly visible and as per specification.
* Verify the input mechanism for coffee ingredients-milk, water, coffee beans/powder, etc.
* Verify that the quantity of hot water, milk, coffee powder per serving is correct.
* Verify the power/voltage requirements of the machine.
* Verify the effect of suddenly switching off the machine or cutting the power. The machine should stop in that situation and in power resumption, the remaining coffee should not get come out of the nozzle.
* Verify that coffee should not leak when not in operation.
* Check for the indicator lights when the machine is switched on-off.
* Verify that the functioning of all the buttons work properly when pressed.
* Verify that each button has an image/text with it, indicating the task it performs.

**Write a scenario of Chair:-**

* Verify that the chair is stable enough to take an average human load.
* Check the material used in making the chair-wood, plastic etc.
* Check if the chair’s leg are level to the floor.
* Check the usability of the chair as an office chair, normal household chair.
* Check if there is back support in the chair.
* Check if there is support for hands in the chair.
* Verify the paint’s type and color.
* Verify if the chair’s material is brittle or not.
* Check if cushion is provided with chair or not.
* Check the condition when washed with water or effect of water on chair.
* Verify that the dimension of chair is as per the specifications.
* Verify that the weight of the chair is as per the specifications.

**To create scenario (Positive & Negative):-**

**Positive Test Scenarios Gmail:-**

* Verify that user can able to login with valid email.
* 2.Verify that user can able to login with valid password.
* 3.Verify that user can able to use forgot password functionality.
* Verify that if user can put the invalid credentials shows an error message user can able to use.
* 5.Verify that user can able to login only with new/change password.
* 6.Verify that user can able click all the tab which is available on login page UI test scenario.
* 7.Verify all color, font size are visible.
* 8.Verify that application UI is responsive.

**Negative Test Scenarios Gmail:-**

* Verify that user can not able to login with valid email.
* 2.Verify that user can not able to login with valid password.
* 3.Verify that user can not able to login with blank field.
* Verify that if user can put the valid credentials shows an error message.
* 5.Verify that user can not able to login with valid password and invalid email.

**Test Scenario of to buy online shopping to buy product(Flipkart)**

* Verify accurate display of product details.
* Test responsiveness to changes in product availability.
* Validate product pricing and discount calculations.
* Verify the accuracy of product specifications.
* Test the responsiveness of the product image gallery.
* Validate the functionality of the “*Buy Now*” button.
* Test adding the product to the shopping cart.
* Verify accurate display of related products.
* Test sharing product details on social media.
* Verify adding a product to the cart.
* Validate the behavior of adding out-of-stock items.
* Test the accuracy of product details in the cart.
* Verify the removal of items from the cart.
* Test responsiveness on changing the quantity of items.
* Verify the accuracy of total price calculations.
* Test adding products with different sizes and color variants.
* Test the Add to Cart functionality across various devices and browsers.
* Verify redirection to the secure payment gateway.
* Test behavior with incorrect payment details.
* Validate responsiveness to changes in payment methods.
* Check the accuracy of applied discounts.
* Test successful payment processing.
* Verify the visibility of transaction details.
* Test behavior with declined transactions.
* Validate responsiveness to changes in user location.
* Test the visibility of saved payment methods.
* Verify accurate calculation of transaction charges.
* Test the behavior of network interruptions during payment.
* Validate the Choosing from various payment options.
* Test the accurate deduction of the transaction amount.
* Check redirection to the first payment gateway page upon forced cancellation of the payment.
* Test the redirection to the Order Confirmation page.

**Write a scenario for Wrist Watch…**

* Verify the type of watch – analog or digital.
* In the case of an analog watch, check the correctness time displayed by the second, minute, and hour hand of the watch.
* In the case of a digital watch, check the digital display for hours, minutes, and seconds is correctly displayed.
* Verify the material of the watch and its strap.
* Check if the shape of the dial is as per specification.
* Verify the dimension of the watch is as per the specification.
* Verify the weight of the watch.
* Check if the watch is waterproof or not.
* Verify that the numbers in the dial are clearly visible or not.
* Check if the watch is having a date and day display or not.
* Verify the color of the text displayed in the watch – time, day, date, and other information.
* Verify that clock’s time can be corrected using the key in case of an analog clock and buttons in case of a digital clock.
* Verify if the dial’s glass/plastic is resistant to minor scratches or not.
* Check the battery requirement of the watch.

**Write a scenario of Lift(Elevator)**

* Verify the dimensions
* Verify the type of door is as per the specification
* Verify the type of metal used in the lift interior and exterior
* Verify the capacity  in terms of the total weight
* Verify the buttons in the lift to close and open the door and numbers as per the number of floors.
* Verify that lifts move to the particular floor as the floor button is clicked.
* Verify that lifts stop when up/down buttons on a particular floor are pressed.
* Verify if there is an emergency button to contact officials in case of any mishap.
* Verify the performance of the floor – the time taken to go to a floor
* Verify that it doesn’t free-fall in case of a power failure and gets halted on the particular floor.
* Verify lifts are working if the door button is pressed before reaching the destination floor.
* Verify that in case the door is about to close and an object is placed between the doors, the object again opens or not.
* Verify the time duration for which the door remains open by default.
* Verify if the lift interior has proper air ventilation
* Verify lighting in the lift
* Verify that at no point should the lift door should open while in motion.
* Verify that there should be a backup mechanism to safely get into a floor or backup power supply in case of power loss.

**Write a scenario of Watsapp Group(generate group)**

* create new group by adding contacts.
* try to send txt and other multimedia messges n check recieved.
* check receiving of txt and multimedia messages from all group members.
* exit group chat.
* delete group.
* admin access testing, adding new admin.
* removing the exist contacts from group.
* add new contacts to group.
* Check for images, videos etc are uploaded properly or not.
* Check if any member want to send message privately to group admin.

**Write a scenario of Watsapp Payment:-**

* Check bank account is add properly or not
* Check your profile related transaction work properly or not.
* Check security for transaction
* Check transaction other options are work properly or not.
* Check display all bank properly or not.
* Check transaction that from one person to another person properly or not.