Module 2 (Manual Testing)

1. What is Error, Defect, Bug and Failure ?

* **Error**

**:** A mistake in coding is called Error.

* **Defect**

**:** An error found by tester is called Defect.

* **Bug**

**:** A defect accepted by development team is called Bug.

* **Failure**

**:** The inability of system to perform its required functions is called Failure

1. What is 7 Key Principles ? Explain in Detail ?

* 7 key principles are following

**1. Testing shows presence of defects**

**2. Exhaustive is impossible**

**3. Early testing**

**4. Defects clustering**

**5. The Pesticide paradox**

**6. Testing is context dependent**

**7. Absence of error fallacy**

1. **Testing shows presence of Defects**

: Testing can show that defects are present, but it can’t prove that there are no defects. Testing reduces the probability of undiscovered defects remaining in the software. However testing can’t prove that there are no defects present.

1. **Exhaustive testing is impossible**

: Testing everything including all combination of inputs and preconditions is not possible and it is very unlikely that the project timescales would allow for this number of tests. So, instead of doing exhaustive testing we can use risk and priorities to focus testing efforts.

1. **Early Testing**

: Testing activity should start as early as possible in the software development life cycle. So it can help to prevent defects from being introduced into code. This activities should be focused on defined objectives.

1. **Defects clustering**

: Defects are not evenly spread in a system, they are clustered. In other words most defects found during testing are usually confined to a small number of modules and are responsible for most operational failures.

1. **The pesticide paradox**

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

1. **Testing is context dependent**

: Testing is basically context dependent. Testing is done differently in different contexts. Different kinds of sites are tested differently.

1. **Absence of error fallacy**

: If the system built is unusable and does not fulfil the user’s needs and expectations then finding and fixing defects does not help. Even after defects have been resolved it may still be unusable .

1. What is traceability matrix ?

: Traceability matrix is a table which is used to trace the requirements during software development life cycle.

1. What determines the level of risk ?

: The level of risk is determined by two main factors :

1.the Probability of the risk occurring

2.the impact if the risk occurred

1. What is Functional system testing?

: Functional system testing is a type of software testing, which is used to verify the functionality of the software application, whether the function is working according to the requirement specification.

1. What is Non functional testing ?

: Non functional testing is a testing of attributes of a system or components that do not related to functionality.

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

|  |  |  |  |
| --- | --- | --- | --- |
| S.N | Quality Assurance | Quality Control | Tester |
| 1. | Activities which ensure the implementation of process, procedures and standards in the context to verification of developed software and intended requirement. | Activities which ensure the verification of developed software with respect to documented requirements. | Activities which ensure the identification of bugs/error/defects in the software. |
| 2. | Focuses on process 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 activities. | It is a preventive activities. |
| 5. | It is a subset of software test life cycle. | QC can be considered as the subset of quality assurance. | Testing is a subset of Quality control. |

8. Explain the difference between functional testing and non functional testing .

|  |  |
| --- | --- |
| Functional testing | Non functional testing |
| Functional testing is performed using the functional specification provided by the clients 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 affective 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. | Non functional 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 non functional testing are   * Performance testing * Load testing * Volume testing * Stress testing * Security testing * Installation testing * Penetration testing * Compatibility testing * migration testing |

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

: Black box testing is a method of software testing that verifies the functionality of application without having specific knowledge of internal structure of a component or a system .

* **There are five black box testing techniques :**
* Equivalence partitioning
* Boundary value analysis
* Decision tables
* State transition testing
* Use-case testing

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

: White box testing is a testing based on an analysis of the internal structure of the component or a system .

* **There are four techniques of white box testing** :
* statement / segment coverage
* Decision / branch coverage
* Condition coverage

1. What is Equivalence partitioning testing?

: It is a software testing technique that divides the input data of a software unit into partitions of data from which test cases can be derived .

1. what is boundary value testing ?

: It is a software testing technique in which tests are tests are designed to include representatives of boundary values.

1. What is exploratory testing?

: it is a black box testing technique performed without planning and documentation. It is usually performed by manual testers.

1. What is integration testing ?

: The phase in software testing in which individual software modules are combined and tested as a group. It is known as integration testing.

1. What is component testing ?

: Component testing is a testing of individual software components. The purpose is to validate that each unit of software performs as designed.

1. Mention what big bang testing is ?

: It is a software testing technique in which all components or modules is integrated simultaneously, after which everything tested as whole.

1. What is GUI testing ?

: It is a process of testing product that uses a graphical user interface, to ensure it meets written specification. This is normally done by a testing team.

1. What is Ad-hoc testing ?

: Ad-hoc testing is a testing performed without planning and documentation , in which the tester tries to ‘break’ the system by randomly trying the system’s functionality.

1. What is Load testing ?

: Load testing is to test the system behaviour under normal workload conditions, and it is just a testing with the actual workload.

1. What is Stress testing ?

: Stress testing is to test the system behaviour under extreme conditions and is carried out till the system failure.

1. When should “Regression testing” should be performed ?

: Regression should be performed :

* When the system is stable and the system or the environment changes.
* When testing bug-fix releases as part of maintenance phase.

1. What is alpha testing ?

: Alpha testing is a type of testing software product or system conducted at the developer’s site.

1. What is Beta testing ?

: Beta testing is the final testing before releasing application for commercial purpose. It is typically done by end users or others.

1. Mention what are the categories of defects ?

**: The defects are categorizes into Five categories :**

1. Data base defects
2. Critical functionality defects
3. Functionality defects
4. Security defects
5. UI (User interface) defects
6. What is the purpose of exit criteria ?

: The 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)

1. What is bug life cycle?

: The bug life cycle is nothing but the various phases a bug undergoes after it is raised or reported.

1. Difference between Smoke and Sanity testing ?

|  |  |
| --- | --- |
| Smoke testing | Sanity testing |
| Smoke testing is performed to ascertain that critical functionality of program is working fine. | Sanity testing is done to check that new functionality / bugs have been fixed. |
| The objective of this testing is to verify the ‘stability’ of the system in order to proceed with more rigorous testing. | The objective of this testing is to verify the ‘rationality’ of the system in order to proceed with more rigorous testing. |
| This testing is performed by developers or testers. | This testing is usually performed by testers. |
| Smoke testing is usually documented or scripted. | Sanity testing is usually not documented or unscripted. |
| Smoke testing is subset of acceptance testing. | Sanity testing is subset of regression testing. |
| Smoke testing exercises the entire system from end to end. | Sanity testing exercises only the particular component of entire system. |

1. Difference between verification and validation.

|  |  |  |
| --- | --- | --- |
| Criteria | Verification | Validation |
| Definition | The process of evaluating work-products 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 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. | To ensure that the product actually meets the user’s needs, and that the specifications were correct in the first place. |
| Question | Are we building the product right? | Are we building the right product? |
| Evaluation items | Plans, requirement specs, design specs, code, test cases. | The actual product/ software. |

1. What is priority ?

: priority is a order in which developer has to fix the bug. If high priority is mentioned then the developer has to fix it at the earliest. The priority status is set based on the customer requirements.

1. What is severity ?

:severity is how seriously the bug is affecting the software. the severity type is defined by the tester based on the written test cases and functionality.

1. Difference between Priority and severity .

|  |  |
| --- | --- |
| Priority | Severity |
| Priority determines the order in which defects should be fixed. | Severity refers to extent of impact a particular defect has on software. |
| It focuses on when a defect needs resolution. | it focuses on how severe the defect is in terms of functionality. |  |
| Priority is relative and business focused. | Severity is absolute and customer focused. |  |

1. Explain types of performance testing ?

**: There are five types of performance testing.**

1. Stress testing
2. Load testing
3. Spike testing
4. Volume testing
5. Scalability testing
6. Stress testing :

* Stress testing is to test the stability and reliability of the system. This test mainly determines the system on its robustness and error handling under extremely heavy load conditions.
* The aim of stress testing is to determine the limit, at which the system or software or hardware breaks.

1. Load testing :

* It is a performance testing to check system behaviour under load. Testing an application under heavy loads, such as testing of website under a range of loads to determine at what point the system’s response time degrades or fails.

1. Spike testing :

* Spike testing is a type of performance testing in which an application receives a sudden and extreme increase or decrease in load.
* The goal of spike testing is to determine the behaviour of a software application when it receives extreme variations in traffic.

1. Volume testing :

* Volume testing is a type of performance testing that examines the stability and response time of a system by transferring huge volume of data, and evaluates a variety of different system components like databases, software, etc.

1. Scalability testing :

* Scalability testing is the testing of a software application to measure its capability to scale up or scale out in terms of any of its non functional capability.
* performance, scalability and reliability testing are usually grouped together by software quality analysts.

1. What is difference between SDLC and STLC :

|  |  |  |
| --- | --- | --- |
| Criteria | SDLC | STLC |
| Scope | SDLC covers the entire software development process, including planning, analysis, design, coding, testing, deployment and maintenance. | STLC is a subset of SDLC and focuses specifically on testing |
| PHASES | It involves six phases: requirement analysis, design, implementation, testing, deployment, maintenance and support. | It consists of five phases: test planning, test case development, test environment setup, test execution, test closure. |
| Relation | SDLC primarily related to software development. | STLC is exclusively related to software testing. |
| Objective | It aims to successfully develop software. | It aims to complete successful testing of software. |
| End result | It helps in developing good quality software. | It ensures that the software is defect free. |

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

|  |  |  |  |
| --- | --- | --- | --- |
| Criteria | Test scenarios | Test cases | Test scripts |
| Definition | A test scenario represents any functionality that need to be tested in software. | A test case is detailed document that specifies the steps to be executed to test a specific aspect of the software product. | A test script is a set of sequential instruction that detail how to execute a core business function. |
| Source | It is derived from SRS or BRS documents. | It is created manually. | It is written as small programs. |
| Focus | It focuses on business processes and flows. | It focuses on specific functionality to be tested. | It focuses on how the tests should be done. |
| means | Manual testing. | Manual testing. | Manual and automation testing. |

1. Bug categories are….

**Bug categories are four :**

* Security bug
* Database bug
* Functionality bug
* User interface bug

1. Advantage of Bugzilla.

**: The advantage of Bugzilla are :**

* It is easy in use and its user interface is understandable for people without technical knowledge.
* It easily integrates with test management instruments.
* It integrates with emailing system.
* It is an open source widely used bug tracker.
* It automates documentation.
* It enhances the communication between the developing team and the testing team.

1. What are the different methodologies in agile development model ?

**: There are six methodologies :**

1. Scrum
2. Extreme programming (XP)
3. Kanban
4. Lean
5. Dynamic system development method
6. Explain what test plan is ? what is the information that should be covered?

: Test plan is a document describing the scope, approach, resources and schedule of intended test activities.

:The following information should be covered in test plan.

* Defining the overall approach of testing, including the definition of the test levels and entry-exit criteria.
* Integrating and coordinating the testing activities into software life cycle activities.
* Making decision about :
* What to test?
* Who do testing?
* When and how the testing activities should be done and when they should be stopped?
* How the test results will be evaluated.
  + - * Define resources to be assigned for the different tasks.
      * Test ware: defining the amount, level of detail, structure and templates for the test documentation.
      * Selecting metrics for monitoring and controlling test preparation and execution, defect resolution and risk issues.

1. Explain the difference between authorization and authentication in web testing. What are the common problems faced in web testing?

: The difference between authorization and authentication is following ;

* authorization verifies a user’s permissions to access resources, while authentication verifies the identity of a user or system.
* Authorization determines what permissions user has. Authentication ensures that the person using the system is a valid user
* After authentication, authorization checks the user’s authorities. During authentication, user’s identities are checked to grant access to the system.
* Common problems faced in web testing is as follows:

1. Cross browser compatibility

* With numerous browsers and versions available, ensuring compatibility across different browsers is challenging.

1. Responsiveness

* Verify that the application adapts properly to different devise resolutions.

1. Cross devise compatibility

* Test across various mobile devices (both iOS and Android) to ensure consistent behaviour.

1. Security

* Test features like online transactions and payment gateways.

1. Performance testing

: assess application performance under different loads.

1. When to use usability testing ?

: here’s when you should consider using usability testing:

1. Early and Often Throughout Development:

* Conduct usability testing early in the design process, even with low-fidelity prototypes.
* Continue testing often as you iterate on your design solutions.

1. Before Launch:

* Ensure a smooth user experience right from the start.
* Validate that your product meets user needs and expectation.

1. After major updates:

* Verify that new features are user-friendly and don’t disrupt existing functionality.
* Usability testing helps maintain a positive user experience even after significant changes.

1. What is the procedure for GUI testing ?

: The procedure for GUI testing is as follows:

1. Understand the requirement:

* Begin by fully understanding the software application’s intended functionalities.
* Define what the GUI elements (buttons, menus, forms, etc.) should do under different conditions.

1. Define test cases:

* Create detailed test cases that cover various scenarios related to GUI.

1. Set up the test environment:

* Prepare the test environment with the necessary hardware, software and configurations.
* Ensure you have access to the application’s GUI for testing.

1. Execute the tests:

* Interact with GUI elements as user would.
* Verify that the buttons, menus, forms and other visual components work as expected.
* Check for responsiveness, alignment, colour consistency, and usability.

1. Report and analyse:

* Document any defects or issues encountered during testing
* Provide clear descriptions of the problems found along with steps to reproduce them.

1. Make necessary changes:

* Developers address the reported issues by fixing defects or improving the GUI
* Retest the modified GUI components to ensure they now function correctly.

1. Review and improvement:

* Conduct a review of the GUI testing Process.
* Identify areas for improvement, such as enhancing test coverage or redefining test cases.

1. Write a Scenario of only WhatsApp chat messages:

* Verify that user can send a text message successfully.
* Test sending messages with different character lengths.
* Verify that user can send or receive images, audio, video, contacts, and emojis In individual chat successfully.
* Verify that user can send or receive the message in the group chat.
* Verify that user can send or receive images, audio, video, contacts, and emojis in group chat.
* Verify that user can check the messages delivered and read the time for a message in the ‘message info’ section.
* Verify that user can send or receive chats In the secondary languages available.
* Verify that user can delete text, contacts, images, and video messages in both (individual chats and group chats).
* Check if forwarding a message to more than five people works as expected.

1. Write a scenario of WhatsApp group (generate group):

* Verify that user can create a new WhatsApp group.
* Check user can set a name for the created group.
* Ensure that the user can add or save a group description.
* Confirm that the admin role can be assigned to a specific group member.
* Test the functionality of a admin- specific permissions such as adding/removing members and changing group settings.
* Verify that only group admin can add or remove people from the group.
* Test the process of adding multiple members to a group.

1. Write a scenario of a pen :

* Verify the type of pen whether it is a Ball pen, ink pen, or a gel pen.
* Test the pen’s ability to write clearly on different type of papers.
* Check the weight of the pen. It should be as per specification.
* Verify the length and diameter of the pen as per specification.
* Verify the strength of the pen’s outer body. It should not be easily breakable.
* Verify the pen is with a cap or without a cap.
* Verify the colour of the ink on a pen.
* Check the Odor of the pen’s ink on writing over a surface.
* Verify the pen writes smoothly on surfaces other than paper.
* Ensure consistent ink flow without leaving blobs.
* Confirm that the pen’s ink does not leak when tilted upside down, or a higher altitudes.
* Test if the text written by the pen is erasable.
* In the case of a ball and gel pen, verify that the user can change the refill of the pen easily.
* In the case of an ink pen, verify that the user is able to refill the pen with all the supported ink types.
* Verify that the functioning of a pen at extreme temperatures-much higher and lower than room temperature.
* Verify the functioning of a pen by applying extreme pressure.

1. Write a scenario a pen stand :

* Verify that the pen stand’s height, width, depth as per specifications.
* Check the material of the pen stand as specified in the requirements.
* Validate that the pen stand has slots or compartments to hold pens securely.
* Confirm that the pen stand’s base is stable and non-slip to prevent tipping over.
* Verify that the pen stand can accommodate different pen sizes.
* Test the ease of inserting and removing pens from the stand.
* Test the pen stand’s durability.
* Verify that the pen stand is easy to clean.
* Attempt to insert oversized pens into slots.
* Test the pen stand’s stability by applying excessive force.
* Verify that the pen’s stand does not scratch pens placed inside.

1. Write a scenario of door :

* Verify if the door is single door or bi folded door.
* Check if the opens inwards or outwards.
* Verify that the dimensions of the doors are as per the specification.
* Verify that the material used In door body its parts as per specifications.
* Verify that colour of the door is as specified.
* Verify the door is sliding door or rotating door.
* Check the position, quality, and strength of hinges.
* Check the type of locks in the door.
* Verify if the door is having peek hole or not.
* Verify the door is having a stopper or not.
* Verify if the door closes automatically or not.
* Verify that door makes noise when opened or closed.
* Check the door condition when used extensively with water.
* Check the amount of force- pull or pushed required to open or close the door.

1. Write scenario of ATM :
   * Verify the type of ATM machine, if it has a touch screen, keypad buttons, or both.
   * Verify that on properly inserting a valid card different banking options appears on the screen.
   * Check that no option to continue and enter credentials is displayed to user when the card is inserted incorrectly.
   * Verify that the touch of ATM screen is smooth and operational.
   * Verify that the user is presented with a 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 limited number 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 than user is not allowed to continue further, and operations like temporary blocking of the card, etc get initiated
   * Check that pin is displayed in mask 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 prompted to enter the amount again in case the amount entered is less than the minimum amount configured.
   * Verify 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.
   * Verify that the user is allowed to do only one transaction per pin request.
   * Verify that the user is not allowed to exceed the one day transaction limit amount.
   * Check that In case the ATM machine run out of money, a proper message is displayed to the user.
   * Verify that the applicable fees gets deducted with the withdrawn amount in case user exceeds the limit of the number of free transaction In a month.
   * Verify that the applicable fees gets deducted with the withdrawn amount In case user uses a card of a bank other than that of an ATM.
   * Check that 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 a null and the amount is not withdrawn from the user’s account.
2. Write a scenario of microwave oven :

* Verify the dimensions of the oven are as per specification provided.
* Verify that the oven’s material is optimal for its use as an oven and as per specifications.
* Verify that the oven heats food at the desired temperature within a specified time durations.
* Verify the ovens functioning with the maximum attainable temperature.
* Verify the ovens functioning with the 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 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 function correctly.
* Verify that the temperature regulator works correctly and is smooth to operate.
* Check the maximum capacity of the oven and test its functioning with that volume of food.
* Check the oven’s functionality with different foods at different temperatures.
* Check 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.

1. Write a scenario of chair :

* verify that the chair is stable enough to take an average load.
* Check the material used in making the chair-wood, plastic etc.
* Check if the chair’s leg are level to the floor.
* Check that usability of the chair as an office 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 colour.
* 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 the chair is as per specification.
* Verify that the weight of the chair is as per specification.
* Check the height of the chair’s seat from the floor.

1. Write a scenario of coffee vending machine :

* Verify that the dimension of coffee machine is as per specification.
* Verify that the outer body, as well as inner part’s material is as per specification.
* Verify that the machine’s body colour as well as brand is correctly visible as per specification.
* Verify that 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 of the machine or cutting the power. The machine should stop in that situation and in power resumption, the remaining coffee should not 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 the indicator lights when machine is switched on-off
* Verify that the functioning of all 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 serving.
* Verify the mechanism to clean the system work correctly.
* 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 not make too much noise when in operation.
* Check the amount of time machine takes to serve a single serving of coffee.
* Check the functioning of the coffee machine when two/multiple times buttons are pressed simultaneously.
* Check the functioning of coffee machine with a lesser or higher voltage then required.
* Check the functioning of coffee machine if the ingredient container’s capacity is exceeded.

1. Write a scenario of Wrist Watch :

* Verify the type of watch- analogue or Digital.
* In the case of an analogue watch, check the correctness time displayed by the second, minute, and hour hand of the watch.
* In the case of digital watch, check the digital display for hours, minute, and seconds is correctly displayed.
* Verify the material of watch and its strap.
* Check if the shape of dial is as per specification.
* Verify the dimension the watch is as per specification.
* Verify the weight of watch.
* Check if the watch is waterproof or not
* Verify that the numbers in the dial are clearly visible or not.
* Verify that clock’s time can be corrected using the key in case of an analogue clock and buttons in case of a digital clock.
* Check if the second hand of the watch makes ticking sound or not.
* Verify 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 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.

1. Write a scenario of Lift (Elevator) :

* Verify the dimensions of the lift.
* Verify the type of door of the lift 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 total weight.
* Verify the buttons in the lift to close and open the door and numbers as per the no 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 a proper 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 user are prompted with current floor and direction information the lift is moving towards- audio/visual prompt.

1. Write a scenario WhatsApp payment :

* Verify that users can link their bank accounts to WhatsApp for payments.
* Test the process of adding payment methods(UPI, debit/credit cards) to the app.
* Ensure that user can set a default payment method for convenience.
* Test the initiation of payments from within a chat.
* Verify that user can send money to contacts seamlessly.
* Check if user receive payment notification when someone sends them money.
* Validate that transaction history accurately reflects sent and received payments.
* Verify that users are prompted for authentication(PIN, Fingerprint, etc.) before making payments.
* Ensure that sensitive Information (such as UPI PIN) is encrypted and protected.
* Check if there are any transaction limits (daily, weekly, monthly) imposed by WhatsApp.
* Check that when user exceeds limit a proper error message shown.
* Verify that international payments (if supported) adhere to relevant regulations.
* Test the process of requesting refunds for failed or cancelled transactions.
* Verify that user can dispute incorrect charges and seek resolution.
* Check if refunds are proceed promptly and accurately.
* Verify that user can pay directly from chat during conversations.
* Verify that payment links or QR codes shared in chats lead to the correct payment flow.
* Test the integration of payments with other WhatsApp features (e.g. business accounts, product catalogues, etc ).