Module–2(Manual Testing)

❖ What is Exploratory Testing?

- Exploratory testing is a type of software testing that is performed in an unstructured and ad-hoc manner. Testers explore the software and try different scenarios, inputs, and interactions to identify bugs and issues without using pre-written test cases.

❖ What is traceability matrix?

- A Traceability Matrix is a document that co-relates any two-baseline documents that require a many-to-many relationship to check the completeness of the relationship.
- It is used to track the requirements and to check the current project requirements are met.

❖ What is Boundary value testing?

- Boundary value analysis generates test cases that highlight errors better than equivalence partitioning.
- Boundary value testing is focused on the values at boundaries.
- This technique determines whether a certain range of values are acceptable by the system or not.
- It is very useful in reducing the number of test cases.

❖ What is Equivalence partitioning testing?

- Equivalence partitioning is a technique that divides test objects into partitions or classes
- This technique is used for testing a range of values, inputs, and outputs
- There are valid and invalid equivalence partitions.
- Valid equivalence partitions contain only the valid values that are accepted by the testing object.
- invalid equivalence partitions contain only invalid values that are rejected by the testing object.
- EP says that by testing just one value we have tested the partition
- (typically a mid-point value is used). It assumes that:
- If one value finds a bug, the others probably will too
- If one doesn't find a bug, the others probably won't eithe
- Plus there are 2 Invalid partitions

❖ What is Integration testing?

- Integration testing is the process of testing the interface between two software units or modules.
- Integration testing is typically performed after unit testing and before system testing.

- The purpose of this level of testing is to expose faults in the interaction between integrated units.
- Integration testing is done by a specific integration tester or test team
- There are 2 levels of Integration Testing:Component Integration Testing,System Integration Testing

❖ What is Alpha testing?

- Beta testing is a type of User Acceptance Testing
- Alpha testing is performed at the developer's site.
- Alpha testing requires a testing environment or a lab.
- Alpha Testing is a type of software testing performed to identify bugs before releasing the product to real users or to the public. Alpha Testing is one of the user acceptance testing.

➢ What is beta testing?

- Beta testing commonly uses black-box testing
- Beta testing is performed at the end-user of the product.
- Beta testing doesn't require a testing environment or lab.

❖ What is component testing?

- Component testing, also known as program or module testing, is done after unit testing.This type of testing those test objects can be tested independently as a component without integrating with other components Mention what bigbang testing is?
- This testing is done by the development team.
- Unit testing is the first level of testing
- Unit tests find problems early in the development cycle.
- Unit testing is performed by using the White Box Testing method.

❖ What is functional system testing?

- Functional testing is based on the business requirement.
- It is based on requirements of customer.
- Functional testing is executed first
- Manual testing or automation tools can be used for functional testing
- It helps to enhance the behavior of the application.
- Functional testing is easy to execute manually.
- It is based on requirements of customer.
- Examples: Unit Testing,SmokeTesting, Whitebox testing, Blackbox testing

❖ What is Non-Functional Testing?

- Non-functional testing is based on the performance requirement.

- It is based on expectations of customer.
- Non functional testing should be performed after functional testing
- it helps to improve the performance of the application.
- It is hard to execute non-functional testing manually.
- It is based on expectations of customer.
- Examples: PerformanceTesting**,**LoadTesting,StressTesting,SecurityTesting

❖ What is GUI Testing?

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

❖ What is Adhoc testing?

- it is also known as Random testing or Monkey testing
- Adhoc testing saves a lot of time
- Adhoc testing is performed randomly.
- It is done after formal testing.
- It follows an unstructured way of testing.
- It can be performed within very limited time.
- Needs good knowledge on product as well as testing concept to perfectly identified the issues in any model.
- Main aim of this testing is to find defects by random checking.

❖ What is load testing?

- Its a performance testing to check system behavior under load.
- Testing an application under heavy loads
- Load testing is a kind of performance testing which determines a system's performance under real-life load conditions.
- This testing helps determine how the application behaves when multiple users access it simultaneously.

❖ What is stress Testing?

- Stress testing is used to test the stability & reliability of the system.
- Stress Testing is done to make sure that the system would not crash under crunch situations.
- Stress testing is also known as endurance testing.

❖ What is white box testing and list the types of white box testing?

- White box testing is also known as structural testing or code-based testing,
- and it is used to test the software's internal logic, flow, and structure.
- It is also called glass box testing or clear box testing or structural testing.
- White Box Testing is also known as transparent testing or open box testing.
- the testers require knowledge of how the software is implemented, how it works.

- Types of white box testing

- White Box Testing types include Unit Testing
- Static and Dynamic Analysis,
- Statement Branch,
- Path Coverage,
- Security Testing,

❖ What is black box testing? What are the different black box testing techniques?

- Black box testing techniques apply to all levels of testing, as well as functional and non-functional testing types.
- The technique of testing without having any knowledge of the interior workings of the application is Black Box testing.
- Example: he Calculator application on our phone. We open the application, enter the numbers we want to calculate as well as the operator and equal sign, and wait for the output

- There are four specification-based or black-box technique:
  1)Equivalence partitioning
  2)Boundary value analysis
  3)Decision tables
  4)State transition testing
  5)Use-case Testing
  6)Other Black Box Testing

❖ Mention what are the categories of defects?

- Bug categories are
- Data Quality/Database Defects: Deals with improper handling of data in the
- database.

- Critical Functionality Defects: The occurrence of these bugs hampers
- the crucial functionality of the application. Examples: - Exceptions

- Functionality Defects: These defects affect the functionality ofthe application.

  - Examples:

- 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 toContinuous execution of loops379

- Types of Defect

- Security Defects: Application security defects generally involve
- improper handling of data sent from the user to the application. These
- defects are the most severe and given highest priority for a fix.

- Examples:

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

- User Interface Defects: As the name suggests, the bugs deal with
- problems related to UI are usually considered less severe.

- Examples:
- Improper error/warning/UI messages
- Spelling mistakes
- Alignment problems

❖ What is the purpose of exit criteria?

- Exit criterion is used to determine whether a given test activity has been completed or NOT.
- you will not enter the next stage until the exit criteria for the previous stage is met. But practically this is not always possible.
- Successful Testing of Integrated Application.
- Executed Test Cases are documented
- All High prioritized bugs fixed and closed
- Technical documents to be submitted followed by release Note
- The major disadvantage is time consuming and difficult to trace the cause of failures because of this late integration
- The major disadvantage is time consuming and difficult to trace the cause of failures because of this late integration

❖ When should "Regression Testing" be performed?

- Regression Testing should be performed when there are changes or updates to the software.
- It ensures that existing functionality remains intact after modifications.
- This testing helps catch any unintended side effects and maintains software quality.

❖ What is 7 key principles? Explain in detail?

1.Testing shows presence of Defects

- Testing can show that defects are present, but cannot prove that there are no defects.
- We test to find Faults As we find more defects, the probability of undiscovered defects
- remaining in a system reduces.
- However Testing cannot prove that there are no defects present

2. Exhaustive Testing is Impossible!

- Testing everything including all combinations of inputsand preconditions is not possible.
- we cannot test everything (i.e. allcombinations of inputs and pre-conditions).
- Why do not Testing Everything?
- requires enormous resources
- is too expensive
- takes too long

3.Early Testing

- Testing activities should start as early as possible in the development life cycle
- that Testing doesn't start once the code has been written!

4.Defect Clustering

- A small number of modules contain most of the defects discovered during pre-release testing
- or are responsible for the most operational failure
- Defects are not evenly spread in a system They are 'clustered'

5. The Pesticide Paradox

- If the same tests are repeated over and over again, eventually the same
- set of test cases will no longer find any new defects.
- "pesticide paradox", the test cases need to be regularly reviewed and revised,
- Testing identifies bugs, and programmers respond to fix them

6.Testing is Context Dependent

- For example Safety – critical software is tested differently from an e-commerce site.
- Testing is basically context dependent.
- Also different industries impose different testing standards

7.Absence of Errors Fallacy

- If the system built is unusable and does not fulfill the user's needs andexpectations then finding and fixing defects does not help.
- Even after defects have been resolved it may still be unusable and/ordoes not fulfil the users' needs and expectations

❖ Difference between QA v/s QC v/s Tester

QA	QC:	Testing
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.
Preventive activities.	It is a corrective process.	It is a preventive process.
Process oriented activities.	Product oriented activities.	Product oriented activities.
Focuses on processes and procedures rather than conducting actual testing on the system.	Focuses on actual testing by executing Software	Focuses on actual testing

❖ Difference between Smoke and Sanity?

Smoke	Sanity
Smoke testing is done to assure that the acute functionalities of program is working fine.	Sanity testing is done to check the bugs have been fixed after the build.
Smoke testing is also called subset of acceptance testing.	Sanity testing is also called subset of regression testing.
Smoke testing is documented.	Sanity testing isn't documented.
Smoke testing is performed by either developers or testers.	Sanity testing is normally performed by testers.
Smoke testing may be stable or unstable	Sanity testing is usually not scripted.
Smoke testing may be stable or unstable	Sanity testing is used in the case of only modified or defect functions of system/products.
Smoke testing is scripted.	Sanity testing is commonly executed manually, not by using any automation approach.

❖ Difference between verification and Validation

Verification	Validation
It includes checking documents, design, codes and programs.	It includes testing and validating the actual product.
Verification is the static testing.	Validation is the dynamic testing.
It does not include the execution of the code.	It includes the execution of the code.
Methods used in verification are reviews, walkthroughs, inspections and desk-checking.	Methods used in validation are Black Box Testing, White Box Testing and non-functional testing.
It checks whether the software conforms to specifications or not.	It checks whether the software meets the requirements and expectations of a customer or not.
It can find the bugs in the early stage of the development.	It can only find the bugs that could not be found by the verification process
Quality assurance team does verification.	Validation is executed on software code with the help of testing team.
It comes before validation.	It comes after verification.
After a valid and complete specification the verification starts.	Validation begins as soon as project starts.
Verification is for prevention of errors.	Validation is for detection of errors.
Verification finds about 50 to 60% of the defects.	Validation finds about 20 to 30% of the defects.
Verification is about process, standard and guideline.	Validation is based on the fact and is often stable.

❖ What is Error, Defect, Bug and failure?

- Error
  - A mistake in coding is called error
  - An error in software testing refers to a slip-up,
  - misunderstanding, or mistake made by a software engineer. In the category of developer,we include software engineers, analysts, programmers, and testers.

- Defect
  - error found by tester is called defect
  - the functionality of an application is not working as per the customer's requirement is known as a defect,
  - It is found during the development phase while unit testing.

- Bug
  - defect accepted by development team then it is called bug
  - A fault in a program which causes the program to perform in an unintended or unanticipated manner
- Failure
  - build does not meet the requirements then it is failure

- The inability of a system or component to perform
- its required functions within specified performance requirements

❖ Difference between Priority and Severity

Priority	Severity
Defined by the impact of a specific problem on any application's functionality.	Defined by the impact on business.
Deals with the technical aspects of the application.	Category decided by developers or product owners.
Deals with the technical aspects of the application.	Deals with the timeframe or order to fix the defects.
The value does not change with time, it's fixed.	The priority value is subjective and may change after comparing with other defects.

❖ What is Bug Life Cycle?

- Bug Life Cycle is the chain of processes through which a bug undergoes, from being reported to being closed/rejected. It is also called the defect life cycle. The bug must pass through a life cycle intended to be closed.

- Usually, software developers categorise bugs into life cycle stages. Subsequently, they consistently communicate about the software bugs they work on when building and updating a software program.
- Two major participants in the defect life cycle are the tester and the developer. The tester either detects a bug or, in the case of a user detecting a bug, the tester would review the bug before conveying it to the development team. Subsequently, a software developer comes up with a solution.

❖ Explain the difference between Functional testing and NonFunctional testing

- Functional testing:

- Functional testing is based on the business requirement.
- It is based on requirements of customer.
- Functional testing is executed first
- Manual testing or automation tools can be used for functional testing
- It helps to enhance the behavior of the application.
- Functional testing is easy to execute manually.
- It is based on requirements of customer.
- Examples: Unit Testing,SmokeTesting, Whitebox testing, Blackbox testing

- NonFunctional testing:

- Non-functional testing is based on the performance requirement.
- It is based on expectations of customer.
- Non functional testing should be performed after functional testing
- it helps to improve the performance of the application.
- It is hard to execute non-functional testing manually.
- It is based on expectations of customer.
- Examples: PerformanceTesting,LoadTesting,StressTesting,SecurityTesting

- To create HLR & TestCase of 1)(Instagram , Facebook) only first page

1.Instagram:[https://github.com/Nishit004/Testing_Workspace/blob/main/Module-2/Project/Instragram.xlsx]

2.Web_Instagram:[https://github.com/Nishit004/Testing_Workspace/blob/main/Module-2/Project/Web-Whatsapp.xlsx]

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

1.Facebook:[https://github.com/Nishit004/Testing_Workspace/blob/main/Module-2/Project/Facebook.xlsx]

2.Web_Facebook:[https://github.com/Nishit004/Testing_Workspace/blob/main/Module-2/Project/Web_Facebook.xlsx]

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

scenarios	test cases	test script

High-level descriptions of functionality to be tested.	Step-by-step instructions for specific scenarios.	Automated test cases in code.
Identify what needs testing.	Precise roadmap for testers.	Efficient, repeatable execution.
“Verify checkout process.”	“Login with valid credentials.”	Python script for login validation.

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

- A test plan is a document that outlines the overall strategy for testing a software project.
- It includes essential information such as:

1.Test Scope: What features or modules will be tested.
2.Test Objectives: Goals of testing (e.g., functionality, performance).
3.Test Schedule: Timeline for testing activities.
4.Test Environment: Hardware, software, and network setup.
5.Test Risks: Potential challenges and mitigation strategies.
In short, a test plan guides testing efforts and communicates the testing approach to stakeholders.

❖ What is priority?

- Severity is a parameter to denote the impact of a particular defect on the software.

❖ What is severity?

- Priority is a parameter to decide the order in which defects should be fixed.

❖ Advantage of Bugzila .

- Centralized Issue Tracking: Bugzilla provides a centralized platform for tracking and managing software issues efficiently.
- Customizable Workflow: It offers customizable workflows to suit the specific needs and processes of different development teams.
- Collaboration: Bugzilla fosters collaboration among team members by providing a shared space to discuss and resolve issues.
- Detailed Reporting: It generates detailed reports on bugs, including status, severity, priority, and resolution progress.
- Version Control Integration: Bugzilla seamlessly integrates with version control systems, facilitating streamlined development workflows.
- Notifications: It sends automated notifications to keep team members informed about updates and changes to bugs they're involved with.

- Access Control: Bugzilla allows administrators to set access controls, ensuring that sensitive information is only accessible to authorized personnel.
- Extensibility: It's highly extensible through plugins and customizations, enabling teams to tailor Bugzilla to their specific requirements.
- Historical Data: Bugzilla maintains a historical record of all bug-related activities, providing valuable insights for future projects and retrospectives.
- Community Support: Being open-source, Bugzilla benefits from a vibrant community of users and contributors who offer support, resources, and enhancements.

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

- The Agile methodology is a way to manage a project by breaking it up into several phases.
- Agile is based on the iterative-incremental model. In an incremental model, we create the system in increments,
- where each increment is developed and tested individuall
  Accommodate changing requirements throughout the development process – The ability to avoid delays when a requirement or feature request changes.

- Frequent delivery of working software – Scrum accommodates this principle since the team operates in software sprints or iterations that ensure regular delivery of working software.

- Collaboration between the business stakeholders and developers throughout the project – Better decisions are made when the business and technical team are aligned.
  Support, trust, and motivate the people involved – Motivated teams are more likely to deliver their best work than unhappy teams.

- Working software is the primary measure of progress – Delivering functional software to the customer is the ultimate factor that measures progress.

- Agile processes to support a consistent development pace – Teams establish a repeatable and maintainable speed at which they can deliver working software, and they repeat it with each release.

- Attention to technical detail and design enhances agility – The right skills and good design ensures the team can maintain the pace, constantly improve the product, and sustain change.

- Simplicity – Develop just enough to get the job done for right now.

❖ Write a scenario of only Whatsapp chat messages

- Verify that text messages are sent and received promptly.
- Check for proper formatting and display of text.
- Verify Emojis and Special Characters:
- Check the usage of emojis and special characters.
- Verify emojis display correctly on both ends.
- Check sending and receiving images, ensuring they load correctly.
- Verify the smooth transmission of videos without lag.
- Checks haring documents (PDFs, Word files) and confirm they can be opened.
- Verify that hyperlinks shared in the chat can be clicked and open in the browser.
- Check reply feature to check if messages can be quoted correctly.
- Verify Ensure replies maintain the conversation flow.
- Check recording and sending voice messages.
- Verify Confirm voice messages are played back without any distortion.
- Check the functionality of group chats, including adding and removing members.
- Verify sending messages to ensure they reach all members.
- Check that status updates (text, images, videos) are displayed correctly.
- Verify that contacts can view and respond to status updates.
- Check for new messages, ensuring they are timely and accurate.
- Verify that the app displays notifications even when it's in the background.
- Check sending messages when one user is offline and verify if they receive the messages upon reconnecting.
- Verify blocking a contact and confirm they cannot send messages or see your status.

❖ Write a Scenario of Pen

- Verify the type of pen, whether it is a ballpoint pen, ink pen, or gel pen.
- Verify that the user is able to write clearly over different types of papers.
- Check the weight of the pen. It should be as per the specifications. In case not mentioned in the specifications, the weight should not be too heavy to impact its smooth operation.
- Verify if the pen is with a cap or without a cap.
- Verify the color of the ink on the pen.
- Check the color of the pen’s ink on writing over a surface.
- Verify the surfaces over which the pen is able to write smoothly apart from paper e.g. cardboard, rubber surface, etc.
- Verify that the text written by the pen should have consistent ink flow without leaving any blob.
- Check that the pen’s ink should not leak in case it is tilted upside down.
- Verify if the pen’s ink should not leak at higher altitudes.
- Verify if the text written by the pen is erasable or not.
- Check the functioning of the pen by applying normal pressure during writing.
- Verify the strength of the pen’s outer body. It should not be easily breakable.
- Verify that text written by pen should not get faded before a certain time as mentioned in the specification.
- Check if the text written by the pen is waterproof or not.

❖ Write a Scenario of Pen Stand

- Verify the material of the pen stand, ensuring it meets the specified quality standards.
- Check Measure and confirm the dimensions of the pen stand, checking against the provided specifications.
- Verify the color of the pen stand matches the specified color in the product description.
- Check the surface finish for any scratches, dents, or imperfections that may affect the product's appearance.
- Verify stability of the pen stand by placing pens and other stationery items to ensure it can hold them securely.
- Check that the weight distribution is balanced to prevent tipping or instability
- Verify a durability test by applying light pressure to various parts of the pen stand to ensure it doesn't break or crack easily.
- Check how easy it is to clean the pen stand and if it retains any stains or marks after cleaning.
- Verify Confirm that different types and sizes of pens fit comfortably in the designated slots.
- Check that the actual product design matches the provided design specifications
- Verify the pen stand requires assembly, ensure that the assembly process is straightforward and all parts are included.
- Verify Inspect the packaging for any damage and ensure all necessary components, including instructions, are included.
- Check if the materials used in the pen stand are environmentally friendly and comply with any relevant standards.
- Verify for the presence and accuracy of the brand logo and any labeling on the pen stand.
- Check the overall user experience with the pen stand, considering ease of use, aesthetics, and functionality.

❖ Write a Scenario of Door

- Verify that the door is made of the specified material, meeting quality and safety standards.
- Check Measure the door's dimensions to ensure they match the provided specifications.
- Verify Confirm that the color and finish of the door match the customer's selection and quality expectations.
- Tests.
- Verify Inspect the locking system for proper functionality, ensuring it securely locks and unlocks without issues.
- Check the presence and effectiveness of weather stripping to prevent drafts and enhance insulation.
- Verify If the door has glass components, inspect them for any cracks, scratches, or imperfections.
- Check Ensure that the threshold is level and properly installed to prevent tripping hazards and maintain a tight seal.
- Verify Conduct a sound insulation test to evaluate the door's ability to minimize noise from the outside.
- Check the ease of opening and closing the door, ensuring it operates smoothly for all users.
- Verify Assess the security features, including the strength of the door material and the reliability of the lock.

- Check the door has fire-resistant properties, confirm that it meets the specified standards for fire safety.
- Verify Evaluate the provided installation guidelines to ensure clarity and completeness for installers
- Check the packaging for any damage during transportation and ensure all necessary components and instructions are included.
- Verify Confirm that the door complies with relevant industry standards and local building codes.

❖ Write a Scenario of ATM

- Check inserting a valid card and confirm it is recognized by the ATM.
- Verify that entering the correct PIN allows access to account options.
- Check the balance inquiry feature to ensure accurate account balance retrieval.
- Verify Attempt a cash withdrawal to confirm the proper dispensing of funds.
- Checkdepositing funds, ensuring the ATM accepts deposits accurately.
- Verify if transaction receipts are printed correctly for all transactions.
- Checkthe card is promptly returned after the transaction to the user.
- Verify Test accessibility options, including audio guidance for visually impaired users.
- Check that users can easily select their preferred language for transactions.
- Verify Assess the speed of transactions, ensuring they are processed efficiently.
- Check the ATM's performance under varying network conditions to avoid transaction failures.
- Verify Verify the effectiveness of security measures, including PIN protection and card encryption.
- Check the ATM's ability to handle cash replenishment and update balance information.
- Verify Intentionally trigger errors, such as insufficient funds, to ensure appropriate error messages and user guidance.
- Check that the ATM is remotely monitored for any irregularities or malfunctions.
- Verify Confirm that the ATM accurately recognizes and dispenses the local currency.
- Check if the ATM enforces transaction limits and displays appropriate notifications.
- Verify If applicable, verify the touchscreen responsiveness for user interaction.
- Check the functionality of cardless transactions using mobile apps or QR codes.
- Verify Confirm that the ATM provides clear instructions in case of emergency or system failure.
- Check Ensure that the screen is designed to prevent shoulder surfing, protecting user privacy.
- Verify for options to select whether a receipt is printed or emailed.
- Check If applicable, test the currency conversion feature to ensure accurate rates.
- Verify the availability and accuracy of the transaction history feature.
- Check Assess the physical security of the cash dispenser to prevent unauthorized access.

❖ Write a scenario of Microwave Owen

- Verify that the oven color as per specification
- Check the tyeps of Microwave Owen
- Verify that the company logo is visiable or not
- Check the dimensions of owen as per specification
- Verify that the glass size is visiable or not
- Check the the handle size & postion are porerly or not
- Verify that the handle grip for easy to use or not

- Check the the how much pressure is apply on handle
- Verify that the how much to take time for opening the door
- Check the tyeps of button are touchable or pressurized
- Verify the much pressure is apply on buttons, if pressurized buttons
- Check the the smoothenss of buttons, if is touchable
- Verify that door open properly or not
- Check the door open properly or not
- Verify that the the display proper visiable or not
- Check the all the buttons working as par their specification
- Verify that the timer ser properly or not
- Check the capacity of as per specification
- Verify that the oven heats desired temperature properly or not.
- Check the oven reach at the maximum and minmum temperature.
- Verify that the size of power corde enough for reach plug
- Check the oven warrranty
- Verify that the user instruction text visiable or not
- Check that the voltage requirements of the machine.

❖ Write a scenario of Coffee vending Machine

- Verify that the Machine color as per specification
- Check the company logo is visible or not
- Verify that the dimensions of Machine as per specification
- Check the types of button are touchable or pressurized
- Verify the much pressure is apply on buttons, if pressurized buttons
- Check the smoothnss of buttons, if is touchable
- Verify that the all the buttons working as par their specification
- Check the how much to take time for making
- Verify that how much ingredients required for making
- Check the test of coffee
- Verify that the size of power corde enough for reach plug
- Check the oven warranty
- Verify that the user instruction text visible or not
- Check that the voltage requirements of the machine.
- Verify that the require size of cup

❖ Write a scenario of chair

- Check the material used in making the chair like wood, plastic etc.
- Verify that the company logo is visible or not
- Check the type of chair like office, gaming, household
- Verify that the color of chair
- Check the dimensions as per specification
- Verify that the which type of material
- Check the chair leg

- Verify that the direction of leg
- Check the chair sitting comfort
- Verify that the cover of the chair
- Check the adjestment of chair at minmum and maximum level properly work or not
- Verifiy that the how much pressure require for adjustment
- Check the capacity of weight
- Verify that the handrest is comfortable or not
- Check the warranty and guarantee if available

❖ To Create Scenario (Positive & Negative)

- Gmail (Receiving mail)

- Verify that the mail notification arrive or not
- Check the mail open their category wise like spam mail open in spam folder
- Verify that the sendr profile and name is visible or not
- Check the sender details show properly or not like date,time,subject
- Verify that the Unsubscrib option is working or not
- Check the Sender time proper accurate or not
- Verify that the mail starred or not
- Check the mail reactable or not
- Verify that the reply is proper working or not
- Check the more option visible or not
- Verify that the all more option work properly or not
- Check the mail is printable or not
- Verify that the user can open mail in new windows or not
- Check the all mail text are visible or not
- Verify that the user can forward mail or not
- Check the user can archived or not
- Verify that the user can report to spam or not
- Check the user can unread or not
- Verify that the user can add to task or not
- Check the user can move to or not
- Verify that the user can add lable or not
- Check the user can download attechment or not

- Negative
- If Attachment Download Failure
- Not open file
- If not receiving notification
- If not react on mail
- If user can not readable teaxt
- If user sent mail but receiver block this user

- Online shopping to buy product (flipkart)

- Check the all items are visible on customer screen or not
- Verify that the customer can search or not
- Check the item show with details or not
- Verify that the customer can add to cart without logging or not
- Check the customer can add items from to wishlist
- Verify that the number of items are show on cart
- Check the all items in cart proper visible or not
- Verify that the customer add out to stock items or not
- Check the save for letter option working or not
- Verify that the remove to cart working or not
- Check the pop message show when remove to car or not
- Verify that the total number of items,price visible or not
- Check the recommendation items show or not
- Verify that the how many items are allow to add
- Check the after add items then allow to buying process or not
- Verify that the customer address choosen option if old customer
- Check the new customer for allow to write address or not
- Verify that the how many paying option are available
- Check the after paying then product notification recevied or not

Negative

- If site not open in user device
- If Login otp not recevied
- If user search none exit product
- if product show wrong price
- If confirmation message not received
- If payment done but order not confirm
- If payment faild & money debit
- If user went to change order address after confirm order
- If user received wrong, damage, messing product
- If user try to connect with support team but delay their response

❖ Write a Scenario of Wrist Watch

- Check the types of watch like analog , smart ,digital etc
- Verify that the watch for male or female
- Check the types of materials used for making it.
- Verify that the color
- Check the type of shape
- Verify that the watch logo visible or not
- Check if smart watch screen is touchable or not
- Verify that the weight as per the specification
- Check the dimension as per the specification.

- Verify that the what is meterials used for belt
- Check if analog,all number are visible or not
- Verify that the second,minute,hour hand
- Check if digital,date,month,year and time proper show or not
- Verify that the watch all function like call,camera etc as per specification
- Check the noice of watch high,low,or quit
- Verify that the watch battery or cell life
- Check the watch come water proof
- Verify that the watch comfortable or not
- Check if the watch comes with any guarantee or warranty.

Negative

- If none waterproof watch fall in water
- If the watch show wrong time
- If user have difficulty for set time
- If cell or battery run out
- if watch belt broken
- IF watch not repairable
- If smart watch function not working properly
- If smart watch display brake
- if fatty guy not able to wear
- If skinny guy not able to war

❖ Write a Scenario of Lift(Elevator)

- Check types of lift like commercial like hospital,things lifting,appartment,personal-usable
- Verify the door type autometic or manual
- Check how many door single or double
- Verify interior and exterior design
- Check ventilation fan ,light and camera
- Verify that user stand front of door when open or not
- if lift door autometic
- Check lift buttons like touch or object foam
- Verify that how many time to take open and close door
- Check that the lift door open & close smoothly and easily
- Verify that all the number are visible or not
- Check that all lift number, which is stopped lift that particular floor or not
- Verify lift buttons, how much to require pressure for press
- the button
- Check Open & Close button working properly or not
- Verify Size of lift as per their specification
- Check capacity of life as per their specification
- Verify the lift speed from one floor to another floor
- Check to stop button working or not
- Verify the lift emergency button working or not

- Check the display of lift which is show current position of floor
- Verify that when emergency occur that time which floor stopped lift
- Check when electricity gone, that time lift working on any power backup or not?
- Verify that when lift failure,that time lift emergency braking system applied

Negative

- Check when object stuck between door
- Check door when jamming
- Check if button not touchable or pressable
- Check if lift main cable is brake
- Check when open the door, while lift is running
- Check earthquake time
- check lift emergency brake failure

❖ Write a Scenario of whatsapp Group (generate group)

- Verify that the user can create group
- Check the user can selectable from their contact list
- Verify that user can make admin or not
- Check the user can remove the grup member
- Verify that the user can add group dp or not
- Check the user can add group description
- Verify that the user can send message or not
- Check the user can video or audio calling
- Verify that the user can group video or audio calling or not
- Check the user can change group permissions like add member,group admin etc
- Verify that the user can change group name or not
- Check the user can invite via link or not
- Verify that the user can send, read,edit,delete message or not
- Check if the user can react or not
- Verify that the user can send emojy,gif or not
- Check the user can set custom notification
- Verify that the user can chat lock or not
- Check the user can add group to a community
- Verify that the user can send some attechment
- like file,poll,location,audio etc or not
- Check the user can payment or not
- Verify that the user can search message or not
- Check the user show number of group member
- Verify that the user can send message privetly of group member or not
- Check the user show sending message whoes read or not
- Verify that the user can forward or starred message
- Check the user can set wallpaper
- Verify that the user can report,exit group or not
- Check the user can export the chat
- Verify that the user can add shortcut or not

- Check the user can download like img,voice,file etc

- Negative

- if user lost networke while sending message
- If user join all group member on video or audio call
- If video or audio call low clarity
- If user try to send large size file
- if user block the group member
- If group member not received on time notification
- If group member send spam,spoil message
- If group admin randomly add any person
- IF group admin randomly remove any perosn
- If group admim mistakely make group admin
- If inviters explicitly declined group invitations
- If group admin need to send personally message to the all member
- If group member change dp,description randomly

❖ Write a Scenario of Whatsapp payment

- Verify that the user can add the back account or not
- Check the user can select the bank or not
- Verify that the user can search of bank on search bar or not
- Check the all bank name or images show properly
- Verify that the user can cancel adding payment method or not
- Check the user can verify the phone number or not
- Verify that the user get varification or not
- Check the user can Select the payment option and enter the number of amount.
- Verify that the user can choose payment method like UPI or debit/credit
- Check the transaction status to see if the payment was processed
- successfully or failed.
- Verify that the user can show payment History or not

- Negative

- If the user can not select the back account
- if the user can not add back account
- If the user can not search bank name

- If the user mobile number not verified
- If user have not UPI or debit/credit
- if payment notification was delay
- If the accors boader person send payment
- If security verification failure
- If the middle of transition network was disappear