**MODULE 2**

**MANUAL TESTING**

**ASSIGNMENT**

** What is Exploratory Testing?**

Exploratory testing is an approach to software testing that is often described as simultaneous learning, test design, and execution.

Exploratory testing is a concurrent process where

* Test design, execution and logging happen simultaneously
* Testing is often not recorded
* Makes use of experience, heuristics and test patterns

** What is traceability matrix?**

* The traceability matrix is to protct against changes you should be able to trace back from every system component to the original requirement that caused its presence.
* There are three types of tracebility matrix

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

** What is Boundary value testing?**

* Boundary value testing is the methodology of design test cases the concentrates that software testing effort on cases near the limits of valid range.
* Boundary value analysis is a method which refine the equivalence partitioning.
* At those points when input values change from valid to invalid errors are most likely to occur.
* Boundary value analysis is generate test cases that highlight errors better than equivalence partitioning.

** What is Equivalence partitioning testing?**

* This is the one type of black box technique.
* It is use for all type of testing.
* Ain is to treat groups of inputs as equivalent and to select one representative input to test them all.
* If we want to test the following IF statement: “If value is between 1 and 100 (inclusive) (e.g. value >=1 and value <=100) Then...”

** What is Integration testing?**

* Integration testing is testing perform to expose defects in the interface and in the interaction between the integrated component or system.
* Integration testing is the level of software testing process where individual units are combine and tested as a group.

** What determines the level of risk?**

* There are two types of risk

1. Product risk
2. Project risk
3. Project risk:

* Example of Project risk is Senior Team Member leaving the project abruptly.
* Every risk is assigned a likelihood i.e. chance of it occurring, typically on a scale of 1 to 10.

Also the impact of that risk is identified on a scale of 1- 10 .

* But just identifying the risk is not enough. You need to identify mitigation. In this case mitigation could be Knowledge Transfer to other team members & having a buffer tester in place.

2.Product Risk:

* Example of product risks would be Flight Reservation system not installing in test environment
* Mitigation in this case would be conducting a smoke or sanity testing. Accordingly you will make changes in your scope items to include sanity testing

** What is component testing?**

* Component testing is also known as Unit testing.
* A minimal software item that can be tested in isolation is known as unit testing.
* Unit testing is smallest testable part of the software.
* The testing of individual software component it also knows as component testing.
* Sometimes known as unit testing,component testing,module testing and program testing.
* Unit testing is the first level of testing and is perform prior to integrated testing .
* A unit testing is performed by using the white box testing methods.

** What is functional system testing?**

* A requirement that specify a function that system or the system component must perform.
* A requirement may exist as a document or model.
* There are three types of approaches

1. Requirement based Functional testing
2. Process based testing

** What is Non-Functional Testing?**

* Testing the attributes of a component or system that do not relate to functionality, e.g. reliability, efficiency, usability, interoperability, maintainability and portability
* Non-functional testing includes, but is not limited to, performance testing, load testing, stress testing, usability testing, maintainability testing, reliability testing and portability testing.
* It is tough to do manual testing.
* Non functional testing describes how good the product work.

** What is GUI Testing?**

* GUI it means Graphical User Interface.
* GUI testing is involves checking the screens with the control button like Menu, buttons, icons and all type of bars like statusbar, menubar, toolbar etc.
* Check error message are display correctly.
* Check font used in application is readable.
* Check the alignment of the text is proper.
* Check the photos are properly aligned.

** What is Adhoc testing?**

* Adhoc testing is informal or unstructured software testing types that aims to break the testing process to find possible defects or errors at an early possible stage.
* Adhoc testing is also known as error guessing.
* It does not follow any test design techniques and create test cases.
* The main aim of testing is to find defects by random checking.
* There are three types of Adhoc testing.

1.Buddy testing

2 Pair testing

3 Monkey testing

** What is load testing?**

* Its a performance testing to check system behavior under load. Testing an application under heavy loads, such as testing of a website under a range of loads to determine at what point the system’s response time degrades or fails.
* Load testing is a kind of performance testing which determines a system’s performance under real-life load conditions. This testing helps determine how the application behaves when multiple users access it simultaneously.

**Need For Load Testing**

* Some extremely popular sites have suﬀered serious downtimes when they get massive traﬃc volumes. E-commerce websites invest heavily in advertising campaigns, but not in Load Testing to ensure optimal system performance, when that marketing brings in traﬃc.

**Goals of Load Testing**

* Loading testing identiﬁes the following problems before moving the application to market or Production:
* Response time for each transaction
* Performance of System components under various loads
* Performance of Database components under diﬀerent loads
* Network delay between the client and the server
* Software design issues
* Server conﬁguration issues like Web server, application server, database server etc.
* Hardware limitation issues like CPU maximization, memory limitations, network bottleneck, etc.

** What is stress Testing?**

System is stressed beyond its speciﬁcations to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load.

* Stress testing is used to test the stability & reliability of the system. This test mainly determines the system on its robustness and error handling under extremely heavy load conditions.
* Stress testing is also known as endurance testing.

**Types of Stress Testing**

* Application Stress Testing:
* Transactional Stress Testing
* Systemic Stress Testing
* Exploratory Stress Testing

**Stress Testing Tools**

* Stress Tester
* Neo Load
* App Perfect

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

* Testing based on an analysis of internal structure of the component or system.
* Structure based technique is also known as white box testing and glass box.
* In this technique the tester require knowledge of how the software is implemented , how it works.
* The tester is concentraing on how the software does it.
* Testing based upon the structure of the code.

There are three types of coverage of white box testing

1. Statement Coverage
2. Condition Coverage
3. Decision Coverage

** What is black box testing? What are the different black box testing techniques?**

* Testing either functional or non functional, without reference to the internal structure of the component or system.
* Specification based on testing is also known as Black box techniques.
* The testers have no knowledge of how system or component is structure inside the box.
* What a system does, rather than how it does it

There are four types of technique of black box

1. Boundry value analysis
2. Equivalance Partitioning
3. State transition Table
4. Decision Table

** Mention what are the categories of defects?**

**Categories of defect:**

* Arithmetic Defects
* Logical Defects
* Syntax Defects
* Multithreading Defects
* Interface Defects
* Performance Defects

** Mention what bigbang testing is?**

In Big Bang integration testing all components or modules is integrated simultaneously, after which everything is tested as a whole. ⚫ Big Bang testing has the advantage that everything is ﬁnished before integration testing starts.

**Advantages:**

* Convenient for small systems.

**Disadvantages**

* Fault Localization is diﬃcult.
* Given the sheer number of interfaces that need to be tested in this approach, some interfaces links to be tested could be missed easily.
* Since the integration testing can commence only after “all” the modules are designed, testing team will have less time for execution in the testing phase.
* Since all modules are tested at once, high risk critical modules are not isolated and tested on priority. Peripheral modules which deal with user interfaces are also not isolated and tested on priority.

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

Purpose of exit criteria is to define when we stop testing either at the:

* End of all testing - i.e. product Go live
* End of phase of testing(e.g. hand over from system test to UAT)

** When should "Regression Testing" be performed?**

Testing of a previously tested program following modiﬁcation to ensure that defect have not been introduced or uncovered in unchanged areas of the software, as a result of the changes made. It is performed when the software or its environment is changed.

**Need of Regression Testing**

* Change in requirements and code is modiﬁed according to the requirement
* New feature is added to the software
* Defect ﬁxing
* Performance issue ﬁx

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

**1.Testing shows presence of Defects**

* Testing can show that defects are present, but cannot prove that there are no defects. Testing reduces the probability of undiscovered defects remaining in the software but, even if no defects are found, it is not a proof of correctness. We test to find Faults As we find more defects, the probability of undiscovered defects remaining in a system reduces.

**2. Exhaustive Testing is Impossible!**

Testing everything including all combinations of inputs and preconditions is not possible. So, instead of doing the exhaustive testing we can use risks and priorities to focus testing efforts

**3. Early Testing**

Testing activities should start as early as possible in the software or system development life cycle, and should be focused on defined objectives. These activities should be focused on defined objectives – outlined in the Test Strategy

**4. Defect Clustering**

Defects are not evenly spread in a system They are ‘clustered’ In other words, most defects found during testing are usually confined to a small number of modules

**5. The Pesticide Paradox**

If the same tests are repeated over and over again, eventually the same set of test cases will no longer find any new defects. To overcome this “pesticide paradox”, the test cases need to be regularly reviewed and revised, and new and different tests need to be written to exercise different parts of the software or system to potentially find more defects.

**6. Testing is Context Dependent**

Testing is basically context dependent. Testing is done differently in different contexts Different kinds of sites are tested differently. Also different industries impose different testing standards

**7. Absence of Errors Fallacy**

If the system built is unusable and does not fulfill the user’s needs and expectations then finding and fixing defects does not help.

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

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

** Difference between Smoke and Sanity?**

|  |  |
| --- | --- |
| **Smoke testing** | **Sanity testing** |
| Smoke Testing is performed to ascertain that the critical functionalities of the program is working fine | Sanity Testing is done to check the new functionality/bugs have been fixed |
| Smoke testing is usually documented or scripted | Sanity testing is usually not documented and is unscripted |
| This testing is performed by the developers or testers | Sanity testing in software testing is usually performed by testers |
| Smoke testing is a subset of Acceptance testing | Sanity testing is a subset of [Regression Testing](https://www.guru99.com/regression-testing.html) |
| Smoke testing is like General Health Check Up | Sanity Testing is like specialized health check up |
| Smoke testing exercises the entire system from end to end | Sanity testing exercises only the particular component of the entire system |
|  |  |

** Difference between verification and Validation**

|  |  |
| --- | --- |
| **Verification** | **Validation** |
| It includes checking documents, design, codes and programs | It includes testing and validating the actual product. |
| Verification is the static testing. | Validation is the dynamic testing. |
| It does *not* include the execution of the code. | It includes the execution of the code. |
| Methods used in verification are reviews, walkthroughs, inspections and desk-checking. | Methods used in validation are Black Box Testing, White Box Testing and non-functional testing. |
| Quality assurance team does verification. | Validation is executed on software code with the help of testing team. |
| It consists of checking of documents/files and is performed by human. | It consists of execution of program and is performed by computer. |

** Explain types of Performance testing.**

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

** Explain the difference between Functional testing and NonFunctional testing**

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

** Difference between Priority and Severity**

|  |  |
| --- | --- |
| **Priority** | **Severity** |
| Defect Priority has defined the order in which the developer should resolve a defect | Defect Severity is defined as the degree of impact that a defect has on the operation of the product |
| Priority is associated with scheduling | Severity is associated with functionality or standards |
| Priority indicates how soon the bug should be fixed | Severity indicates the seriousness of the defect on the product functionality |
| Priority is driven by business value | Severity is driven by functionality |
| Priority status is based on customer requirements | Severity status is based on the technical aspect of the product |
| Priority is categorized into three types   * Low * Medium * High | Severity is categorized into five types   * Critical * Major * Moderate * Minor * Cosmetic |

** What is Error, Defect, Bug and failure?**

**Error** “A mistake in coding is called error,

**Defect**  error found by tester is called defect,

**Bug** defect accepted by development team then it is called bug,

**Failure** build does not meet the requirements then it is failure

** What is Bug Life Cycle?**

* “A computer bug is an error, flaw, mistake, failure, or fault in a computer program that prevents it from working correctly or produces an incorrect result. Bugs arise from mistakes and errors, made by people, in either a program’s source code or its design.”
* When a bug is discovered, it goes through several states and eventually reaches one of the terminal states, where it becomes inactive and closed.

**The number of states that a defect goes through varies from project to project. Below lifecycle diagram, covers all possible states**

* **New:** When a new defect is logged and posted for the first time. It is assigned a status as NEW.
* **Assigned:** Once the bug is posted by the tester, the lead of the tester approves the bug and assigns the bug to the developer team
* **Open:** The developer starts analyzing and works on the defect fix
* **Fixed:** When a developer makes a necessary code change and verifies the change, he or she can make bug status as “Fixed.”
* **Pending retest**: Once the defect is fixed the developer gives a particular code for retesting the code to the tester. Since the software testing remains pending from the testers end, the status assigned is “pending retest.”
* **Retest:** Tester does the retesting of the code at this stage to check whether the defect is fixed by the developer or not and changes the status to “Re-test.”
* **Verified**: The tester re-tests the bug after it got fixed by the developer. If there is no bug detected in the software, then the bug is fixed and the status assigned is “verified.”
* **Reopen**: If the bug persists even after the developer has fixed the bug, the tester changes the status to “reopened”. Once again the bug goes through the life cycle.
* **Closed**: If the bug is no longer exists then tester assigns the status “Closed.”
* **Duplicate**: If the defect is repeated twice or the defect corresponds to the same concept of the bug, the status is changed to “duplicate.”
* **Rejected**: If the developer feels the defect is not a genuine defect then it changes the defect to “rejected.”
* **Deferred:** If the present bug is not of a prime priority and if it is expected to get fixed in the next release, then status “Deferred” is assigned to such bugs
* **Not a bug:** If it does not affect the functionality of the application then the status assigned to a bug is “Not a bug”.



** What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)?**

|  |  |
| --- | --- |
| **SDLC** | **STLC** |
| Development Life Cycle | Testing Life Cycle |
| The main object of SDLC life cycle is to complete successful development of the software including testing and other phases. | The only objective of the STLC phase is testing. |
| In SDLC the business analyst gathers the requirements and create Development Plan | In STLC, the QA team analyze requirement documents like functional and non-functional documents and create System Test Plan |
| In SDLC, the development team creates the high and low-level design plans | In STLC, the test analyst creates the Integration Test Plan |
| The real code is developed, and actual work takes place as per the design documents. | The testing team prepares the test environment and executes them |
| SDLC phase also includes post-deployment supports and updates. | Testers, execute regression suits, usually automation scripts to check maintenance code deployed. |

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

|  |  |  |
| --- | --- | --- |
| **TEST SCENARIO** | **TEST CASES** | **TEST SCRIPT** |
| Is any functionality that can be tested | Is a setoff actions executed to verify particular features or functionality | Is a set of instructions to test an app automatically |
| Is derived from test artifacts like Business Requirement Specification and Software Requirement Specification | Is mostly derived from test scenarios | Is mostly derived from test cases |
| Helps test the end-to-end functionality in an Agile way | Helps in exhaustive testing of an app | Helps to test specification things result |
| Is more focused on what to test | Is focused on what to test and how to test | Is focused on the expected result |
| The main task is to check the full functionality of a software application | The main task is to verify compliance with the applicable standards, guidelines, and customer requirement | The main task is to verify that nothing is skipped, and the results are true as the desired testing plan |
| Allows quickly assessing the testing scope | Allows detecting errors 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 document describing the scope, approach, resources and schedule of intended test activities Determining the scope and risks, and identifying the objectives of testing.
* Scheduling test analysis and design activities.
* Scheduling test implementation, execution and evaluation.
* All projects require a set of plans and strategies which define how the testing will be conducted.

** Bug categories are…**

* Security,
* Database,
* Functionality (Critical/General),
* UI.

** Advantage of Bugzila .**

Advanced search capabilities

* E-mail Notifications
* Modify/file Bugs by e-mail
* Time tracking
* Strong security
* Customization Localization

** What are the different Methodologies in Agile Development Model?**

The Agile methodology is a way to manage a project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement at every stage.

* Once the work begins, teams cycle through a process of planning, executing, and evaluating. Agile is a philosophy, i.e., a set of values and principles to make a decision for developing software.
* Agile is based on the iterative-incremental model. In an incremental model, we create the system in increments, where each increment is developed and tested individually

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

|  |  |
| --- | --- |
| **Authorization** | **Authentication** |
| Process of specifying access rights/privileges to resources related to information security | Process of confirming the truth of an attribute of a single piece of data claimed true by an entity |
| Checks a user’s privileges to access resources | Checks a person’s details to identify him |
| Verifies user’s permissions | Verifies user’s credentials |
| Occurs after authentications | Occurs before authorization |
| EX. he can access lecture slides and other learning material of the courses based on the permissions given to him | EX.A student can authenticate himself before accessing the Learning Management Systems of a University |

** When to used Usablity Testing?**

Usability Testing identifies usability errors in the system early in development cycle and can save a product from failure. Aesthetics and design are important. How well a product looks usually determines how well it works.

** What is the procedure for GUI Testing?**

Check all the GUI elements for size, position, width, length and acceptance of characters or numbers. For instance, you must be able to provide inputs to the input fields.

* Check you can execute the intended functionality of the application using the GUI
* Check Error Messages are displayed correctly
* Check for Clear demarcation of different sections on screen
* Check Font used in application is readable
* Check the alignment of the text is proper
* Check the Colour of the font and warning messages is aesthetically pleasing
* Check that the images have good clarity Check that the images are properly aligned

** Write a scenario of only Whatsapp chat messages**

1. Check the chat window that contains the entire chat list
2. Check the chat window display the contact numbers whose number are not save on mobile
3. Check the chat window displayed with all contact with DP or without DP
4. Check the chat page displayed on the group chat list
5. Check the chat window display the last updated chatting time
6. Check to click on one contact that should be open new page with history
7. Check that user can see all send and received messages
8. Check the user can see the read or send time of messages
9. Check the user can send and receive text messages in the individual chatbox
10. Check the user can send and receive documents in the individual chatbox
11. Check the user can send and receive photos in the individual chatbox
12. Check the user can send and receive videos in the individual chatbox
13. Check the user can send and receive audio in the individual chatbox
14. Verify the user can send and receive emotional icons in the individual chat box
15. Check the user can send and receive contacts in the individual chatbox
16. Check the user can send and receive location in the individual chatbox
17. Check the user can send and receive GIFs in the individual chatbox
18. Check the user can send and receive Stickers in the individual chatbox
19. Verify the user can delete text, video, audio, location, and documents in the individual chat
20. Check the user can delete the entire chat history in the individual chat
21. Verify the user can search specific chat history using the search option in the chat
22. Check the user can mute the individual in the individual chatbox
23. Check the user can change the DP
24. Check the user can change the wallpaper
25. Check the user have options like report, block, clear chat, add shortcut so on

** Write a Scenario of Pen**

|  |
| --- |
| 1. Verify the types of pen, whether it is a ballpointpen, gel pen or ink pen. |
| 1. Verify that the user is able to write clearly over different types of paper. |
| 1. Verify if the pen is with a cap or without cap |
| 1. Verify the colour of the ink of pen |
| 1. Verify that the text written by the pen should have consistent ink flow without leaving any blob. |
| 1. Check the that pen’s ink should not leak in case it is tilted upside down. |
| 1. Verify if the pen’s ink should not leak at higher altitudes. |
| 1. Verify if the text written by the pen is erasable or not |
| 1. Check if the pen can support multiple refills or not |
| 1. Verify the working of pen point is working properly or not |

** 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 the dimension of the doors are as per the specifications
4. Verify that the material used in the door body and its parts is as per the specifications
5. Verify that color of the door is as specified
6. Verify if the door is sliding door or rotating door
7. Check the position, quality and strength of hinges
8. Check the type of locks in the door
9. Check the number of locks in the door interior side or exterior side
10. Verify if the door is having peek-hole or not
11. Verify if the door is having stopper or not
12. Verify if the door closes automatically or not – spring mechanism
13. Verify if the door makes noise when opened or closed
14. Check the door condition when used extensively with water
15. Check the door condition in different climatic conditions- temperature, humidity etc
16. Check the amount of force- pull or push required to open or close the door

** Write a Scenario of ATM**

* 1. Verify the ‘ATM Card Insertion Slot’ is as per the specification  
     2. Verify the ATM machine accepts card and PIN details  
     3. Verify the error message by inserting a card incorrectly  
     4. Verify the error message by inserting an invalid card (Expired Card)  
     5. Verify the error message by entering an incorrect PIN  
     6. Verify that the user is asked to enter the PIN after inserting a valid ATM Card  
     7. Verify that PIN is encrypted  
     8. Verify that there is an action like blocking of card occurs when the total no. of incorrect PIN attempts get surpassed  
     9. Verify the user is allowed to do only one cash withdrawal transaction per PIN request  
     10. Verify the machine logs out of the user session immediately after successful withdrawal  
     11. Verify the message when there is no money in the ATM  
     12. Verify the language selection functionality  
     13. Verify the cash withdrawal functionality by entering some valid amount  
     14. Verify the cash withdrawal functionality by entering an amount less than 100  
     15. Verify the cash withdrawal functionality by entering an amount greater than the total available balance in the account.  
     16. Verify the cash withdrawal functionality by entering an amount greater than per day limit  
     17. Verify the user is allowed to enter the amount again in case the amount entered is not valid. A proper message should be displayed.  
     18. Verify the ATM machine successfully takes out the money.  
     19. Verify the ATM machine takes out the balance printout after the withdrawal  
     20. Verify the font of the text displayed in ATM screen  
     21. Verify the text on the screen buttons visible clearly.  
     22. Verify the functionality of all the buttons on the keypad  
     23. Verify the text on the buttons visible clearly.  
     24. Verify that touch of the ATM screen is smooth and operational  
     25. Verify the user is allowed to choose different account types like Savings, Current etc.,  
     26. Verify the different combinations of operation and check if there will be an electricity loss in the middle of the operation. If there is an electricity loss in the middle of the transaction then the transaction should be marked as null and the amount shouldn’t be disclosed to others.  
     27. Verify the functionality of the cash dispenser  
     28. Verify the functionality of the receipt printer  
     29. Verify whether the printed data is correct or not in the receipt  
     30. Verify how much time the system takes to log out.

**Write a scenario of Microwave Owen**

1. Verify that the dimensions of the oven are as per the specification provided.
2. Verify that the oven’s material is optimal for its use as an oven and as per the specification.
3. Verify that the oven heats the food at the desired temperature properly.
4. Verify that oven heats food at the desired temperature within a specified time duration.
5. Verify the ovens functioning with maximum attainable temperature.
6. Verify the ovens functioning with minimum attainable temperature.
7. Verify that the oven’s plate rotation is speed is optimal and not too high to spill the food kept over it.
8. Verify that the oven’s door gets closed properly.
9. Verify that the oven’s door opens smoothly.
10. Verify the battery requirement of the microwave oven and check that it function’s smoothly at that power.
11. Verify that the text written over the oven’s body is clearly readable.
12. Verify that the digital display is clearly visible and functions correctly.
13. Verify that the temperature regulator is smooth to operate.
14. Verify that the temperature regulator works correctly.
15. Check the maximum capacity of the oven and test its functioning with that volume of food.
16. Check oven’s functionality with different kinds of food – solid, liquid.
17. Check the oven’s functionality with different food at different temperatures.
18. Verify the oven’s functionality with different kinds of container material.
19. Verify that the power cord of the oven is long enough.
20. Verify that the usage instruction or user manuals have clear instructions

**Write a scenario of Coffee vending Machine**

1. UI scenario – Verify that the dimension of the coffee machine is as per the specification
2. Verify that outer body, as well as inner part’s material, is as per the specification
3. Verify that the machine’s body color as well brand is correctly visible and as per specification
4. Verify the input mechanism for coffee ingredients-milk, water, coffee beans/powder, etc
5. Verify that the quantity of hot water, milk, coffee powder per serving is correct
6. Verify the power/voltage requirements of the machine
7. 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.
8. Verify that coffee should not leak when not in operation
9. Verify the amount of coffee served in single-serving is as per specification
10. Verify that the digital display displays correct information
11. Check if the machine can be switched on and off using the power buttons
12. Check for the indicator lights when the machine is switched on-off
13. Verify that the functioning of all the buttons work properly when pressed
14. Verify that each button has an image/text with it, indicating the task it performs
15. Verify that complete quantity of coffee should get poured in a single operation, no residual coffee should be present in the nozzle
16. Verify the mechanism to clean the system work correctly- foamer
17. Verify that the coffee served has the same and correct temperature each time it is served by the machine
18. Verify that system should display an error when it runs out of ingredients
19. Verify that pressing the coffee button multiple times leads to multiple serving of coffee
20. Verify that there is the passage for residual/extra coffee in the machine
21. Verify that machine should work correctly in different climatic, moistures and temperature conditions
22. Verify that machine should not make too much sound when in operation
23. Performance test – Check the amount of time the machine takes to serve a single serving of coffee
24. Performance test – Check the performance of the machine when used continuously until the ingredients run out of the requirements
25. Negative Test – Check the functioning of the coffee machine when two/multiple buttons are pressed simultaneously
26. Negative Test – Check the functioning of coffee machine with a lesser or higher voltage than required
27. Negative Test – Check the functioning of the coffee machine if the ingredient container’s capacity is exceeded

** Write a scenario of chair**

1. Verify that the chair is stable enough to take an average human load
2. Check the material used in making the chair-wood, plastic etc
3. Check if the chair’s leg are level to the floor
4. Check the usability of the chair as an office chair, normal household chair
5. Check if there is back support in the chair
6. Check if there is support for hands in the chair
7. Verify the paint’s type and color
8. Verify if the chair’s material is brittle or not
9. Check if cushion is provided with chair or not
10. Check the condition when washed with water or effect of water on chair
11. Verify that the dimension of chair is as per the specifications
12. Verify that the weight of the chair is as per the specifications
13. Check the height of the chair’s seat from floor

**Write a Scenario of Wrist Watch**

1. Verify the type of watch – analog or digital.
2. In the case of an analog watch, check the correctness time displayed by the second, minute, and hour hand of the watch.
3. In the case of a digital watch, check the digital display for hours, minutes, and seconds is correctly displayed.
4. Verify the material of the watch and its strap.
5. Check if the shape of the dial is as per specification.
6. Verify the dimension of the watch is as per the specification.
7. Verify the weight of the watch.
8. Check if the watch is waterproof or not.
9. Verify that the numbers in the dial are clearly visible or not.
10. Check if the watch is having a date and day display or not.
11. Verify the color of the text displayed in the watch – time, day, date, and other information.
12. 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.
13. Check if the second hand of the watch makes ticking sound or not.
14. Verify if the brand of the watch and check if its visible in the dial.
15. Check if the clock is having stopwatch, timers, and alarm functionality or not.
16. In the case of a digital watch, verify the format of the watch 12 hours or 24 hours.
17. Verify if the watch comes with any guarantee or warranty.
18. Verify if the dial has glass covering or plastic, check if the material is breakable or not.
19. Verify if the dial’s glass/plastic is resistant to minor scratches or not.
20. Check the battery requirement of the watch.

** Write a Scenario of Lift(Elevator)**

1. Verify the dimensions of the lift
2. Verify the type of door of the lift is as per the specification
3. Verify the type of metal used in the lift interior and exterior
4. Verify the capacity of the lift in terms of the total weight
5. Verify the buttons in the lift to close and open the door and numbers as per the number of floors
6. Verify that lift moves to the particular floor as the button of the floor is clicked
7. Verify that lift stops when up/down buttons at particular floor are pressed
8. Verify if there is an emergency button to contact officials in case of any mishap
9. Verify the performance of the floor – the time is taken to go to a floor
10. Verify that in case of power failure, lift doesn’t free-fall and get halted in the particular floor
11. Verify lifts working in case button to open the door is pressed before reaching the destination floor
12. Verify that in case door is about to close and an object is placed between the doors if the doors sense the object and again open or not
13. Verify the time duration for which door remain open by default
14. Verify if lift interior is having proper air ventilation
15. Verify lighting in the lift
16. Verify that at no point lifts door should open while in motion
17. Verify that in case of power loss, there should be a backup mechanism to safely get into a floor or a backup power supply
18. Verify that in case multiple floor number button is clicked, lift should stop at each floor
19. Verify that in case of capacity limit is reached users are prompted with warning alert- audio/visual
20. Verify that inside lift user are prompted with current floor and direction information the lift is moving towards- audio/visual prompt

** Write a Scenario of Whatsapp payment**

1. Verify the whatsapp open properly or not
2. Verify the payment option should be open or when we go to right side on whatsapp and tick on the three dots
3. Verify the show the payment page
4. Verify the security protect functionality should be work properly or not
5. Verify the add payment method functionality should open
6. Check the all bank name show into the page
7. Verify that the we can select the multiple bank in a whatsapp payment
8. Check the verification should properly or we get the otp in verify the mobile number
9. Verify that the we can see the transaction history
10. Verify the QR code generate properly or not
11. Verify that how many time get for transaction payment
12. Verify that the internet speed is affect on payment time

** Write a Scenario of instagram ( video call with chat )**

1. Verify that the Instagram application is install or not
2. Verify that the camera should be available on the mobile phone
3. Verify that the Internet connection should be on both person
4. Check the voice clarity during videocall
5. Verify that the user can send the massage when video call is on
6. Verify that the picture clarity during the video call
7. Verify that the front camera and top camera available or not
8. Verify that the video call with microphone is working or not
9. Verify that the group video call is available or not
10. Verify that the user can recorded video call or not
11. Verify that the user can send video during the videocall
12. If you tab the message when the video call is on, the video call stops
13. Verify that use can invite people during the video call
14. Check the how many user get authority of add other person
15. Check the minimum user get admin authority.

** Write a Scenario of whatsapp Group (generate group)**

1. Check whether the user can create a new one or note
2. Check the user can add multiple contacts from the contact list
3. Verify the user can insert the group name and select an image for DP
4. Check the user can add and remove contacts from the group
5. Check the user is able to delete group
6. Check the user can send and receive text message in the group
7. Check the user can send and documents in the group chat box
8. Check the user can send and receive photos in the group chat box
9. Check the user can send and receive videos in the group chat box
10. Check the user can send and receive emotion icons in the group chat box
11. Check the user can send and receive contact
12. Check the user can send and receive the live location
13. Check the user can send and receive GIFs
14. Check the user can delete text, video, audio, location and documents
15. Check the user can send the recorded voice
16. Check the user is able to invite or add multiple video call
17. Verify the user can see the group contact information from group info in the chat box
18. Check the user is able to search specific chat history using the search option in the group chat box
19. Check the user is able to mute the group in the group chat box
20. Check the users have options like Report, block, clear chat, export chat and add shortcut
21. Check the how many user get authority of add other person
22. Check the minimum user get admin authority.

** To Create Scenario (Positive & Negative)**

** 2. facebook Chat on Mobile**

1. Verify that the massager app install in mobile
2. Verify that the successfully login on messenger
3. Check the received messages count should be displayed on the Facebook Messages icon
4. Check the user gets all received messages in his inbox
5. Check that only message contact will display on the left hand side of the message box
6. Check the Active users display with a green got in the message box
7. Check the unread messages are highlighted so that the user can identify it
8. Check the user can send or recived message from massager app
9. Check the user can search contact in the message box
10. Check the user can delete the message or not
11. Check the user can show the all contact profile pic in massager app
12. Check the user can send or received text, picture ,documents, videos, audio so on
13. Check the user can call audio or video
14. Check the user should get message after uploading an images or file of an unsupported type
15. Check that the user is able to send messages to other offline users
16. Check the user can open more account in massager app
17. Check the user should successfully logout or not

**Gmail(Receiving mail)**

|  |
| --- |
| 1. Check that the user can receive email are correctly displayed or not 2. Check that the recently received unread emails is highlighted and bold in the Inbox section 3. Check the user can get the notification of receiving mail 4. Check the user should open the receiving mail properly or not 5. Check the user can get the information ID who receving mail 6. Check the attached documents of the email are download or not 7. Check the already read emails should not be the highlight 8. The number of unread email counts should be displayed beside the inbox test box 9. Check if the count is increased asper the number of new emails as unread 10. Check the name are visible to all the user whose names are present in CC and To section 11. Check those name and emails are present in the BCC section and should not display to other 12. Check that the you can receive emails from other domains like yahoo, skype, twiter |

**Online shopping to buy product (flipkart)**

1.Verify that user can search easily as per your choice

1. Verify that user get proper product displayed which they search on the flipcart
2. Verify that the user can all information about the product
3. Verify that the user can easily add or not in add to cart
4. Verify that the user can show the all product which they add in to the card with details
5. Verify that the user can easily buy the one product from cart
6. Verify that the user should logon successfully
7. Verify that the user get more option for payment
8. Verify that the user get product order details
9. Verify that the user get the product into the delivery time
10. Check the return policy is available or not for the product
11. Verify that user get product on right address
12. Verify that the user can see the review of the product