

Module-2

Assignment

- **What is Exploratory testing?**

Exploratory testing is a concurrent process where Test design, execution and logging happens simultaneously. Testing are not recorded. In this era where the industry pushes over to the automation testing, exploratory testing is new way of thinking. Automation has limits.

Exploratory testing is not random testing, but ad hoc testing with purpose of finding bugs. It is teachable and manageable process.

Exploratory testing is a part of Grey box testing.

- **What is traceability matrix?**

Test condition should be able to linked back to their sources in Test Basis, That is known as Traceability.

A software process that helps you to keep up your virtual data up to date.

There are # types of Traceability matrix:

- 1) Forward traceability: mapping of requirement to test cases*
- 2) Backward traceability: mapping of test cases to requirement*
- 3) Bidirectional traceability: work as both forward and direction.*

- **What is Boundary Value testing?**

Boundary value analysis is methodology for designing test cases that concentrates on software testing effort on cases near the limits of valid range. It refines the equivalence partitioning technique. Boundary value analysis generate the test case that highlights errors better than Equivalence partitioning. Boundary value analysis is performing with the valid as well as invalid entries.

- **What is equivalence partitioning testing?**

The aim is to treat group of inputs as equivalent and to select one representative to test them all. EP can be used for all levels of testing.

EP says that by testing the one value from partition, it assumes that

- *If the one value finds a bug, other will too.*
- *If one doesn't find a bug, others will not.*

- **What is integration testing?**

Testing performs to expose the defects in the interfaces and interaction of different integrated components of software system is known as Integration testing.

We need to perform integration technique because the code of different modules write by different people means the people who have different mindsets from each other. And in integration technique when we merge all this individual module to one group it may behave differently from User requirement. So to satisfy the user requirement it is necessary to perform Integration Testing

The purpose of this testing is to find the faults into interaction between the integrated modules.

- *There are two levels of Integration testing*
 - 1) *Component Integration testing: In component integration testing, it exposes the defects in interaction between the different integrated individual modules of Software application. All individual components should be tested prior before System Integration testing.*
 - 2) *System Integration Testing: In system integration testing it exposes the defects in interaction and interfaces between the different systems.*

Integration testing is done by Integration tester or special test team.

- *There are 2 methods in Integration testing*
 - 1) *Big Bang Integration Testing:*

In Big Bang integration testing, all components or modules integrated simultaneously, after that everything is tested as whole. Big bang technique has advantage that everything is tested before integration testing starts. The major disadvantage in this is that it is time consuming.
 - 2) *Incremental integration testing:*

- i) *Top down Approach: In top down approach testing flows from Top to down.*
- ii) *Bottom up approach: In bottom up approach testing flows from down to upwards.*

- **What determines the level of risk?**

Risk could be any future event that could be result in negative consequences.

There are 2 types of risk

- 1) *Project Risk: Example of Project risk is that the senior manager leave the firm abruptly.*
- 2) *Product risk: The example of Product risk is when the Flight reservation system not installing environment feature.*

- **What is Alpha testing?**

Alpha testing technique is the type of User Acceptance testing

- *Alpha Testing is performed by developer at the development site.*
- *It can be also performed by Independent testing team.*
- *Alpha testing is not open to the market or public.*
- *It is conducted for software application or project.*
- *It is always performed in Virtual Environment.*
- *It is always performed within the organization.*
- *It is the form of Acceptance testing.*
- *It comes under the category of both white box testing and Black box testing.*
- *Alpha testing is always performed at the development side by the developer team.*

- **What is Beta testing?**

Beta testing is a type of User acceptance testing.

- *Beta testing is performed by customers at their own site.*
- *Beta testing is open to market or public.*
- *It is not performed by independent testing team.*
- *Beta testing is performed in Live environment.*
- *It is always performed outside the organization.*
- *It is form of acceptance testing.*

- *Beta testing is performed by user or customer at their own site using their own testing data.*
- *It is kind of Black box testing.*

- **What is component testing?**

Component Testing is also known as Unit testing, Module Testing or Program Testing. Where 'Unit' means smallest testable part of Software application. In Component testing we test individual components of Software Application.

Component testing is performed by software developer.

The result of Unit testing is worked as an input to Integration Testing. Component testing verify that each module work properly as per the specification.

In component testing the tests are written before the coding.

Component testing is run by using White box testing techniques.

- **What is functional system testing?**

A requirement that specifies the functionality that a system should perform, is called Functional System Testing. Functional system testing is performing by Black Box testing Techniques, it does not have any relation with code of software. In functional testing the tester verifies that the all the functionalities of software system are working properly. A software requirement may exist in form of written text document or model.

Function means what a system does. Functional testing determines that each function of Software system work in accordance with the requirement specification. Each and every functionality of software is checked by implying various input, record their output and prepare the comparison of actual and expected result. The testing can be done either manual or automation testing technique.

- **What is non-functional system testing?**

Testing the attribute of a component or system, that does not relate with the functionality is called Non Functional testing. e.g., reliability, interoperability, efficiency, accuracy, maintainability, usability. May be performed at all the test levels.

Non-functional testing includes stress testing, load testing, performance testing etc.

- **What is GUI testing?**

Graphical User Interface (GUI) testing is the process of testing system's GUI for system under test GUI checking involves the checking of user screen resolutions, menus, drop down menus, if error message is display correctly or not.

There are 3 approaches of GUI Testing:

- 1) Manual Testing*
- 2) Record and replay*
- 3) Model based testing*

- **What is ad hoc testing?**

Ad hoc testing is an informal testing type with aim to break the system. It does not follow any test design technique to create the test case or test data. In Ad hoc testing it does not create test case altogether. For testing in ad hoc testing, the tester should have to depth knowledge of testing techniques. The tester tests the application without any requirement document or test cases.

Main aim of this testing is to find defects by random checking. In ad hoc testing tester could pick any random part of software application to test. Ad hoc testing can be achieved through the technique called error guessing. The error guessing is a technique in which experienced and skill testers are encouraged to think of situation in which software could not be able to cope.

There are 3 types of Ad hoc testing

- 1) Buddy testing: Two buddies manually work on the identifiable defects of one module. In this two buddies, one is from Development team and another will be from testing team. The developer can do modifications in test designs.*
- 2) Pair testing: Tow testers are assigned module, they share their ideas and designs with each other. From these two testers one will perform the tests on the component or system and other will make a note of findings.*

3) *Monkey testing: Randomly test the component without any knowledge of use requirement or test cases with an aim to break the system.*

- **What is load testing?**

Load testing is the type of performance testing to check the behavior of system under load. Testing the system with more load, for example with varying load how the system response time change or system got failed. The load testing explains that the how the system will perform when multiple user use the system simultaneously.

- **What is stress testing?**

The system is stressed beyond its specification to check how and when it fails, is called stress testing. Stress testing is also known as Endurance testing. Stress testing mainly perform to check which condition can break the hardware or software system. In stress testing, we check when the system crash under crunch conditions. Stress testing is performing to check systems stability and scalability.

- **What is white box testing and list the types of white box testing?**

Testing based on analysis of internal structure of system or component is known as White Box Testing. White box testing is also known as Glass Testing or Open Box Testing, because the tester requires knowledge of internal structure of Component or system, how the software is implemented and work.

White box testing is a detailed investigation of internal logic of source code.

The tester needs to look inside the code and check which chunk or unit of software system is working inappropriately.

Types of White box testing

- Test/ Code coverage
- Segment/Statement coverage
- Branch/Decision Coverage
- Condition coverage

- **What is Black box techniques? What are the different Black box testing?**

Testing, either functional or non-functional, without reference to the internal structure of component or system is called Black Box Testing.

The testers have no knowledge of how the component or system is structured inside the box. In Black box testing technique tester just concentrate on what the system does, in spite of how it does it.

The tester is oblivious to the system architecture and does not access the source code.

Typically, when performing Black box testing the tester interacts with the User Interface of system or component by providing various inputs, and examining the result obtained. It does not have any relation that where all these inputs are going, and how the system gives output for input.

Different Black Box Testing Techniques are:

- *Equivalence Partitioning*
- *Boundary Value Analysis*
- *Decision Tables*
- *State Transition testing*
- *Use case testing*

- **Mention what are the categories of defect?**

The categories of defect are

Data quality/Database defect: The defect in database, either the deletion or addition entries doesn't reflect in database.

Critical Functionality Defects: The defects affect the critical functionality of application.

Functionality Defect: The defects affect the functionality of application.

Security Defect: It is necessary to secure your application from internal and external threats. These defects have given higher priority to resolve.

User interface defects: The defects related to UI of application are called User Interface defects. The error message does not display properly.

- **Mention what big bang testing is?**

In Big Bang integration testing, all components or modules integrated simultaneously, after that everything is tested as whole. Big bang technique has advantage that everything is tested before integration testing starts. The major disadvantage in this is that it is time consuming.

- **What is the purpose of exit criteria?**

The purpose of Exit criteria is to define when can we stop testing

- *End of all testing (deliver the product)*
- *End of Phase testing (from one phase to another i.e; system testing to UAT)*

- **When should “Regression Testing” be performed**

When the system is stable, but change in work environment.

When testing bug fix releases as a result of maintenance phase.

It should apply at all the test levels.

New features added to the system.

- **What is the 7 key principal? Explain in detail.**

1. *Testing shows the presence of defect: Testing shows the presence of defect, but it doesn't mean that there is no defect remaining in the system. With the help of testing we can detect the defects in the system, but it could not guarantee you that there is no any defect presence in system until it found.*
2. *Exhaustive testing is impossible: Testing each and everything is not possible. We could not test anything with all the possible combinations, because it could be waste of time and money. So testing done on the basis of risk analysis.*
3. *Early testing: It is advisable to start testing early in the life cycle. So one should have to start testing from the requirement gathering stage to avoid future negative consequences.*

4. *Defect clustering: A small number of module contain most of the defects, discovered during the pre-release testing. Defects are not evenly spread in such cases, they are CLUSTERED.*
5. *The Pesticide Paradox: When we are using same testing methods again and again the same method will not be able to find the defects from the system. So we need to tailored our existing methods to best fit as per the project requirement. That is called Pesticide Paradox.*
6. *Testing is context dependent: Each and every project need to be handle differently. We can't proceed by the same testing techniques with 2 different kind of applications, because their functionalities are different. For example; we could not test the pharmacy application and banking application with the same testing methods.*
7. *Absence of error fallacy: When our project is not able to meet the customer requirement, there is no use to find the defects from the system. That is known as absence of error fallacy.*

- **Difference between QA vs QC vs Testing**

QA	QC	Testing
<i>Focuses on processes and procedures rather than actual testing.</i>	<i>Do actual testing, finding and reporting the testing with help of testing</i>	<i>Do actual testing.</i>
<i>Preventive activities</i>	<i>It is corrective process</i>	<i>It is a preventive process</i>
<i>Process Oriented Activities</i>	<i>Product Oriented activities</i>	<i>Product Oriented activities</i>
<i>QA is a subset of Software Test Life Cycle</i>	<i>QC can be considered as a subset of Quality Assurance</i>	<i>Testing can be considered as a subset of Quality Control</i>

- **Difference between smoke and sanity testing**

<i>Smoke Testing</i>	<i>Sanity Testing</i>
<i>Smoke testing is performed after software build to ascertain that critical functionality of Software build is working fine.</i>	<i>After receiving the software build with minor changes in Code or functionality, sanity testing ascertain that all the bugs are fixed, and there is no further issues introduce due to these changes</i>
<i>Smoke testing is performed on Unstable builds</i>	<i>Sanity testing is performed on relatively stable builds.</i>
<i>The main objective of this testing is to verify the “stability” of the system in order to proceed with rigorous testing</i>	<i>The main objective of Sanity Testing is to verify the “rationality” of system to proceed with rigorous testing.</i>
<i>This testing is performed by the developer or tester</i>	<i>This testing is performed by the tester.</i>
<i>Smoke testing is usually documented or scripted</i>	<i>Sanity testing is usually not documented or un scripted.</i>
<i>Smoke testing is performed prior sanity testing</i>	<i>Sanity testing is performed after Smoke testing</i>
<i>Smoke testing is subset of Regression testing</i>	<i>Sanity testing is a subset of Acceptance testing</i>

- **Difference between verification and validation.**

- *Verification is a development process*
Validation is a testing process
- *Verification handle by developer team*
Validation done by Tester team.
- *Verification is done with the help of Inspections, Walkthroughs and review methods.*
Validation is done with testing.

- *The phases of validation: Business Requirement Analysis, Software requirement analysis, Technical Specification, Program Specification.*

The phases of Verification: Unit Testing, Integration Testing, System Testing, User Acceptance Testing.

- **Explain the types of performance testing?**

Software Performance testing is a means of Quality Assurance. It involves testing software application to ensure they will perform well under their expected workload.

Types of Performance Testing

- 1) *Stress Testing: System is checked beyond its specification to check how and when it fails. Stress testing is done to ensure that the system would not crush under crunch conditions. Stress testing is also known as Endurance testing. The most prominent use of stress testing is to determine the limit at which system breaks.*
- 2) *Load testing: is the type of performance testing to check the behavior of system under load. Testing the system with more load, for example with varying load how the system response time change or system got failed. The load testing explains that the how the system will perform when multiple user use the system simultaneously.*
- 3) *Endurance Testing*
- 4) *Spike Testing*
- 5) *Volume testing*
- 6) *Scalability testing*

- **What is error, bug, defect and failure?**

Error: Error is something that can be occurred because of the programming mistake. It is basically the mistake in coding done by the developer.

Detect: When the mistake of developer found by the tester than its known as Defect.

Bug: The developer team accept the mistake found by tester is known as bug.

Failure: The build does not meet the requirement than its called failure.

- **Difference between priority and Severity?**

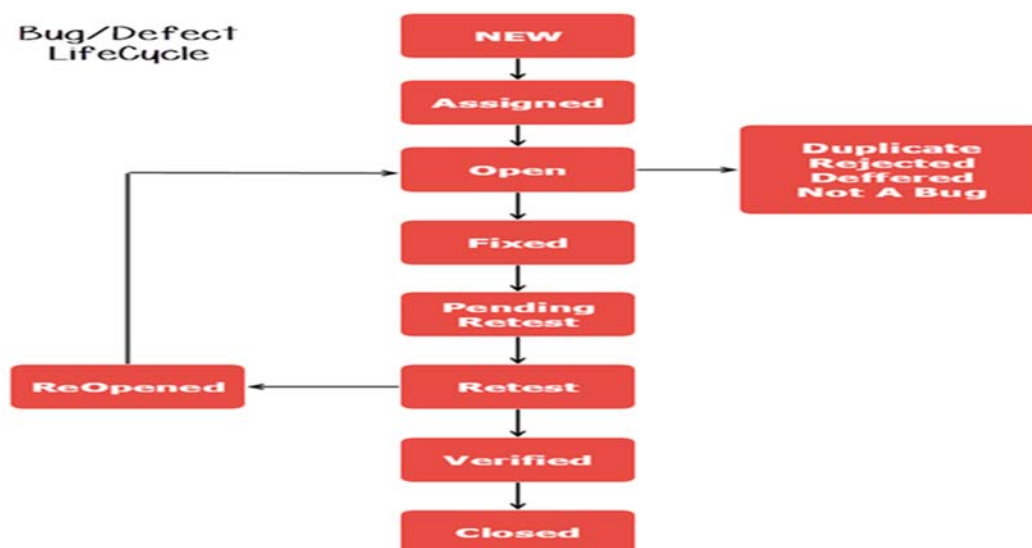
Severity	Priority
Severity is absolute and customer focused	Priority is Relative and business focused
Severity is the extent to which the defect can affect the software program	Priority defines the order in which we can resolve the defect

- **What is the Bug life cycle?**

A computer bug is error, flow, mistake or fault which may stop application from work properly or give incorrect output.

The time span when the Bug introduces in the system till the time it closed, deferred, or rejected is called Bug Life Cycle.

When the bug introduced to the system, it passes through various states till reached the close state.



he stages in Bug life cycle is as below:

- *New: When the bug is introduced to the system.*
- *Assigned: Once the bug is posted by the tester. The tester team approves the bug and assigned it to the developer team.*
- *Open: The developer starts analyzing and fixing process of bug.*
- *Fixed: When a developer makes necessary change to the code, and he or she can make the bug status as “fixed”.*

- *Pending retest: Once the defect is fixed, the developer team gives the code to tester team for check the fixing of defects. At that time its retesting is pending so assigned as "pending retest".*
- *Retest: The tester does retest on the changing code received from developer side.*
- *Verified: The tester received the code from developer after fixing the defect, when the tester retests the code and there is no defect present in the code, then it assigned as verified.*
- *Closed: If the bug not exist in the system, then it's called as closed.*
- *Re-opened: When the tester gives the code to tester after fixing the bugs, tester retest the code, if it's still has a bug then is assigned as re-open. Once again the code goes through bug life cycle.*
- *Duplicate: If the defect is repeated twice, then its assigned as Duplicate.*
- *Deferred: If the bug is not the prime priority, then it could get fixed in the next release, so the status assigned is as "deferred".*
- *Not a bug: If it does not affect the functionality of system, it referred to as "Not a Bug".*
- *Rejected: If the developer feels that the defect is genuine to the developer its referred to as "Rejected".*

• **Explain the difference between Functional Testing and Non Functional Testing?**

<i>Functional Testing</i>	<i>Non Functional testing</i>
<i>Testing the functionality specification of component or system is calls Functional Testing</i>	<i>Testing the attributes of Component or system that does not relate with functionality is called Non-functional Testing.</i>
<i>Functional testing is performed with the specification provided by the client and check if it meets with the user specification or not.</i>	<i>Non Functional testing checks the scalability, performance and other non-functional aspects of software system.</i>
<i>Functional testing performs prior</i>	<i>Non-functional testing should be performed after functional testing.</i>
<i>Manual and automation testing tools is useful for functional testing</i>	<i>Using tools will be effective for this testing.</i>

<i>Business requirements are the input of Functional testing</i>	<i>Performance parameter like speed, performance, scalability is working as a input to Non-functional testing.</i>
<i>Functional testing describes what the system does</i>	<i>Non-functional testing describe how good the product works.</i>
<i>Easy to do manual testing</i>	<i>Hard to do manual testing.</i>
<i>Types of Functional testing are</i> <i>Unit testing</i> <i>System testing</i> <i>Integration testing</i> <i>Black box testing</i> <i>Smoke testing</i> <i>Sanity testing</i> <i>White box testing</i> <i>User acceptance testing</i> <i>Regression testing</i>	<i>Types of Non-functional testing are</i> <i>Performance testing</i> <i>Load testing</i> <i>Stress testing</i> <i>Volume testing</i> <i>Security testing</i> <i>Installation testing</i> <i>Compatibility testing</i> <i>Migration testing</i>

- To create HLR and test case of
 1. (Instagram, Facebook) only first page
 2. Facebook login page:

<https://github.com/alpachavda76/alpa-testing>

- What is the difference between the STLC and SDLC?

SDLC	STLC
<i>Concept: Business analyst gathers requirements. Development team analyze the requirements</i>	<i>Concept: No active involvement but can participate on meeting</i>

<i>Requirement: the development team starts analyzing from the architecture and the design perspective</i>	<i>Requirement: Read requirement document try to understand as requirement: clear, consistence and testable.</i>
<i>Design stage: the architecture of SDLC helps you develop a high level and low level design of the software based on the requirements</i>	<i>Design phase: Read design documents, writing test cases based on design prototype of application.</i>
<i>Development stage: Development team starts developing the software. Integrate with different systems. Once all Integration is done, a ready to test software or product is provided.</i>	<i>Development Stage: Black box tester not active but white box tester do testing.</i>
<i>Testing stage: The actual testing is carried out in this phase. It includes Unit testing, Integration testing, System testing.</i>	<i>Testing stage: all testing needs to do in this stage with manage defect life cycle.</i>
<i>Implementation and Maintenance: Once signed off is received from various testing team, application is deployed in prod environment for real end users.</i>	<i>Implementation and Maintenance: do regression test if any update version or defect fix.</i>

- **What is the difference between test scenario, test case and test script.**

Test Script: The set of sequential instructions that detail how to execute a core business function

Test case: Test case involves the set of steps, inputs and test conditions, which can be used while performing the testing tasks.

Test Scenario: A scenario is any functionality that can be tested. It is also called Test Condition or Test Possibility.

- **Explain what Test Plan is? What is the information that should cover.**

All the projects require set of plans and strategies which define how testing will be conducted.

Test Planning is a document that describing the Scope, approach and schedule for intended test activities. Determine the scope, risks and identifying the objectives of testing.

Define overall testing level process, from entry point to exit point.

Taken decision on, who will do testing? How the testing activities should be performed, how the test results will be evaluated, the list of test activities should be done.

- **What is Priority?**

Priority is Relative and Business focused.

Priority defines the order in which we should resolve the defect.

- **What is Severity?**

Severity is Absolute and Customer Focused

Impact of defect on software system.

- **Bug Categories are**

The categories of defect are

- *Data quality/Database defect: The defect in database, either the deletion or addition entries doesn't react in database.*
- *Critical Functionality Defects: The defects affect the critical functionality of application.*
- *Functionality Defect: The defects affect in functionality of application.*
- *Security Defect: It is necessary to secure your application from internal and external threats. These defects have given higher priority to resolve.*
- *User interface defects: The defects related to UI of application is called User Interface defect. The error message does not display properly.*

- **Advantage of Bugzila.**

- *Advanced search capabilities.*
- *E-mail notifications*
- *Modify/File bugs by Email*
- *Time tracking*
- *Strong security*
- *Customization*
- *Localization*

- **Difference between Priority and severity.**

<i>Severity</i>	<i>Priority</i>
<i>Severity is absolute and customer focused</i>	<i>Priority is Relative and business focused</i>
<i>Severity is the extent to which the defect can affect the software program</i>	<i>Priority defines the order in which we can resolve the defect</i>

- **What are the different methodology in Agile Development model.**

Agile Methodology is a way to manage the project by breaking it up into several phases. It involves constant collaboration with stakeholders and continuous improvement in each phase

Agile Methodology:

- *Individual and interaction over processed and tools: Suppose the team find a bug in Computer Software then they search for another process or tool for resolving the issue. But in Agile it is preferable to interact with the client, manager regarding the issue and resolve the bug.*
- *Customer Collaboration over Contract negotiation: Agile believe that Customer collaboration is more important than Contract Negotiation. Contract Negotiation is important because it gives budget and other requirements for the development of Software Application. But sometimes when you stuck with*

requirements we cannot follow the contract for procedures. We need to collaborate with customer for solution.

- *Working Software over comprehensive document: Agile says Document is important but not as important as Working Software. It does not mean that Document is not mandatory it is also important. For example, if you have 20 pages document but not a single prototype ready than it won't convince. But sometimes customer demand for document as well.*
- *Responding to change over following the plan: In waterfall model, each phase is completed at a time, and the deliverables are predefined. While in agile, requirements can be change in versatile way.*

- **Explain the difference between Authorization and Authentication in Web Testing. What are the common problems face in web testing?**

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

- *Authentication: Accepting invalid username and password.*
- *Authorization: Accessibility of accessing the pages though permission not given.*

- **To create HLR and test case of Web Based**

1. **Whatsapp web (<https://web.whatsapp.com/>)**
2. **InstaGram Web(<https://www.instagram.com/accounts/login/>)**

<https://github.com/alpachavda76/alpa-testing>

- **To create HLR and testcase on this link: <https://artoftesting.com/>**
<https://github.com/alpachavda76/alpa-testing>

- **Write a scenario of only whatsapp chat messages.**

<https://github.com/alpachavda76/alpa-testing>

- **Write a scenario of Pen.**
<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of Pen Stand.**
<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of Door**
<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of ATM**
<https://github.com/alpachavda76/alpa-testing>

- **When to use Usability Testing?**

For working better, the software application should look better. If the user gets confused while using the application, it could not provide result as expected.

Usability testing identifies usability errors in the system in development phase, so it can save a product from failure.

Usability testing performs to check

- *Effectiveness of the system*
- *Accuracy*
- *Efficiency*
- *User Friendliness*

- **What is the procedure for GUI Testing?**

GUI Testing can be processed in 3 ways

- 1) Manual based testing: In manual based testing the GUI tested manually by the tester with reference of the User Requirement document.*
- 2) Record and Replay: In record and replay GUI testing, the testing procedure is recorded, and in replay this procedure replay to execute the tests.*
- 3) Model based testing: In model is a graphical description of system's behavior. It helps us to understand system.*

- **Write a scenario of Microwave oven.**
<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of Coffee Vending Machine.**
<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of chair.**
<https://github.com/alpachavda76/alpa-testing>
- **To create scenario (Positive, Negative)**
 1. Facebook chat on mobile.
 2. Gmail (Receiving mail)
 3. Online shopping to buy product (Flipkart).<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of wrist watch.**
<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of Lift (Elevator).**
<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of Whatsapp Group (generate group).**
<https://github.com/alpachavda76/alpa-testing>
- **Write scenario of Instagram (video call with chat).**
<https://github.com/alpachavda76/alpa-testing>
- **Write a scenario of Whatsapp payment.**
<https://github.com/alpachavda76/alpa-testing>