**1)What is Exploratory Testing?**

It’s a type of testing where testers explore the application freely, without following pre-written test cases. They learn about the app and test is at the same time.

**2)What is Traceability Matrix?**

A documented that shows the connection between requirements and test cases. It helps make sure all requirements are tested.

**3)What is Boundary Value Testing?**

Testing values at the edges or limits. For Example, if input should be 1 to 100, we test 0,1,100 and 101.

**4)What is Equivalence Partition Testing?**

Divide input data into groups (valid and invalid) and testing only one value from each group. The saves time and still gives good coverage.

**5)What is Integration Testing?**

After combining two or more modules, this testing checks whether they work well together and exchange data properly.

**6)What Determines the Level of Risk?**

* How Likely a problem will happen
* How serious the problem will be if it happens
* How often the system is used

These helps decide how risky a part of the software is.

**7)What is Alpha Testing?**

Testing done by the internal team (Developers/Testers) before releasing the product to real users.

**8)What is Beta Testing?**

Testing done by real users in the real world before the final release, to get feedback and find issues.

**9)What is Component Testing?**

Testing Individual Part (Components) of the software separately to check if they work properly.

**10)What is Functional System Testing?**

Testing the whole system to check if all features work as expected based on the requirement.

**11)What is Non-Functional Testing?**

Testing things like speed, security, usability, and performance – not the features, but how well the system works.

**12)What is GUI Testing?**

Testing the graphical user interface like Buttons, Screen, Color, Layout) to check if it looks and works correctly.

**13)What is Adhoc Testing?**

Adhoc Testing is an informal and unstructured type of software testing where testers randomly test the application without following any test cases or documentation.

**14)What is Load Testing?**

Check how the system performs under expected user load – like 100 or 1000 users using the app at once.

**15)What is Stress Testing?**

Pushing the system beyond its limits (more users than normal) to see how it handles high pressure or if it crashes.

**16)What is White Box Testing and list the types of white box testing?**

White box testing is a technique where the testers known the internal code, logic, and structure of the program. It is usually done by developers to check if the internal operations are working correctly.

* Types of Coverage:

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

**17)What is Black Box Testing? What are the different black box testing technique?**

Black box testing is a technique where the tester does not know the internal code or structure.

The tester only checks the functionality of the system from the user’s point of view.

This type of testing is usually done by testers.

* Black Box Testing Techniques:

1. Equivalence Partitioning
2. Boundary Value Analysis
3. Decision Tables
4. State Transition Testing
5. Use-case Testing
6. Other black-box Testing
7. Syntax or Pattern Testing

**18)Mention what are the categories of defects?**

1)Data Quality/Database Defects:

Related to incorrect data handling in the database.

Examples:

* Values not properly inserted/deleted.
* Wrong/null values saved.

2)Critical Functionality Defects:

Bugs that affect the most important functions of the application.

Examples:

* Exceptions that break main functionality.

3)Functionality Defects:

Affect how the system works (Functional behavior).

Examples:

* JavaScript errors.
* Buttons not working (e.g., Save, Cancel).
* Missing or Incorrect features.
* Infinite loops.

4)Security Defects:

Related to authentication and authorization problems.

These are high priority defects.

Examples:

* Logging in with invalid credentials.
* Access to restricted pages.

5)User Interface (UI) Defects:

Issues with design, layout, and text.

Less severe but affect user.

Examples:

* Wrong error messages.
* Spelling Mistake.
* Alignment issues.

**19)Mention what big bang testing is?**

Big Bang Testing is an integration testing approach where all modules are combined and tested at once, instead of testing them step by step.

It saves time in writing test plan.

But it becomes difficult to find the root cause of defects when something fails.

**20)What is purpose of exit criteria?**

Exit criteria are the conditions that must be met before ending the testing process.

* Purpose:
* To ensure enough testing has been done.
* To confirm that the product is stable and ready for release.
* To check that all major defects are fixed and test cases are passed.

**21)When should “Regression Testing” Be performed?**

Regression Testing should be done when any changes are made to the code, such as:

* After fixing bugs.
* After adding new features.
* After any updates or enhancements

It helps to make sure existing functionality is still working after the changes.

**22)What is 7 key principles? Explain in Details?**

These are the fundamental ideas that guides all software testing activities.

1)Testing Shows Presence of Defects:

Testing can show bugs exist, but not that there are no bugs.

2)Exhausting Testing is Impossible:

You can’t test everything; Test what matters most.

3)Early Testing:

Start testing early in the development cycle.

4)Defect Clustering:

Most bugs are found in a few modules.

5)Pesticide Paradox:

Repeating the same tests won’t find new bugs update test cases regularly.

6)Testing is context dependent:

Testing approach depends on the type of software (e.g., Web, Mobile, Banking).

7)Absence of Error Fallacy:

A bug-free system is useless if it doesn’t meet user needs.

**23)Difference Between QA v/s QC v/s Tester.**

|  |  |  |
| --- | --- | --- |
| **(QA)**  **Quality Assurance** | **(QC)**  **Quality Control** | **Tester** |
| Focus on improving and managing the process of software development and testing. | Focuses on checking the product to ensure it meets quality standard. | Focuses on executing tests and identifying bugs in the software. |
| It is Process oriented activities | It is Product oriented activities | It is Product oriented activities |
| It is Preventive activities | It is Corrective process | It is Preventive process |
| It is a subset of Software Test Life Cycle (STLC) | QC can be considered as the subset of Quality Assurance | Testing is the subset of Quality Control |
| Creating a company-wide testing process and checklist | Reviewing the final app to check if features work properly. | Testing login, Payment, or search functions and reporting errors. |

**24)Difference between Smoke and Sanity.**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Smoke Testing** | **Sanity Testing** |
| Purpose | To check basic functionality of the build | To verify specific functionality after changes. |
| When | After a new build is received | After bug fixes or minor changes |
| Depth | Shallow testing-checks overall working | Deep testing – checks focused features deeply |
| Example | Does the app open? Can you log in? | Is the search function working after fixing the bug? |
| Time | Quick and broad | Quick but more focused |
| Who Performs | Testeror developer | Mainly testers |

**25)Difference between Verification and Validation**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Verification** | **Validation** |
| Definition | Verification is the process of checking whether the software is being built correctly by following requirements, design & coding. | Validation is the process of checking whether the developer software meets user need & expectation. |
| Happens | Verification happens before testing | Validation happens after and during testing. |
| Process Type | It is a Static process type. | It is a Dynamic process type. |
| Who Performs | Verification does developers and analysts | Validation is does QA and Tester teams. |
| Activities | Activities like Review, Walkthrough and Inspection. | Activities like Actual Testing (Manual and Automation) |
| Evaluation Items | Include like Plan, Requirement Specification, Design Specification, Code, Test cases. | Include like Test the actual Product/Software. |

**26)Explain types of performance testing.**

Performance testing is checks how fast, stable, and responsive an application is under a specific workload.

1)Load Testing:

Check how the system performs under expected user load.

**Ex.** 100 or 1000 users using the app at once.

2)Stress Testing:

Pushing the system beyond its limits (more users than normal) to see how it handles high pressure or if it crashes.

**Ex.** Increasing users to 10,000 to check if the system crashes or handles the pressure.

3)Spike Testing:

To see how the system reacts to sudden and extreme changes in load.

**Ex.** Suddenly increasing users from 100 to 5000 in few seconds.

4)Endurance Testing:

To check system performance over a long period under a normal load.

**Ex**. Running a web application with 500 users for 24 hours to find memory leaks or showdowns.

5)Scalability Testing:

To check if the system can handle increased load by adding resources (like more servers).

**Ex.** Adding more users gradually to see if performance remains stable when scaling up.

6)Volume Testing:

To test the system’s ability to handle a large amount of data.

**Ex.** Uploading 10 lakh records into a database to see if the system can process and retrieve data efficiently.

**27)What is Error, Defect, Bug and Failure.**

A mistake in coding is called Error, Error found by tester team it called Defect, and defect is accepted by developers’ team it’s called Bug, and product or system it’s not meets the requirement it’s called Failure.

**28)Difference between priority and severity.**

|  |  |
| --- | --- |
| **Priority** | **Severity** |
| How soon the bug should be fixed | How serious the impact of the bug is. |
| Set by Tester/Manager. | Set by Tester. |
| Focus on business need. | Focus on Technical issues |
| Ex. Spelling Mistake on home page (high priority, low severity | Ex. App crashes when clicking a button (high priority, low severity (if rarely used)) |

**29)What is Bug life Cycle?**

The duration or time span between the first-time defects are found and the time that it is closed successfully, rejected, postponed or deferred is called as ‘Defect Life Cycle’.

**New**

**Assigned**

**Duplicate Rejected Deferred Not A Bug**

**Open**

**Retest**

**Reopened**

**Fixed**

**Pending Retest**

**Closed**

**Verified**

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

**30)Explain the difference between Function and Non-Functional Testing.**

|  |  |
| --- | --- |
| **Functional Testing** | **Non-Functional Testing** |
| Verifies the action and operation of the application, | Verifies performance, reliability, scalability of the application. |
| Based on Business logic and requirements. | Based on performance standards and quality attributes. |
| Manual or automation testing is done. | Mostly tool-based testing (e.g., performance testing tools). |
| Ex. Smoke testing, Regression Testing, Integration Testing. | Ex. Performance Testing, Load Testing, Stress Testing. |

**31)To create HLR and Testcase of (Instagram, Facebook) first page and chat functionality**

1)Instagram: [“Click here”](https://docs.google.com/spreadsheets/d/1oFYYuMpknFUArDPtAQbsH4bnIDoTPTCZ/edit?usp=sharing&ouid=101736461453958108480&rtpof=true&sd=true)

2)Facebook:[“Click Here”](https://docs.google.com/spreadsheets/d/1ISCh3uzOxPS2PVMIm5a_SGhI95SEhwh8/edit?usp=sharing&ouid=101736461453958108480&rtpof=true&sd=true)

**32)Difference Between STLC and SDLC.**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **SDLC (Software Development Life Cycle)** | **STLC (Software Testing Life Cycle)** |
| Focus | Entire Software development process | Only the software testing process |
| Goal | Develop a complete, functional software product | Ensure the product is tested and meets quality standards |
| Phases include | Requirement Gathering  Design  Coding  Testing  Maintenance | Requirement Analysis  Test Planning  Test Case Design  Execution  Closure |
| Start when | Project Start | After requirements are defined |
| Involves | Developers, testers, business analysts, project managers | Mainly testers and QA team |
| Deliverable | Working software application | Tested product with test reports and defect logs |

**33)What is difference between test scenario, test cases and test script?**

|  |  |  |
| --- | --- | --- |
| **Test Scenario** | **Test Cases** | **Test Script** |
| High-level functionality to be tested | Detailed steps to verify a scenario | Code to automate the test case |
| Identify all testable areas | Validate specific condition and workflow | Automate execution for efficiency |
| Broad and general | Specific and detailed | Very detailed and technical |
| Written in simple language | Written in step-by-step format | Write in programming/scripting language. |
| Used by Testers and QA leads | Used by Testers | Used by Automation testers/developers |
| EX. Check User login | EX. Open login page-Enter valid details-click login-check redirection | Selenium code to perform login and validate successful login |

**34)Explain what Test Plan is? What is the information that should be covered.**

A Test Plan is a document that describe the scope, approach, resource and schedule of testing activities. It guides the testing team on what to test, how to test and who will test it.

* The information that should be covered in a Test Plan includes:
* Test Plan ID
* Introduction
* Objective of testing
* Scope (in-scope and out of scope features)
* Test Strategy (Manual/Automation approach)
* Test environment details
* Entry and exit criteria
* Test deliverables
* Roles and responsibilities
* Test schedule/timeline
* Risk and mitigation plans
* Tools used
* Approval and sign-off

**35)What is priority?**

It shows how soon a defect should be fixed. Its set from a business or project point of view.

* Priority can be of following types:

1. Low:Defect can be fixed later it has minimal impact and doesn’t affect major functions. Ex. Type on about page.
2. Medium**:** Defect should be fixed soon but is not critical it affects some functions but has workaround. Ex. Profile pic upload fails.
3. High:Defect must be fixed quickly it affects important features or customer requirements. Ex. Payment crash.
4. Critical:Defect needs an immediate fix it blocks key functions or system use. Ex. App won’t open.

**36)What is severity?**

It shows how serious the defects impact is on the system. Its set from a technical or tester point of view.

* Severity can be of following types:

1. Major:System or component stops data may be corrupted but workaround exists. Ex. System crash, data loss.
2. Moderate:System gives wrong or incomplete results, no crash. Ex. Main feature broken, no workaround.
3. Minor:Small issue, system works fine with a workaround. Ex. Small issues, has a workaround.
4. Cosmetic:Only affects look/feel not functionality. Ex. Very small issues, like UI alignment or spelling mistake.

**37)Bug categories are…**

1)Data quality/Database defects:Deals with improper handling of data in the database.

**Ex.**

* Values not deleted/inserted into the database properly
* Improper/wrong/null values inserted in place of actual values.

2)Critical Functionality Defects**:** The occurrence of these bugs hampers the crucial functionality of the application.

**Ex.**

* Exceptions

3)Functionality Defects:These defects affect the functionality of the application.

**Ex.**

* All JavaScript errors
* Buttons like Save, Delete, Cancel not performing their intended functions
* A missing functionality or a feature not functioning the way it is intended to
* Continuous execution of loops

4)Security Defect: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.

**Ex.**

* Authentication: Accepting an invalid username/password
* Authorization: Accessibility to pages though permission not given.

5)User interface defects:As the name suggests, the bugs deal with problems related to UI are usually considered less severe.

**Ex.**

* Improper error/warning/UI message
* Spelling mistake
* Alignment problems

**38)Advantage of Bugzilla.**

* It is free and open-source.
* Easy to report and track bugs.
* Supports email notifications for updates.
* Allows setting bug priorities and severity.
* Helps manage projects and track progress.
* Offers good search and filter options.
* Supports multiple users and permissions.
* Generates useful bug and activity reports**.**

**39)Difference between Priority and severity.**

|  |  |  |
| --- | --- | --- |
| **Aspect** | **Priority** | **Severity** |
| Meaning | Urgency to fix the bug | Impact of the bug on functionality |
| Focus | Business or customer urgency | Technical seriousness of the defect |
| Decided by | Project Manager, Client, Product Owner | Tester, Developer, QA lead |
| Ex1 | Wrong company name on homepage->High | Login button not working->Critical |
| Ex2 | Minor UI issue on main page->Medium | App crashes on specific action->Major |
| Ex3 | Small typo on contact page->Low | Incorrect calculation in billing->Major |
| Levels | High, Medium, Low, Critical | Major, Moderate, Minor, Cosmetic |

**40)What are the different Methodologies in Agile Development Model?**

1. Scrum
2. Kanban
3. Extreme Programming (XP)
4. Lean
5. Crystal
6. Dynamic Systems Development Method (DSDM)
7. Feature-Driven Development (FDD)

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

* **Authentication**: Checks who the user is (example: username, password, OTP).
* **Authorization**: Checks what the user is allowed to do (example: user can view profile but not admin panel).

**Common problems faced in web testing:**

* Broken links
* Slow page loading
* Poor UI on different devices or browsers
* Security issues (like SQL injection, XSS)
* Session timeout or logout not working
* Form input validation errors
* Incorrect error or success messages
* Compatibility issues on different browsers or screen sizes

**42)To create HLR and Testcase of web Based (WhatsApp web)**

WhatsApp Web: [“Click here”](https://docs.google.com/spreadsheets/d/1eXkELXvpvX96hPJXdoX7cxfH1APOXQik/edit?usp=sharing&ouid=101736461453958108480&rtpof=true&sd=true)

**43) Create Testcases on WhatsApp Group chat.**

WhatsApp Group chat: [“Click here”](https://docs.google.com/spreadsheets/d/110sZkLHUhHZIF6NtMY4OERBS7ZuVRQw7/edit?usp=sharing&ouid=101736461453958108480&rtpof=true&sd=true)

**44)Write a scenario of only WhatsApp chat message.**

WhatsApp chat message: [“Click here”](https://docs.google.com/spreadsheets/d/1CABKwJk8M_QuQQjNSty4CM47pjVKrACM/edit?usp=sharing&ouid=101736461453958108480&rtpof=true&sd=true)

**45)Write a scenario of pen.**

* Check the pen type - is it a ballpoint, ink pen, or gel pen?
* Check if the pen writes well on different papers -like smooth or rough paper.
* Check the weight of the pen
* Check if the pen has a cap or is without a cap.
* Check the ink color -blue, red, pink
* Check the smell of the ink -when writing, it should not have a bad or strong smell.
* Check if the pen writes on other surfaces - like cardboard or rubber, not just paper.
* Check if the writing is erasable or permanent.
* Check if the pen works well under normal pressure - no need to press hard.
* Check the strength of the pen body - it should not break easily if dropped or bent.
* Check the grip — it should feel comfortable and not slip from fingers.
* Check if the pen supports multiple refills.

**46)Write a scenario of pen stand.**

* Put different pens (ball pen, ink pen, gel pen) inside the pen stand.
* Check if the pens fit well and don’t fall or tilt.
* Try putting other items like pencils, markers, or scissors into the stand.
* Check if the pen stand stays steady on the table and doesn’t fall over.
* Take the pens out and put them back in several times to check if the stand stays strong.
* Check if the stand can hold the number of pens, it says it can hold.
* Look at the material (plastic, metal or wood) and check if it feels strong.
* Check if the pen stand is easy to clean (remove dust or ink marks easily).
* Check the design - make sure there are no sharp or loose parts that can hurt someone.
* Check if the pen stand is light and easy to move.

**47) Write a scenario of Door.**

* Check if the door is a single door or a bi-folding door.
* Check if the door opens inside or opens outside.
* Check if the size of the door matches the given measurements.
* Check if the material (like wood, metal, glass, etc.).
* Check if the door is a sliding door or a rotating door.
* Check what type of lock the door (key lock, latch, etc.).
* Check how many locks are on the door -both on the inside and outside.
* Check if the door has a peephole (small hole to look outside) or not.
* Check if the door has a stopper or not.
* Check if the door can close automatically using a spring or similar system.
* Check if the door makes noise when you open or close it.
* Check how the door holds up when used with lots of water (for example, in rain or cleaning).
* Check how the door works in different weather conditions — like hot, cold, or rainy weather.

**48) Write a scenario of ATM.**

* Check if the ATM has a touch screen, buttons or both.
* Check if the ATM shows options when the valid card is inserted.
* Check if the touch screen is smooth.
* Check if the user can choose the language.
* Check if the user must enter a PIN before seeing account details.
* Check if the ATM limits the number of incorrect PIN attempts.
* Check if the ATM blocks the card after too many incorrect PIN attempts.
* Check if the PIN is shown as dots when entered.
* Check if the user can select account types like savings or current.
* Check if the user can view account balance.
* Check if the ATM only allows withdrawals in specific amounts (like ₹100, ₹500).
* Check if the user can’t withdraw more than available balance.
* Check if the user can get a transaction receipt.
* Check if the user can’t exceed the one-day withdrawal limit.
* Check if the ATM doesn’t work with expired cards and shows an error.
* Check if the transaction is cancelled if there’s a power failure before cash is withdrawn.

**49)When to used Usability Testing?**

Usability Testing is used to ensure a product is easy to use and provides a positive user experience. Its done to identify and fix any issues with navigation, design or features, especially before launch, during redesigns, or when adding new features. It helps improve user satisfaction and ensures the product is effective and user-friendly.

**50)What is the procedure of GUI Testing?**

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

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

**51) Write a scenario of Microwave Owen**

* Check if the oven size matches the given specifications.
* Check if the oven material is correct and safe for cooking.
* Check if the oven heats food at the right temperature.
* Check if the oven heats food within the right time.
* Check the oven at its maximum temperature.
* Check the oven at its minimum temperature.
* Check if the plate spins at the right speed without spilling food.
* Check if the oven door closes properly.
* Check if the oven door opens smoothly.
* Check the power or battery requirement and if it works smoothly.
* Check if the text on the oven is easy to read.
* Check if the digital display is clear and working.
* Check the oven with maximum food capacity.
* Test with different types of food like solid and liquid.
* Test with different food at different temperatures.
* Check if the user manual has clear and easy instructions.

**52) Write a scenario of Coffee vending Machine**

* Check if the machine size matches the specifications.
* Check if the color and brand name are correct and easy to see.
* Check how you add ingredients (milk, water, coffee).
* Check if the right amount of milk, water, coffee is used per cup.
* Check the power or voltage needed for the machine.
* Check what happens if you turn off power suddenly -machine should stop, no extra coffee should come out.
* Check that coffee does not leak when not in use.
* Check the digital display shows correct info.
* Check the power button works to switch on/off.
* Check the indicator lights work when on/off.
* Check all buttons work when pressed.
* Check the coffee temperature is correct and same every time.
* Check the system shows an error if ingredients are over.
* Check there’s a way for extra coffee to drain.  
   Check the machine works in different weather and moisture.
* Check the machine doesn’t make too much noise.
* Check how long it takes to make one cup.
* Check what happens if you press two or more buttons together.
* Check what happens if you overfill ingredient containers.

**53) Write a scenario of chair**

* Check if the chair is stable and can hold the weight of an average person.
* Check the material of the chair like wood, plastic, etc.
* Check if the chair can be used as an office chair or home chair.
* Check if the chair has back support.
* Check if the chair has hand support.
* Check the type and color of the paint.
* Check if the material is strong or if it breaks easily.
* Check how the chair reacts when washed with water.
* Check if the chair size matches the specifications.
* Check if the chair weight is as per specifications.
* Check the height of the seat from the floor.

**54) Create Test Cases on Compose Mail Functionality.**

Compos mail test cases: [“Click Here”](https://docs.google.com/spreadsheets/d/1idciCSWTTv5IgAZ2t3i7xxMoPrl61ecF/edit?usp=sharing&ouid=101736461453958108480&rtpof=true&sd=true)

**55) Online shopping to buy product (Flipkart)**

Online shopping product test cases: [“Click here”](https://docs.google.com/spreadsheets/d/1VdmHQOldhGshIaMNGd0M7-OwPABry_Lt/edit?usp=sharing&ouid=101736461453958108480&rtpof=true&sd=true)

**56) Write a Scenario of Wrist Watch**

* Check if the watch is analog or digital.
* For analog, check if the second, minute, and hour hands show the correct time.
* For digital, check if hours, minutes, seconds are shown correctly.
* Check the material of the watch.
* Check if the dial shape matches the specification.
* Check the weight of the watch.
* Check if the watch is waterproof.
* Check if the watch shows day and date.
* Check the color of the text (time, date, day) on the display.
* Check if the brand name is visible on the dial.
* Check if the watch has stopwatch, timer, alarm features.
* For digital, check if it shows 12-hour or 24-hour format.
* Check if the watch has a warranty or guarantee.
* Check if the dial cover is glass or plastic and if it can break easily.
* Check the battery type and requirement.

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

* Verify the dimensions of the lift.
* Check if the type of lift door matches the specifications.
* Check the type of metal used inside and outside the lift.
* Verify the weight capacity of the lift.
* Check if the buttons for door open/close and floor numbers are working.
* Make sure the lift goes to the right floor when the button is pressed.
* Check if the lift stops when up/down buttons are pressed on any floor.
* Verify if there is an emergency button to call for help.
* Verify that the lift stops safely during a power failure.
* Check if the lift works when the door open button is pressed before reaching the floor.
* Check how long the door stays open by default.
* Verify if the lift has good air ventilation.
* Check if the lift has proper lighting.
* Make sure the lift doors don’t open while moving.
* Verify if the lift shows a warning when it is overloaded.

**58)** **Write a Scenario of WhatsApp payment**

* Verify that the WhatsApp app is updated to the latest version.
* Verify that the payment option is visible in the chat or attachment menu.
* Verify that the user’s bank account is linked to WhatsApp.
* Check if the user can select a contact to send payment.
* Verify that the user can enter the amount correctly.
* Verify that the app asks for the UPI PIN before sending money.
* Check if the payment completes successfully after entering the correct PIN.
* Verify that the user and receiver both get payment confirmation messages.
* Check if the payment history shows the correct details.
* Verify that if the user enters the wrong PIN, the payment fails with a proper error message.
* Check if payment fails when the internet is off or slow, and the user sees an error.
* Verify that the app shows a proper message if the bank server is down or busy.