### Project:

It will be developed based on the client’s requirements; once it is developed it will be delivered to the client. Based on the client’s need the team members (End users) will use it.

Ex: Spicejet.com, selenium4testing.com, Constructing own house with our requirements.

### Product:

It will be developed based on the company’s requirements. Once it is developed, it will be released in the market based on the customer needs they will choose the product.

Ex: Mobile App, Calculator, Facebook, Yahoo, MS Office, Mac operating system, Windows operating system, Gmail, etc…

### Testing types:

Testing Tools



Functional Testing Non-Functional Testing

Load Testing Manual Testing Automation Testing Performance testing



Selenium, win runner Stress Testing

QTP, RFT, silktest, Water, Watin Soak Testing

## SOFTWARE DEVELOPMENT LIFE CYCLE(SDLC):

It consists of the below phases,

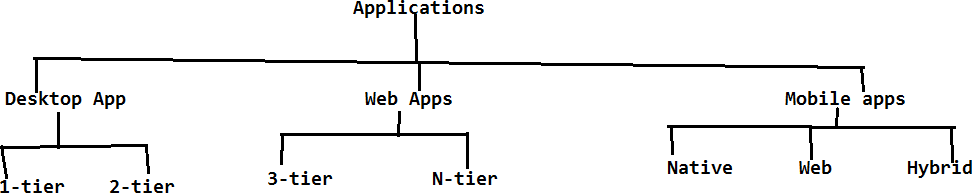
1. Requirement Phase
2. Analysis Phase
3. Design phase
4. Coding Phase
5. Testing phase
6. Delivery and Maintenance Phase.

**Delivery:** Once the build is stable in the test environment the testing team (TL) will send an email to the project manager saying that the build is stable then the project manager will deliver the build to the client.

#### Maintenance:

*‘Live’* means where the client or end-users are using the application. Maintenance will be continued as long as the application is on live.

#### Q. What kind of applications you have tested?



#### TYPES OF APPLICATIONS:

There are three types of applications which can be tested;

1. Web Applications
2. Desktop Applications
3. Mobile Applications

DEV - Development code (D1+D2+D3) at least application URL redirecting to, login logout.

- Build process of converting the source code into .exe file/tar file/ war file

- the application will be accessible within the dev environment

QA/ Staging/ Testing - Gmail application STLC test cases, test execution, defect

UAT - alpha within the client-side team, beta testing end-user side/ client-side

PPT - Pre Production Test / Infrastructure behavior load test, stress test sales

- Load Runner, Jmeter

PRD - Production

#### TESTING METHODOLOGIES:

##### Q: Who is responsible for testing. At what level the Test Engg.. will involve in testing

They are of three types

1. Black box testing
2. White box testing
3. Grey box testing

**a)Black box testing:**

If the resource is performing testing on the functional part of the application, then he will be treated as “*Black box tester*”.

Functional part means whether the developed application is as per the client’s requirements or not. Testers will perform black box testing in test environment and stage env (Pre production env)

**b)White box testing:**

If the resource is testing the structural part (programming) of the application, then he will be treated as *“white box tester*”. Developers are responsible for white box testing in development environment.

**C)Grey box testing:**

If the resource is having the experience on both testing (white box testing and black box testing). Then he will be treated as “*Grey box tester*”.

## LEVELS OF TESTING:

If one project has to go from the signoff stage to live(production), it has to undergo the below levels of testing.

1. **Unit level of testing**
2. **Module level testing**
3. **Integration level of testing**
4. **UAT(User acceptance testing)**
5. **System testing**

**1)Unit level of testing:** *Unit* means the smallest flow or scenario in the application.

* + Developer is responsible for Unit level testing.
  + He will divide the assigned module to multiple units and develops the code for all the units.
  + He is responsible to check whether each and every unit is working as expected or not.

**2)Module level testing:**

* + From Module level testing, both testing team and development team is responsible.
  + The developer will combine all the related units to form a module.
  + Once the module is developed, the developer is responsible for white box testing in development environment.
  + Once the module is released to the testing team, they are responsible for Black box testing in testing environment.

**3)Integration level testing:**

* 1. The process of combining all the developed modules is known as *integration*.
  2. Check whether the data flow is navigating from one module to other is known as

*integration level testing*.

* 1. Both development team and testing team is responsible for integration level testing.

Ex: Create one account in Gmail, check that whether you are able to login into the application with created account. Then compose mail and send it, check that whether it is properly delivered or not.

* While integration if any mandatory moduleis missing then the development Lead will replace the mandatory module with some dummy code is known as stub or driver.

+ + + +

D1

D2

D3

D4

+

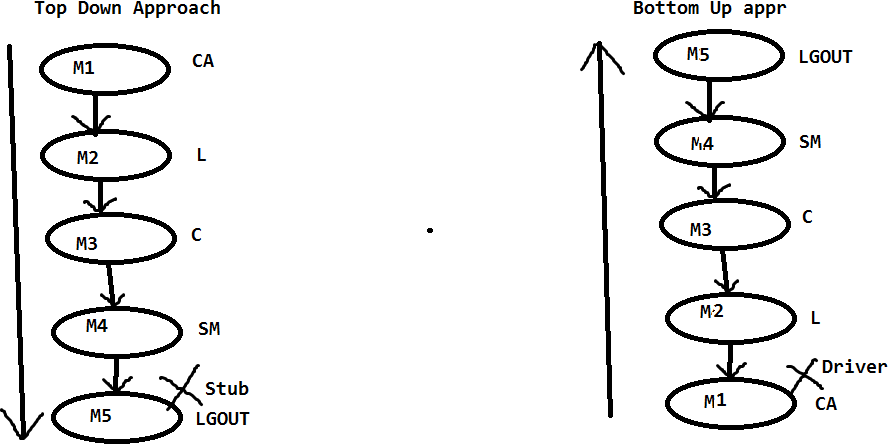
D5

Credentials Login Compose Send Logout



**Stub/Driver:**Both are nothing but a dummy code, it doesn't contain any functionalities.

* If the development lead is using top down approach to integrate the modules, while integration if any mandatory module is missing then it will be replaced by *Stub*.
* If the development lead is using bottom up approach to integrate the modules, while integration if any mandatory module is missing then he will replace with *Driver.*



**4)User Acceptance Testing:**

* 1. It is known as *user/client acceptance testing*. Once the build is stable in test environment then we willplanto deliver the build to the client. Before delivering the build to the client, the client will send User acceptance test cases to the testing team for execution.
  2. The testing team will execute all the UA test cases in test environment; if all are passed then the project manager will deliver the build to the client.
  3. Client again will execute all the UA test cases in the client’s environment (stage environment). If all are passed then the client will deploy the build in the Live or Production environment.
  4. UAT is of two types:

**a.**Alpha Testing

**b.** Beta Testing **UAT**



**Alpha testing Beta testing** (UATCS)

(UATCs) Test environmentStage Environment

1. **Alpha Testing:**Executing all the UA test cases in a test environment by the testing team is known as ‘*Alpha testing’*.
2. **Beta Testing:** Executing all the UA test cases in the clients Stage environment by the client’s team or the testing team is known as ‘*Beta testing’*.

Once the Beta testing is passed then the client will go to the live environment. Test Environment Client



UATCS

Start

Testing Team Build

Build (UATCS) pass Deliver

#### **5)System Testing:**

* 1. It is also known as *non-functional testing*. Once the application is stable, then we can go for non-functional testing.
  2. In non-functional testing performance (response time) of the application will be identified.
  3. The time taken between the request and response is known as *response time*. It will be identified with the help of multiple non-functional testing types like Load testing, Performance testing, Stress testing, Breaking point testing.

## SOFTWARE DEVELOPMENT MODELS:

##### Q. What process you have used to develop your project

The models are as follows.

1. **Waterfall model**
2. **Spiral model**
3. **V-model**
4. **Fish Model**
5. **Agile process**

#### 1)Waterfall Model:

It was the initial process or model introduced for software development (old process). The sequential execution of all the phases in SDLC is known as water fall model. Once the phase is completed, high level management will analyze that phase.

**NOTE:** How waterfalls from one level to another, in the same way, the phases of SDLC will be implemented.

##### Advantages:

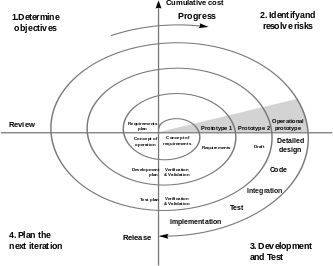
It is very easy to implement the project because it is Sequential Execution.

##### Disadvantages:

* + The risksk can't be identified at the early stage of the life cycle so it can't be prevented.
  + It is time consuming process as well as costly process.
  + We can't accept the requirements change in the middle of the project. If still needs to be accepted then we will accept the requirement change in the form of CRs-change requests.Change requests are done at the end of the project and CRs charged by the company.

#### **2) Spiral Model**:

* + Spiral model is a combination of waterfall model and prototype.



* + Instead of collecting all the requirements once, the BA collects few requirements; it will be analyzed and designed with the help of the prototype. Then it will be given to the development.
  + Once the developer develops the build then it will be released to the testing team. The same process will be continued for all the requirements.
  + Once all the requirements are completed and the build is stable, then the build will be delivered to the client.

##### Advantages:

* + We can save the time and cost, because we are executing all the phases in parallel.
  + The risk can be identified at the early stage of the SDLC and it can be prevented at the early stage of the life cycle.
  + The requirement change can be accepted at the middle of the process.

##### Disadvantages:

* + It is having the huge delivery risk, because of the aggressive time lines(less time).
  + Cannot accept requirement change at the end stage of the project to avoid delivery risk.

#### **2)V-Model(Verification and Validation model):**

##### Validation:

It is also known as “*QC” (Quality control*). The testing team is responsible for validation. Testing team will check whether the developed software is as per the client’s requirement or not.

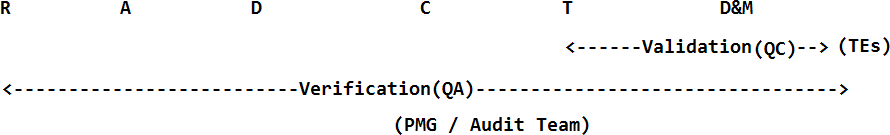
Test engineers are validators.

##### Verification:

Def1: Check whether each and every phase outcome document is as per the company and client guidelines or not.

Def2**:**Check whether each and every role in the organization is working as per the Companies and clients guide lines or not. Verification is also known as *QA (Quality assurance).*

The project/Process management group (PMG) or audit group are responsible for verification.



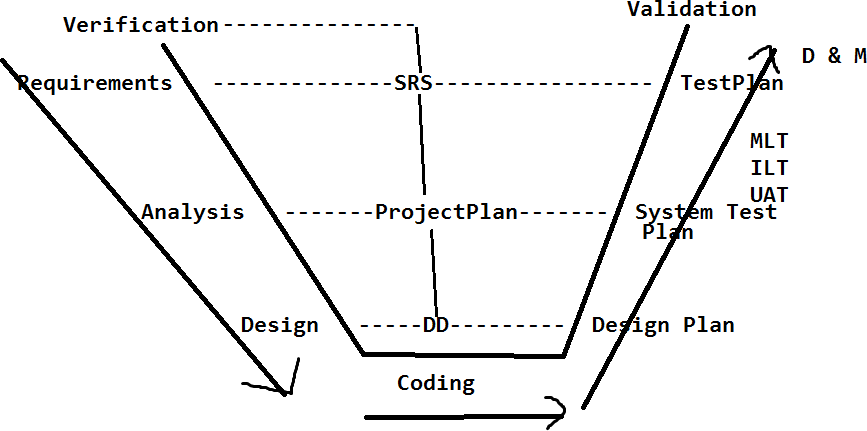
* From V-model onwards even the testing team will participate in collecting requirements.
* BA is responsible to collect the requirements,parallel the testing team will be analyzing all the requirements to check whether it is possible to test or not.
* Once the SRS is baselined, validation team is responsible for test plan
* Based on the analysis and design phases the validation team is preparing the system test plan and design plan.
* Once the code is developed they will release the build to the testing team where they will perform all the levels of testing. Once the build is stable it will be delivered to the client.

##### Advantages:

* The testing activities are started from the requirement phase onwards so that we can ensure for quality.
* For each and every phase the verification team and validation team will check the phases so that we can ensure for quality.
* The risk can be identified at the early stage because we have a parallel testing activity, so it can be prevented.
* We can accept the requirement change at the middle of the phase.

##### Disadvantages:

* It's a time consuming and costly process but we can ensure for quality.



#### Agile process:

* + It is having multiple sub models like adaptive, Scrum, iterative model etc…The model which we are going to use is scrum model.
  + It is an iterative and incremental model. Scrum model is having the below activities.

1. Scrum master
2. User stories
3. Scrum meeting/scrum calls/DSM
4. Sprint plan
5. Sprint meeting
6. Backlogs

##### a)Scrum master:

The scrum master is, who is going to lead the project. The project manager or the client will acts as a scrum master. Scrum master is responsible for scrum meetings and sprint meetings.

##### User stories:

The requirements will be captured in the form of end user used flows (end user used ways). Hence we will call it as *User stories*. BA is responsible to collect

##### Scrum meeting:

On daily basis all the team members will participate in a quick meeting where they will describe what activities were performed yesterday and what tasks are planned to perform today and is there any challenges.

* + All the team members including the scrum master and client have to describe.
  + The main purpose of the scrum meeting is to track the resources and also to maintain the transparency.

##### Sprint plan:

* + Sprint is fixed time period can beone week/two weeks/three weeks etc. It will be decided by the scrum master.
  + Sprint plan is, to collectuser stories, analyze, develop, test and deliver to the client.
  + During the sprint if we are unable to complete any of the requirements the sprint won’t be extended. And the pending requirements should be carried to the next sprint. Sprint is a fixed time period

##### Sprint meeting:

Once the sprint is completed the next sprint plan will be decided under the sprint meeting. They will discuss, if the current sprint is delivered successfully or not, is there any challenges faced.

##### Backlogs:

During the sprint plan if any user stories are unable to accomplish, those will be taken as Backlogs. These backlogs have to be completed in the next sprint.

It is of two types,

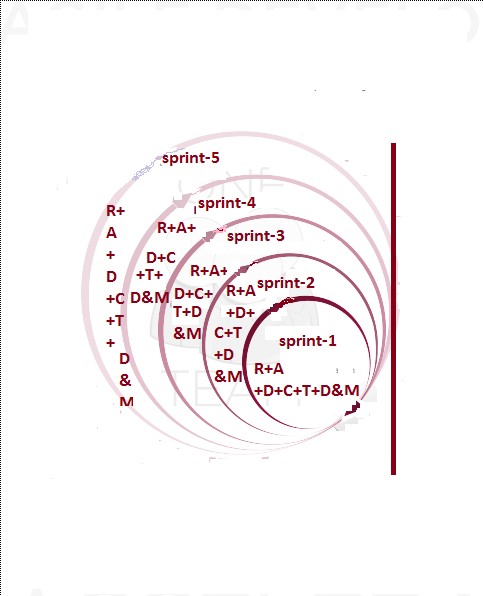
1. Product Backlog
2. Sprint backlog

Product Backlog: The Requirements (user stories) which we are going to collect, develop, test and deliver it to the client as a part of sprint plan is known as *product backlogs.*

Sprint Backlog: The Requirements which are not completed as part of the sprint plan will be treated as sprint backlog.

##### Advantages:

* Each and every sprint will be tested multiple times by the testing team and client, so we can ensure for quality.
* All the phases in SDLC are performed parallel y so we can save time and cost.
* The requirement change can be accepted at any stage of the project (even after delivery of sprint).
* Risk can be identified at the early stage and it can be prevented
* We can maintain transparency of the project.
* The client will also participate in scrum meetings, so we can get the information very quickly.
* Each and every sprint is delivered to the client so we doesn’t have delivery risk.
* We can gain the customer satisfaction by delivering all the sprints to the client.
* Sprint is also known as iterative. Its and iterative and incremental model



##### Disadvantages:

Maintaining all the sprint related information is a very challenging task, but we can overcome with the help of test management tools like Scrum wise, Quality center, JIRA and test link etc.

### Functional testing types (or) Black box testing types:

#### Smoke testing:

* Once the build is developed and deployed it in any environment then the initial testing will be performed, that is known as smoke testing. Initially the development team will deploy the build in development environment, and perform smoke test. They will check each and every module related field is properly navigating their pages or not and checks the main functionality of the application. The objective of smoke test is to check whether the build is ready for further testing or not. The developer will concentrate on white box testing
* If all these fields are properly navigating to the related pages then they will conclude that smoke test is passed.

#### Sanity testing:

* Once the build is deployed in the test environment, the testing team will perform the smoke test in test environment. It is known as sanity testing.
* In sanity test the testing team will perform at least one round of the main flow functionality and check whether its properly working or not.
* If sanity test is passed then the testing team will execute all the test cases if it fails they will reject the build to the development team.

Ex for Main flow: Create an account in Gmail and login into that account and compose email and send it to one valid email and check that whether it is properly delivered or not.

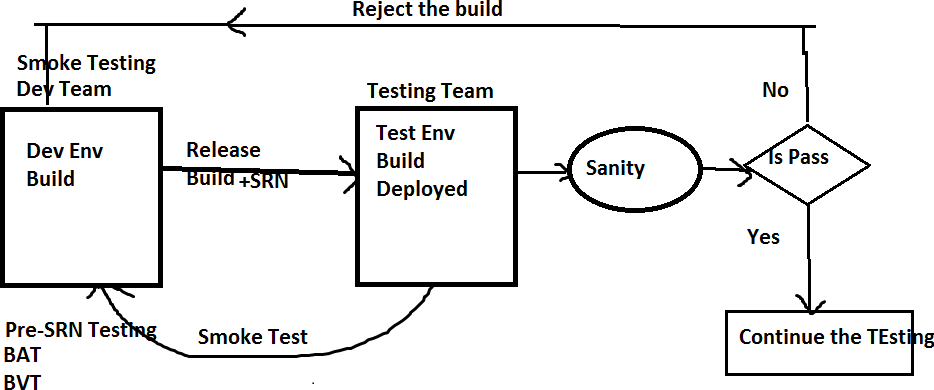
**Note:**Once the sanity test is performed the testing team (test lead) has to send an email with sanity test results to the development team.

1. **Pre SRN testing:**SRN - Software Release Notes

* It contains the build status like, number of modules available in the build for testing.
* Number of modules which are under development.
* Number of stubs and drivers are available in the build.
* Number of bugs which are fixed and available in the build.
* Number of bugs which are under development
* Deployment guidelines etc..
  + Before releasing the SRN document along with the build to the testing team, the testing team will perform the smoke test in development environment, is known as Pre-SRN Testing
  + It is also known as *Build acceptance testing (BAT) or Build verification testing (BVT).*

*Note:* Once the build is released to the testing team, the test engg's will review the SRN doc to know the build status (what build contains). Then the testing will perform sanity test.

1. The order is initially the Dev team will perform Smoke testing, then the testing team will perform Pre-SRN testing in Dev Env. If both are pass then the Dev team will release the build to testing team then the testing will perform sanity testing

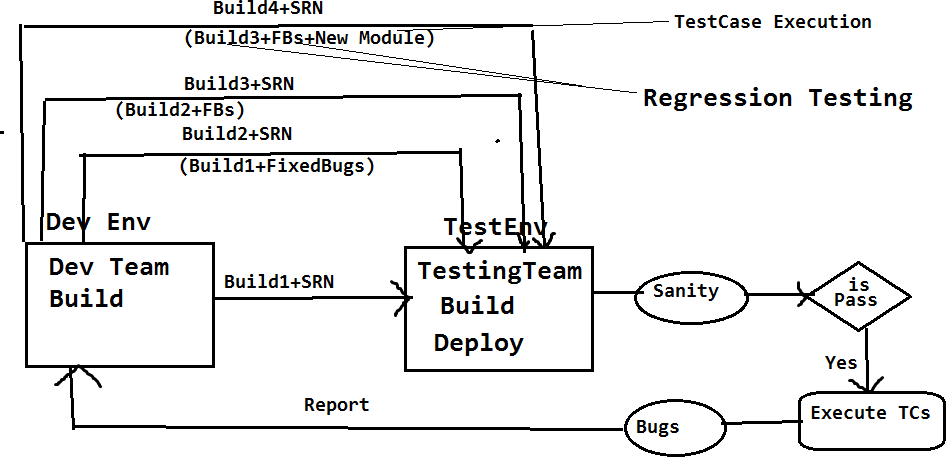


#### GUI/UI Testing:

Graphical user interface/user interface testing. The below five activities will be tested in GUI.

* + Check the spellings of all the fields.
  + Check the font of all the fields where it should maintain the consistency.
  + Check the color and alignments of all the fields it should maintain the consistency.
  + Check the overall look and feel of the page

#### Regression testing:



Performing testing on already tested functionalities on the iterative and incremental builds is known as ‘*Regression Testing’*.

It will be performed in two ways:

* Whenever any bug is identified it will be reported to the developer, the developer will fix it then he will releasing the fixed bug in the form of new build(Build2) to the testing team.The test engineer will test again, to check whether the bug is really fixed or not.
  + The test cases which are passed on the old build will be executed again on the new build and check that whether these are working same as previous or not.

Testing already tested functionalities is *regression testing.* Testing the new functionalities is not the regression testing. It comes under test case execution.

**Note:**If any code update is there,then that new code may affect the existing code, so we are performing the regression testing.

#### Retesting:

* + Perform testing on the same functionalities again and again with multiple sets of different test data on the same build is known as ‘*Retesting*’.
  + Executing the failed test cases on the iterative and incremental builds is also known as “Re testing”.

**Test data:**The data which the testing team is using for testing is known as “*Test data*”.

**Ex: 1.** Test the login functionality of Gmail with multiple sets of different credentials.

2. Test the spicejet one way search with the multiple sets of different origins and different passengers.

Q: What is the difference between Regression Testing and Retesting

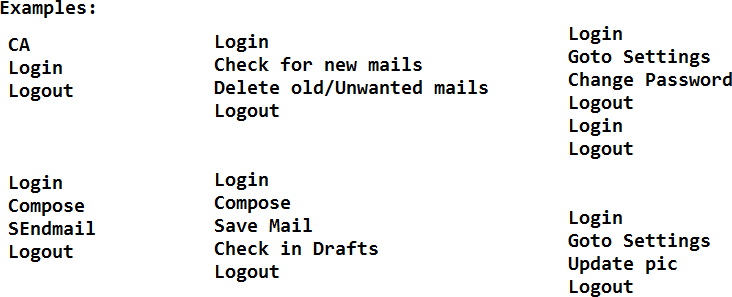
Q: What is the difference between Regression Testing and integration Level Testing

#### End to End Testing:

The test engineer has to identify all the end user used scenarios of the application, and check that whether the End to End Scenarios are properly working or not

By performing End To End testing we can achieve Integration level testing

**Ex**: The end to end scenario for Gmail.



#### Compatibility Testing:

* + Test whether the application is working same as expected in all the targeted

environments or not is known as ‘*compatibility testing*’. Environment is combination of OS, Browser, Server, DB etc.

* + Compatibility testing is two types ‘*cross browser testing’* and ‘*cross platform testing’*.
  + Test whether the web application is working as expected in all the targeted browsers like firefox,safari,google chrome,IE etc. is known as ‘*cross browser testing’.*
  + Test whether the desktop application is working as expected in different platforms or operating systems like windows,LINUX,MAC,Solaris etc. is known as ‘*cross platform testing’.*

**Ex** for Cross browser testing: Test whether the spicejet is working in the below environments or not.

Windows – Internet explorer, Firefox, Google chrome, Safari, Opera Linux - Internet explorer, Firefox, Google chrome, Safari, Opera MAC - Firefox, Google chrome, Safari, Opera

**Ex** for Cross platform testing:Test whether the skype is working in the below platforms or environments

Windows Linux

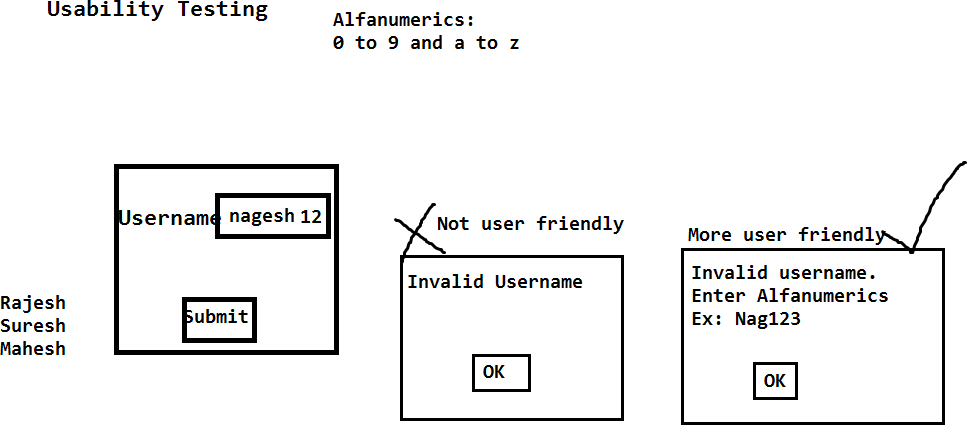
MAC and Mobile

**Note:** Whenever we are performing compatibility testing we need to concentrate more on GUI of the application

#### Usability testing:

* + *Usability* means user friendliness. The test engineer has to provide usability to the application for the end user satisfaction.
  + Depends on the application we have to provide the usability.

**Ex:** For Banking (secured) application we have to provide more security whereas for social sites (Face book, twitter)**,** we need to provide more user friendliness.



#### Adhoc Testing:

* + Adhoc means in our own way.
  + *Adhoc testing* means test theapplication in your own way, After understanding all the requirements and at least one round of manual testing is completed on the application
  + The main purpose of adhoc testing is to provide usability to the application.

#### Exploratory Testing:

* + *Exploratory* means identifying the new requirement / new Feature. Once the build is stable the domain experts will test the application as per their domain knowledge, while testing they will explore ifthe existing requirements are sufficient, if not they will provide the new requirements.
  + The main purpose of exploratory testing is to provide usability and security to the application.

#### Monkey testing/Gorilla testing:

* + Once the application is stable then we can go for monkey testing.
  + Perform testing on the application by doing some abnormal actions is known as

*Monkey/Gorilla testing*.

* + Abnormal actionsmeans continuously click on some field for long period of time and check that whether the application is crashing or not.
  + Test the application with invalid data like HTML tags (<html>) and check that whether the application is crashing or not.

**Note**: The objective of monkey testing is to check whether the application is crashing or not (Means will get server not found exception)

#### Static testing:

* + Testing the application without performing any action is known as *static testing*.

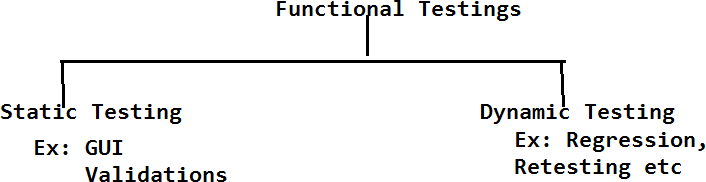
**Ex:**1. GUI testing

2. Validations:- checking the availability of the fields in the page comes under static testing.

#### Dynamic Testing:

* + Test the application by performing some action is known as *dynamic testing*.

**Ex:** Regression testing, Retesting, Adhoc testing etc…



#### Authentication testing:

* + Authentication means check whether the secured credentials/data is available in the database or not.
  + *Authentication testing* means test the application with multiple sets of valid and invalid data, for valid data it should display the homepage, whereas for invalid data, it should display the proper authentication message (error message).

**Ex:** Test the login functionality of HMS with multiple sets of valid and invalid credentials. It has to authenticate the application properly.

#### Direct URL testing:

* + Login into a secure page and take the URL of the secured page and access that URL in a new browser. Where it should not be accessible if it is accessible then the application is not secured.

**Ex:** Login into Gmail.com, copy the URL of the home page. Open in new browser and access the URL directly, where it should not be accessible.

#### Firewall Leakage Testing:

* + Login into the application as one level of user and try to access the data beyond your role limitation.If it is accessible then we conclude that the application is not working as per the role (It is having the firewall leakage).

**Ex:** Login into the HMS application as Jr. Doctor and try to access the data of Sr. Doctor, where it should not be accessible

#### Database Testing:

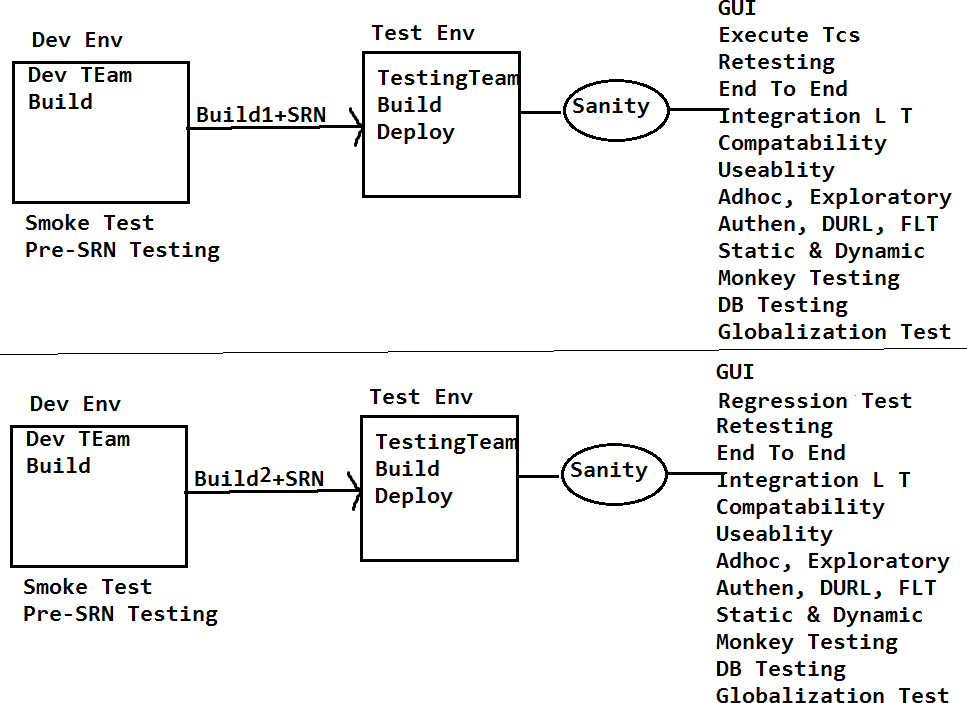
* + Test whether the data is properly inserting into the database of all the tables or not is known as *database testing*. With the help of SQL queries we can perform DB testing.

**Ex:** Whenever we are creating the permanent registration in HMS, all the patient details will be stored in HMS database, as a test engineer we have to login into the database and check that whether the data is properly inserted in all the tables or not.

**Deployment Testing/ Installation Testing:** The deployment team or Test lead will deploy the build in multiple environments like dev, testing, stage1,stage2, production etc and check that whether its properly deploying or not

##### Q: Once the build is released, how you will test the build

**A: Initially we will perform sanity testing, if it is pass then we will execute all the test cases then will perform all the functional testing types as below.By performing all the below testing then we can ensure for quality for the application**



**Functional Testing types – Function testing’s execution flow on the Build, once its released to**

**testing team**

**Note:** For any application all the above testing's will be performed to ensure, whether the application is fulfilling the clients requirements, quality and its useful for enduser or not.

2. If it is a desktop top application Direct URL Testing and cross browser testing is not possible to perform.

#### Review Report Template:

Review the SRS document of spice jet and provide the review report in the below template.

##### Requirement ID Requirement Comments by TE Comments Description

1. Adult, child and infant drop 1. What is the difference between Downs should be available. Child, adult and infant

2. What values the adult, child, infant fields?

#### Globalization testing:

It is two of types

1. Localization testing.
2. Internationalization testing.

##### Localization testing:

* + Test the application in all the local languages which are selective to our country like Hindi, Bengali, Telugu, etc. is known as *localization testing*.
  + It supports maximum of 10 languages for single integration. Hence we will call it as ‘*L10N’ testing*.

**Ex:**1. Test Google.co.in in all the local languages like Hindi, Bengali, Telugu etc…

* 1. Test the ATM machine in local languages likes Hindi, Telugu and English.

##### Internationalization testing:

* + Test the application in all the international languages like Japanese, Chinese and Spanish etc.is known as *internationalization testing*. It supports maximum of 18 languages for a single integration. Hence we will call it as ‘*I18N’ testing*.

**Ex:** Test Gmail.com in all the international languages like, Japanese, Spanish and Chinese etc…

## SOFTWARE TESTING LIFE CYCLE:

It contains the below phases:

* + 1. Software test plan.
    2. Software test design.
    3. Test Execution.
    4. Result analysis.
    5. Reporting & BLC.
    6. Delivery and maintenance.
    7. Test summary report/ Build postmortem report.

#### Software test plan:

* + Plan is a strategic document which describes how to perform a task in an effective and efficient way.
  + Software test plan is also a strategic document which describes how to perform testing in an effective and efficient way. The test plan will be prepared by the test lead; once it is prepared it will be sent to the testing team for review.
  + Based on the test plan we are responsible to perform testing.
  + It contains below activities or Index.

### Test plan Index

1. Objective
   1. Scope of testing
2. Reference documents
3. Test Items
   1. Features to be tested
   2. Features not to be tested
4. Test strategy
   1. Testing types
      1. Functional testing types
5. Test environment
6. Test pass/Fail criteria
7. Defect analysis and closure
8. Test Deliverables
9. Automation testing
10. Risks and contingencies
11. Hardware and software requirements
12. Resource plan
13. Test summary report/ Build postmortem report.

##### Objective:

The main purpose of the test plan will be described here. It contains scope of testing.

##### Scope of testing:

What kinds of testing’s the testing team is responsible to test on the application is known as scope of testing?

**Ex**: Testing team is responsible for manual testing and automation for the project.

##### Reference Documents:

The list of documents which the test lead used to prepare the test plan will be described here. Test lead will use SRS documents to prepare the test plan.

##### Test Items:

* 1. **Features to be tested:**

The list of functionalities or modules which the team is responsible for, will be

described here and also the list of testing’s which the testing team is performing on the modules will be described here.

**Ex**: Testing team is responsible for Book a Flight, Book a hotel and managemy booking.

For the above modules they are responsible for manual testing and automation.

##### Features not to be tested:

The list of modules and testing’s which the testing team is not responsible for will be described here.

**Ex:** Testing team is not responsible for payment modules and also they are not responsible for performance testing, Load testing, Stress testing.

##### Test strategy:

*Strategy* means the list of steps which we are going to take to accomplish the plan.

* The test strategy means the list of functional testing types. Which the testing team is going to take to perform testing is known as test strategy.
* We will perform all the functional testing types like regression, re testing, etc… on the application
* **In short, plan means what to do. Strategy means how to achieve the plan.**

##### Test Environment:

Environment means the system which we are going to use to deploy the build and to test the application is known as the test environment.

**Ex:**Machine type : Windows server enterprise OS : Windows

Processor : Intel Xeon CPU

Memory : 4GB/2.13 GHZ

Hard disk : 150GB

Data base : Microsoft SQL server 2008 standard edition Web server : IIS 7.0

Client : Microsoft internet explorer, Firefox, Google chrome

##### Test pass fail/criteria:

If any test case is deviating from the expected result, then it will be treated as failure or bug.

Every bug is having the criteria or bug type.

It is of five types

1. Blocker
2. Very High
3. High
4. Medium
5. Low

##### Defect Analysis Closure:

At the time of delivering the build if any bugs/defects are available it will be analyzed by the testing team with project manager. If any bug is not necessary to be fixed then it will be closed.

##### Test Deliverables:

The list of modules which we are going to deliver to the client known as *deliverable*s.

All the modules will be divided into multiple phases and also the lead will be providing the targeted deadline (delivery date).

|  |  |  |
| --- | --- | --- |
| **Phase No** | **Modules** | **Dead Lines (Date of Delivery)** |
| **1** | 1. **BookaFlight** 2. **ManageMy Booking** 3. **PNR Status** | **30th Jun** |
| **2** | **4. Flight Schedules** | **31st July** |
| **3** | 1. **Corporate Benefit** 2. **Spice connect** | **30th Sept** |

##### Automation testing:

The number of modules which the testing team is going to automate will be described here and also the automation tool and strategy which the test engineers are going to follow will be described here.

##### Risks and contingencies:

The list of risks which the team is going to face while executing the project and also with the related solution will be described here.

|  |  |
| --- | --- |
| **Risks** | **Contingencies** |
| **Resource shortfall** | **Maintain buffer resources** |
| **Continuous Requirement Changes** | **Analyze the requirements** |
| **Lack of peer reviews** | **Monitor Peer reviews** |

##### Hardware & Software requirements:

The number of machines like laptops, mobiles, printers etc… required for the testing with related software will be described here.

##### Resource Plan:

The numbers of resources required for manual testing, automation testing, database testing will be described here.

##### Test summary report/Build postmortem report:

Once the testing is completed the test lead has to prepare the test summary report, it contains the summary of the testing.

#### Software Test Design:

The process of writing the test cases on the test case template after understanding all the requirements is known ‘*software test design’*.

* Every organization will be having their own template based on that template; the test engineer is responsible to write the test cases.
* We are having the below templates to write the test cases. It contains CoverSheet, Test cases, Test data, Traceability matrix and Test Report

##### Cover sheet:

Module name :

Total no. of test cases :

No. of P1 test cases :

No. of P2 test cases :

No. of P3 test cases :

No. of P4 test cases :

**Requirement ID:** The requirement number for which we are writing the test cases will be described here.

**Test types:** The test case type is known as test type. It is of five types.

* + GUI
  + Validation
  + Positive test case (or) Functional positive test case.
  + Negative test case (or) Functional negative test case
  + Database test case

**Positive test case:** Test the application with all the valid data is known as positive test case.

**Negative test case:** Test the application with at least one invalid data is known as Negative test case.

**Priority:** It describes how important the test case is.It is below types: P1, P2,P3 and P4.

**P1:** If the test case is describing the main functionality of the application/module then it will be treated as P1.

The main functionality means if the test case failed we can’t continue the testing further, so that priority is ‘P1’.

**P2:** If the test case is describing the field level functionality then the priority is ‘p2’.

Field level test case means, if it is failed we can continue the testing but it is important to be there in the application as per client requirement.

**P3:** All GUI test cases are comes under P3 priority.

**P4:**Test engineer is having the option to give the suggestion to the application. Those suggestions will be captured in the form of test cases and then the priority is ‘P4’.

**Test case ID:** The serial number of the test case will be described here.

**Test Scenario:**Scenario means a flow or the end user used way. The requirement will be divided into all the end user used flows or scenarios and those will be described here. The test engg has to identify the maximum possible flows(Scenrios) for the requirement or user story

**Pre-condition:**The condition which is required to test the scenario will be described here.

**Test Steps:**The list of steps which are required to execute the scenario will be described here. Based on the test steps the test engg will execute on the application or build

**Expected Result:**At the time of writing the test cases we won’t be having the application with us. So we will be expecting the result for the scenario. That expected result will be updated inthe expected result column.

**Actual Result:**It will be updated at the time of executing the test cases.The test engineer will observe the actual behavior of the application for the scenario and it will be updated here.

**Result:**Once the test execution is completed the test engineer will compare the actual result with expected result, if both are matching then he will update the result as pass, if not he will update it as fail.

**Comments:**The BA or client will provide the comments here.

Refer Gmail login TCs Document

**Test design techniques:**

To perform testing in an effective and efficient way, we need to follow the below test design techniques.

* 1. Boundary value analysis (BVA)
  2. Equivalence class partition (ECP)
  3. Error Guessing
  4. Decision Making

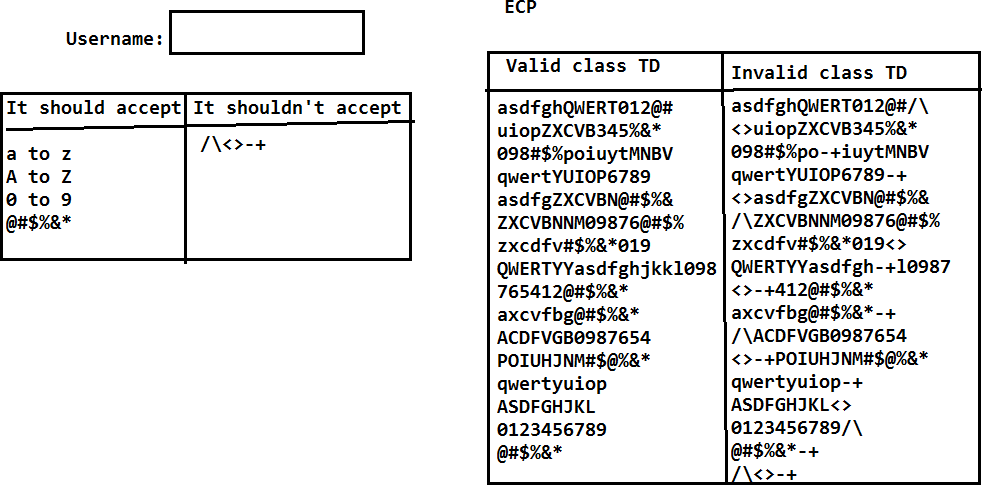
##### Equivalence class partition(ECP):

Whenever we are having the special requirement like check whether the field (user name or password) is accepting the characters like a to z, A to Z, 0 to 9 and #%@$&\*. At the same time the field should not accept the special characters like <>-+/\.

* + In this scenario it’s not possible to perform the exhaustive testing with all the characters. So we need to follow the ECP technique.
  + Divide equally the test data into two classes.

a. Valid test data class b. Invalid test data class

* + Prepare the test data with all the possible ways.
  + To perform positive testing, tests the field with valid test data. Where it has to accept. (Its +ve Test case)
  + To perform negative testing, test the field with invalid test data. Where it should not accept.(Its -Ve Test case)
  + If it is working as expected in the above we can conclude that it working as per the requirement.



##### Error guessing:

Whenever any bug is identified by the test engineer then it should be reported to the developer where he will fix it and send it back to the testing team. The test engineer will check if the bug is really fixed or not. At the same time he has to guess the errors or bugs in the related functionalities. He has to perform the testing in the related functionalities also. It is known as ‘Error guessing’.

Ex: In PR page alert msg is not displaying, it was fixed by developer and tested by the tester. Where the alert msg is properly working in PR page. Now the test engg.. has to go the related functionalities like Admission advice and admission then look (guess) for the similar kind of bug.

##### Requirements Traceability Matrix: RTM

It is to track back whether the test engineer has covered all the test cases for the entire requirements or not.

Based on the traceability matrix the lead or the client will track whether the test engineer has covered all the test cases or not.

|  |  |  |  |
| --- | --- | --- | --- |
| **Req ID** | **No of TCs** | **Test case id** | **Comments** |
| 1 | 1 | 1 |  |
| 2 | 1 | 2 |  |
| 3 | 1 | 3 |  |
| 4 | 1 | 4 |  |
| 5 | 1 | 5 |  |
| 6 | 1 | 6 |  |
| 7 | 1 | 7 |  |
| 8 & 9 | 5 | 8 to 12 |  |
| 10 |  | Not yet Implemented | Requirement is not clear. Waiting for BA comments |

#### Test Execution:

* The process of executing the test cases on the build in test environment is known as test execution. Whenever the build is released to the testing team the test engineer has to review the SRN document to know the build status.
* Based on the SRN document the test lead will deploy the build and the testing team will perform sanity test.
* Once the sanity test is completed, the sanity test results are mailed to the developer.
* If sanity test is passing the testing team will continue to execute the test cases, if sanity test is failed, the testing team will reject the build back to the development team.
* While executing the test cases the test engineer will observe the actual behavior of the application for the scenario and it will be updated under the actual result field. The same will be continued for all the test cases.

#### Result analysis:

* While executing the test cases the test engineer will update the actual result field then he will compare the actual result with expected result, if both are matching then he will provide the result as pass else he will update as fail.
* For pass we will give the green color, whereas for the fail we will provide the red color. Test execution and result analysis, both are parallel process.

**Note:** Once the test cases execution is completed we are responsible to execute all the types of functional testing's on the application to identify the bugs.

##### How many test cases can we write in a day?

It all depends on all the requirements and test engineer but an average we can write around 40-50 test cases in a day. It means we are taking approximately 8-10 minutes for one test case

to analyze the requirement and to prepare the test case on the test case template with the test data.

##### How many test cases we can execute in a day?

It also depends on the test cases and the application but on an average we can execute 50-60 test cases in a day because to review the test case and execute it on the application.

We are taking around 5-8 minutes to execute one test case on an average.

#### Reporting:

* + The process of reporting/sending the bugs (failed test cases) to the developer is known as Reporting.
  + It is two types.
  1. Reporting the Bugs by using XL files.
  2. Reporting the bugs by using reporting tools. JIRA, ALM, BugZilla

##### Report the bugs by using XL file:

It was the old process we used to have the below template to update the bug and send it to the developer.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Bug ID | Bug title/ Summary | Status | Severity | Prior ity | Bug descrip tion | Screens hot | Build Versio n | Report ed  By | Assigned  to | Comments |
| 1. |  |  |  |  |  | | | | | |

##### Bug ID:

The serial number of the bug will be described here.

##### Bug title/Summary:

The actual result of the bug will be described here.

##### Status:

Based on the bug the test engineers as well as the developer are responsible to give the status. It is below of types.

New - TL, DL

OPEN - DL, DM

ACCEPTED/REJECTED/DEFERRED/NEED MORE INFO

In Progress

Fixed

Ready for Retest

Reopen/ Closed

ACCEPTED

In Progress

Fixed

Ready for Retest

Closed

##### New:

Whenever the test engineer identifies any bug. Initially the status of the bug is New. The new bug will be reported to the developer.

##### Open:

The developer will check that whether the new bug is really a bug or not. If yes then we will update the status from new to open.

##### Fixed/Verified:

Developer will take some time to fix the open bug once it is fixed he will update the status from open to fixed. Fixed bug will be sent to the test engineer.

##### Closed:

Test engineer will check whether the fixed bug is really working as expecting or not. If it is working then we will update the status from fixed to closed. Closed state is the end of the Bug.

##### Re open:

The fixed bug will be tested by the test engineer; if it is not working as expected then he will update the status from fixed to reopen and the reopen bug will be sent back to the developer.

The developer will check that whether it is really a bug or not, if yes he opens it, fix the bug and send it back to the testing team.

##### Rejected/Not a Bug/Hold/Differed:

When the test engineer identified any bug it will be reported to the developer with new status. If developer is not accepting a bug then he will update a status from new to Rejected/Not a bug and it will be sent back to the testing team.

##### Severity:

It describes how seriously the bug impacted the application on testing. Severity means seriousness of the bug. It is below types.

##### Blocker:

If the bug is blocking the entire testing of the module then the severity or type of a bug is Blocker.

##### Very high:

If the bug is blocking partially the testing of the module then the severity of the bug is very high.

##### High:

If the bug is blocking only the specific scenario of the module then the severity is high.

##### Medium:

All GUI bugs severity is medium.

##### Low:

Test engineer is having the option to give the suggestion also. So the suggestion will be reported in the form of bug, where the severity is low.

##### Priority:

Priority describes in which order the bug has to be fixed by the developer. Based on the severity the test engineer will provide priority to the bug as below

|  |  |
| --- | --- |
| **Severity** | **Priority** |
| Blocker/Urgent/critical Very high  High  Medium Low | P1 P2 P3 P4  P5 |

##### Description:

The detailed steps to produce/get the bug will be described here. Based on the steps developer will check that whether it is really a bug or not.

##### Screenshot:

The test engineer will capture the screenshot of the bug and it will be uploaded in the bug template. It is to prove the reported bug is valid and also to understand about the bug.

##### Build version:

The build number on which the test engineer identified the bug will be described here.

##### Reported by:

The test engineer who identified the bug will describe here.

**Assign to:**The developer name or the developer lead name, who is going to fix the bug will be described here.

##### Comments:

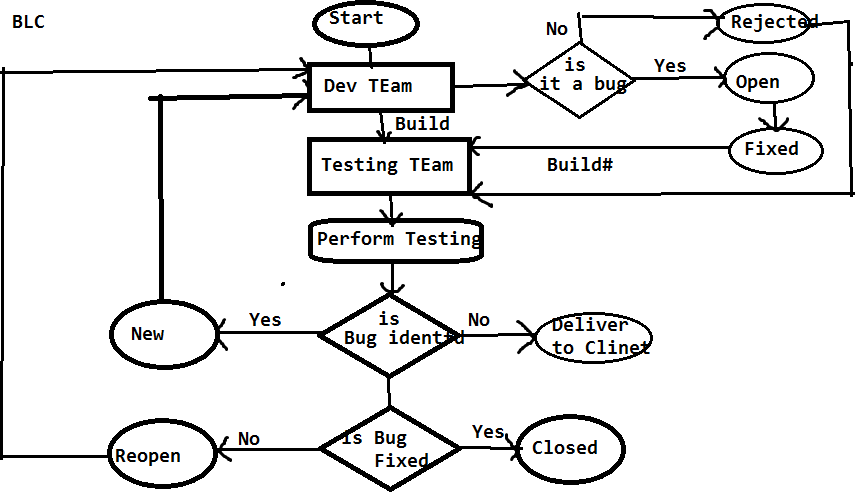
Both test engineers, developer can ask the questions in the form of comments.

**Note:**The XL file reporting will consume lots of time so our plan is to use the reporting tools.

Ex:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Phases** | **BUG ID** | **Bug Title / Summary** | **Status** | **Severity** | **Priority** | **Bug Description** | **Screenshot** |
| I | 1 | Application is not displaying both date  pickers | New | Blocker | P1 | 1. Open Spicejet.com 2. Click on Roundtrip radio button 3. Application is not displaying   both the date pickers | D:\Nagesh\SPicejet |
| II | 2 | Spicejet name is displaying as spacejet | New | Medium | **P4** | 1. Open Spicejet.com 2. Observe the Spicejet logo 3. Its displaying as spacejet | D:\Nagesh\SPicejet |
| III | 3 | Oneway radio button is not  displaying | New | VeryHigh | P2 | 1. Open Spicejet.com 2. Oneway radio button is not available | Path |
| I | 4 | Student check box is not  available | New | High | P3 | 1. Open Spicejet.com 2. Student check box is not available | Path |
| II | 5 | Cheange the color of spicejet home page  to blue | New | Low | P5 | 1. Open Spicejet.com | Path |
| III | 6 | Spicejet club link is not navigating  to Spicejet club page | New | Blocker | **P1** | 1. Open Spicejet.com 2. Click on Spiceje Connect link 3. Spiceje Connect link is not navigating to MySpicetrip page | Path |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| I | 7 | Application is not maintaing GUI | New | Medium | P4 | 1. Open <http://selenium4testing.com/hms> 2. Login into the application 3. Click on Search Registration 4. Application is not maintaining GUI | [Spicejet\_GUI.png](file://localhost/D:/Nagesh/Spicejet/Spicejet_GUI.png) |
| II | 8 | Admission worklist GUI is not maintaining  properly | New | Medium | p4 | 1. Open <http://selenium4testing.com/hms> 2. Login into hms with user1/user1 3. Click on ADT 4. Click on Admission worklist 5. Observe the GUI, its not maintaining properly | D:\Nagesh\hms\_AD |
| III | 9 | Adult field is not displaying on the  page | New | Blocker | P1 | 1. Open [http://spicejet.com](http://spicejet.com/) 2. Observe all the fields 3. Adult dropdown is not   available | D:\Nagesh\Spicejet |
| I | 1 | Application is not displaying Hyderabad and Bangalore | New | VeryHigh | P2 | 1. Open [http://spicejet.com](http://spicejet.com/) 2. Click on LeavingFrom field 3. Application is not displaying Hyderabad and Bangalore | D:\Nagesh\Spicejet |



Q: what is the difference between severity and Priority

Q: what is thedifference between Priority in test cases and Priority in bug template

**Q.**If the developer is not accepting your bug then how you will prove that yours is valid bug?

**A:** Based on the bug description, SRS document, screenshot we will try to prove that the bug is valid if is not accepting it then I will take the sever log to prove the bug is valid, if still not accepting it and then I will send it to a BA, project manager and finally client.

**Q.** Explain the scenario where the bug is having high severity with low priority and low security with high priority?

##### A:Severity Priority

Blocker - P1

High security Very high - P2 High priority High - P3

Medium - P4

Low security Low - P5 Low priority

We have two bugs one is Blocker another one is medium. The blocker will be having high priority and medium will be having low priority.

Based on the severity the test engg.. will provide priority. Based on the priority the dev team is responsible to fix

But the development lead is having the option to change the priority, depends on the situation.

* The bugs which are related to the current phase delivery will be converted to high priority irrespective of the severity.
* The bugs which are not part of the current delivery will be converted to low priority irrespective of the severity.

##### Phase Bug Id Bug title/Summary Status Severity Priority

* 1. **1.** Spice jetname is displaying New Medium P4---P1

As space jet

1. 2. Spice jet connect link is not New Blocker P1 P4

Navigating spice jet connect page

##### Test Report/Build status report:

Once the test case execution is completed on the build then test engineer is responsible to send a test report to the lead as well as client. It below format **Build status Report/Test Report**

|  |  |
| --- | --- |
| **Build Status Report / Test Report** | |
| **Test Engg Name:** |  |
| Build No | 1 |
| Login Credentials |  |
| Browser | FF, IE GoogleChrome |
|  |  |
| **Test Matrics** |  |
| Total no of testcases | 200 |
| No of Test cases executed | 150 |

|  |  |
| --- | --- |
| No of Test cases pending | 50 |
| No of Test cases Pass | 100 |
| No of Test cases Fail | 50 |
| No of Test cases Skipped | 10 |
| No of Bugs Reported | 3 |

##### Test Metrics:

Metrics means measurement of the task. Test metrics means measurement of the testing.

##### Pending:

If the developer has not giving functionality at all then those test cases can’t be executed. It’s comes under pending.

##### Skipped:

Developer has given the functionality, but we are unable to test the functionalities, because of the dependent functionalities failed.

Ex: if Login failed, we can’t execute compose. Compose test cases comes under skipped.

* The reporting will be continued until the build is stable, stable means majority of the test cases are pass and no blocker bugs are available in the reporting tool.
* The stable build will be delivered to the client.

##### Q: Explain me the reporting structure in your organization

Report the Bugs By using Reporting Tools:

* Any reporting tool having two kinds of users: One is admin user and another one is End user.
* Admin user: The admin user is responsible to create the project, create users like test engineers, developers…etc. He will assign the user to the project
* End user: He is responsible to use (report) or receive the bugs ex: test engineers, developers…etc.

Ex: QC, JIRA, Bugzilla, Redmine, Test manager etc… BugZilla:

* Access the Bugzilla by using selenium4testing.com
* Then click on Bugzilla.
* Login into the Bugzilla as a test engineer ([jan30selenium@gmail.com](mailto:jan30selenium@gmail.com)& password : selenium)
* By using Bugzilla we can perform below activities.

1. Reporting a Bug.
2. Search for the bugs.
3. We can take the report.
4. Preference.

API testing - Application Programming interface

Postman, SoapUI, Swagger

CRUD - Create, Retrieve, Update, Delete

HTTPS Methods

GET - fetch the data from Server similar to SELECT Command in DB

POST - to create the data in server similar to INSERT command in DB

PUT - if data exists in server it will update else will create new data in server similar to update command

PATCH - update the data for particular value similar to update command with condition

DELETE - delete the data from server similar to DROP/DELETE/TRUCANTE command in DB

Mostly we used GET and POST methods

HTTP STATUS CODE

1XX - Information message

2XX - Success Message (200 Success) (201 Data created successfully in server)

3XX - Redirectional Message 301 ..

4XX - Client side errors (400 Bad request) (404 NOT FOUND) (401 Unauthorized)

5XX - Server Side errors (500 Internal Server Error)(503 Service Unavailable) (504 - Gateway time out)