**Assignment: - 2**

1. What the difference between functional and non-functional testing?

|  |  |
| --- | --- |
| **Functional Testing** | **Non-functional Testing** |
| Functional Testing is performed using the functional specification provided by client and verifiers the system against the functional requirement. | Non-functional Testing checks the performance, reliability, scalability and other non-functional aspects of the system. |
| Functional testing execute first. | Non-functional testing should be execute after functional testing. |
| Manual testing or automation testing tool can be used for functional testing. | Using tool will be effective for this testing. |
| Business requirements are inputs for this testing. | Performance parameters like speed, scalability are the inputs for this testing. |
| Functional testing describe what the product does. | Non-functional testing describes how good product works. |
| Easy to do manual testing. | Tough to do manual testing. |
| Types of Functional testing are  ∙ Unit Testing  ∙ Smoke Testing  ∙ Sanity Testing  ∙ Integration Testing  ∙ White box testing  ∙ Black Box testing  ∙ User Acceptance testing  ∙ Regression Testing | Types of Nonfunctional testing are  ∙ Performance Testing  ∙ Load Testing  ∙ Volume Testing  ∙ Stress Testing  ∙ Security Testing  ∙ Installation Testing  ∙ Penetration Testing  ∙ Compatibility Testing  ∙ Migration Testing |

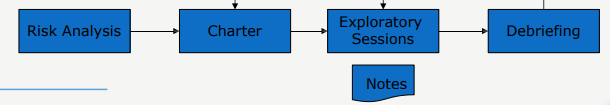
1. Difference between priority and severity?
   * Priority is a term that define how fast we need to fix that defect.
   * Severity is basically a parameter that denotes the total impact of given defect on system.
2. What is error, defect, bug and failure?

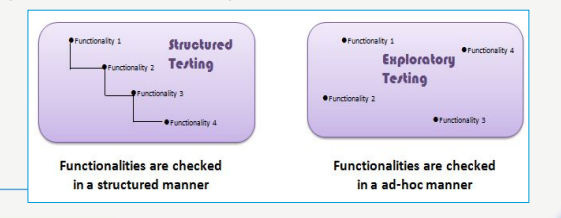
* Failure: The inability of a system or component to perform its required functions within specified performance requirements. See: bug, crash, exception, and fault.
* Bug: A fault in a program which causes the program to perform in an unintended or unanticipated manner. See: anomaly, defect, error, exception, and fault. Bug is terminology of Tester.
* Fault: An incorrect step, process, or data definition in a computer program which causes the program to perform in an unintended or unanticipated manner. See: bug, defect, error, exception.
* Defect: Commonly refers to several troubles with the software products, with its external behavior or with its internal features.

1. Type of performance testing?
   * Load Testing
   * Stress Testing
   * Endurance Testing
   * Spike Testing
   * Volume Testing
   * Scalability Testing
2. Difference between verification and validation?

|  |  |
| --- | --- |
| **Verification** | **Validation** |
| Verification is the process of evaluating the artifacts of software development in order to ensure that the product being developed will comply with the standards. | Validation is the process of validating that the developed software product conforms to the specified business requirements. |
| It is a static process of analyzing the documents and not the actual end product. | It involves dynamic testing of a software product by running it. |
| Verification is a process oriented approach. | Validation is a product-oriented approach. |
| Answers the question – “Are we building the product right?” | Answers the question – “Are we building the right product?” |
| Errors found during verification require lesser cost/resources to get fixed as compared to be found during the validation phase. | Errors found during validation require more cost/resources. Later the error is discovered higher is the cost to fix it. |
| It involves activities like document review, test case review, walk-throughs, inspection etc. | It involves activities like functional testing, automation testing etc. |

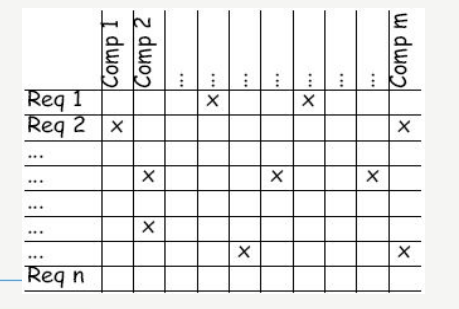
1. What is Exploratory Testing?
   * Exploratory testing is a concurrent process where
     1. Test design, execution and logging happen simultaneously
     2. Testing is often not recorded
     3. Makes use of experience, heuristics and test patterns
     4. Testing is based on a test charter that may include
        1. Scope of the testing (in and out)
        2. The focus of exploratory testing is more on testing as a “thinking” activity.
        3. A brief description of how tests will be performed
        4. Expected problems
     5. Is carried out in time boxed intervals
   * More structured than Error guessing
   * Though the current trend in testing is to push for automation, exploratory testing is a new way of thinking. Automation has its limits
     1. Is not random testing but it is Adhoc testing with purpose of find bugs
     2. Is structured and rigorous
     3. Is cognitively (thinking) structured as compared to procedural structure of scripted testing. This structure comes from Charter, time boxing etc.
     4. Is highly teachable and manageable
     5. Is not a technique but it is an approach. What actions you perform next is governed by what you are doing currently





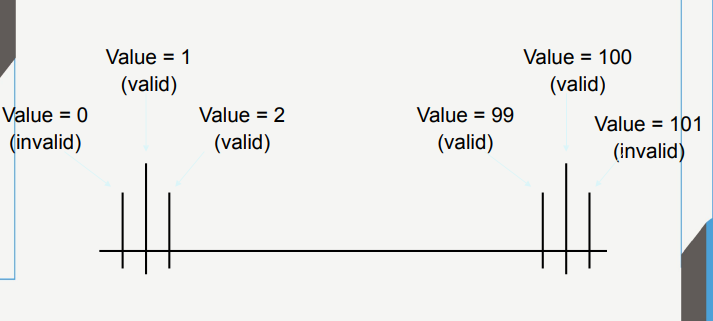
1. What is traceability matrix?

* **Traceability**
  + Test conditions should be able to be linked back to their sources in the test basis, this is known as traceability.
  + Traceability can be horizontal through all the test documentation for a given test level (e.g. system testing, from test conditions through test cases to test scripts) or it can be vertical through the layers of development documentation (e.g. from requirements to components).
* **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.
  + A software process should help you keeping the virtual table up-to-date.
  + Simple technique may be quite valuable (naming convention)



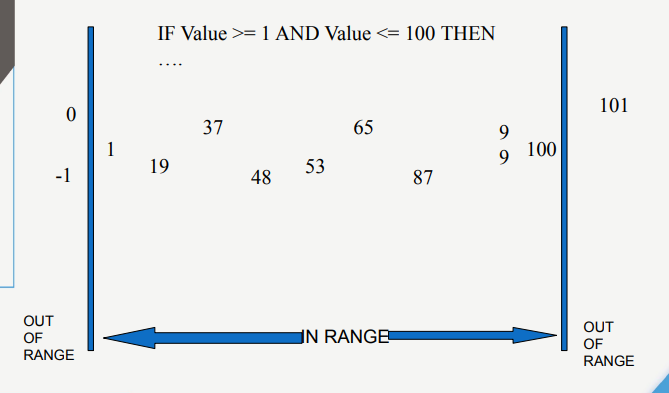
1. What is Boundary value testing?

* 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.
* Boundary value analysis generates test cases that highlight errors better than equivalence partitioning.
* The trick is to concentrate software testing efforts at the extreme ends of the equivalence classes.
* At those points when input values change from valid to invalid errors are most likely to occur.
* Boundary Value Analysis (BVA) uses the same analysis of partitions as EP and is usually used in conjunction with EP in test case design



1. What is Equivalence partitioning testing?

* Aim is to treat groups of inputs as equivalent and to select one representative input to test them all
* EP can be used for all Levels of Testing
* Equivalence partitioning is the process of defining the optimum number of tests by:
  + Reviewing documents such as the Functional Design Specification and Detailed Design Specification, and identifying each input condition within a function,
  + Selecting input data that is representative of all other data that would likely invoke the same process for that particular condition.
* If we want to test the following IF statement: “If value is between 1 and 100 (inclusive) (e.g. value >=1 and value <=100) Then...”
* We could put a range of numbers as shown in the below figure.



1. What is Integration testing?

* Integration Testing - Testing performed to expose defects in the interfaces and in the interactions between integrated components or systems
* Integration Testing is a level of the software testing process where individual units are combined and tested as a group.
* The purpose of this level of testing is to expose faults in the interaction between integrated units. Test drivers and test stubs are used to assist in Integration Testing.
* Integration testing tests integration or interfaces between components, interactions to different parts of the system such as an operating system, file system and hardware or interfaces between systems.
* Integration testing is done by a specific integration tester or test team.
* Components may be code modules, operating systems, hardware and even complete systems
* There are 2 levels of Integration Testing
  + Component Integration Testing
  + System Integration Testing

1. What determines the level of risk?
   * A properly designed test that passes, reduces the overall level of Risk in a system
   * 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
   * The Quality of systems can be improved through Lessons learned from previous projects
   * Analysis of root causes of defects found in other projects can lead to Process Improvement
   * Process Improvement can prevent those defects reoccurring
   * Which in turn, can improve the Quality of future systems
   * Testing should be integrated as one of the Quality assurance activities
2. 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.
   * It is always performed within the organization.
   * It is the form of Acceptance Testing.
   * Alpha Testing is definitely performed and carried out at the developing organizations location with the involvement of developers.
   * It comes under the category of both White Box Testing and Black Box Testing.
3. 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 usually conducted for software product.
   * It is performed in Real Time Environment.
   * It is always performed outside the organization.
   * It is also the form of Acceptance Testing.
   * Beta Testing (field testing) is performed and carried out by users or you can say people at their own locations and site using customer data.
   * It is only a kind of Black Box Testing.
4. What is component testing?
   * Component (Unit) – A minimal software item that can be tested in isolation. It means “A unit is the smallest testable part of software.”
   * Component Testing – The testing of individual software components.
   * Unit Testing is a level of the software testing process where individual units/components of a software/system are tested. The purpose is to validate that each unit of the software performs as designed.
   * Unit testing is the first level of testing and is performed prior to Integration Testing.
   * Sometimes known as Unit Testing, Module Testing or Program Testing
   * Component can be tested in isolation – stubs/drivers may be employed
   * Unit testing frameworks, drivers, stubs and mock or fake objects are used to assist in unit testing.
   * Functional and Non-Functional testing
   * Unit tests are typically written and run by software developers to ensure that code meets its design and behaves as intended with debugging tool.
   * A unit is the smallest testable part of an application like functions/procedures, classes, interfaces.
   * The goal of unit testing is to isolate each part of the program and show that the individual parts are correct.
   * A unit test provides a strict, written contract that the piece of code must satisfy.
   * As a result, it affords several benefits.
   * Unit tests find problems early in the development cycle.
   * Unit testing is performed by using the White Box Testing method.
   * Test Approach :
   * Test-First/Test-Driven approach – create the tests to drive the design and code

Construction!

* + Instead of creating a design to tell you how to structure your code, you create a test that defines how a small part of the system should function.
  + Three steps:
    1. Design test that defines how you think a small part of the software should behave (Incremental development).
    2. Make the test run as easily and quickly as you can. Don't be concerned about the design of code, just get it to work!
    3. Clean up the code. Now that the code is working correctly, take a step back and refactor to remove any duplication or any other problems that were introduced to get the test to run.
  + Unit testing in Extreme Programming involves the extensive use of testing frameworks. A unit test framework is used in order to create automated unit tests. Unit testing frameworks are not unique to extreme programming, but they are essential to it.
  + Below we look at some of what extreme programming brings to the world of unit testing:
    1. Tests are written before the code
    2. Rely heavily on testing frameworks
    3. All classes in the applications are tested
    4. Quick and easy integration is made possible

1. What is functional system testing?
   * Functional System Testing : A requirement that specifies a function that a system or system component must perform
   * A Requirement may exist as a text document and/or a model
   * There is two types of Test Approach
     1. Requirement Based Functional Testing
     2. Process Based Testing
   * Functional System Testing Functionality As below:

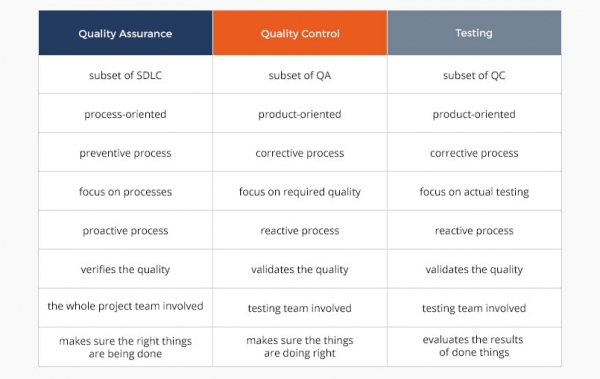
|  |  |
| --- | --- |
| **Accuracy** | Provision of right or agreed result or effects |
| **Interoperability** | Ability to interact with specified systems |
| **Compliance** | Adhere to applicable standards, conventions, regulations or laws |
| **Auditability** | Ability to provide adequate and accurate audit data |
| **Suitability** | Presence and appropriateness of functions for specified tasks |

1. What is Non-Functional Testing?
   * It is the testing of “how” the system works. Non-functional testing may be performed at all test levels.
   * The term non-functional testing describes the tests required to measure characteristics of systems and software that can be quantified on a varying scale, such as response times for performance testing.
   * To address this issue, performance testing is carried out to check & fine tune system response times. The goal of performance testing is to reduce response time to an acceptable level
   * Hence load testing is carried out to check systems performance at different loads i.e. number of users accessing the system
2. 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.
   * WHAT DO YOU CHECK IN GUI TESTING?
     + Check all the GUI elements for size, position, width, length and acceptance of characters or numbers. For instance, you must be able to provide inputs to the input fields.
       1. Check you can execute the intended functionality of the application using the GUI
       2. Check Error Messages are displayed correctly
       3. Check for Clear demarcation of different sections on screen
       4. Check Font used in application is readable
       5. Check the alignment of the text is proper
       6. Check the Color of the font and warning messages is aesthetically pleasing
       7. Check that the images have good clarity
       8. Check that the images are properly aligned
       9. Check the positioning of GUI elements for different screen resolution.
3. What is Adhoc testing?
   * Adhoc 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.
   * Testers randomly test the application without any test cases or any business requirement document.
   * Adhoc Testing does not follow any structured way of testing and it is randomly done on any part of application.
   * Main aim of this testing is to find defects by random checking.
   * Adhoc testing can be achieved with the testing technique called Error Guessing.
   * Error guessing can be done by the people having enough experience on the system to “guess” the most likely source of errors.
   * The Error guessing is a technique where the experienced and good testers are encouraged to think of situations in which the software may not be able to cope.
   * Some people seem to be naturally good at testing and others are good testers because they have a lot of experience either as a tester or working with a particular system and so are able to find out its weaknesses.
   * This is why an error guessing approach, used after more formal techniques have been applied to some extent, can be very effective.
   * It also saves a lot of time because of the assumptions and guessing made by the experienced testers to find out the defects which otherwise won’t be able to find.
     + 1. Using experience to postulate errors.
       2. Use Error Guessing to Complement Test Design Techniques.
4. What is load testing?
   * Load testing - It’s 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 degrades or fails.
   * Load testing is a kind of performance testing which determines a system’s performance under real-life load conditions. This testing helps determine how the application behaves when multiple users access it simultaneously.
   * This testing usually identifies –
     + The maximum operating capacity of an application
     + Determine whether current infrastructure is sufficient to run the application
     + Sustainability of application with respect to peak user load
     + Number of concurrent users that an application can support, and scalability to allow more users to access it.
     + It is a type of non-functional testing. Load testing is commonly used for the
     + Client/Server, Web based applications – both Intranet and Internet.
5. What is stress Testing?
   * Stress testing - System is stressed beyond its specifications to check how and when it fails. Performed under heavy load like putting large number beyond storage capacity, complex database queries, continuous input to system or database load.
   * Stress testing is used to test the stability & reliability of the system. This test mainly determines the system on its robustness and error handling under extremely heavy load conditions.
   * 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 not crash under crunch situations.
   * Stress testing is also known as endurance testing.
   * Under Stress Testing, AUT is be stressed for a short period of time to know its withstanding capacity.
   * Most prominent use of stress testing is to determine the limit, at which the system or software or hardware breaks.
   * It also checks whether system demonstrates effective error management under extreme conditions.
   * The application under testing will be stressed when 5GB data is copied from the website and pasted in notepad.
   * Notepad is under stress and gives ‘Not Responded’ error message.
   * Examples of stress conditions include:
     + 1. Excessive volume in terms of either users or data; examples might include a denial of service (DoS) attack or a situation where a widely viewed news item prompts a large number of users to visit a Web site during a three-minute period.
       2. Resource reduction such as a disk drive failure.
       3. Application components fail to respond.
6. 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.
   * In white-box testing the tester is concentrating on how the software does it.
     + For example, a structural technique may be concerned with exercising loops in the software.
   * Different test cases may be derived to exercise the loop once, twice, and many times. This may be done regardless of the functionality of the software.
   * Structure-based techniques are also used in system and acceptance testing, but the structures are different.
     + For example, the coverage of menu options or major business transactions could be the structural element in system or acceptance testing.
   * Testing based upon the structure of the code
   * Typically undertaken at Component and Component Integration Test phases by development teams
   * White box testing is the detailed investigation of internal logic and structure of the code.
   * White box testing is also called glass testing or open box testing. In order to perform white box testing on an application, the tester needs to possess knowledge of the internal working of the code.
   * The tester needs to have a look inside the source code and find out which unit/chunk of the code is behaving inappropriately
   * **Type of White-box Testing**
     + Experience Based Testing
     + Grey-box Testing
     + Adhoc Testing
     + Exploratory Testing
7. 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.
   * Specification-based testing technique is also known as ‘black-box’ or input/output driven testing techniques because they view the software as a black-box with inputs and outputs.
   * The testers have no knowledge of how the system or component is structured inside the box. In black-box testing the tester is concentrating on what the software does, not how it does it.
   * Specification-based techniques are appropriate at all levels of testing (component testing through to acceptance testing) where a specification exists.
     + For example, when performing system or acceptance testing, the requirements specification or functional specification may form the basis of the tests.
   * **Black-box Testing Technique**
     + Equivalence Partitioning (E.P.)
     + Boundary Value Analysis (B.V.A.)
     + Decision Table
     + State Transaction Testing
8. Mention what are the categories of defects?
9. Mention what big bang testing is?
   * In Big Bang integration testing all components or modules is integrated simultaneously, after which everything is tested as a whole.
   * Big Bang testing has the advantage that everything is finished before integration testing starts.
   * The major disadvantage is that in general it is time consuming and difficult to trace the cause of failures because of this late integration.
     + Here all component are integrated together at once, and then tested.
10. What is the purpose of exit criteria?
    * Purpose of exit criteria is to define when we STOP testing either at the:
      + End of all testing – i.e. product Go Live
      + End of phase of testing (e.g. hand over from System Test to UAT)
11. When should "Regression Testing" be performed?
    * Regression testing should be carried out:
      + when the system is stable and the system or the environment changes
      + when testing bug-fix releases as part of the maintenance phase
12. What is 7 key principles? Explain in detail?

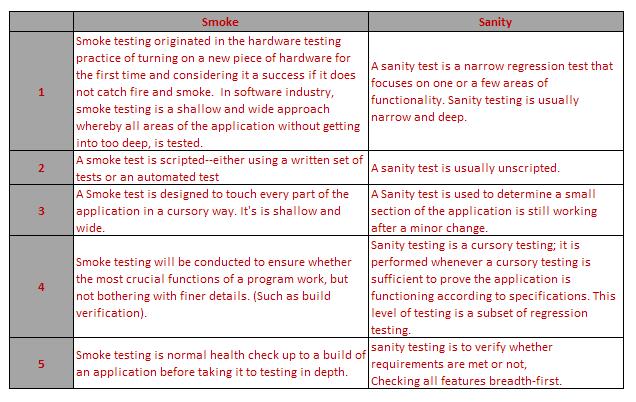
* **Testing shows presence of defects: -**
  + - * Testing can show that defects are present, but cannot prove that there are no defects.
* **Exhaustive testing is impossible: -**
  + - * Testing everything including all combinations of inputs and preconditions is not possible.
      * It’s mean that you cannot test all combinations form your side for example, in an application in one screen there are 15 input fields, each having 5 possible values, then to test all the valid combinations you would need 30178125 (515) tests.
* **Early testing: -**
  + - * Testing activities should start as early as possible in SDLC, and should be focused defined objectives.
* **Defect clustering: -**
  + - * If tester found a defect in one stage he should find more defect in coverage area
      * Defect are not evenly spread in a system, because they are “clustered”
* **The pesticide paradox: -**
  + - * If the same tests are repeated over and over again, eventually the same set of test cases will no longer find any new defects. You should try new test case.
* **Testing is context dependent: -**
  + - * Testing is done differently in different contexts, means gaming testing is different website testing is different application testing is different.
* **Absence of errors fallacy: -**
  + - * That is not possible to build 100% accurate software.

Even after defects have been resolved it may still be unusable and/or does not fulfil the users’ needs and expectations.

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



1. Difference between Smoke and Sanity?



1. What is bug life cycle?
   * A bug life cycle in software testing is a set of statuses designed to coordinate defect management. A bug status helps keep all the members of the development team posted on the progress. The cycle starts when a QA engineer reports a new issue found in the tested software and finishes when this issue is sold.
     1. New defect
     2. Assign defect
     3. Open defect by dev
     4. If bug fixed
     5. Or dev can comment on bug like duplicate, not a bug, deffered, rejected
     6. Retest
     7. Reopen if defect was found again
     8. Verified
     9. Closed
2. **HLR & Test case for Instagram and Facebook first page?**
   * Instagram HLR
     1. By clicking on login button after entering data user should redirect to the system
     2. By clicking on login with Facebook button user should redirect to the Facebook login page
     3. By clicking on forget password button user should redirect to the account recovery page
     4. By entering valid email address or username and then clicking on send button user should get recovery link on email address
     5. By clicking on can’t reset password user should redirect to the help center of an application
     6. By clicking on create new account button user should redirect to the create new account page
     7. By entering valid data and then clicking on sign up button user should redirect to the system
     8. By clicking on login button user should redirect to the system login page
     9. By clicking on back to login button user should redirect to the login page of a system
     10. By clicking on sign up button user should redirect to the create new account page of a system
   * Instagram Test case

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Test Scenario I'D | Test Scenario Description | Test Case I'D | Test Case Description | Prerequisite | Test Data | Expected Output | Actual Output | Note | Result(Pass/Fail) |
| 1 TS | verify That user is able to download instagram application | IN-RS-TC-1 | check that user can download the application from playstore | Already app available on Playstore | Click to install | App should show "downloaded sucessfully" | App is downloaded sucessfully |  | pass |
|  |  | IN-RS-TC-2 | check that user can download the application from appstore | Already app available on Appstore | Click to install | App should show "downloaded sucessfully" | app is shown in downloded folder |  | pass |
|  |  | IN-RS-TC-3 | check that instagram app sucessfully install to your mobile | App is already downloaded | find application name in device | App should shown in mobile | app is showing in app managaer in mobile |  | pass |
| 3 TS | Check that the login functionality is working | IN-RS-TC-4 | check that if it is valid username and password | App is already installed | username : shreyansh\_agola password : Shreyansh@\*\*\*\*\*\*\* | User should login into application and redirect to the home page | user is redirect to the home page |  | pass |
|  |  | IN-RS-TC-5 | check that if it is invalid username and password | App is already installed | username : shreyansh\_patel password : Bittu@\*\*\*\*\*\*\* | user should get error and he/her can't login, it shows error message | user is getting error |  | pass |
|  |  | IN-RS-TC-6 | check that if username is valid and password is invalid | App is already installed | username : shreyansh\_agola password : Bittu@\*\*\*\*\*\*\* | user should get error and he/her can't login, it shows error message | user is getting error |  | pass |
|  |  | IN-RS-TC-7 | check that if username is invalid and password is valid | App is already installed | username : shreyansh\_patel password : Shreyansh@\*\*\*\*\*\*\* | user should get error and he/her can't login, it shows error message | user is getting error |  | pass |
|  |  | IN-RS-TC-8 | check that user can able to logout | App is already installed | click to logout button | user can logout into cuurent account | user is able to logout from application |  | pass |
| 18 TS | verify that user is able to edit his/her profile or not | IN-RS-TC-9 | check that user can able to upload profile picture | App is already installed | [..\..\Persnol\Picture\IMG\_20230115\_122100.jpg](file:///C:\Persnol%20Files\Persnol\Picture\IMG_20230115_122100.jpg) | user can able to upload profile pic | user can change profile pic |  | pass |
|  |  | IN-RS-TC-10 | check that user can able to change his/her username | App is already installed | shreyansh\_agola | user can able to save this name | user is able to save his/her name sucessfully |  | pass |
|  |  | IN-RS-TC-11 | check that if new username is same as any other's profiles existing username then user can't save that username | App is already installed | other users can't use this username : shreyansh\_agola | user can't save this name and shows error message : username is already exist | user is getting error |  | pass |
|  |  | IN-RS-TC-12 | check that user is able to delete his/her profile details | App is already installed | delete : bio text | user can able to delete text from bio | user can delete text from bio sucessfully |  | pass |

1. What is priority?
   * Defect priority refers to the urgency of fixing a defect, based on the business value, customer expectations, and project deadlines.
2. What is severity?
   * Defect severity refers to the impact of a defect on the software functionality, usability, or performance.
3. Bug categories?
   * Low Severity Bugs: ...
   * Medium Severity Bugs: ...
   * High Severity Bugs: ...
   * Critical Bugs
4. Advantage of Bugzilla?
   * Bugzilla is an open-source issue/bug tracking system that allows developers to keep track of outstanding problems with their product.
5. Different between priority and severity?

| **Priority** | **Severity** |
| --- | --- |
| Defect Priority has defined the order in which the developer should resolve a defect | Defect Severity is defined as the degree of impact that a defect has on the operation of the product |
| Priority is associated with scheduling | Severity is associated with functionality or standards |
| Priority indicates how soon the bug should be fixed | Severity indicates the seriousness of the defect on the product functionality |
| Priority of defects is decided in consultation with the manager/client | QA engineer determines the severity level of the defect |
| Priority is driven by business value | Severity is driven by functionality |
| Its value is subjective and can change over a period of time depending on the change in the project situation | Its value is objective and less likely to change |
| High priority and low severity status indicates, defect have to be fixed on immediate bases but does not affect the application | High severity and low priority status indicates defect have to be fixed but not on immediate bases |
| Priority status is based on customer requirements | Severity status is based on the technical aspect of the product |
| During UAT the development team fix defects based on priority | During SIT, the development team will fix defects based on the severity and then priority |
| Priority is categorized into three types   * Low * Medium * High | Severity is categorized into five types   * Critical * Major * Moderate * Minor * Cosmetic |

1. Door negative?
   * It is possible that if the door weight is too high then it is difficult to open and close it.
   * It is possible that the door sensors are not working properly then a human will be stuck behind the door.
   * It is possible that the door is made of wood then it will expand in monsoon and create problems for human.
   * It is possible that the senor door closing speed have to be normal otherwise it will hurt someone.
   * It is possible that some doors are too noisy when they open and close it.
   * It is possible that if the door is made of iron then it is required to maintain it.
   * It is possible that if the door is transparent then someone will hurt from it.
   * It is possible that the door hasn't something on it to open and close then it is difficult to go through it.
   * It is possible that the door material is low quality then someone will break it easily.
   * It is possible that some door is open only one side the other side was blocked.
2. Lift negative?
   * It is possible that some Lift is so fast or some is so slow. it should balance transportation speed.
   * It is possible some lift doesn't have emergency button for stopping lift, ex: industrial lift.
   * It is possible that lift cannot start if it has over capacity.
   * It is possible that some lift fans are too noisy.
   * It is possible sometimes lift button is not working.
   * It is possible that maintenance of lift is too costly.
   * It is possible that some lift get a shock when it start or the end of destination.
   * It is possible that some time lift sensors are not working properly.
   * It is possible some of lifts are noisy when it is transmitting.
   * It is possible that some lifts announcement system are not working then it is difficult to transmitting for blind person.
3. Water bottle negative?
   * It is possible that some plastic bottles can be melted at high temperatures.
   * It is possible that if bottel cap is not fitted properly then bottel should be leack.
   * It is possible that different water bottle carrying different capacity of water.
   * Some plastic bottles have bad impact on environment.
   * It is possible that we can't recycle all types of Metirial.
   * It is possible that some bottles can't perform their activities in the long term (the coldest bottle can't be cold the lequid to entire life).
   * It is possible that bottle material is hard to break by humans with hands only.
   * It is possible when we store bottles for a long time that impact while drinking that lequid which was stored in bottles and harmful for humans.
   * It is possible that some plastic bottlea can be performe differenet activity at very low temperatures.
   * To recycle Plastic water bottles are very expensive.
4. What is GUI testing?
   * There are two types of interfaces for a computer application. Command Line Interface is where you type text and computer responds to that command. GUI stands for Graphical User Interface where you interact with the computer using images rather than text.
   * GUI Testing is a software testing type that checks the Graphical User Interface of the Software. The purpose of Graphical User Interface (GUI) Testing is to ensure the functionalities of software application work as per specifications by checking screens and controls like menus, buttons, icons, etc.
   * GUI is what the user sees. Say if you visit guru99.com what you will see say homepage it is the GUI (graphical user interface) of the site. A user does not see the source code. The interface is visible to the user. Especially the focus is on the design structure, images that they are working properly or not.
   * Check all the GUI elements for size, position, width, length, and acceptance of characters or numbers. For instance, you must be able to provide inputs to the input fields.
   * Check you can execute the intended functionality of the application using the GUI
   * Check Error Messages are displayed correctly
   * Check for Clear demarcation of different sections on screen
   * Check Font used in an application is readable
   * Check the alignment of the text is proper
   * Check the Color of the font and warning messages is aesthetically pleasing
   * Check that the images have good clarity
   * Check that the images are properly aligned
   * Check the positioning of GUI elements for different screen resolution.
5. Why need of usability testing?
   * It helps uncover usability issues before the product is marketed.
   * It helps improve end-user satisfaction
   * It makes your system highly effective and efficient
   * It helps gather true feedback from your target audience who actually use your system during a usability test. You do not need to rely on “opinions” from random people.
6. Whatsapp web HLR?

|  |  |  |
| --- | --- | --- |
| Functionality ID | Functionality Name | Functionality Description |
|  |  |  |
|  | Website URL |  |
| 100 | website url | by entering url in browser it open webpage. |
|  |  |  |
|  | QR Code |  |
| 200 | QR code | after a certain time by drage cursor on QR code it offer to reload QR code. |
|  |  |  |
|  | Need Help to Get Started? |  |
| 300 | need help to get started button | by clicking on need help to get started it redirect to the help center of an application. |
|  |  |  |
|  | Play Button |  |
| 400 | play button | by clicking on play button it start playing video for instucution about how to start application. |
| 401 | pause button | by clicking on pause button it stop video |
| 402 | volume button | by clicking on volume button user can mute volume or adjust volume. |
| 403 | full screen button | by entering fullscreen button video will continue in full screen. |
| 404 | more option icon button | by entering more icon button user can adjust playback speed and minimize video. |
| 405 | drage on line | by dragging on line user can fast forward or backward in video. |

1. What is different between SDLC and STLC?

| **Parameter** | **SDLC** | **STLC** |
| --- | --- | --- |
| Origin | Development Life Cycle | Testing Life Cycle |
| Objective | The main object of SDLC life cycle is to complete successful development of the software including testing and other phases. | The only objective of the STLC phase is testing. |
| Requirement Gathering | In SDLC the business analyst gathers the requirements and create Development Plan | In STLC, the QA team analyze requirement documents like functional and non-functional documents and create System Test Plan |
| High & Low-Level Design | In SDLC, the development team creates the high and low-level design plans | In STLC, the test analyst creates the Integration Test Plan |
| Coding | The real code is developed, and actual work takes place as per the design documents. | The testing team prepares the test environment and executes them |
| Maintenance | SDLC phase also includes post-deployment supports and updates. | Testers, execute regression suits, usually automation scripts to check maintenance code deployed. |

1. Write test scenarios for only WhatsApp chats?
   * Test Scenarios Positive:
     1. Verify that user can sends and receive message or not.
     2. Verify that user can sends text message, sticker, audio, video, photos, document, files, and many more or not.
     3. Verify that user can see other’s last online time or not.
     4. Verify that user can set privacy for chatting or not.
     5. Verify that user can able to change background image or not.
   * Negative:
     1. Sometime user have to wait for some certain time to receive message.
     2. Sometime user can’t sends a message because of internet issue.
     3. User can’t change text style.
     4. User can’t translate language on application.
     5. If user are forward the message then it indicate third person with forwarded message.
2. Write scenarios for pen?
   * Positive:
     1. Verify that the pen’s type is writeable or not.
     2. Verify that pen’s cap is missing or not.
     3. Verify that pen have refile or not.
     4. Verify that pen weight is normal or not.
     5. Verify that pen Height is normal or not.
     6. Verify that pen have grip or not.
   * Negative:
     1. Sometime if pen is gel pen then it get its dark mark on paper.
     2. Sometime it is possible if cap is missing then it’s too risky because ink could be overflow at any time.
     3. Sometime pen’s ball could be broke.
     4. It is possible all material is not possible to recycle.
3. Write scenarios for door?
   * Positive:
     1. Verify that door material is proper or not.
     2. Verify that door is smooth to open and close or not.
     3. Verify that door size is proper or not.
     4. Verify that door have handle or not.
     5. Verify that door is noisy or not.
     6. Verify that door hardness is proper or not.
   * Negative:
     1. It is possible that if the door weight is too high then it is difficult to open and close it.
     2. It is possible that the door sensors are not working properly then a human will be stuck behind the door.
     3. It is possible that the door is made of wood then it will expand in monsoon and create problems for human.
     4. It is possible that the senor door closing speed have to be normal otherwise it will hurt someone.
     5. It is possible that some doors are too noisy when they open and close it.
     6. It is possible that if the door is made of iron then it is required to maintain it.
     7. It is possible that if the door is transparent then someone will hurt from it.
     8. It is possible that the door hasn't something on it to open and close then it is difficult to go through it.
     9. It is possible that the door material is low quality then someone will break it easily.
     10. It is possible that some door is open only one side the other side was blocked.
4. Test scenarios on chair?
   * Positive:
     1. Verify that hardness of chair is good or not.
     2. Check that chair is comfortable to sit or not.
     3. Verify that chair is limp or not.
     4. Verify that is there back in the chair or not.
     5. Check that the chair is stable to carry enough human load or not.
     6. Verify that the chair material is worth to price or not.
     7. Check that the chair usability is different or not.
     8. Verify that the chair color is user-friendly or not.
     9. Verify that the chair height from floor is good or not.
     10. Verify that water is effecting chair or not.
     11. Verify that the chair is washable or not.
     12. Verify that the chair material is brittle or not.
     13. Verify that the dimension of the chair is as per the specification or not.
     14. Verify that the weight of the chair is as per the specification or not.
     15. Verify that the paint’s type & color is attractive or not.
     16. Verify that user can sit comfortably on chair or not.
     17. Verify that the chair is recline or not.
     18. Verify that the chair size is user friendly or not.
   * Negative:
     1. Sometime chair is not comfortable to sit.
     2. It is possible some chair is noisy.
5. Write a test scenarios for online shopping application?
   * Positive:
     1. Verify that user is able to register his/her self or not.
     2. Verify that user is able to login with his/her register Id and Password or not.
     3. Verify that user is redirect to the home page after login or not.
     4. Check that user is able to add product he/her want to buy in my basket or not.
     5. Verify that user is able to remove product in my basket or not.
     6. Check that user is able to add product in wish list or not.
     7. Verify that user is able to remove product in my basket or not.
     8. Verify that if user is purchased membership than he/her got a special discount or not.
     9. Verify that if user have membership than he/her get fastest delivery or not.
     10. Verify that if user apply coupon code than he/her gets a discount or not.
     11. Verify that all price details of payment process is accurate and visible or not.
     12. Check that user is able to apply filter for shopping or not.
     13. Verify that user is able to buy product with shop by category or not.
     14. Verify that user is able to see more different offers or not.
     15. Verify that user is able to make a payment or not.
     16. Verify that users are able to make a payment on cash on delivery or not.
     17. Check that user is able to make a payment online/credit card or not.
     18. Verify that users could return delivered item or not. If he/her have complain on it.
     19. Verify that users are able to search product alphabetically in search bar or not.
     20. Check that user can review a product or not.
     21. Verify that system is working smoothly or not.
     22. Verify that system user friendly or not.
     23. Check that system user interface is attractive or not.
     24. Verify that if anyone click to search bar then system is showing search history or not.
     25. Verify that if mouse is move to shop by category then system automatically open menu or not.
     26. Check that is mouse is move to menu item then system is showing submenu item or not.
     27. Verify that system logo is as same as requirement or not.
     28. Check that system fonts is same as requirement or not.
     29. Verify that system style is same as requirement or not.
     30. Verify that system easy to use or not.
6. Write a test scenarios for lift (Elevator)?
   * Positive:
     1. Verify that lift capacity is as per requirement or not.
     2. Verify that lift have emergency switch or not.
     3. Verify that all button on the lift is working or not.
     4. Verify that user can open and close lift’s door easily or not.
     5. Verify that lift is smooth while it start and end or not.
     6. Verify that lift space is as per capacity or not.
     7. Verify that lift doors are manually controlled or not.
   * Negative:
     1. It is possible that some Lift is so fast or some is so slow. it should balance transportation speed.
     2. It is possible some lift doesn't have emergency button for stopping lift, ex: industrial lift.
     3. It is possible that lift cannot start if it has over capacity.
     4. It is possible that some lift fans are too noisy.
     5. It is possible sometimes lift button is not working.
     6. It is possible that maintenance of lift is too costly.
     7. It is possible that some lift get a shock when it start or the end of destination.
     8. It is possible that some time lift sensors are not working properly.
     9. It is possible some of lifts are noisy when it is transmitting.
     10. It is possible that some lifts announcement system are not working then it is difficult to transmitting for blind person.
7. Test scenarios on WhatsApp?
   * Positive:
     1. Verify that website is working on different browser or not.
     2. Verify that a QR code is valid for user to login into the system or not.
     3. Verify that tutorial for user is working or not.
     4. Verify that logout button is working or not.
     5. Verify that user should login into the system after scanning QR code.
     6. Verify that user should get all his/her recent chats in system.
     7. Verify that users are able to edit their profile or not.
     8. Verify that community icon button is working or not.
     9. Verify that user can create new community or not.
     10. Verify that user can see all community which are added them or not.
     11. Verify that user can see all contacts status or not.
     12. Verify that user can add their status or not.
     13. Verify that user can add their comment on other's status or not.
     14. Verify that users are able to see watching on their status or not.
     15. Verify that user can set privacy on status or not.
     16. Verify that user can pick photo or video from gallery or not.
   * Negative:
     1. Sometimes website are not working on different browser.
     2. After a certain time QR code will be change so user and user's device need to so fast.
     3. It happen sometime need help is not working well.
     4. Sometime users are haven't time to click on logout button. Because of login function was working speedily.
     5. Sometime users are stuck in loading scenarios between after scanning QR code and before logged in into the system.
     6. Recent chats are restricted for some time ago only.
     7. Users are able to see only saved contacts status.
     8. User cannot see blocked contacts status.
     9. User can add their status for maximum 30 sec of duration.
     10. Status will be shared for only saved contacts.
     11. User cannot set privacy as no one can see their status.
     12. It is not possible to add status from browser search.