Software Testing

Module–2(Manual Testing)

1. **What is Exploratory Testing?**
   - Exploratory testing is a software testing method that involves simultaneously learning, designing, and executing tests without using predefined test cases.

2. **What is traceability matrix?**
   - A **traceability matrix** is a tool used in software testing and project management to ensure that all requirements are covered by test cases and to track the progress and relationships between requirements, test cases, and other project artifact.

3. **What is Boundary value testing?**
   - Boundary value testing is a black-box testing method that identifies errors in a software program by testing the boundaries of the input domain.

4. **What is Equivalence partitioning testing?**
   - Aim is to treat groups of inputs as equivalent and to select one representative input to test them all
   - EP can be used for all Levels of Testing

5. **What is Integration testing?**
   - Integration Testing - Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems
   - Integration Testing is a level of the software testing process where individual units are combined and tested as a group.

6. **What determines the level of risk?**
   - The level of risk in software testing is determined by assessing the likelihood of a risk occurring and the potential impact of its consequences. The risk level is used to prioritize testing efforts.

7. **What is Alpha testing?**
   - Alpha Testing is definitely performed and carried out at the developing organizations location with the involvement of developers.

8. **What is beta testing?**
   - Beta Testing (field testing) is performed and carried out by users or you can say people at their own locations and site using customer data. It is only a kind of Black Box Testing.
   - Beta Testing is always performed at the time when software product and project are marketed.

9. What is component testing?

- Component(Unit) – A minimal software item that can be tested in isolation. It means “A unit is the smallest testable part of software.”
- Component Testing – The testing of individual software components.

10.What is functional system testing?

- Functional System Testing: A requirement that specifies a function that a system or system component must perform
- A Requirement may exist as a text document and/or a model

11. What is Non-Functional Testing?

- 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

12. What is GUI Testing?

- Graphical User Interface (GUI) testing is the process of testing the system’s GUI of the System under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons, and all types of bars – tool bar, menu bar, dialog boxes and windows etc.

13. What is Adhoc testing?

- The Error guessing is a technique where the experienced and good testers are encouraged to think of situations in which the software may not be able to cope.
- Adhoc testing is an informal testing type with an aim to break the system.

14. What is load testing?

- Load testing - Its a performance testing to check system behavior under load. Testing an application under heavy loads, such as testing of a web site under a range of loads to determine at what point the system’s response time degrades or fails.
- 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.

15. What is stress Testing?

- 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.

16. What is white box testing and list the types of white box testing?

- White Box Testing: Testing based on an analysis of the internal structure of the component or system.
- White box testing is also called glass testing or open box testing.
- White box testing is the detailed investigation of internal logic and structure of the code.

Types of Coverage

- Statement coverage
- Decision coverage
- Condition coverage

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

- Black-box testing: Testing, either functional or non-functional, without reference to the internal structure of the component or system.
- The testers have no knowledge of how the system or component is structured inside the box

Techniques of Black Box Testing

There are four specification-based or black-box technique:

- Equivalence partitioning
- Boundary value analysis
- Decision tables
- State transition testing
- Use-case Testing
- Other Black Box Testing (Syntax or Pattern Testing)

18. Mention what are the categories of defects?

- Defect is the variance from a desired product attribute (it can be a wrong, missing or extra data).
- It can be of two types –
- Defect from the product or a variance from customer/user expectations.
- It is a flaw in the software system and has no impact until it affects the user/customer and operational system.
- Defects can be categorized into different types basing on the core issues they address.
- **Data Quality/Database Defects:** Deals with improper handling of data in the database.
  Examples: Values not deleted/inserted into the database properly
  Critical Functionality Defects: The occurrence of these bugs hampers the crucial functionality of the application. Examples: - Exceptions
- **Functionality Defects:** These defects affect the functionality of the application. Examples: All JavaScript errors
- **Security Defects:** Application security defects generally involve improper handling of data sent from the user to the application. These defects are the most severe and given highest priority for a fix.
  Examples: Authentication: Accepting an invalid username/password
- **User Interface Defects:** As the name suggests, the bugs deal with
- problems related to UI are usually considered less severe.
  Examples: Improper error/warning/UI messages

19. Mention what bigbang testing is?

- Big Bang Testing is a type of integration testing approach where all the components or modules of a system are integrated at once, and the system is tested as a whole after all parts have been integrated. The idea behind this testing approach is that instead of integrating and testing parts of the system incrementally, the entire system is integrated and tested all at once

20. 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)

21. When should "Regression Testing" be performed?

- Regression Testing: Testing of a previously tested program following modification to ensure that defects 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.

22. What is 7 key principles? Explain in detail?

1. Testing shows presence of Defects
2. Exhaustive Testing is Impossible!
3. Early Testing
4. Defect Clustering
5. The Pesticide Paradox
6. Testing is Context Dependent
7. Absence of Errors Fallacy

1. Testing shows presence of 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.

2**. Exhaustive Testing is Impossible!**

- All combinations of inputs and preconditions is not possible

3. Early Testing

- Testing activities should start as early as possible in the development life cycle

4. Defect Clustering

- A small number of modules contain most of the defects discovered during pre-release testing, or are responsible for the most operational failures.
- Defects are not evenly spread in a system
- They are ‘clustered’
- 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.
- Testing identifies bugs, and programmers respond to fix them

6. Testing is Context Dependent

- Different kinds of sites are tested differently.
  For example
  - Safety – critical software is tested differently from an e-commerce site.

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.
- If we build a system and, in doing so, find and fix defects ....
  - It doesn't make it a good system

23. Difference between QA v/s QC v/s Tester

S.No.	Quality Assurance	Quality Control	Testing
1	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
2	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.
3	Process oriented activities.	Product oriented activities.	Product oriented activities.
4	Preventive activities.	It is a corrective process.	It is a preventive process
5	It is a subset of Software Test Life	It is a subset of Software Test Life	It is a subset of Software Test Life

24. 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
The objective of this testing is to verify "stability" of the system in order to the with more rigorous testing	The objective of the testing is to verify the "rationality" of the system in order proceed to proceed with more rigorous testing
This testing is performed by the developers or testers	Sanity testing is usually performed by testers
Smoke testing is a subset of Regression testing	Sanity testing is a subset of Acceptance testing
Smoke testing exercises the entire system from end to end	Sanity testing exercises only the particular component of the entire system
Smoke testing is like General Health Check Up	Sanity Testing is like specialized health Check Up

25. Difference between verification and Validation

Criteria	Verification	Validation
Definition	The process of evaluating work-products (not the actual final product) of a development phase to determine whether they meet the specified requirements for that phase.	The process of evaluating software during or at the end of the development process to determine whether it satisfies specified business requirements.
Objective	To ensure that the product is being built according to the requirements and design specifications. In other words, to ensure that work products meet their specified requirements.	To ensure that the product actually meets the user's needs, and that the specifications were correct in the first place. In other words, to demonstrate that the product fulfills its intended use when placed in its intended nvironment.
Question	Are we building the product right?	Are we building the right product?
Evaluation Item	Plans, Requirement Specs, Design Specs, Code, Test Cases	The actual product/software.
Activities	· Reviews · Walkthroughs · Inspections	Testing

26. Explain types of Performance testing.

Load testing

- 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

Stress testing

- Stress Testing is done to make sure that the system would not crash under crunch situations.
- Stress testing is also known as endurance testing.

Endurance testing

- To evaluate the system's behavior under a constant load over an extended period.
- It checks for memory leaks, resource depletion, or any performance degradation over time. The goal is to see how the system holds up when used for prolonged periods, often days or weeks.

Spike testing

- To determine how the system reacts to sudden, extreme increases in load.
- It simulates a sudden spike in traffic, often much higher than typical usage, to evaluate how well the system recovers or handles rapid increases in load.

Volume testing

- To evaluate the system's performance when handling a large volume of data.
- It checks how the system handles large datasets, such as databases or file systems, to ensure performance remains optimal.

Scalability testing

- To determine how the system can scale up or down to accommodate changes in load.
- It evaluates the system's ability to handle growth in terms of traffic, users, or data volume. Scalability testing is important for systems that expect future g

27. 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 requirement then it is failure.

28. Difference between Priority and Severity

	Priority	**Severity**
Definition	Urgency of fixing the defect based on business needs	Impact of the defect on the system
Focus	Business and user perspective on how soon the issue needs resolution	Technical impact on functionality or performance
Measurement	Measures how soon the defect should be fixed	Measures the seriousness of the defect
Example	Urgent bug affecting payment process (High priority) or minor visual issue (Low priority)	System crash (Critical severity) or UI glitch (Minor severity
Assigned by	Project managers or stakeholders (based on release schedules and business needs)	Testers or developers (based on technical analysis)
Impacts	Determines how soon the defect needs attention to avoid delays in project delivery	Determines the level of impact on the system's operation

29. 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."

30. Explain the difference between Functional testing and Nonfunctional testing

Functional testing	Non-functional testing
Functional testing is performed using the functional specification provided by the client and verifies the system against the functional requirements.	Non-Functional testing checks the Performance, reliability, scalability and other non-functional aspects of the software system.
Functional testing is executed first	Non functional testing should be performed after functional testing
Manual testing or automation tools can be used for functional testing	Using tools will be effective for this testing

Business requirements are the inputs to functional testing	Performance parameters like speed , scalability are inputs to non-functional testing.
Functional testing describes what the product does	Nonfunctional testing describes how good the product works
Easy to do manual testing	Tough to do manual testing
Types of Functional testing are ➢ Unit Testing ➢ Smoke Testing ➢ Sanity Testing ➢ Integration Testing ➢ White box testing ➢ Black Box testing ➢ User Acceptance testing ➢ Regression Testing	Types of Nonfunctional testing are • Performance Testing • Load Testing • Volume Testing • Stress Testing • Security Testing • Installation Testing • Penetration Testing • Compatibility Testing • · Migration Testing

31…To create HLR & TestCase of

1)(Instagram, Facebook) only first page

2) Facebook Login Page: https://www.facebook.com/

https://github.com/Lalit693/Software-Testing

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

	SDLC (Software Development Life Cycle)	STLC (Software Testing Life Cycle)
Definition	A framework for developing software from start to finish.	A process for testing and validating the software.
Focus	Entire development process, including design, coding, and deployment.	Focuses on the testing phase of the software.
Phases	Requirement gathering, design, coding, testing, deployment, maintenance.	Requirement analysis, test planning, test design, execution, defect reporting, closure.
Involved Teams	Developers, designers, testers, business analysts, etc.	Primarily testers, but also works with developers.
Goal	Deliver a working software system.	Ensure software quality and functionality.
Duration	Longer, covers all stages of software development	Shorter, only during the testing phase.

33.What is the difference between test scenarios, test cases, and test script?

Test Scenario

➢ A Scenario is any functionality that can be tested. It is also called

Test Condition, or Test Possibility.

- Test Scenario is 'What to be tested'
- Test scenario is nothing but test procedure.
- The scenarios are derived from use cases.
- Test Scenario represents a series of actions that are associated together.
- Scenario is thread of operations

Test Case

- Test cases involve the set of steps, conditions and inputs which can be used while performing the testing tasks.
- Test Case is 'How to be tested'
- Test case consist of set of input values, execution precondition, expected Results and executed post-condition developed to cover certain test Condition.
- Test cases are derived (or written) from test scenario.
- Test Case represents a single (low level) action by the user.
- Test cases are set of input and output given to the System.

Test Script

- A set of sequential instruction that detail how to execute a core business function
- One script is written to explain how to simulate each business scenario
- Written to a level of detail for which someone else (other than the script writer) would be able to easily execute
- Identifies the test condition that is being satisfied for each step, if applicable
- Identified the input/test data that should be entered for each transaction
- Identifies the expected results for each step, if applicable
- Should demonstrate how the system can support the HCA warehouse business processes

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

- A Test Plan is a detailed document that outlines the strategy, approach, resources, schedule, and scope of testing activities for a specific project or product. It serves as a roadmap to guide the testing process and ensures that testing is thorough, organized, and aligned with the overall project objectives. The goal of a test plan is to ensure that the system or application being developed meets the specified requirements and quality standards.

Key Information That Should Be Covered in a Test Plan:

- Test Plan Identifier:
- Test Objectives:
- Scope of Testing:
- Test Approach:
- Test Items:
- Test Deliverables:
- Test Environment:
- Test Schedule:
- Resource Planning:
- Roles and Responsibilities:
- Entry and Exit Criteria:
- Test Case Design and Execution:

- Risk Management:
- Defect Management:
- Approval and Sign-off:

36. What is priority?

- Priority in the context of software testing refers to the urgency or importance of addressing a defect, issue, or task. It determines the order in which defects should be fixed or features should be tested based on their impact on the project, users, or business operations. Priority is typically assigned by the product manager, project manager, or the client, as they are more familiar with the business and functional implications.

37. What is severity?

- severity in the context of software testing refers to the impact or seriousness of a defect or bug on the system, application, or user experience. It defines how much the issue affects the functionality, performance, or usability of the system. Severity is typically assessed based on the defect's effect on the software's behavior and how critical it is for the application's operation or the end user.

38.Bug categories are…

- Bug categories are classifications used to organize and describe different types of defects or issues found in software during testing. These categories help testers, developers, and project managers understand the nature of the bug, its impact, and the area of the application it affects. Bugs are typically categorized based on their cause, behavior, or the area of the system they impact.

39. Advantage of Bugzila .

1. Open Source and Free
2. Customizable
3. Robust Search Capabilities
4. Advanced Reporting and Analytics
5. Email Notifications and Alerts
6. Integration with Other Tools
7. Access Control and Permissions
8. User-Friendly Interface
9. Scalability
10. Security
11. Multi-Language Support
12. Community Support

40.What are the different Methodologies in Agile Development Model?

- Scrum: SCRUM is an agile development method which concentrates particularly on how to manage tasks within a team based development environment.
- Kanban is a very popular framework for development in the agile software development methodology.

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

	Authentication	Authorization
Purpose	Verifying the identity of a user/system	Determines what actions or resources the user can access
Process	Requires credentials like username/password, biometrics.	Grants access based on roles/permissions after authentication.
Order	Happens first in the access control process.	Happens after authentication, based on user roles.
Focus	Confirms "Who are you?"	Determines "What are you allowed to do?"
Example	A user logs in using a username and password.	After login, a user with admin privileges can edit content while a regular user can only view it.

Web testing is critical for ensuring that web applications are functional, secure, and user-friendly. However, several challenges can arise during this process:

42.To create HLR & Test Case of WebBased (WhatsApp web , Instagram) 1. WhatsApp Web : https://web.whatsapp.com/

https://github.com/Lalit693/Software-Testing

43.Write a scenario of only WhatsApp chat messages

- Verify that on downloading the WhatsApp application, users can register using a new mobile number.
- Verify that for a new mobile number user will get a verification code on his mobile and filling in the same verifies the new user account.
- Check the maximum number of incorrect attempts allowed while filling out the verification code.
- Verify that registering an existing mobile number for new user account registration is not allowed.
- Verify that on successful registration all the contacts in the user's contact directory get imported to the Whatsapp contact list.
- Verify that the user can set DP and status on Whatsapp.
- Verify that the user can update the existing DP and Whatsapp status.
- Verify that the user can send messages to any individual selected from his contact list.
- Verify that 'Chats' window contains all the chat list with DP and name and last message preview of the other person with whom chat was initiated.
- Verify that clicking a chat in the chat list opens a new window containing all the chats received and sent with the other person.
- Verify that the user can check the message delivered and read the time for a message in the 'Message Info' section.
- Verify that the user can share or receive contact with the other person.
- Verify that the user can create a group by adding multiple people from his contact list.
- Verify that the user can send and receive the message in group chats.
- Verify that users can send and receive images, audio, video, and emoticons in the chat with individuals.
- Verify that users can send and receive images, audio, video, and emoticons in group chats.

- Verify that the user can send and receive chats in the secondary languages available.
- Verify that users can delete text, images, audio, and video messages within a chat.
- Verify that users can clear their complete chat history in an individual or group chat.
- Verify that users can archive chats in an individual or group chat.
- Verify that users can block a user to prevent any message from getting received from the blocked contact.
- Verify that the user makes WhatsApp calls to the person in his contact list.
- Verify that the user can receive WhatsApp calls from the person in his contact list.
- Verify that users can mark chats as favorites and access all chats marked as favorites from the 'Favorites' section.

44.Write a Scenario of Pen

- Verify the type of pen, whether it is a ballpoint pen, ink pen, or gel pen.
- Verify that the user is able to write clearly over different types of papers.
- 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.
- Verify if the pen is with a cap or without a cap.
- Verify the color of the ink on the pen.
- Check the odor of the pen's ink on writing over a surface.
- Verify the surfaces over which the pen is able to write smoothly apart from paper e.g. cardboard, rubber surface, etc.
- Verify that the text written by the pen should have consistent ink flow without leaving any blob.
- Check that the pen's ink should not leak in case it is tilted upside down.
- Verify if the pen's ink should not leak at higher altitudes.
- Verify if the text written by the pen is erasable or not.
- Check the functioning of the pen by applying normal pressure during writing.
- Verify the strength of the pen's outer body. It should not be easily breakable.
- Verify that text written by pen should not get faded before a certain time as mentioned in the specification.
- Check if the text written by the pen is waterproof or not.
- Verify that the user is able to write normally by tilting the pen at a certain angle instead of keeping it straight while writing.
- Check the grip of the pen, and whether it provides adequate friction for the user to comfortably grip the pen.
- Verify if the pen can support multiple refills or not.
- In the case of an ink pen, verify that the user is able to refill the pen with all the supported ink types.
- For ink pens, verify that the mechanism to refill the pen is easy to operate.
- In the case of a ballpoint pen, verify the size of the tip.
- In the case of a ball and gel pen, verify that the user can change the refill of the pen easily.

45. Write a Scenario of Pen Stand

- Verify pen stand is reusable o Verify that pen stand have proper structure
- o Verify the different material of pen stand
- Verify that for different types of pen
- Verify the usability of pen stand in any weather
- Verify the transportability of pen stand
- Verify that it can stand on any type of surface

- Verify that pen stand have different compartment
- Verify that it can contain other things small size diary

46. Write a Scenario of Door

- Verify if the door is single door or bi-folded door.
- Check if the door opens inwards or outwards.
- Verify that the dimension of the doors are as per the specifications.
- Verify that the material used in the door body and its parts is as per the specifications.
- Verify that color of the door is as specified.
- Verify if the door is sliding door or rotating door.
- Check the position, quality and strength of hinges.
- Check the type of locks in the door.
- Check the number of locks in the door interior side or exterior side.
- Verify if the door is having peek-hole or not.
- Verify if the door is having stopper or not.
- Verify if the door closes automatically or not – spring mechanism.
- Verify if the door makes noise when opened or closed.
- Check the door condition when used extensively with water.
- Check the door condition in different climatic conditions- temperature, humidity etc.
- Check the amount of force- pull or push required to open or close the door.

47.Write a Scenario of ATM

- Verify the type of ATM machine, if it has a touch screen, both keypad buttons only, or both.
- Verify that on properly inserting a valid card different banking options appear on the screen.
- Check that no option to continue and enter credentials is displayed to the user when the card is inserted incorrectly.
- Verify that the touch of the ATM screen is smooth and operational.
- Verify that the user is presented with the option to choose a language for further operations.
- Check that the user is asked to enter a pin number before displaying any card/bank account detail.
- Verify that there is a limited number of attempts up to which the user is allowed to enter the pin code.
- Verify that if the total number of incorrect pin attempts gets surpassed then the user is not allowed to continue further. And operations like temporary blocking of the card, etc get initiated.
- Check that the pin is displayed in masked form when entered.
- Verify that the user is presented with different account type options like- saving, current, etc.
- Verify that the user is allowed to get account details like available balance.
- Check that the correct amount of money gets withdrawn as entered by the user for cash withdrawal.
- Verify that the user is only allowed to enter the amount in multiple denominations as per the specifications.
- Verify that the user is prompted to enter the amount again in case the amount entered is less than the minimum amount configured.
- Check that the user cannot withdraw more amount than the total available balance and a proper message should be displayed.
- Verify that the user is provided the option to get the transaction details in printed form.
- Verify that the user’s session timeout is maintained.

- Check that the user is not allowed to exceed one transaction limit amount.
- Verify that the user is not allowed to exceed the one-day transaction limit amount.
- Verify that the user is allowed to do only one transaction per pin request.
- Check that in case the ATM machine runs out of money, a proper message is displayed to the user.
- Verify that the applicable fee gets deducted along with the withdrawn amount in case the user exceeds the limit of the number of free transactions in a month.
- Verify that the applicable fee gets deducted along with the withdrawn amount in case the user uses a card of a bank other than that of an ATM.
- Check that the user is not allowed to proceed with the expired ATM card and that a proper error message gets displayed.
- Verify that in case of sudden electricity loss before withdrawing cash, the transaction is marked as null and the amount is not withdrawn from the user's account.

48. When to used Usablity Testing?

- Usability Testing identifies usability errors in the system early in development cycle and can save a product from failure

49 What is the procedure for GUI Testing?

- Graphical User Interface (GUI) testing is the process of testing the system's
- GUI of the System under Test. GUI testing involves checking the screens
- with the controls like menus, buttons, icons, and all types of bars – tool bar,
- menu bar, dialog boxes and windows etc.

50. Write a scenario of Microwave Owen

- 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 oven's plate rotation speed is optimal and not too high to spill the food kept over it.
- Verify that the oven's door gets closed properly.
- Verify that the oven's door opens smoothly.
- Verify the battery requirement of the microwave oven and check that it function's smoothly at that power.
- 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.
- Check the oven's functionality with different food at different temperatures.
- Verify the oven's functionality with different kinds of container material.
- Verify that the power cord of the oven is long enough.
- Verify that the usage instruction or user manuals have clear instructions.

51. Write a scenario of Coffee vending Machine

- UI scenario – Verify that the dimension of the coffee machine is as per the specification.
- 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.
- Verify the amount of coffee served in single-serving is as per specification.
- Verify that the digital display displays correct information.
- Check if the machine can be switched on and off using the power buttons.
- 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.
- Verify that complete quantity of coffee should get poured in a single operation, no residual coffee should be present in the nozzle.
- Verify the mechanism to clean the system work correctly- foamed.
- Verify that the coffee served has the same and correct temperature each time it is served by the machine.
- Verify that system should display an error when it runs out of ingredients.
- Verify that pressing the coffee button multiple times leads to multiple serving of coffee.
- Verify that there is the passage for residual/extra coffee in the machine.
- Verify that machine should work correctly in different climatic, moistures and temperature conditions.
- Verify that machine should not make too much sound when in operation.
- Performance test – Check the amount of time the machine takes to serve a single serving of coffee.
- Performance test – Check the performance of the machine when used continuously until the ingredients run out of the requirements.
- Negative Test – Check the functioning of the coffee machine when two/multiple buttons are pressed simultaneously.
- Negative Test – Check the functioning of coffee machine with a lesser or higher voltage than required.
- Negative Test – Check the functioning of the coffee machine if the ingredient container's capacity is exceeded.

52.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.
- Check the height of the chair's seat from floor.

53.To Create Scenario (Positive & Negative)

1.Email

- Check if all the elements of the received email are correctly displayed or not.
- Check whether the user clicks on the new email; it redirects the user to the email content.
- Check if the email content is displaying in the proper format or not.
- Check the attached documents of the email are downloadable.
- Check the already-read emails should not be the highlight.
- The number of unread email counts should be displayed beside the inbox text box.
- Check if the count is increased as per the number of new emails you are received.
- Check the count is increased when you mark an email as unread.
- Check after opening or make as read an email. The count should be decreased.
- Check the names are visible to all the users whose names are present in CC & To section.
- Check those names or emails are present in the BCC section and should not display to others.
- Check that Gmail open properly
- Check that it shows proper mail home page
- Check that mail list shows with oldest and newest message
- Check the scrolling functionality of mail list
- Check that it shows different category of mail(primary, social,spam)
- Verify that it shows sender details
- Check the compatibility of receive mail functionality
- Check the receiving mail functionality without internet
- Check the movement functionality of mail on same page
- Check that it shows option of search data from sender

2.Online shopping to buy product (flipkart)

- Check that open website properly
- Check that product is available on flip kart
- Check that it shows proper page of product
- Check that it shows price of product properly
- Check the available offers option
- Check the terms & condition option of offers
- Check that shows proper product details
- Check that it shows product specifications properly
- Check the product image option is working properly
- Check buy now option of product is working properly
- Check that it shows pincode for delivery option of product
- Check that on the product page, and a user can select the desired attribute of the product, e.g., size, color, etc.
- Check that user can add to the cart one or more products.
- Check that user can add products to the wish list.
- Check that users can buy products added to the cart after signing in to the
- application (or as per the website's functionality).
- Check that user can successfully buy more than one products that were added to his/her cart.

- Check that the limit to the number of products a user can buy is working correctly by displaying an error message and preventing the user from buying more than the threshold.
- Check the availability of products at desired locations.
- Check that the Cash on the Delivery option of payment is working fine.
- Verify that the different pre-paid methods of payment are working fine.
- Check that the product return functionality works fine.
- Check that the Cancel Order option is present.
- Check the buy products functionality without internet
- Check that product price is available in different currecy
- Check that all offers can apply at a same time
- Check that it can verify multi pincode at a same time
- Check the availability of delivery out of country option

54.Write a Scenario of 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.
- Check if the second hand of the watch makes ticking sound or not.
- Verify if the brand of the watch and check if its visible in the dial.
- Check if the clock is having stopwatch, timers, and alarm functionality or not.
- In the case of a digital watch, verify the format of the watch 12 hours or 24 hours.
- Verify if the watch comes with any guarantee or warranty.
- Verify if the dial has glass covering or plastic, check if the material is breakable or not.
- Verify if the dial's glass/plastic is resistant to minor scratches or not.
- Check the battery requirement of the watch.

55.Write a Scenario of Lift(Elevator)

- Verify the dimensions of the lift.
- Verify the type of door of the lift is as per the specification.
- Verify the type of metal used in the lift interior and exterior.
- Verify the capacity of the lift 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 the lift moves to the particular floor as the button of the floor is clicked.
- Verify that the lift stops when the 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 in case of power failure, the lift doesn't free-fall and gets halted on the particular floor.
- Verify lifts working in case the button to open the door 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 if the doors sense the object and again open or not.
- Verify the time duration for which the door remains open by default.
- Verify if the lift interior is having proper air ventilation.
- Verify lighting in the lift.
- Verify that at no point the lift door should open while in motion.
- Verify that in case of power loss, there should be a backup mechanism to safely get into a floor or a backup power supply.
- Verify that in case the multiple floor number button is clicked, the lift should stop on each floor.
- Verify that in case of capacity limit is reached users are prompted with a warning alert- audio/visual.
- Verify that inside lift users are prompted with the current floor and direction information the lift is moving towards- audio/visual prompt.

56. Write a Scenario of WhatsApp Group (generate group)

- Check whether the user can create a new one or not.
- heck the user can add multiple contacts from the contact list.
- Verify the user can insert the group name and select an image for DP.
- Check the user can add and remove contacts from the group.
- Check that whatsapp setting and generate a new group
- Check that it shows new group option
- Verify the available of contact list
- Check that it shows search contact option
- Check that choose and add contact in new group
- Check its multi selection functionality of member
- Check functionality like set group name, group icon
- Verify the disappearing message functionality
- Check new group generate functionality without internet
- Verify the maximum member add limit in new group
- Verify functionality of adding unsaved member in new group
- Verify that delete all data in group admin

57.Write a Scenario of WhatsApp payment

- verify that user is able to see Scan code screen on mobile phone or not
- verify that QR code is scan from WhatsApp pay
- verify that user can get message for payment
- Check that it shows payment option
- Verify that it shows add bank account functionality
- Check the availability of bank list
- Check the search bank functionality in payment option
- Verify that we can send money to any valid UPI id
- Check that payment is securely done through UPI pin
- Check payment functionality without internet
- Verify the maximum transferrable limit of payment
- Check the multi payment functionality at a time
- Verify the availability of payment option in different currency