# Modual 2 assignment 1.what is exploratry testing?

>Exploratory testing is a concurrent process where

Test design, execution and logging happen simultaneously.

>The focus of exploratory testing is more on testing as a “thinking”

activity

**2.what is traceability matrix.**

>To protect against changes you should be able to trace back from every System

component to the original requirement that caused its presence.   
  
**3.What is Integration testing?  
>**Integration Testing *-* Testing performed to expose defects in the

interfaces and in the interactionsbetween integrated components

or systems.

Integration Testing is a level of the software testing process

where individual units are combined and tested as a group.  
There are 2 types.  
 Component Integration Testing

 System Integration Testing

1. **What determines the level of risk?**

A factor that could result in future

negative consequences; usually expressed as impact and

Likelihood.

When testing does find defects, the Quality of the software

system increases when those defects are fixed

**5• What is Alpha testing?**

It is always performed by the developers at the software

development site.

Sometimes it is also performed by Independent Testing Team.

Alpha Testing is not open to the market and public

It is conducted for the software application and project.

It is always performed in Virtual Environment.

**6• What is beta testing?**

It is always performed by the customers at their own site.

It is not performed by Independent Testing Team.

Beta Testing is always open to the market and public.

It is performed in Real Time Environment

1. **What is component testing?**

Component Integration Testing: Testing performed to expose

Defects in the

Interfaces and interaction between integrated components

**8• What is functional system testing?**

Functional System Testing: A requirement that specifies a function

That a system or system component must perform

1. **What is Non-Functional Testing?**

Testing the attributes of a component or system that

do not relate to functionality, e.g. reliability, efficiency, usability,

interoperability, maintainability and portability

1. **What is GUI Testing?**

Graphical User Interface (GUI) 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

1. **What is Adhoc testing?**

Ad hoc testing is an informal testing type with an aim to break the system.

It does not follow any test design techniques to create test cases.

In fact is does not create test cases altogether!

This testing is primarily performed if the knowledge of testers in

the system under test is very high.

1. **What is load testing?**

Its a performance testing to check system behavior under load.

Testing an application under heavy loads, such as testing of a web site

under a range of loads to determine at what point the system’s response time degradesorfail

1. **What is stress Testing?**

It even tests beyond the normal operating point and evaluates how the system

works under those extreme conditions.

 Stress Testing is done to make sure that the system would notcrash under

crunch situations.

 Stress testing is also known as endurance testing

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

White Box Testing: *Testing based on an analysis of the*

*Internal structure of the component or system.*

 Structure-based testing technique is also known as ‘white

box’ or ‘glass-box’ testing technique because here the testers

require knowledge of how the software is implemented, how it

works.

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

Black-box testing: Testing, either functional or non-functional,

without reference to the internal structure of the component or

System.

There are four specification-based or black-box

> Equivalence partitioning

Aim is to treat groups of inputs as equivalent and to select one

representative input to test them all

> Boundary value analysis   
Boundary value analysis is a methodology for designing test cases

that concentrates software testing effort on cases near the limits of valid

ranges Boundary value analysis is a method which refines equivalence

partitioning.

> Decision tables

The techniques of equivalence partitioning and boundary value analysis

are often applied to specific situations or inputs.

> State transition testing

A back box test design technique in which test case design to execute valid and invalid state transition.

1. **Mention what are the categories of defects?**

Functional Defects:

Logic Errors: Flaws in the design or implementation of a system that lead to incorrect functionality.

Calculation Errors: Mistakes in mathematical computations or algorithms.

Performance Defects:

Speed Issues: Slow response times or delays in system performance.

Scalability Issues: Inability of a system to handle an increasing amount of load or users.

Security Defects:

Vulnerabilities: Weaknesses that can be exploited by malicious entities.

Authentication Issues: Problems related to the verification of user identity.

Authorization Issues: Flaws in granting or denying access to resources.

Data Defects:

Data Corruption: Errors leading to the loss or alteration of data.

Data Integrity Issues: Problems with the accuracy and consistency of data.

Documentation Defects:

Incomplete Documentation: Missing or insufficient information in system documentation.

Usability Defects:

User Interface Issues: Problems with the design or functionality of the user interface.

Accessibility Issues: Difficulties for users with disabilities to access or use the system.

Reliability Defects:

Fault Tolerance Issues: System's inability to recover from failures.

Error Handling Problems: Inadequate mechanisms for handling and reporting errors.

Regression Defects:

Unintended Side Effects: Changes in the code that cause previously working functionality to fail.

**16Mention what bigbang testing is?**

 In Big Bang integration testing all components or modules is

integrated simultaneously, after which everything istested as a

whole

1. **What is the purpose of exit criteria?**

How do we know when to stop testing?

Run out of time?

Run out of budget?

The business tells you it went live last night!

Boss says stop?

All defects have been fixed?

When out exit criteria have been met?

**18. When should "Regression Testing" be performed?**

Code Changes:

Integration of Code:

Bug Fixes:

System Upgrades or Patches:

Environment Changes:

Automated Builds:

Periodically in the Development Cycle:

Before Release:

**19What is 7 key principles? Explain in detail?**

 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, t

 Exhaustive Testing is Impossible.

Testing everything including all combinations of inputsand preconditions

is not possible.

 Early Testing.

Testing activities should start as early as possible in the software

or system development life cycle, and should be focused on defined objectives.

 Defect Clustering.

A small number of modules contain most of the defects discovered during

pre- release testing, or are responsible for themost operational failures.

The Pesticide Paradox.

If the same tests are repeated overland 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 needto be written to exercise

different parts of the software or system topotentially

 Testing is Context Dependent.

Testing is done differently in different contexts.

Different kinds of sites are tested differently.

For example

Safety– critical software is tested differently from an

E-commerce site.

 Absence of Errors Fallacy

If the system built is unusable and does not fulfill the user’s needs and

Expectations then finding and fixing defects does not help.

**20.Difference between QA v/s QC v/s Tester**

**QA**

|  |  |
| --- | --- |
| focus on process and procedure rather than conducting actuall testing on the system | Focuses on actual testing by executing Software with intend to identify bug/defect through implementation of procedures and process. |
| Process oriented activities. | Product oriented activities. |
| Preventive activities. | It is a corrective process. |
| It is a subset of Software Test Life Cycle (STLC). | QC can be considered as the subset of Quality Assurance. |
|  |  |
|  |  |

**21.Difference between Smoke and Sanity?**

|  |  |
| --- | --- |
| **smoke** | **sanity** |
| Smoke Testing is performed after software build to ascertain that the critical  functionalities of the program is working fine. | After receiving a software build, with minor changes in code, or functionality,  Sanity testing is performed to ascertain that the bugs have been fixed and no  further issues are introduced due to thesechanges |
| It is executed "before" any detailed functional or regression tests are  executed on the software build. | The goal is to determine that the proposed functionality works roughlyas  expected. |
| In Smoke Testing, the test cases chosen cover the most important  functionality or component of the system. | If sanity test fails, the build is rejected to save the time andcosts involved in a  more rigorous testing. |

1. **Difference between verification and Validation**

|  |  |
| --- | --- |
| **varification** | **validation** |
| 1.verifying documents, design, code  2. Not involve executing code.  3. Usually documents and file  are check by people  4. Uses methods like inspections, do reviews, decision making etc.  5. Check if the software meets the specification.  6. It can catch errors that validation cannot catch and it is low level exercise.  7. It involves the requirements specification, application, software architecture, high level, complete design, and database design etc. | VALIDATION (have we built the right software)  1. Validating and testing of the actual product done by series of dynamic mechanism.  2. Execution of the code.  3. Execution of program using  computer.  4. Uses methods in testing the product. Ex.black box functional testing, gray box testing etc.  5. Check if the software meets the customer expectations and requirements or their satisfaction.  6. Can catch errors that verification cannot catch and it is high level of exercise.  7. Target is the actual product. A module, unit, effective final product. |

1. **Explain types of Performance testing.**

Load Testing:

Objective: Determine the system's behavior under expected load conditions.

Method: Gradually increase the number of concurrent users or transactions until the system reaches its maximum capacity.

Stress Testing:

Objective: Evaluate the system's stability and robustness under extreme conditions.

Endurance Testing (Soak Testing):

Objective: Assess system performance over an extended period under normal or

Focus: Ensure the system can handle continuous usage without deterioration in performance.

Volume Testing:

Objective: Evaluate the system's ability to handle a large volume of data.

Scalability Testing:

Objective: Determine the system's ability to scale up or down in terms of user load, transactions, or data volume.

Isolation Testing:

Objective: Evaluate the performance of individual components or modules in isolation.

Resilience Testing:

Objective: Assess the system's ability to recover from failures or disruptions.

**24What is Error, Defect, Bug and failure?**

**>**A mistake in coding is called error

,> error found by tester is called defect,

>defect accepted by development team then it is called bug,

>build does not meet the requirements then it is failure

**25.Difference between Priority and Severity**

|  |  |
| --- | --- |
| Severity  Defect Severity is specified as the degree of impact that a defect has on the operation of the product  Severity means the seriousness of the defect in the product functionality.  The test engineer determines the severity level of the defect.  It is driven by functionality  Severity status is established on the technical aspect of the product. | Priority  Defect Priority has specified the order in which the developer should fix a defect.  Priority means how soon the bug should be fixed.  Priority of defects is decided in discussion with the manager/client.  It is driven by business value.  Priority status is established on customer requirements. |
|  |  |

**26What is Bug Life Cycle?**

The bug life cycle, also known as the defect life cycle, describes the stages that a software bug goes through from its identification to its resolution. The bug life cycle helps in managing and tracking the progress of bug fixes and ensures that the development

**27.Explain the difference between Functional testing and Non Functional testing**

**Function testing**

> Its execute first.

>Menual or automation tools can be effect for function testing.

>Buisness requirment is input for function testing.

>Easy to do menual testing.

Types of function testing .

1. unit testing
2. Smoke testing
3. Sanity testing

**Non Function testing**

>Its execute after function testing

>Using tools will be effective fort this testing

>Performance parameter like speed ,scability are input for this non function testing.

Types of function testing .

1. performance
2. Load
3. security
4. ** To create HLR & Test Case of 1)(Instagram , Facebook) only first page 2) Facebook Login Page : <https://www.facebook.com/>**

**Facebook**

**<https://docs.google.com/spreadsheets/d/1K_Rs3S-GPznVj2HaoeY7IBeT5dHXiOPY/edit?usp=drive_link&ouid=106021056548910488293&rtpof=true&sd=true>**

**Instagramm**

**<https://docs.google.com/spreadsheets/d/1y9BcKiFXgSwZxUqywppjsiOMvXsU8vOI/edit?usp=drive_link&ouid=106021056548910488293&rtpof=true&sd=true>**

29 What is the difference between the STLC (Software Testing Life Cycle) and SDLC (Software Development Life Cycle)?

|  |  |
| --- | --- |
| sdlc | stlc |
| STLC is related to software testing. | The SDLC is primarily concerned with software development. |
| STLC fewer people are involved. | SDLC a more people involved in all processes, (number of developers). |
| STLC ensures that anything we produce meets customer needs and that the products are of high quality. | The SDLC ensures that we are building the correct thing in the correct manner. |
| STLC is a testing life cycle. | SDLC is a development life cycle. |
| STLC is concerned with both the development and testing processes, but it is primarily concerned with the testing process. | Software development life cycle assures that we deliver high quality software which is as per client needs. |

30. What is the difference between test scenarios, test cases, and test script?

|  |  |  |
| --- | --- | --- |
| Test Scenario  Is any functionality that can be Insted  is derived from test artifacts like Business Requirement Specification (BRS) and Software Requirement Specification (SRS).  Helps test the end-to-end functionality in an Agile way  Is more focused on what to test  Takes less time and fewer  resources to create  Includes an end-to-end functionality to be tested | Test Case  is a set of actions executed to verify particular features o functionality  s mostly delved from test scenarios  Helps in exhaustive testing o an app  Is focused on what to lest and how to test  Requires more resources and time.  Includes test steps, data expected results for testing. | Test Script  is a set of instructions to test an app automatically  Is mostly derived from test Cases  Helps to test specific things repeatedly  is focused on the expected result  Requires less time for testing but more resources for scripts creating and updating  Includes different commands to develop a script |



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

A Test Plan is a comprehensive document that outlines the overall approach, scope, resources, schedule, and activities required for testing a particular system, application, or software.

Test Scope:

In-scope and out-of-scope items for testing.

Features, functionalities, and modules that will be tested.

Test Objectives:

Clear and measurable goals that the testing effort aims to achieve.

Test Strategy:

High-level approach to testing, including methodologies, techniques, and testing levels (e.g., unit testing, integration testing, system testing, acceptance testing).

Test Environment:

Description of the testing environment, including hardware, software, network configurations, and any other dependencies.

Configuration management and version control details.

Test Entry and Exit Criteria:

Criteria that must be met for testing to begin (entry criteria) and for testing to be considered complete (exit criteria).

Test Deliverables:

List of documents and artifacts that will be produced as part of the testing process (e.g., test cases, test scripts, test reports).

Test Schedule:

Timeline for the testing activities, including start and end dates for each phase of testing.

Resource Planning:

Roles and responsibilities of team members involved in testing.

Hardware, software, and human resources required for testing.

Test Execution Plan:

Details on how test cases will be executed, including the order, dependencies, and any specific conditions.

Test Data:

Requirements and strategies for creating, managing, and using test data.

Risks and Contingencies:

Identification of potential risks that may impact the testing process.

Strategies and contingency plans for mitigating risks.

Defect Management:

Procedures for reporting, tracking, and managing defects or issues discovered during testing.

Approvals:

Sign-offs and approvals required at various stages of the testing process.

References:

Any additional reference materials, standards, or guidelines that the testing team should follow.

32. What is priority?

In the context of software testing and project management, "priority" refers to the relative importance or significance assigned to a particular task, feature, or defect. It is a way of determining the order in which work should be addressed based on its importance to the overall project goals.

There are typically three common terms associated with priority:

High Priority:

Issues or tasks that have a significant impact on the project or its users.

Critical functionalities or defects that need immediate attention to ensure the project's success.

Medium Priority:

Issues or tasks that are important but may not require immediate attention.

Functionality or defects that, while significant, may not be critical for the project's immediate success.

Low Priority:

Issues or tasks that have minimal impact on the project's success.

Enhancements or non-critical defects that can be addressed when higher-priority work is complete.

33. **What is severity?**

Severity, in the context of software testing and defect management, refers to the impact or degree of harm that a defect or issue can cause to the system's functionality. It is a measure of how serious or critical a particular problem is in the context of the overall system or application. Severity is independent of the priority assigned to the defect.

Here are common severity levels used in many software development and testing processes:

Critical or Blocker:

Defects that cause a complete system failure or prevent a crucial part of the system from functioning.

These issues usually require immediate attention and resolution before the system can be considered usable.

Major or High:

Defects that have a significant impact on the system's functionality but do not cause a complete failure.

These issues may result in the system not meeting certain requirements or behaving unpredictably.

Minor or Medium:

Defects that have a relatively low impact on the system's functionality.

These issues are generally tolerable and do not significantly affect the overall performance of the system.

Cosmetic or Low:

Defects that have minimal or no impact on the system's functionality.

These issues are usually related to the appearance or user interface and do not affect the core functionality of the system.

34. **Bug categories are…**

Bug categories, also known as defect categories, are classifications used to categorize and organize different types of bugs or defects identified during software testing.

process, here are some common bug categories:

Functional Bugs:

These bugs are related to the incorrect behavior of a function or feature. They may include issues with calculations, data processing, or the overall functionality of a particular component.

Interface Bugs:

Bugs that occur at the interface level, involving the interactions between different modules or components. This can include issues with data exchange, communication protocols, or integration points.

Performance Bugs:

Bugs related to the performance of the software, such as slow response times, resource utilization problems, or bottlenecks that impact the system's efficiency.

Compatibility Bugs:

Issues that arise when the software behaves differently on various platforms, browsers, or operating systems. Compatibility bugs can affect the user experience on different devices.

Usability Bugs:

Bugs that impact the overall user experience, including issues with the user interface, navigation, accessibility, and other elements that affect how easily users can interact with the software.

Security Bugs:

Bugs that pose security risks, such as vulnerabilities, unauthorized access points, or other weaknesses that could be exploited by malicious entities.

Data Bugs:

Bugs related to incorrect data handling, storage, retrieval, or processing. These bugs may lead to data corruption, loss, or inaccuracies.

Installation Bugs:

Bugs that occur during the installation or uninstallation process of the software. This may include issues with configuration, file placement, or system requirements.

Regression Bugs:

Bugs that occur when a new version of the software introduces issues that were not present in the previous version. These bugs may result from code changes or updates.



**35. Advantage of Bugzila . **

ugzilla is an open-source bug tracking system that provides several advantages for software development and quality assurance processes. Here are some key advantages of using Bugzilla:

Bugzilla is an open-source tool, which means it is freely available for use and can be customized according to specific project needs. This can be particularly beneficial for small to medium-sized projects with budget constraints.

Customization and Flexibility:

Bugzilla is highly customizable, allowing teams to define custom fields, workflows, and user permissions to match the specific requirements of their development process. This flexibility ensures that Bugzilla can adapt to various project structures and methodologies.

Web-Based Interface:

Bugzilla provides a web-based interface that allows users to access the bug tracking system from any location with an internet connection. This accessibility facilitates collaboration among team members, including those working remotely or in different locations.

Comprehensive Search and Reporting:

Bugzilla offers powerful search and reporting capabilities, enabling users to create custom queries, filter bugs based on various criteria, and generate detailed reports. This functionality is crucial for project managers and teams to analyze and track the status of bugs and issues.

Email Notifications and Alerts:

Bugzilla can send email notifications to relevant stakeholders when changes occur in the bug tracking system. This helps in keeping team members informed about updates, status changes, or new bug reports without having to check the system constantly.

**36.Difference between priority and priority**

|  |  |
| --- | --- |
| Priority  Defect Severity is specified as the degree of impact that a defect has on the operation of the product.  Severity means the seriousness of the defect in the product functionality.  The test engineer determines the severity level of the defect.  It is driven by functionality.  Severity status is established on the technical aspect of the product. | priority  Defect Priority has specified the order in which the developer should fix a defect.  Priority means how soon the bug should be fixed.  Priority of defects is decided in discussion with the manager/client.  It is driven by business value.  Priority status is established on customer requirements. |

**37.What are the different Methodologies in Agile Development Model? **

Agile is a software development approach that emphasizes flexibility, collaboration, and customer satisfaction.

Scrum:

Key Characteristics:

Iterative and incremental development.

Fixed-length iterations called sprints (usually 2-4 weeks).

Kanban:

Key Characteristics:

Visualizes the workflow on a Kanban board.

Focuses on continuous delivery and flow of work.

No fixed iterations; work is pulled as capacity allows.

Extreme Programming (XP):

Key Characteristics:

Emphasizes technical excellence and frequent releases.

Pair programming (two developers working together at one workstation).

Feature-Driven Development (FDD):

Key Characteristics:

Emphasizes modeling and feature-centric development.

Develops a feature list, and each feature is a short, time-boxed project.

Lean Software Development:

Key Characteristics:

Based on lean manufacturing principles.

Focuses on delivering value to the customer.

Dynamic Systems Development Method (DSDM):

Key Characteristics:

Emphasizes user involvement throughout the project.

Time-boxed and incremental development.

Crystal:

Key Characteristics:

Adaptable and lightweight methodology.

Tailored to the specific characteristics of a project.

Adaptive Software Development (ASD):

Key Characteristics:

Emphasizes continuous adaptation to changing circumstances.

**38.Explain the difference between Authorization and Authentication in Web testing.What are the common problems faced in Web testing?**

|  |  |
| --- | --- |
| **Authentication**  **I**t determines whether users are who they  are claiming to be  Requests the user to validate their  credentials that could be present in the formm of passwords, PIN codes, voice or other biometrics, etc.  It is done before authorization  The data is moved through data tokens  Authentication is visible to the user  This process is changeable by the user  Authentication finds out if the person is a user or not | **Authorization**  It determines the access that should be given or denied to an employee/user  Determines whether the user is allowed access to a resource based on the work policies  It is done after the users successfully authenticate themselves  The data is moved through access tokens  Authorization is not visible to the user  It is not changeable by the user  It determines the permissions that the user |

39. Write a Scenario of Pen

|  |  |
| --- | --- |
| positive | negitive |
| Verify the type of pen ,whether it is a ball pen ,gel pen and ink pen. | en holding is difficult because of not proper designed |
| verify that the pen is able to write in different paper.  verify that the pen writes smothly in  light and hard pressure verify the weight of pen it should be light weight to write properly.  verify the pen have cap or not.  verify the color of the ink.  verify if the pen's ink should not leak at higher altitudes.  verify if the text writeen by a pen is  erasable or not verify that the pen grip is comfortable  or not  verify that the pen parts fit together  securly after reassemble.  verify that ink not dries on paper after  some time  check the visiblity of pen ink on bot ligh and drak-colored paper.  check the opcity of the ink.  verify that text written by pen should not get faded before some time  verify the text written by the pen is  waterproof or not  verify that the mechanism of refill the  pen is easy to operate.  verify if pen can support multipal refils or not  verify that the user of gel pan can easil  change the refile easily.  type of refile or not.  verify that the pen we use it's change  verify that the pen clip can hold the pe  on pocket or not | if pen point not made then how we write. sometime pen weight is heavy so is difficult to  write  sometime pen grip is not there so we don't  write proper  sometime pen refile is empty so we can't write. sometime pen is useless because pen ink is overflow sometime.  if we throw the pen the pointer of pen is broken then the pen useless. sometime pen stop between write because ink  of pen is cold in winter  sometime pen is not useful because some  types of pen not change the refile.  sometime pen in dry if not used for a while. some pen are harmful for the enviroment like  plastic pen etc.  at some plastic pens can easily break in pockets or bages.  pen caps be dangerous for small childrean because the cap of pen is too small and is swallow by the child.  some inks don's flow well in extreme cold ink leaks sometime spoil some document and clothes.  if the cap is left off the link can dry out many pens don't show ink levels, so they can run out un expectedly.  some pens don't write good at high altitudes Il some times inks smudging can ruin written work  some chip pen have poor build and not good  ink quality |

40. Write a Scenario of Pen Stand

|  |  |
| --- | --- |
| Positive  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.  Verify if the pen is with a cap or without a cap.  Verify the color of the ink on the pen.  Check the odor 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.  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.  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.  Verify that the user is able to write normally by tilting the pen at a certain angle instead of keeping it straight while writing.  Check the grip of the pen, and whether it provides adequate friction for the user to comfortably grip the pen.  Verify if the pen can support multiple refills or not. | Negitive  Verify the functioning of a pen at extreme temperatures – much higher and lower than room temperature.  Verify the functioning of a pen at extreme altitude.  Check the functioning of a pen at zero gravity.  Verify the functioning of the pen by applying extreme pressure.  Verify the effect of oil and other liquids on the text written with a pen.  Check if the user is able to write with a pen when used against gravity i.e. upside down.  Verify the functioning of a pen when a user tries to write on unsupported surfaces like glass, plastic, wood, etc.  Verify if the pen works normally or not when used after immersing in water or any other liquid for some period of time.  Verify that the text written by the pen should have consistent ink flow without leaving any blob.  Check the functioning of the pen by applying normal pressure during writing.  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. |

 41.Write a Scenario of Door

|  |  |
| --- | --- |
| positive test scenario  verify the type of doors like single door, bi-  folded door, double door.  verify if the door opens inwards and  outwards  verity door's dimension as per requirement.  verify the matireal of door whether it is wooden, iron, fiber etc  verify the height of the door is proper or  door  verify the width of the door is proper or not  verify the color of the door is specified.  verify the position of hings of the door  verify the quality of hings of the door  verify the strength of hings of the door  verify the type of locks whether it's  simple, digital, bio-matric verify that the door have both side locking  system or not  verify that the door have stopper or not. verify that the door have a spring or not to  automatic closing the door  verify the amount of force requried to open and close then door.  verify that door have sticky system on wall verify that door have a bottom side  stopping system for keeping door open verify the wooden door has polished perfact  or not  verify that the metal door is properly  finished.  verify that painting or design on door. | negative test scenario  It must be strong, otherwise its broken by thief.  Its handles or locking system is not broken, otherwise not lock properly.  verify if digital locking system not working so we can't open the door  verify if biometric system not working so we  can't open the door the wooden door is swelled in raininy season  so sometimes door stuck  iron door is corroded in rainy season. due to moisture the door panel is bent or twisted.  The handle or knob moves or rotates more than it should bacause the handle of door in not fit properly  the door's paint ar finish in coming off in rainy season  physical damage to door's surface sometimes  it's dented or scratched because of smothing  hit by door  sometime door sticks when opening or closing because of not proper installlation  For doors that are supposed to be fire-  rated, because it's help in fire condition Rot in the door frame because the bottom the door jamb is the place where water is most likely to pool  crackes in doors during the humid months. Sometime door frame is splits because one side of wooden door frame dries out  the door dose not adequately block sound. the door is sticks when opening and closing is likely cause by loose hinges  sometimes door loose or missing hardware like stripped screws or inferior quality. Improper threshold in door because of poor  installation  door is discalored because of uv exposure sometimes water staining, after ages |

42. Write a Scenario of ATM

|  |  |
| --- | --- |
| positive  verify the tape of atm machine  whether it's touch screen or keypad or truth  verifyt that the am machine properly inserting a valid cant or not  verifyy that the atm machine accepteit different Banking card or not  verify that the atm machine connected with the cable or will enfyn if the card is not inserted that is no option for continue от естет  verify that the button of atm working properly or not  verifyy that the touch of aim machine worting smooth or nat  verifythat there is option of chessing different language.  verify that there is an option to enter the inno  verify that them in limit number of attempt to enter the number  heck the pin is displayed in masked form.  verify that the user is atted to enter different account type option like aving of current account  Verify that the war is asked to enter different option ite withdrawal  heck balance or mini statement  heck that the user get same number of amount that i enter by m verify that the user ale-wed to enter amount in multiple Be-nominations as per require  verify that the weer not withdraw more amount than the total available balance  verify that there is option to get transaction details in panted fome,  verify that the uter session timeout is working  Verify that there is an of transaction amount on daily basis verify if the atm machine have no money su it's show anti-per message | negitive  sometime ate machine nut working due to power  sometime atm machine due to network connectivity  problems  some time case dispenser of atm in stuck due to  physical obstruction    sometime ate machine not acepted valid due to Punctity empr  sometime atm machine show income account  balance stue to software and database error  sometime atm machine touchscreen hot woning and unresponsiveness due to harware issue and display  sometime key patte button of atm machine t  worning due to testinal sometime atm machine not give proper print of  reciept because of receipit printer in fallure sometime atm machine not the reciept of  withdrawal because of paper jam in redirect printer sometime pie pad of atm machine is nut showing  number of pad properly because of poor quality alm machine not selected language properly it's result  in customers contingent difficulties when sing atm machine  Arm machine cash case settel not lock securely  due to  atm machine not clean properly due to lack of  inregidor maintenance  imaruper mariance  sometime alm machine software update can Introduce bugs and glitches  sometime atm amputated invalide and expire card  sometime user card details can show after wer the card  l atm machine not show proper out of  money massage due to improper functionality machine finds wrong pin due to software and dantiase  atom machine accepted less then 100 amount due to  software atm machine not have transitionally basis |



43.When to used Usablity Testing? Write

Usability testing should be incorporated into the software development lifecycle at various stages to ensure that a product is user-friendly, intuitive, and meets the needs of its intended users.

Early Design Phase:

Usability testing can be conducted on low-fidelity prototypes to gather initial feedback on the basic design and user interactions.

Mid-Development Phase:

Iterative Testing: Conduct usability tests during development iterations to address and refine specific features or components.

Pre-Launch Phase:

Beta Testing: Before a public release, usability testing in a beta environment can uncover issues with a larger user base and diverse usage scenarios.

Post-Launch Phase:

Ongoing Monitoring: Continuously monitor user feedback and behavior after the product is launched to identify and address emerging usability issues.

Redesign or Major Updates:

Significant Changes: When making substantial changes to the product, conduct usability testing to ensure that users can easily adapt to the new features or interface.

Usability Issues Identification:

Task-Specific Testing: Focus usability testing on specific tasks or user journeys to identify issues related to those activities.

Competitive Analysis:

Comparative Usability Testing: Evaluate the usability of your product against competitors to identify areas for improvement and competitive advantages.

**44. What is the procedure for GUI Testing?**

GUI testing is a type of software testing that focuses on verifying whether the graphical interface of a software application is functioning correctly and meeting the specified requirements.

Understanding Requirements:

Begin by thoroughly understanding the GUI design specifications and requirements.

Test Planning:

Develop a comprehensive test plan that outlines the scope, objectives, resources, and schedule for GUI testing.

Environment Setup:

Set up the testing environment with the necessary hardware, software, and configurations to replicate the end-user environment.

Functional Testing:

Verify that all GUI components function as intended. This includes checking buttons, links, menus, input fields, and other interactive elements for proper behavior.

Layout and Design Consistency:

Confirm that the GUI adheres to design specifications and standards.

Check for consistency in layout, font styles, colors, and other visual elements across different screens and resolutions.

Compatibility Testing:

Test the GUI across various browsers, operating systems, and devices to ensure compatibility.

Error Handling:

Evaluate how the application handles errors. Check for error messages, warnings, and prompts to guide users in case of incorrect inputs or system issues.

Performance Testing:

Assess the responsiveness of the GUI under different load conditions.

Verify that the GUI remains responsive and functions correctly even when the application is under stress.

Usability Testing:

Evaluate the overall user experience by observing how users interact with the GUI.

Accessibility Testing:

Ensure that the GUI is accessible to users with disabilities by testing with screen readers, keyboard navigation, and other assistive technologies.

Verify that the GUI does not expose any security vulnerabilities, such as input validation issues or unauthorized access to sensitive information.

Documentation and Reporting:

Document test cases, test results, and any issues encountered during GUI testing.

Regression Testing:

Perform regression testing after making changes to the GUI or the underlying code to ensure that existing functionality remains intact.

 45.Write a scenario of Microwave Owen

|  |  |
| --- | --- |
| verify the dimesion of the ovan as per  reuired  verify that the ovan material is good  verify that the ovan heat food properly verify that the oven head food within spacefied time.  check the oven functioning at maximum  temperature.  verify the oven functioning at minimum  temperature  verify that the plate speed of the oven in  optimal.  verify that the plate speed of the oven in  optimal and not to high to spill the food kept  over it  verify the the oven door closed properly  verify that oven door opend properly.  verify the weight of the ovan as per  specification verify the battery requirement of the  microwave oven.  verify that the text writen on the ovan is  viable or not  verify that the digital display is visable or not  verify that the temperature regulator is  working or not  verify that that the temperature regulator is  working smoothly or not  check the ovan kind of food  functionality with different  check the ovan functionality with different kind of food with different temp.  check the power cord of ovan is long  enough  check the company name on the oven is  visible or not | Negative  sometime oven not heating food properly due less power supply  plate of he ovan is not spinning properly due to  motor issue or power issue verify that the plate of the oven is not broked due to  overheat  sometime oven is not turn  on due to pluging error microwave door not closed properly due to hinges of  doors  microwave door not opend somthely due to hinges of  doors  microwave oven not stop running if timers runs out because of system error oven touch pad is not working proper because of un responsive button  oven button not working  properly due to control panel and functionalist issue Display of the oven is not working proper due poor  display uses  Display of the ovan is not brighter due to long time uses or poor quality display  uses  sometime ovan is  overheated because of long time of uses  sometime ovan smell badly due to food can burn and  produce smell  sometime text written on the ovan is not visable because of poor qulity  color of the oven is faded due to long time uses or poor quality |

46.Write a scenario of Coffee vending Machine

|  |  |
| --- | --- |
| Positive  verify that the dimension of the coffee machine verify the the body and quality of machine as per require  verify the color of coffee vending machine  verify that the brand is correctly visable or not verify the input for coffee ingredients like milk, water and powder etc.  verify that the power require for machine  verify the weight of the machine verify that the coffee should not leak if when not is used  verify that the display show correct information. check the machine in on and off using power button  check the indicator light is working or not  check the all functionality in working or not  verify that the each button have an image so it's indicating what is perform  verify that the complate quantity of a coffee should get poured in single operation  verify that the functionally to clean system working  properly  verify that system should display error if there have  some missing ingredients  verify that the machin should not make to much sound  check the amount of time taken to make a single  coffee  check the cup holder dimension as per market  standrad  check the machine performance in low voltage and  high voltage | Negative  sometime dimension of coffee vending machine is not proper due not properly made  sometime color of machine in fadedd because  of long time uses or poor color uses  sometime brand name of machine is not show properly due to poor quality  sometime machine is not on due to electric  connection issue sometime machine make to much noise due to  user for a long time  sometime coffee tastes bad due to stale coffee beans or dirty components  sometime coffee is week due to incorrect  brewing time  sometime marching is heating because of long time of uses or sometime system is faliure sometime machine is frezzing due to cooling  system is failure  machine display error because of software and  screen issue  display is not show brighter because of poor  display quality or software issue Sometime cup is stuck in machine due ro  incorrect cup or misaligned cup dispenser  sometime button of machine is not working due to poor funcionlity  sometime image on the button in not visable  properly due poor quality or long time issue sometime water in the machine is overflow  because of water sensor issue Sometime machine has some hygiene issue  because of improper maintance sometime display screen is flicking due to losse cable  sometime machine brewed coffee drips after cup removal because of some sensor problem check the sensor off machine if there is no cup s placed.  sometime coffee is heat to much due heating Functionlity problem |



47.Write a scenario of chair

|  |  |
| --- | --- |
| positive test scenario  verify the type of chair whether it's plastic, iron, steel, wood etc  verify that the chair is enough to take an average human load  verify that the cushion provide with the chair or not  verify that the chair cushion if fitted proper or not  sit on the chair with average force to chack the quality of chair  check the gently punch on chair to ensure the stability of chair  check the leg of chair is perfectly level to the floor or not verify the usability of chair whether it's office chair or normal home use chair  verify that the chair has back support or not  check that there is support for hands or not in the chair  check the height of the chair as per sanctification.  check the weight of the chair  check the color of the chair  check the chair material is brittle or not  check the dimension of chair as per specification  check if the chair has an adjustment functionality or not.  check the functionality of office chair for sit up or down  verify that the wheel of the office chair is working good or  not verify that the wheel of chair in do not stuck between moving from one side to another  check the logo of company is printed properly or not | negative test scenario  check the balance of the chair with one arm  check the balance of the chair with three  leg  some times chair have weak and unstable  joint because of poor quality some times chair have unirven legs length. because of poor joint connection  chair has a poor finish or low quality wood  chair has a fading color because of low  quality paint  sometimes chair have weak armrests because of thin materials  chair has a defect wheels for rolling because  of jammed and poor quality chair has a uneven padding because of  manufacturing error  sometimes chair have a sharp edges reason incompetence fishing and broken.  parts  sometimes plasic chair in crcked because of  low quality material and store chair in  extreme temp sometimes chair tilts to one side because of  uneven weight distribution  chair have rough surface  reason:poor finshing  sometime chair is difficult to clean because of fabric of chair is not good  chair missing branding or lables because of manufacturing problem  sometimes remote contori chair is not working properly because of functionlity  error sometimes cushion of chair is tattered  because of poor quality some times switch to adjust the chair is  nor working because of functionality  problem  sometimes chair have a stiff backrest tilt  because of poor lubricated some light weight chair is unstable because of the use of light weight matrall |

**To create HLR & TestCase of WebBased (WhatsApp web , Instagram)**

**Whats web**

**<https://docs.google.com/spreadsheets/d/1N5LmtZP0KdnTAMGjF6CHxA24ufu6WpdH/edit?usp=drive_link&ouid=106021056548910488293&rtpof=true&sd=true>**

Instagramm

**<https://docs.google.com/spreadsheets/d/1-X7FMGbB2p6yCVgBZlPQFIhr7JvGdYrh/edit?usp=drive_link&ouid=106021056548910488293&rtpof=true&sd=true>**

1. To create HLR and TestCase on this Link. https://artoftesting.com/.

**<https://docs.google.com/spreadsheets/d/1m27Z9bFnkU16538J0IA3T1HltDJt-Cu3/edit?usp=drive_link&ouid=106021056548910488293&rtpof=true&sd=true>**

**50.To Create Scenario (Positive & Negative)**

Gmail

<https://docs.google.com/spreadsheets/d/1TvAONa8ZIeWL3__Ghn43qn3WvfQXmNMh/edit?usp=drive_link&ouid=106021056548910488293&rtpof=true&sd=true>

Flipkart

<https://docs.google.com/spreadsheets/d/1Hqy3I09JyPGdmPluwlP8orcFYom1M1cP/edit?usp=drive_link&ouid=106021056548910488293&rtpof=true&sd=true>

51.Write a scenario of only Whatsapp chat messages

|  |  |
| --- | --- |
| positive | negitive |
| Verify that on downloading the Whatsapp application, users can register using a new mobile number.  Check the maximum number of incorrect attempts allowed while filling out the verification code.  Verify that registering an existing mobile number for new user account registration is not allowed.  Verify that on successful registration all the contacts in the user’s contact directory get imported to the Whatsapp contact list.  Verify that the user can set DP and status on Whatsapp.  Verify that the user can update the existing DP and Whatsapp status.  Verify that the user can send messages to any individual selected from his contact list.  Verify that ‘Chats’ window contains all the chat list with DP and name and last message preview of the other person with whom chat was initiated.  Verify that clicking a chat in the chat list opens a new window containing all the chats received and sent with the other person.  Verify that the user can check the message delivered and read the time for a message in the ‘Message Info’ section.  Verify that the user can share or receive contact with the other person.  Verify that the user can create a group by adding multiple people from his contact list.  Verify that the user can send and receive the message in group chats.  Verify that users can send and receive images, audio, video, and emoticons in the chat with individuals.  Verify that users can send and receive images, audio, video, and emoticons in group chats.  Verify that the user can send and receive chats in the secondary languages available.  Verify that users can delete text, images, audio, and video messages within a chat.  Verify that users can clear their complete chat history in an individual or group chat.  Verify that users can archive chats in an individual or group chat.  Verify that users can block a user to prevent any message from getting received from the blocked contact.  Verify that the user makes WhatsApp calls to the person in his contact list.  Verify that the user can receive WhatsApp calls from the person in his contact list.  Verify that users can mark chats as favorites and access all chats marked as favorites from the ‘Favorites’ section. | You can try to send more than 30 images at a time.  Check that are you able to send large videos(exceeds limited size) or not.  Try to send hundreds of messages at the same time to a person and then check the behaviour of this application.  Try to register with the same number on two different devices at the same time.  Check that are you able to forward a message to more than five people.  Try to add more than 256 members in a group.  Members might share spam, chain messages, or irrelevant content, leading to a cluttered and unpleasant group experience.  In large groups, especially those with people you don't know well, there's a risk of sensitive information being shared unintentionally, leading to privacy concerns.  False information and rumors can spread quickly in group chats, potentially causing confusion or panic among members.  In active groups, constant notifications can become overwhelming, disrupting members' daily activities and causing frustration.  Differences in opinions or misunderstandings can escalate into conflicts, and the text-based nature of messaging might exacerbate the situation.  Group dynamics can lead to exclusionary behavior, forming cliques within the group and making some members feel left out.  Some members may use the group as a platform for personal issues or grievances, invading the personal space of others.  If the group admin is not vigilant, there's a risk of unauthorized people gaining access to the group, leading to potential data breaches or misuse.  Excessive use of WhatsApp groups can contribute to smartphone addiction and distract individuals from their work or other important activities.  Legal Issues:  In some cases, group members might engage in activities or discussions that could lead to legal consequences, putting both individuals and the group at risk. |

52Write a Scenario of whatsapp Group (generate group)

|  |  |
| --- | --- |
| positive | negitive |
| Verify that on downloading the Whatsapp application, users can register using a new mobile number.  Verify that for a new mobile number user will get a verification code on his mobile and filling in the same verifies the new user account.  Verify the maximum number of incorrect attempts allowed while filling out the verification code.  Verify that registering an existing mobile number for new user account registration is not allowed.  Verify that on successful registration all the contacts in the user’s contact directory get imported to the Whatsapp contact list.  Check that the user can create a Whatsapp group.  The user can set a name for the created group.  Check that the user can add and save the group description.  Verify that the user can make multiple people as group Admin.  Check that only group admins can add people to the group.  Is there any option to mute group notifications for some time?  Is there any option to add people to the group by sharing a link?  The admin can delete users from this group.  If a person is removed from the group, he/she will not be able to see any updates.  Check that user can exit the group by clicking on the exit button or not.  Verify that the admin can delete the group.  Check that are you able to create a group without adding any member. | Members may share false information, rumors, or unverified news, leading to confusion and misinformation.  Chain messages and excessive forwarding of content can clutter the group and annoy members.  Sensitive information may be shared unintentionally, raising privacy and security concerns.  Members may find the continuous flow of notifications disruptive and overwhelming.  Differences in opinions may escalate into conflicts, leading to a negative atmosphere within the group.  Group dynamics may result in exclusionary behavior, forming cliques and leaving some members feeling isolated.  Invasion of Personal Space:  Some members may use the group to address personal issues, invading the personal space of others.  Members might use the group to promote products or services without consent, leading to spam-like behavior.  Inadequate security measures may pose risks of unauthorized access or data breaches.  Inactivity and Lack of  Some members may not actively participate, leading to a lack of engagement and meaningful interaction.  Irrelevant discussions and off-topic conversations can waste members' time and distract from the group's purpose.  Instances of impersonation or fake accounts may lead to confusion and mistrust among group members.  N  Group rules and guidelines may be ignored, resulting in a chaotic and unregulated environment.  Bullying and Harassment:  Negative interactions, bullying, or harassment can occur, creating a toxic atmosphere within the group.  Group discussions or activities may inadvertently involve members in legal issues, causing complications and potential consequences. |

53. Write a Scenario of Whatsapp payment

|  |  |
| --- | --- |
| positive | negitive |
| Verify that Users can make payments directly within the WhatsApp app, eliminating the need to switch between multiple apps or platforms.  Verify that The user-friendly interface simplifies the process of sending and receiving money, making it accessible to a wide range of individuals.  Verify that Transactions are processed in real-time, allowing for quick and immediate transfers between users.  Verify that Payments are seamlessly integrated with the user's contact list, making it easy to send money to friends and family.  Verify that WhatsApp Payments incorporates security measures such as UPI (Unified Payments Interface) to ensure the safety of financial transactions.  Verify that In many cases, WhatsApp Payments does not charge additional fees for transactions, making it a cost-effective option for users.  Verify that Users can conveniently pay utility bills, mobile recharge, and other expenses directly through the app.  Verify that Group members can split bills for shared expenses like meals or gifts, simplifying the process of settling expenses within a group.  Verify that Small businesses can receive payments from customers through WhatsApp, providing a convenient and efficient payment solution.  Verify that WhatsApp Payments can contribute to financial inclusion by providing individuals who don't have traditional bank accounts with access to digital transactions.  Verify that Users can easily track their transaction history within the app, promoting transparency and record-keeping.  Verify that Users receive notifications for successful transactions and reminders for pending payments, enhancing financial accountability.  Verify that In some regions, WhatsApp Payments allows users to make international transactions, facilitating global financial transactions.  Integration with Business Accounts:  For businesses, WhatsApp Payments can be integrated into their business accounts, streamlining transactions with customers.  Encouraging Cashless  Verify that By providing a convenient and secure platform for digital payments, WhatsApp Payments encourages users to adopt cashless transactions, contributing to a digital economy. | Check the Attempt to make a payment with incomplete or invalid transaction details, ensuring the system handles such cases appropriately.  Check the Test making a payment with an account that has insufficient funds and verify that the system correctly handles and communicates this situation.  Check the Introduce network connectivity issues during a transaction to check how the system handles interruptions and whether it recovers gracefully.  Check the Allow a transaction to reach the timeout period to ensure that the system handles timeouts appropriately without compromising security.  Check the Attempt to make duplicate transactions and verify that the system detects and prevents the processing of duplicate payments.  Check the Scan or input an invalid QR code during a payment to verify that the system detects and rejects invalid codes.  Check the Enter an incorrect PIN multiple times during a transaction and verify that the system enforces lockout policies and security measures.  Check the Test scenarios where user authentication fails due to incorrect credentials or other issues, ensuring the system responds appropriately.  Check Test making payments on unsupported devices or outdated operating systems to ensure the app provides clear messages and prevents transactions.  Check the Intentionally crash the WhatsApp or payment app during a transaction and observe how the system recovers, ensuring no data loss or security vulnerabilities.  Test cross-border transactions with currencies or accounts that are not supported, ensuring the system handles these cases appropriately.  Attempt transactions exceeding set limits to verify that the system correctly enforces transaction limits and communicates them to users.  Test making payments over unsecured or public Wi-Fi networks to identify any vulnerabilities related to data security.  Verify that the payment app doesn't request or access unnecessary permissions, ensuring user privacy is maintained.  Simulate server downtime during a transaction to check if the system gracefully handles such situations and maintains data integrity. |