Software Testing Assignment

Module 2

1) What is Exploratory Testing?

- **Exploratory Testing** is a type of software testing in which the tester is free to select any possible methodology to test the software. Exploratory testing checks the functionality and operations of the software as well as identify the functional and technical faults in it. The aim of exploratory testing is to optimize and improve the software in every possible way.

2) What is Traceability matrix?

- **Traceability Matrix** is a document that is used in software development to trace requirements**.** It is a table that shows the relationship between requirements and other components or functionalities of the software. It can be used to verify that each requirement is being developed and tested, and to track any changes or defects.

3) What is Boundary value testing?

- **Boundary Value** testing is a software testing technique that involves testing the boundary values of valid and invalid partitions. It is used when it is impractical to test a large pool of test cases individually.
- **Example**: consider a system that accepts ages from 18 to 56. The valid boundary values are 18 and 56, while the invalid boundary values are 17 and 57.

4) What is Equivalence partitioning testing?

- **Equivalence Partitioning** also called as equivalence class partitioning. It is abbreviated as ECP. It is a software testing technique that divides the input test data of the application under test into each partition at least once of equivalent data from which test cases can be derived.

5) What is Integration testing?

- **Integration Testing** is a software testing technique that focuses on verifying the interactions and data exchange between different components or modules of a software application. Integration testing is typically performed after unit testing and before system testing . Any testing technique (Blackbox, Whitebox, and Grey box) can be used for Integration Testing.

6) What determines the level of risk?

- **Product Risk** = Product risk refers to the set of things that could go wrong with the service, software or whatever is being produced by the project. It is the chance that the product fails in relation to the expected damage if this occurs. The chance of failure is determined by the chance of defects and the frequency of use . In other words, product risk is a type of risk that occurs when a product (component) contains a defect.

- **Project Risk** = A risk is an uncertain event or condition that can have affect the outcome of a project. The term “project risk” refers to the sum of all individual project risks that might affect a project.

7) What is Alpha testing?

- **Alpha Testing** is a type of software testing performed to identify bugs before releasing the software product to the real users or public. It is a type of acceptance testing.The main objective of alpha testing is to refine the software product by finding and fixing the bugs that were not discovered through previous tests.

8) What is Beta Testing?

- **Beta testing** is a type of user acceptance testing where the software is made available to the intended customer base or a segment of real users to try out and provide feedback in a real environment.

9) What is Component Testing?

- **Component testing** is performed by testers. ‘Unit Testing’ is performed by the developers where they do the testing of the individual functionality or procedure. After Unit Testing is performed, the next testing is component testing. Component testing is done by the testers.

10) What is Functional system testing?

- **System Testing** = is a form of black-box testing that assesses the complete functionality and performance of a fully integrated software system. It is conducted after integration testing and before acceptance testing. System testing aims to identify any flaws in the integrated components of the system and to guarantee that the system complies with the specified requirements and operates as expected by the end users.

11) What is Non-Functional Testing?

- **Non-Functional Testing** is a type of software testing that checks the non-functional aspects of a software application, such as performance, usability, reliability, Stress Testing, Scalability Testing, Volume Testing , Security Testing.

12) What is GUI Testing?

- **GUI Testing** is a software testing type that checks the Graphical User Interface of the Software. The purpose of GUI Testing is to ensure the functionalities of software application work as per specifications by checking screens and controls like menus, buttons, icons, etc. In GUI testing, we check all the GUI elements for size, position, width, length, and acceptance of characters or numbers.

13) What is Adhoc Testing?

- **Adhoc Testing** is an informal and unstructured software testing technique that aims to break the testing process to discover potential flaws or faults at the earliest possible stage . It is usually an unplanned activity that does not follow any documentation or test design techniques to create test cases . Ad hoc testing is done randomly and does not follow any structured way of testing.

14) What is Load testing?

- **Load Testing** is a type of performance testing that simulates a real-world load on a system or application to see how it performs under stress. The goal of load testing is to identify bottlenecks and determine the maximum number of users or transactions the system can handle .
- **Tools** : Apache JMeter, LoadRunner, Gatling, and Grinder can be used to simulate load and measure system performance .

15) What is Stress Testing?

- **Stress Testing** is a software testing technique that determines the robustness of software by testing beyond the limits of normal operation.
- Stress testing is particularly important for critical software but is used for all types of software. Stress testing emphasizes robustness, availability, and error handling under a heavy load rather than what is correct behaviour under normal situations.

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

- **White Box Testing** is a software examining technique that involves testing the product's underlying structure, design, and coding in order to verify input-output flow and improve design, usability, and security.

Types of White Box Testing

- Unit Testing
- Testing for Memory Leaks
- White Box Penetration Testing
- White Box Mutation Testing

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

- **Black box** testing is a software testing technique where the internal workings or code structure of the system being tested are not known to the tester.
- The tester focuses solely on the external behaviour of the software, without having access to its internal source code.

Different Black Box Testing Techniques:

- Equivalence partitioning
- Boundary value analysis
- Testing based on a decision table
- State transition testing
- Use case based testing
- All-pair testing
- Cause-effect graph technique
- Error guessing technique
- User story technique
- Orthogonal array testing

18) Mention what are the categories of defects?

- **Minor Defects** mean that the item does not fully comply with product specifications but is still usable.
- **Major Defects** affect the core and major functionalities of a software product, but do not result in complete failure of the system.
- **Critical Defects** directly affect the critical and essential functionalities of a software product, and require immediate attention and treatment.

19) Mention what big bang testing ?

- **Big Bang** Testing is an approach of integration testing where integration of all or major components of the system are tested. The Big Bang method is very effective for saving time in the integration testing process. It helps amateur developers to find integration related defects earlier than actual integration testing.
- However, if the test cases and their results are not recorded properly, the entire integration process will be more complicated and may prevent the testing team from achieving the goal of integration testing.

20) What is the purpose of exit criteria?

- **Exit Criteria** are the defined requirements within software testing that must be met in order to determine that testing has been completed . They are used to verify that a requirement has been met and the service, product, or process can move to the next step. Exit criteria are used to report against and to plan when to stop testing .

21) When should "Regression Testing" be performed?

- When a new functionality is added to the system and the code has been modified to absorb and integrate that functionality with the existing code.
- When some defect has been identified in the software and the code is debugged to fix it.
- When the code is modified to optimize its working.
- Regression Testing should be performed when there are changes made to the software, such as adding new functionality, fixing defects, or optimizing code.

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

- Below is the list of 7 key principles

1. Testing shows the presence of defects.

We test software to discover issues, so that they can be fixed before they are deployed to live environments – this enables us to have confidence that our systems are working.

2. Exhaustive testing is impossible

you can then cover vast areas, while making sure you are testing the most important functions. With careful planning and assessment, your test coverage can remain excellent and enable that necessary confidence in your software, without requiring that you test every single line of code.

3. Early testing
Testing early is fundamentally important in the software lifecycle. This could even mean testing requirements before coding has started.

4. Defects cluster
This is the idea that certain components or modules of software usually contain the most number of issues, or are responsible for most operational failures.

5. The pesticide paradox
This is based on the theory that when you use pesticide repeatedly on crops, insects will eventually build up an immunity, rendering it ineffective.

6. Testing is context dependent
Testing is ALL about the context. The methods and types of testing carried out can completely depend on the context of the software or systems.

7. Absence-of-errors is a fallacy
If your software or system is unusable (or does not fulfill users' wishes) then it does not matter how many defects are found and fixed – it is still unusable.

23)Difference between QA v/s QC v/s Tester?

Quality Assurance (QA)	Quality Control (QC)	Testing
1) Process-oriented focuses on making the process of creating software better.	**1)** A product-oriented approach is a way to make sure the software meets all its requirements.	**1)**Testing the software system is about finding any mistakes or issues.
2) It works with the development process to help stop mistakes and ensure the software is of good quality.This means setting up and keeping standards, processes, procedures, and tools in place to ensure we're consistently producing high-quality software.	**2)** It's done after the development process and involves running test cases and seeing how the software reacts.	**2)** This usually happens after the software has been created, and it's all about ensuring that the software's quality is up to standard.
3) The goal is to keep improving our software development process for the best possible results.	**3)** The goal is to find any defects or errors in the software and fix them.	**3)** It involves running tests and looking at what comes out of them, finding any problems with the software, and ensuring that it does everything it's supposed to do.
4) Preventive activities	**4)** It is corrective process	**4)** It is preventive process
5) Process -Oriented activities	**5)** Product-Oriented activities	**5)** Product-Oriented activities

24) Difference between Smoke and Sanity?

SMOKE TESTING	SANITY TESTING
1) Smoke Testing is performed to ascertain that the critical function-lities of the program is working fine.	**1)** After receiving a software build, with minor change in code or functionality sanity testing is performed to ascertain that the bugs have been fixed and no further issues are introduced due to these changes.
2) The objectives of this testing is to verify “stability” of the system by performing testing.	**2)** The objectives of this testing is to verify the “rationality” of the system by performing testing.
3) Smoke Testing is performed by either developers or testers.	**3)** Sanity Testing is performed by testers alone.
4) Smoke Testing is the subset of acceptance testing.	**4)** Sanity Testing is the subset of regression testing.
5) Smoke Testing is usually documented or scripted.	**5)** Sanity Testing is usually not documented and is unscripted.
6) There is end to end system verification done in smoke testing	**6)** A specific component gets verified in sanity testing.
7) For every new build release smoke testing is carried out	**7)** Sanity Testing is carried out when in dept testing is not possible because of short time.

25) Difference between verification and Validation ?

VERFICATION	VALIDATION
1) Verification is the process which is performed at the development level	**1)** Validation is a process which is performed at the Testing level.
2) Verification phases are: ➢ Business/User Requirement ➢ System Requirement ➢ Technical Requirement ➢ Program Specification	**2)** Validation phases are: ➢ Unit Testing ➢ Integration Testing ➢ System Testing ➢ Acceptance Testing
3) Verification can be achieved by asking- **Are you building a product right**	**3)** Validation cab be achieved by asking- **Are you building a right products**
4) It is the process of evaluating product of development to check whether the specified requirement meet or not.	**4)** It is the process of evaluating the products of development to check whether it satisfied business requirement or not.
5) Verification activities are reviews and inspection.	**5)** Validation activities is Testing.

26) Explain types of Performance testing ?

- **Performance Testing** is a type of software testing that ensures software applications to perform properly under their expected workload. It is a testing technique carried out to determine system performance in terms of sensitivity, reactivity and stability under a particular workload.

Types of Performance Testing:

1. **Load Testing:**
   It checks the product's ability to perform under anticipated user loads. The objective is to identify performance congestion before the software product is launched in market.
2. **Stress Testing:**
   It involves testing a product under extreme workloads to see whether it handles high traffic or not. The objective is to identify the breaking point of a software product.
3. **Endurance Testing:**
   It is performed to ensure the software can handle the expected load over a long period of time.

4. **Spike Testing:**
   It tests the product's reaction to sudden large spikes in the load generated by users.
5. **Volume Testing:**
   In volume testing large number of data is saved in a database and the overall software system's behaviour is observed. The objective is to check product's performance under varying database volumes.
6. **Scalability Testing:**
   In scalability testing, software application's effectiveness is determined in scaling up to support an increase in user load. It helps in planning capacity addition to your software system.

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 than it is called bug.
- **Failure** = Build does not meet the requirements then it is called failure.

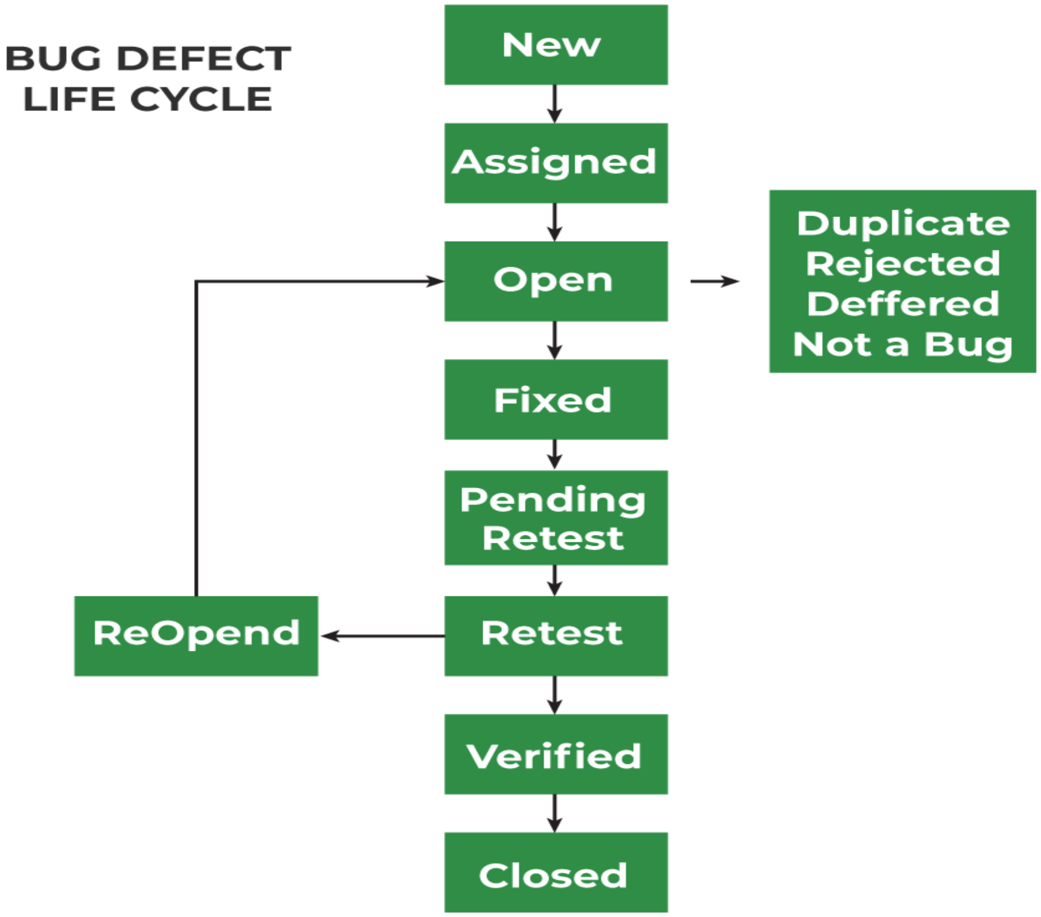
28) Difference between Priority and Severity ?

Priority	Severity
1) Defect Priority has defined the order in which the developer should resolve a defect	**1)** Defect Severity is defined as the degree of impact that a defect has on the operation of the product
2) Priority is associated with scheduling	**2)** Severity is associated with functionality or standards
3) Priority indicates how soon the bug should be fixed	**3)** Severity indicates the seriousness of the defect on the product functionality
4) Priority of defects is decided in consultation with the manager/client	**4)** QA engineer determines the severity level of the defect
5) Priority is driven by business value	**5)** Severity is driven by functionality
6) Its value is subjective and can change over a period of time depending on the change in the project situation	**6)** Its value is objective and less likely to change
7) High priority and low severity status indicates, defect have to be fixed on immediate bases but does not affect the application	**7)** High severity and low priority status indicates defect have to be fixed but not on immediate bases
8) Priority status is based on customer requirements	**8)** Severity status is based on the technical aspect of the product

Priority	Severity
9) During UAT the development team fix defects based on priority	**9)** During SIT, the development team will fix defects based on the severity and then priority
10) Priority is categorized into three types • Low • Medium • High	**10)** Severity is categorized into five types • Critical • Major • Moderate • Minor • Cosmetic

29) What is Bug Life Cycle?

- ➢ The bug life cycle or defect life cycle in testing is a process that involves different stages through which a defect progresses during its lifetime. It starts when a tester identifies a new defect while testing the application. The cycle continues until the tester finds a resolution and closes the bug to prevent its recurrence.



30) Explain the difference between Functional testing and Non Functional.

FUNCTIONAL TESTING	NON-FUNCTIONAL TESTING

- Functional Testing is the type of software testing and a black box testing whereby the system is tested against the functional specification provided by the clients.	- Non-Functional Testing is a types of software testing which is used to check performance, reliability, scalability and other non-functional aspects of the software system.
- The purpose of Functional testing is to test each function of the software application by providing appropriate input, verifying the output against the functional requirements.	- The primary purpose of non-functional Testing is to test the reading speed of the software system as per non-functional parameters.
- It is based on the business requirement	- It is based on the performance requirement
- Functional Testing is easy to execute manually	- It is hard to execute non-functional testing manually
- Functional Testing describes "what the product does".	- Non-functional testing describes how good the product works
- Functional Testing helps to enhance the behaviour of the application.	- Non-functional testing helps to enhance the performance of the application
- Unit testing, smoke & sanity testing, Integration testing, white & black box testing, User acceptance testing and Regression testing are the types of functional testing	- Performance testing, Load testing, volume testing, Stress & security testing, Installation testing, penetration testing and compatibility testing are the types of functional testing.

31) To create HLR & Test Case of

1. Instagram, Facebook only first page

HLR & TEST CASE of Instagram	**Click Here**
HLR & TEST CASE of Facebook	**Click Here**

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

HLR & TEST CASE of Facebook login page	**Click Here**

32) 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.
➢ 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.
➢ 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.
➢ It helps in developing good quality software.	➢ It helps in making the software defects free.
➢ SDLC phases are completed before the STLC phases.	➢ STLC phases are performed after SDLC phases.

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

➢ **Test Scenario**

i) A Test scenario is any functionality that cab be tested.
ii) Test scenario provides a small description, mostly one-line statements.
iii) It is also called Test condition or Test possibility
iv) It focus on "what to be tested".
v) Test scenario is derived from Use case and SRS.
vi) It requires fewer resources to write test scenario.

➢ **Test Cases**

i) Test case is a detail document that describes step by step process to execute a test.
ii) Test cases are more detailed with number of parameters
iii) It focus on "How to be tested".

iv) Test cases are derived from Test scenario.
v) It requires more resources for documentation and execution.

- **Test Script**
  i) A Test script is a set of instructions that is performed on a system under test to verify that the system performs as expected.
  ii) It is a line-by-line description that contain information about system functions that must be perform to verify an application or system under test.
  iii) It is used for automation testing that aims to validate the functionality of the software.

34) Explain what Test Plan is? What is the information that should be covered ?
Test Plan is a document that describes the strategy and objectives for a testing software product or system. It usually includes information such as the schedule, scope, resources, criteria and risks of the testing process. The Test plan serves as a blueprint to conduct software testing activities as a defined process, which is minutely monitored and controlled by the test manager. It guides the testing efforts and helps to find and resolve errors.

- **Test plan should cover the following information:**
  i) The roles and responsibilities of the test team and stakeholders.
  ii) The general timelines and schedules for testing activities
  iii) The levels and types of testing to be performed
  iv) The test coverage, methods and responsibilities
  v) The test environment and tools
  vi) The test criteria and metrics
  vii) The test deliverables and reports

35) What is priority?

- **Priority** are the aspects of software testing that measure the impact and urgency of a defect. It refers to how quickly the fault should be rectified and it is decided by the manager or client.
  Priority is basically a parameter that decides the order in which we should fix the defects.

36) What is Severity?
Severity are the aspects of software testing that measure the impact and urgency of a defect. It refers to how important the flow is to product’s functionality and it is determined by the QA Engineers.
Severity is basically a parameter that denotes the total impact of a given defect on any software.

37) Bug Categories ?

- **Low**: bugs that cause minor problems for the users. These bugs reduce the quality of experience, but the software is still functional

- **High**: bugs that limit the use of the software. The user gets limited in functional terms by the bug, but the software is still usable
- **Critical**: bugs that make software useless.

38) Advantage of Bugzila ?

- It is an open-source widely used bug tracker.
- It is easy to use and its user interface is understandable for people without technical knowledge.
- It easily integrates with test management instruments.
- It integrates with an e-mailing system.
- It automates documentation.
- It permits software and hardware issue monitoring and troubleshooting to be performed within a cloud-based workspace.
- It delivers powerful capabilities that can enhance how developers and IT professionals work and collaborate on bugs.
- It makes it possible for specific organizations to come up with high-quality software and hardware products.

39) What are the different Methodologies in Agile Development Model?

- **Agile Development Model** is a software development approach that emphasizes flexibility, speed, and customer satisfaction. It is different from the traditional waterfall approach, which delivers large products in long periods of time and is less responsive to changing requirements.

- **Different Methodologies in Agile Development Model**
  1) **Scrum**: Scrum is one of the most widely used Agile methodologies. It is a prescriptive framework that excels at managing iterative and incremental projects.
  2) **Kanban**: Kanban is a visual management tool that helps teams work more efficiently by visualizing work in progress.
  3) **Lean**: Lean is an Agile methodology that focuses on reducing waste and maximizing value.
  4) **Extreme Programming (XP)**: XP is an Agile methodology that emphasizes teamwork, communication, and simplicity

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

Authentication	Authorization
➢ In the authentication process, the identity of users are checked for providing the access to the system.	➢ While in authorization process, a the person's or user's authorities are checked for accessing the resources.
➢ In the authentication process, users or persons are verified.	➢ While in this process, users or persons are validated.
➢ It is done before the authorization process.	➢ While this process is done after the authentication process.
➢ It needs usually the user's login details.	➢ While it needs the user's privilege or security levels.
➢ Authentication determines whether the person is user or not.	➢ While it determines What permission does the user have?
➢ Generally, transmit information through an ID Token.	➢ Generally, transmit information through an Access Token.
➢ The OpenID Connect (OIDC) protocol is an authentication protocol that is generally in charge of user authentication process.	➢ The OAuth 2.0 protocol governs the overall system of user authorization process.

➢ **Common issues in web testing include** :

- Functional issues
- Problems that occur while navigating an application
- Usability issues like broken links, form fields missing default focus, tab key not working, and all keyboard shortcuts not fully functional
- Cross-browser compatibility
- Responsiveness

- Cross-device compatibility
- Integration testing

41) To create HLR & TestCase of WebBased (WhatsApp Web , Instagram Web)

HLR & TEST CASE WhatsApp web	Click Here
HLR & TEST CASE Instagram web	Click Here

42) To create HLR and Test Case on this Link. https://artoftesting.com/

HLR & TEST CASE Art of Testing	Click Here

43) Write a scenario of only WhatsApp chat messages?

1. Verify that the user can set a chat wallpaper.
2. Verify that the user sets privacy settings like turning on/off last seen, online status, read receipts, etc.
3. Verify that the user can update notification settings like – notification sound, on/off, and show preview for both group and individual chats.
4. Verify that the user can take the complete chat backup of his chats.
5. Verify that the user can update the phone number that is used by the WhatsApp application.
6. Verify that the user can disable/delete his WhatsApp account.
7. Verify that the user can check data usage by images, audio, video, and documents in WhatsApp chat

44) Write a Scenario of Pen ?

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 colour 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.
6. Verify the type of pen, whether it is a ballpoint pen, ink pen, or gel pen.
7. Verify that the user is able to write clearly over different types of papers.
8. 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.
9. Verify if the pen is with a cap or without a cap.
10. Verify the color of the ink on the pen.
11. Check the odor of the pen's ink on writing over a surface.

12.Verify the surfaces over which the pen is able to write smoothly apart from paper e.g. cardboard, rubber surface, etc.

13.Verify that the text written by the pen should have consistent ink flow without leaving any blob.

14.Check that the pen’s ink should not leak in case it is tilted upside down.
15.Verify if the pen’s ink should not leak at higher altitudes.
16.Verify if the text written by the pen is erasable or not.
17.Check the functioning of the pen by applying normal pressure during writing.
18.Verify the strength of the pen’s outer body. It should not be easily breakable.
19.Verify that text written by pen should not get faded before a certain time as mentioned in the specification.
20.Check if the text written by the pen is waterproof or not.

45) Write a Scenario of Pen Stand ?

1. Pen stand should be in different categories in term of colour, Design, materials, size etc
2. Different material should be used for making stand like wood, glass or plastic, fibre and any other material.
3. The Pen stand should not be easily breakable.
4. It should be washable or not affect by water after washing.
5. it should be of maximum quantity where you can put lots of pen in one stand.
6. It should be able to hold pen accurately.
7. it should be adjustable for all types of pens depend on its manufactured

46) 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 is 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

47) Write a Scenario of ATM?

1.Verify that all the labels and controls including text boxes, buttons, images, and links are present on the screen.

2.Check the informative text written displayed on the screen is clearly visible and legible.
3.Verify that the size, color, and UI of the different objects are as per the specifications.
4.Verify that the application's UI is responsive i.e. it should adjust to different screen resolutions of ATM machines.
5.Verify the type of ATM machine, if it has a touch screen, both keypad buttons only, or both.
6.Verify that on properly inserting a valid card different banking options appear on the screen.
7.Check that no option to continue and enter credentials is displayed to the user when the card is inserted incorrectly.
8.Verify that the touch of the ATM screen is smooth and operational.
9.Verify that the user is presented with the option to choose a language for further operations.
10.Check that the user is asked to enter a pin number before displaying any card/bank account detail.
11.Verify that there is a limited number of attempts up to which the user is allowed to enter the pin code.
12.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.
13.Check that the pin is displayed in masked form when entered.
14.Verify that the user is presented with different account type options like- saving, current, etc.
15.Verify that the user is allowed to get account details like available balance.
16.Check that the correct amount of money gets withdrawn as entered by the user for cash withdrawal.
17.Verify that the user is only allowed to enter the amount in multiple denominations as per the specifications.
18.Verify that the user is prompted to enter the amount again in case the amount entered is less than the minimum amount configured.
19.Check that the user cannot withdraw more amount than the total available balance and a proper message should be displayed.
20.Verify that the user is provided the option to get the transaction details in printed form.

48) When to used Usablity Testing ?

- **Usability testing** is a type of testing that is done from an end user's perspective to determine if the system is easily usable. It is generally the practice of testing how easy a design is to use on a group of representative users .

- Usability testing should be performed throughout the system and acceptance testing levels. It is recommended to implement usability testing in the early stage of the Software Development Life Cycle (SDLC) .

<u>The following are the phases involved in usability testing</u> :

1. Prepare your product or design to test

2. Choose a representative group of users
3. Create a list of tasks for the users to perform
4. Observe the users as they complete the tasks

49) What is the procedure for GUI Testing?

- **GUI Testing** is a software testing type that checks the Graphical User Interface of the Software. The purpose of GUI Testing is to ensure the functionalities of software application work as per specifications by checking screens and controls like menus, buttons, icons, etc[1]. GUI stands for Graphical User Interface where you interact with the computer using images rather than text . In GUI testing, we 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.

<u>Here are some examples of GUI testing test cases </u>:

- Testing the size, position, width, height of the elements.
- Testing of the error messages that are getting displayed.
- Testing the different sections of the screen.
- Testing of the font whether it is readable or not.
- Testing of the screen in different resolutions with the help of zooming in and zooming out like 640 x 480, 600x800, etc.

50) 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 the oven heats food at the desired temperature within a specified time duration.
5. Verify the ovens functioning with the maximum attainable temperature.
6. Verify the ovens functioning with minimum attainable temperature.
7. Verify that the oven's plate rotation 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 the oven's functionality with different kinds of food – solid, and 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.

51) Write a scenario of Coffee vending Machine ?

1.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.

52) Write a scenario of chair?

1. Check the material used for making the chair is as per the requirement document.
2. Check if the dimension of the chair is as per the specification document.
3. Check if the dimension of the weight is as per the specification document.
4. Check if the dimension of the height is as per the specification document.
5. Check the number of legs of a chair.
6. Check the chair backrest option.
7. Check whether all legs of the chair on a plane surface are equal.
8. Check if the chair is compatible for taking a rest.
9. Check whether a human can sit comfortably or not on a chair.
10. Check if the chair has an adjustment functionality or not.

11. Check the sitting space as per mentioned in the requirement document.
12. Check whether the legs of the chair have any wheels or not.
13. Check if the chair is good enough to handle a specified load.
14. Check what the maximum amount of load the chair is handling is.
15. Check the date is stable enough to take any human load.
16. Check the color of the table is as per the SRS documents.
17. Check the type of chair, for example, Office chair, Dining room chair, Dentist chair,
18. Check the balance of the chair with one arm.
19. Check the balance of the chair with three legs.
20. Check the stress testing of the chair by dropping the Chair down from the practical height.

53) To Create Scenario (Positive & Negative) Gmail

- **Positive**

1. Verify that all the read and unread emails are displayed in the inbox

2. Verify that the recently received email or unread emails are highlighted in bold in the Inbox section.

3. Verify that the recently received email has correct sender's name or email id, subject of the email, its preview, and date or time.

4. Verify that the recently received email's sender's name or email id, subject of the email, and date or time should be in bold, and preview text shouldn't be in bold.

5. Verify that the attachment icon is displayed next to the preview text of the email if the email has any attachment.

6. Verify that the Archive, Delete, Mark as read, Snooze options are displaying on hovering the unread email.

7. Verify that the Archive, Delete, Mark as unread, Snooze options are displaying on hovering the read email.

8.Verify that the Email id, Add to contacts, Open detailed view, Send email, Send message, Start video call, Schedule event options are displaying when we hover on the name/email of the read/unread email.

9.Verify that the user is navigated to the email content when clicking on the email in the inbox.

10.Verify that the content of the email is displayed correctly without any formatting issues.

11.Verify that the attachment in the email is downloadable or not.

12.Verify that the attachments can be downloaded as a single zip file.

13.Verify that the attachments can be downloaded individually.

14.Verify that the attachments can be viewable in the browser itself without downloading.

15.Verify that the attachment is downloading in zip format if the attachment size is more than 1 MB.

16.Verify that the attachments are scanned for viruses once we try to download the file.

17.Verify that the Reply and Forward buttons are displaying at the bottom of the email content.

18.Verify that all the read emails are not highlighted.

19.Verify that the unread emails count is displayed beside 'Inbox' text in the left sidebar of Gmail.

20.Verify that the unread emails count is increased as per the number of new emails we received.

- **Negative**

1.Verify that all the read and unread emails are displayed in the inbox.

2.Verify that the recently received email or unread emails are highlighted in bold in the Inbox section.

3.Verify that the recently received email has correct sender's name or email id, subject of the email, its preview, and date or time.

4.Verify that the recently received email's sender's name or email id, subject of the email, and date or time should be in bold, and preview text shouldn't be in bold.

5.Verify that the attachment icon is displayed next to the preview text of the email, if the email has any attachment.

6.Verify that the Archive, Delete, Mark as read, Snooze options are displaying on hovering the unread email.

7.Verify that the Archive, Delete, Mark as unread, Snooze options are displaying on hovering the read email.

8.Verify that the Email id, Add to contacts, Open detailed view, Send email, Send message, Start video call, Schedule event options are displaying when we hover on the name/email of the read/unread email.

9.Verify that the user is navigated to the email content when clicking on the email in the inbox.

10.Verify that the content of the email is displayed correctly without any formatting issues.

11.Verify that the attachment in the email is downloadable or not.

12.Verify that the attachments can be downloaded as a single zip file.

13.Verify that the attachments can be downloaded individually.

54) Online shopping to buy product (flipkart) Scenario

1. Verify that the company logo and name are clearly visible.
2. Verify that the user is able to navigate through all the products across different categories.
3. Verify that all the links and banners are redirecting to the correct product/category pages and none of the links are broken.
4. Verify that all the information displayed – product name, category name, price, and product description is clearly visible.
5. Verify that all the images – product and banner are clearly visible.
6. Verify that category pages have a relevant product listed, specific to the category.
7. Verify that the correct count of total products is displayed on the category pages.
8. Search – Verify that on searching, all the products satisfying the search criteria are visible on the search result page.
9. Search – Verify that on searching, products get displayed on the basis of their relevancy.
10. Search – Verify that the count of products is correctly displayed on the search result page for a particular search term.
11. Filtering – Verify that filtering functionality correctly filters products based on the filter applied.
12. Filtering – Verify that filtering works correctly on category pages.
13. Filtering – Verify that filtering works correctly on the search result page.
14. Filtering – Verify that the correct count of total products is displayed after a filter is applied.
15. Filtering – Verify that the correct count and products get displayed on applying multiple filters.
16. Sorting – Verify that all the sort options work correctly. On sorting the products based on the sort option chosen.

17. Sorting – Verify that sorting works correctly on the category pages.
18. Sorting – Verify that sorting works correctly on the search result page.
19. Sorting – Verify that sorting works correctly on the pages containing the filtered result, after applying filters.
20. Sorting – Verify that the product count remains the same irrespective of the sorting option applied.

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

56) Write a Scenario of Lift

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 the lift moves to the particular floor as the button of the floor is clicked.
7. Verify that the lift stops when the up/down buttons on a 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 taken to go to a floor.
10. Verify that in case of power failure, the lift doesn't free-fall and gets halted on the particular floor.

11. Verify lifts working in case the button to open the door is pressed before reaching the destination floor.
12. 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.
13. Verify the time duration for which the door remains open by default.
14. Verify if the lift interior is having proper air ventilation.
15. Verify lighting in the lift.
16. Verify that at no point the lift 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 the multiple floor number button is clicked, the lift should stop on each floor.
19. Verify that in case of capacity limit is reached users are prompted with a warning alert- audio/visual.
20. Verify that inside lift users are prompted with the current floor and direction information the lift is moving towards- audio/visual prompt.

57) Write a Scenario of WhatsApp Group (generate group)

1. check if an admin can add others as admin.
2. check admin can remove it from the group.
3. check admin can add users to the group.
4. check admin can restrict users.
5. check admin can remove others from admin.
6. check if the admin can add people.
7. check if the admin can add 250 people to group.
8. check the admin user able to add people with the invite link.
9. check the admin can delete people and add them back to the group.
10. check the admin user can be able to delete people.
11. check the admin user able to delete all people in the group.
12. check the admin user cab be able to ban users.
13. check the contact details shows the name and profile photos of the contacts.
14. check the maximum and minimum length of the test field.
15. check the total number of characters the text field can allow.

58) Write a Scenario of WhatsApp payment ?

1. Open the chat window of the contact you want to send or receive payments from.
2. Click on the attachment icon in the chat window.
3. Select the Payment option from the list of available options.
4. Enter the amount you want to send or request.
5. Add a note (optional).
6. Click on the Send button.
7. Verify that the payment is successfully sent or received.
8. Verify that the payment details are displayed in the chat window.
9. The payment is successfully sent or received.
10. The payment details are displayed in the chat window.