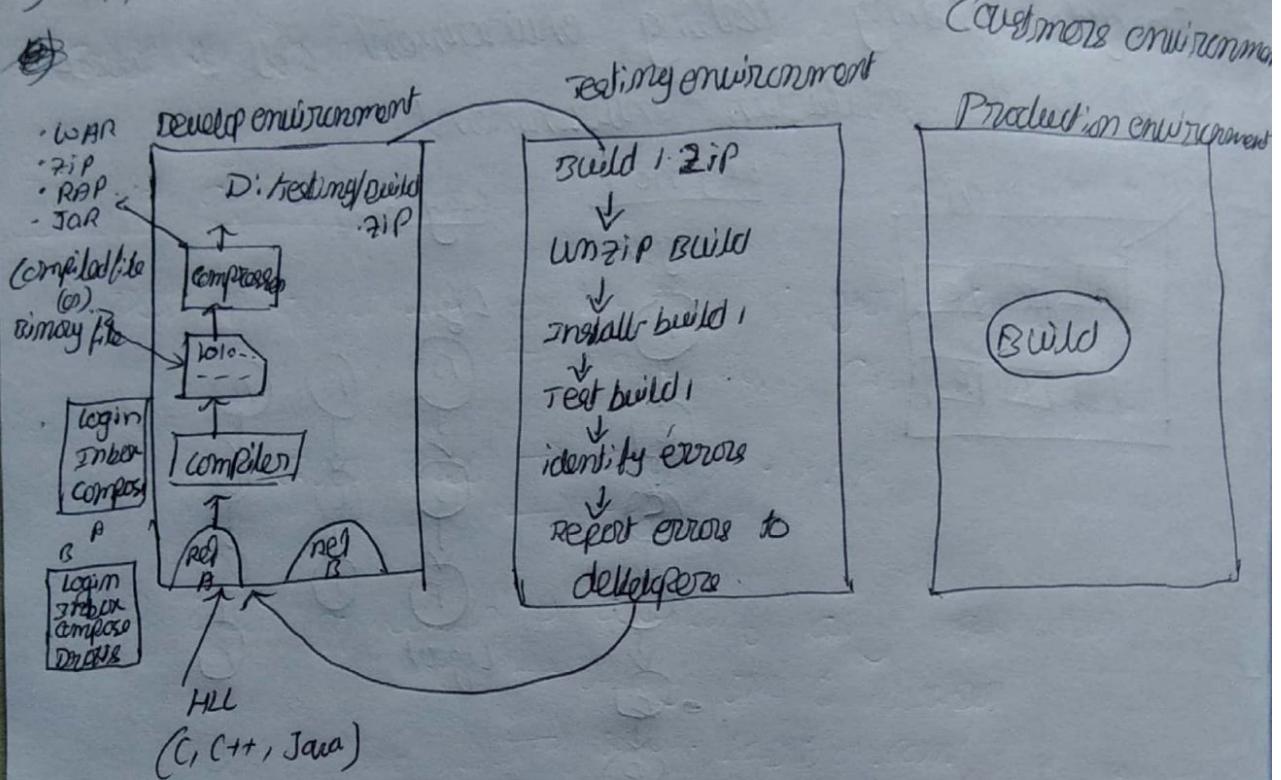


Types of environments

- 1) Development environment
- 2) Testing environment
- 3) Production environment. (a) Client environment (b)



Development environment (a) server

The environment where only the development activities will be performed, we call it as development environment or development server.

Testing environment

The environment where only the testing activities will be performed called it as testing environment (a) testing server

Production environment

In this environment, the developed and tested software will be installed

→ customers will come to the company & gives requirements & the developer will start writing the source code (a) main programs for requirements. Now this high level code is given as an input to the compiler, where in compiler converts high level code into corresponding machine level code and this file is called a binary file (b) compiled file. Now this binary file is given as an input to compressor, where the compressor compresses the obtained binary file into either of these four formats.

(war, .zip, .RAR, .JAR) and obtained file are called it as build.

Build

Build is a piece of software which is compiled & compressed and is installed into the testing environment.

Types of software

1) Application software which ever the software that is used to perform a task that is assigned by the user is called it as application software.

e.g. calc app, fb, slack → These are coded in Java.

→ System software the software that is used to run application software is called a system software.

e.g. O.S, drivers → These are coded in C, C++

3) Scientific software

The software that are used for scientific purposes are called as scientific software.

e.g. CADD, S/W in rocket launching

Coded as C, C++.

Types of application

1) Desktop applications
→ NC media player, calculator, note pad
→ no internet connection.

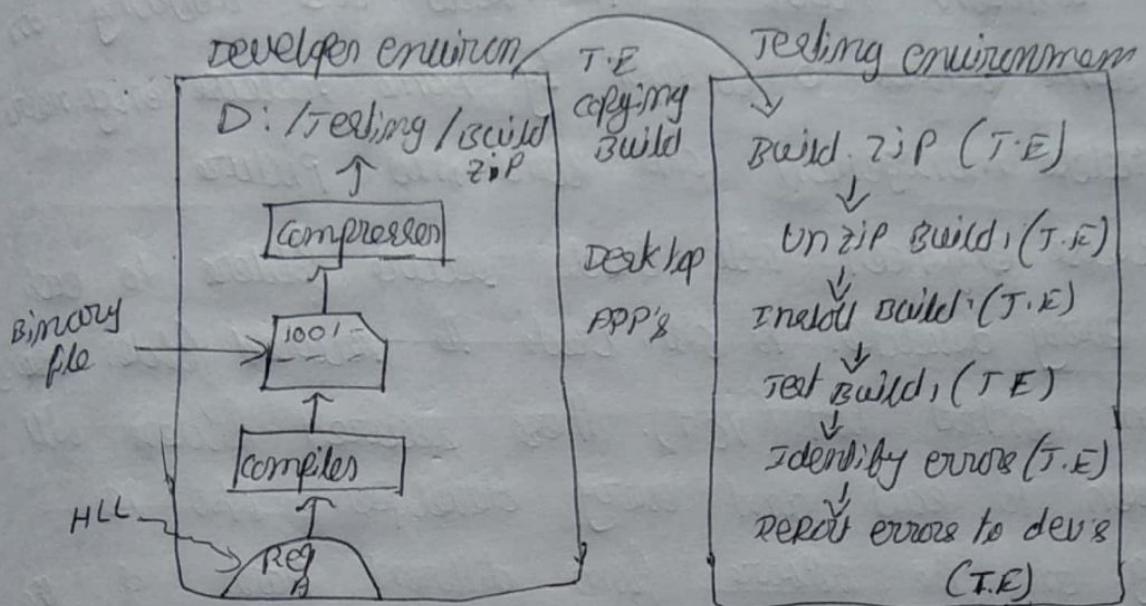
2) Client-Server Based
→ Gotoapp, TikTok.
→ Icons.

3) Web-based application
→ accessing any application
→ using URL & Browser
→ www.Facebook.com, www.gmail.com

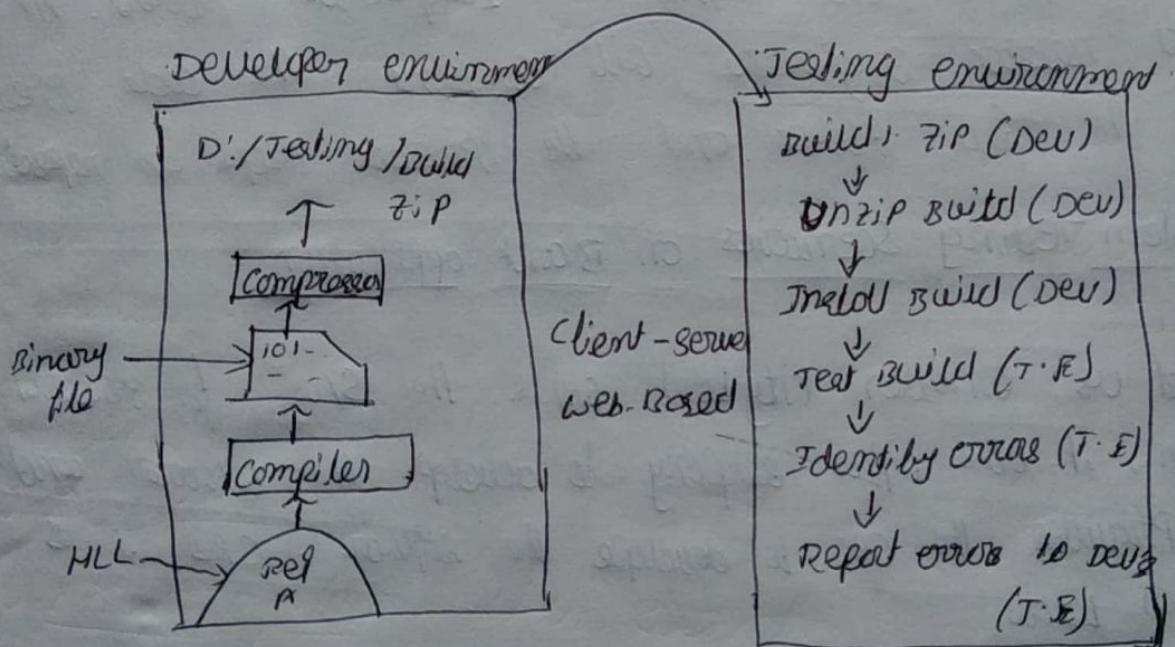
Different people who are involved in carrying the build from development environment to testing environment

- 1) Developers copying the build
- 2) Test engineers copying the build
- 3) Release engineers copying the build

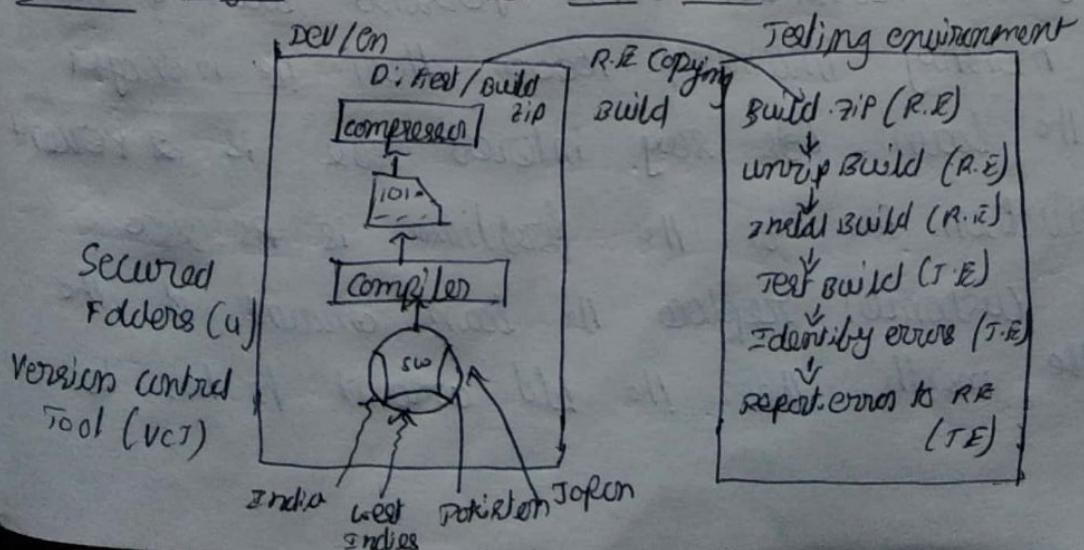
Test Engineer copying build



Developers copying the build



Release Engineers copying the build



→ whenever there is a software, where developers from different parts of the world are working on the same software or concept by name release engineer, (or) release management will come into picture.

Release engineer will create secure folders to each and every developer he will create the secured folder using version control tool (VCT). These secured folders will be given to each and every developer who is working on the software from different parts of the world. now the duty of this developer is to upload their respective code into the respective folders. at last release engineer will combine the code present in the four folders and the process keeps on repeating.

System Testing Scenarios on Bank application

Ex:-

Let us consider city bank wants the SW for personal loan, it ask wipro company to develop the software and it provides the CRS to develop the software. CRS is as shown below.

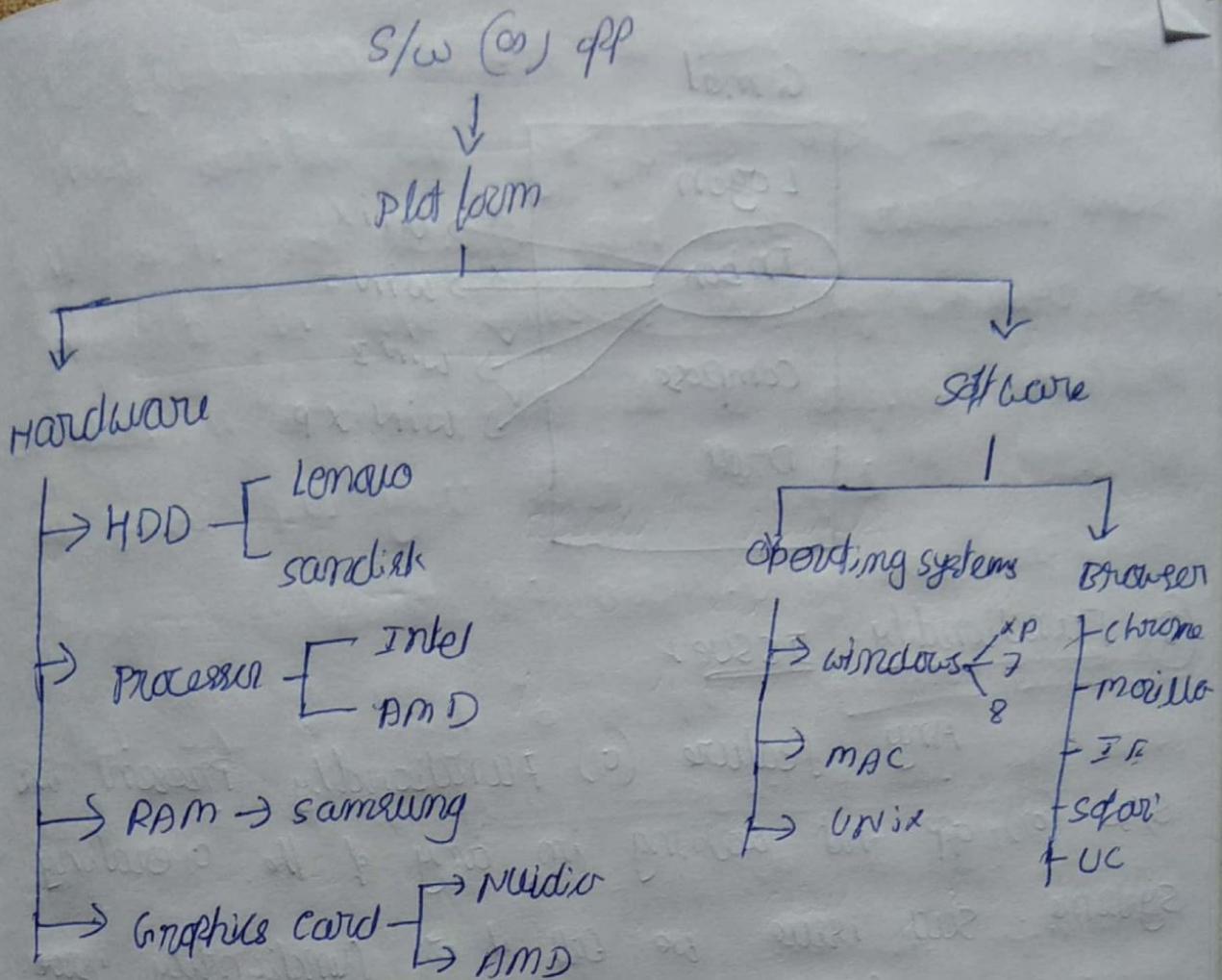
→ if city bank customer wants personal loan of 20000 over is monthly income of 10000, then the manager approves the loan. Let's say interest rate is 2 percent and activation fee for the first time is RS. 250. when the customer replace the loan amount of the end of the month, then the total amount he has to

4) ~~new~~ ~~new~~ ~~new~~ ~~new~~ ~~new~~ ~~new~~
on the software and the total time is
5 days. On 6th day the software should be
delivered to the customer. The test engineer will perform
deep testing on all the functionalities. And on the 8th
day the test engineer will identify one error for which
developer will take 3 days extra to fix it. Because of this
reason the delivery of the software has been postponed to
the customer by three days. This is not the correct approach
of testing.

Initially we should test the important functions
so that it doesn't affect the delivery process. Once
the basic functionalities are tested then we ~~should~~
can perform deep testing (functional testing, integration &
system testing).

- smoke testing is positive testing.
- Smoke testing is performed either manually or
by using automation tools.

Compatibility Testing: Testing the software (or) application
in different hardware or software platforms is
only called compatibility testing.

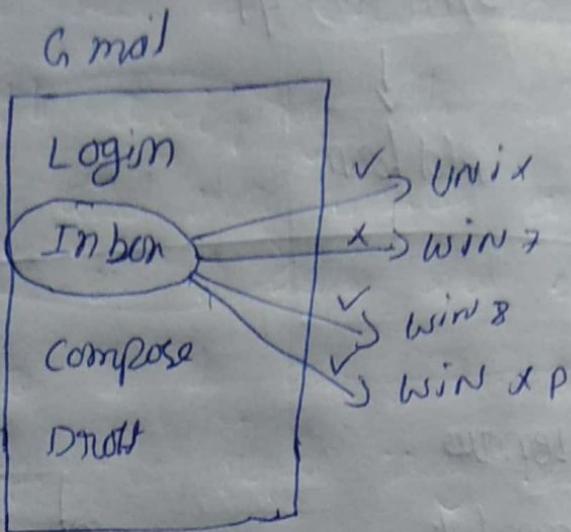


After performing compatibility testing, He test engineer two type of issues.

- 1) compatibility issue.
- 2) functionality issue.

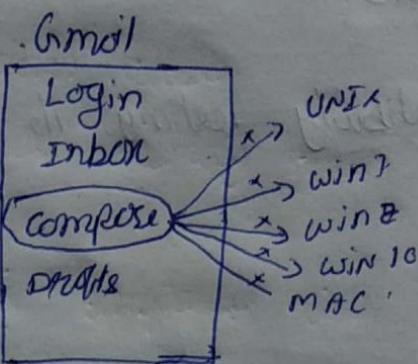
① Compatibility ISSUE

Any Feature (or) functionality present in the S/w (OS) application which is not working in one operating system(OS) but it is working fine all the other os, such issue called as compatibility issue.



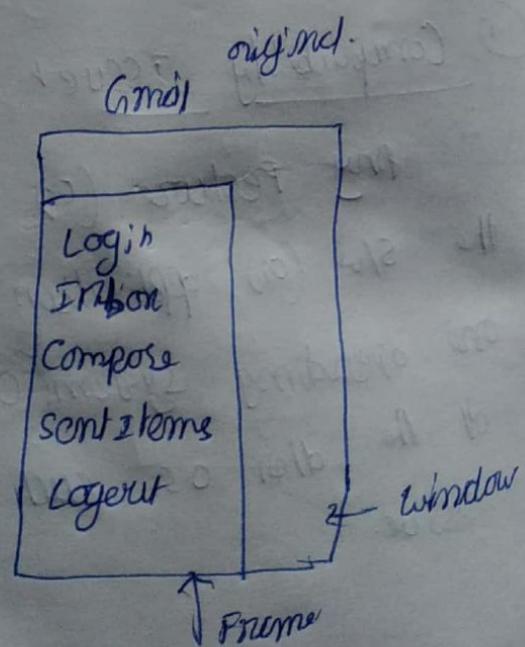
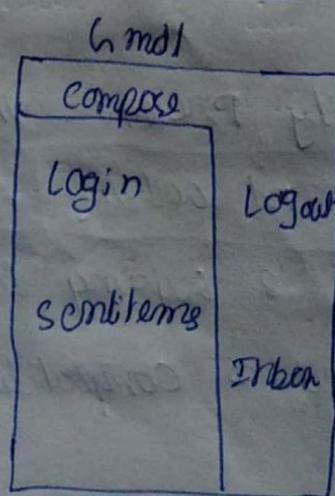
② Functionality ISSUE

Any feature (a) functionality present in S/w (b) off not working in any of the operating systems. Such issue we call it as functionality issue.

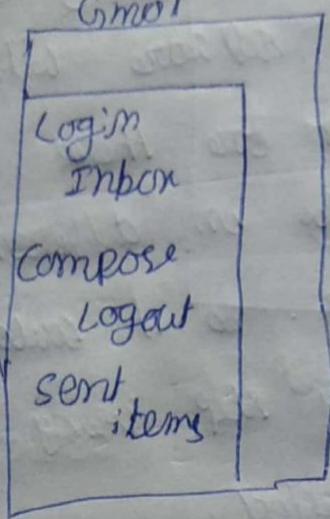


List of some compatibility issues

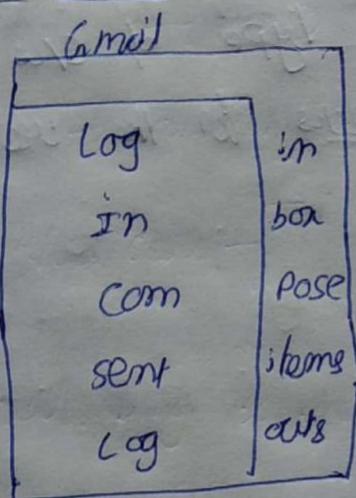
① Scattered contents



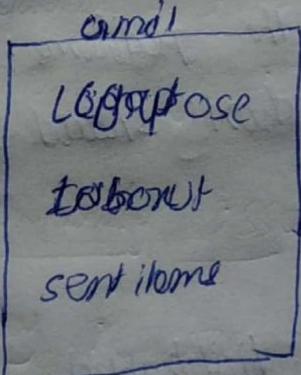
2) Alignment Issues



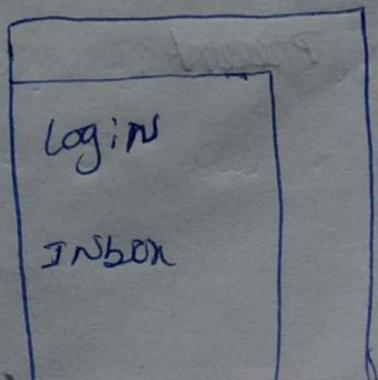
3) Broken Frames



4) Text overlapping



5) Change in font



* why developers need to perform compatibility testing?

→ Developers while developing the software with some hardware and more chances are there that we user might use the same s/w in different platforms compare to the developers so it could lead to defects or bugs so people may stop using the s/w, thus business interest will effected.

Usability testing - It is a type of s/w testing where the testing engineer checks for the user friendliness of any s/w (or) application

- 1) Simple in use
- 2) Freely downloadable
- 3) Occupies less memory space
- 4) Faster
- 5) Should be compatible with all the platforms
- 6) Font colors used in the application should flexible to the users
- 7) Should be add free
- 8) important features should be highlighted
- 9) secured
- 10) Help option should be present.

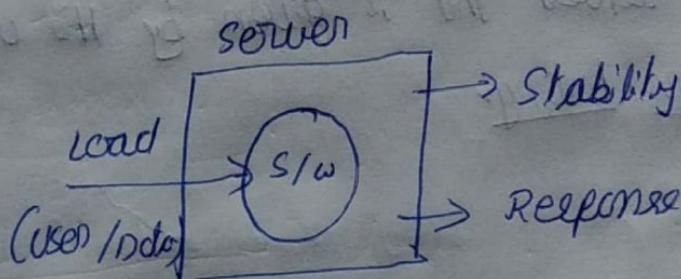
Accessibility Testing

It is a type of S/W testing, where the test engineer check whether the S/W or application can be used by the disable people with colour blindness deaf and dumb.

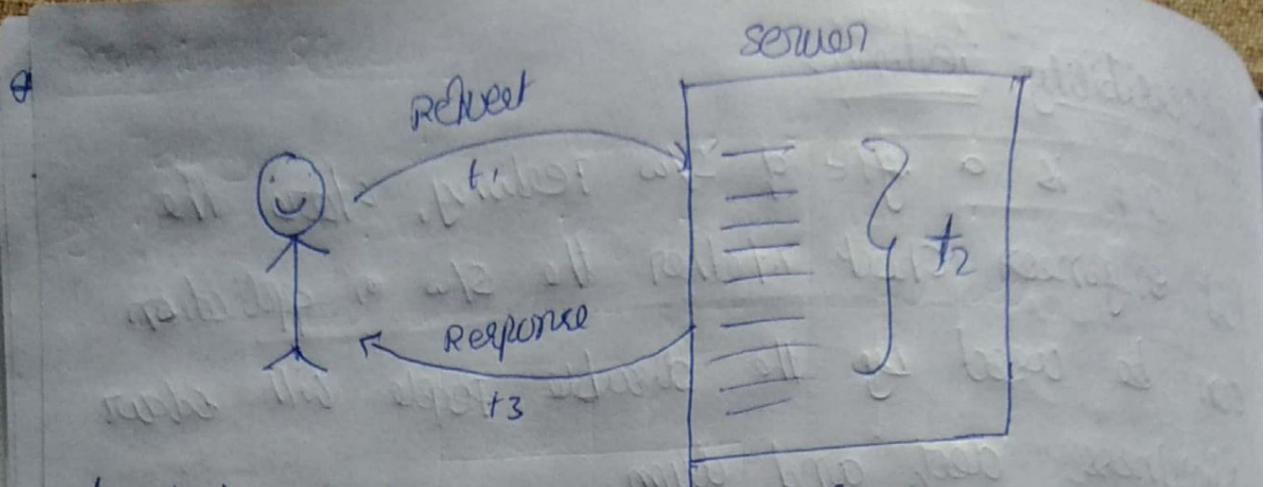
Checking features for accessibility testing are :-

- 1) Checks whether voice recognition functionality is present (o) not.
- 2) Checks whether audio / video content is present (o) not
- 3) Font colours used in the S/W should be flexible to users.
- 4) most of the features should be flexible with keyboard.

Performance Testing



Checking the stability and response time of the software by given load (as input (users)) is called as Performance Testing.



$t_1 = t_r$ is the time taken by the request to reach the server.

t_2 = time taken by the server to process the request

t_3 = time taken by the response to reach the client

$$\boxed{\text{Response time} = t_1 + t_2 + t_3}$$

Response time → The total time that is taken by the customer to send the request, the time taken by the server to process the request and the time taken for response to reach the client is called response time.

Stability → It is the ability of the application to handle load.

Crash → The situation at which the server will be unable to respond to the request that is given by the user.

Types of Performance testing

- 1) Load Testing
- 2) Stress Testing
- 3) Scalability Testing
- 4) Soak / endurance Testing
- 5) Volume Testing.

1) Load testing: Testing the stability and the response time of the S/W by applying the load (users/data) less than or equal to the desired no. of users until desired goal is achieved is called load testing.

load
by
developer \Rightarrow 100 users
5 sec

no of users	R.T
100	10 sec
75	8 sec
50	5 sec

but for
it 50 users
5 sec

stress testing: Testing the stability and the response time of S/W by applying the load (user/data) greater than or equal to the desired ~~goal~~ until crush point is arrived

no of users	Response time
100	10 sec
200	20 sec
500	50 sec
1000	(crash)

scalability:

upward downward.

upward scalability:

no of users	R.T
100	10 sec
200	20 sec
300	30 sec
400	40 sec
500	crash

Testing the stability and the response time of the software by applying the load (data/user) in fixed interval greater than or equal to the desired load until crush point.

Down ward scalability

Testing the stability and the response time of the S/W by applying the load in short interval less than or equal to the desired load until desired goal is achieved.

No. of users	R.T
100	10 sec
90	8 sec
80	5 sec

Scal Testing / endurance

Testing the stability and the response time of the S/W by applying the load continuously for the particular duration of the time.

Volumer

Testing the stability and response time of the S/W by applying huge volumes of data.

Ex, Examint results.com

Some of the automachine tools in order to performance

- 1) new load
- 2) ~~new~~ win runner
- 3) J meter → it is freely downloadable
- 4) QTP.

Test cases

Test scenarios

- It is a high level document which consists of all the possible ways that is required in order to test any software (or) application.
- T.S are written in ms word document
- TS will be written by reference to SRS document

Test cases

- It is a low level document which consists of input that is required to test any software (or) application.
- It is written in ms excel document.
- Test cases to be written by reference of SRS and TS documents

Note Test cases will be written by author (test engineer).

Header section

- 1) Project name - Gmail
- 2) module name / component name - sign in
- 3) Test Case Type - F.T
- 4) Requirement no - 1
- 5) Reference Document - SRS & TS
- 6) Pre-Condition - Browser
- 7) Test - Data - www.gmail.com

Body section.

Comments

Test Case ID	Procedure (C) TS	Inputs (E) T.C	E.O	A.O	Status	Severity	Priority Defect no.
000 001	open the browser and enter URL present in test data and click enter	www.gmail.com	Signin page of Gmail should be opened				
000 002	Enter valid Email/Phone no in Email/Phone no T.F and click enter	Subtu 143 anu@gmail.com	ask user to enter the password				
000 003	Leave blank space in Email/Phone no T.F & click enter	Nothing	error message should be displayed				

Footer Sec

- 1) Author - XYZ
- 2) Date of creation of T.C - 13/9/19
- 3) Reviewer - ZYX
- 4) Date of reviewing ~~b~~ of T.C - 13/9/19
- 5) Approved by - YZX
- 6) Date of approval of T.C - 13/9/19
- 7) Executed by -
- 8) Date of execution of T.C -

Header section

- 1) Project name - Project - 1
- 2) module name / component name - URL
- 3) Test case type - F.T
- 4) Requirement no - 0
- 5) Reference document - T.S
- 6) Pre condition - Browser
- 7) Test - Data

Body section

Test case Id	procedure (a) TS	Inputs	E.O	A.O	Status	Security	Ticket no/ Deleted no	Comments
000001	open the browser enter char in browser address bar	ABC.com	entered char should be visible					
000002	Enter first letter of URL in brow ser address bar	www. gmail	suggestions regarding to URL should be displayed					
000003	Enter invalid URL in B.A.N and click enter	www. gmaile. com	E90201					

Footer section

- 1) Author - XYZ
- 2) Date of creation of T.C - 13/9/19
- 3) Review -
- 4) Date of reviewing of T.C
- 5) Approved by
- 6) Date of approval of T.C
- 7) Executed by
- 8) Date of execution of T.C

Test case design Techniques

In order to derive limited set of test cases or inputs we require Testcase design techniques.

1) Boundary value Analysis (BVA) + $(0-120)$ range

$$\text{min value} - 1 = -1 \quad (\times)$$

$$\text{min value} + 1 = 1 \quad (\checkmark)$$

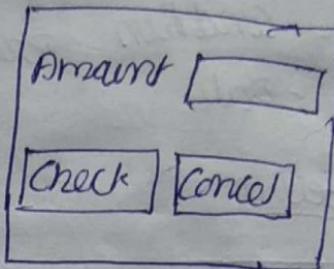
$$\text{min value} = 0 \quad (\checkmark)$$

$$\text{Avg value} = \frac{0+120}{2} = 60 \quad (\checkmark)$$

$$\text{max value} = 120 \quad (\checkmark)$$

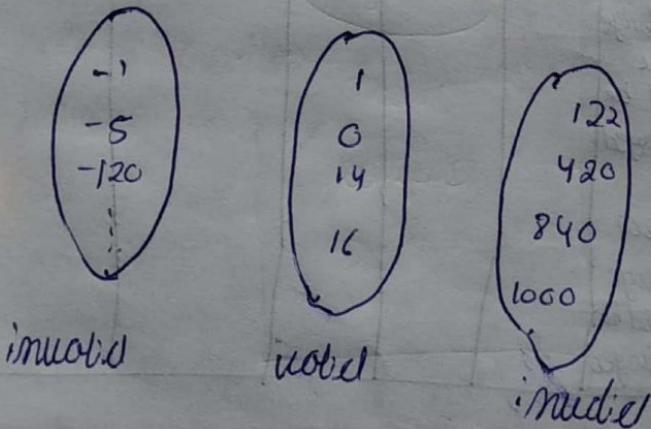
$$\text{max value} + 1 = 121 \quad (\times)$$

$$\text{max value} - 1 = 119 \quad (\checkmark)$$



2) Equivalence Class / Partitioning

$-1, 14, 420$ Inputs

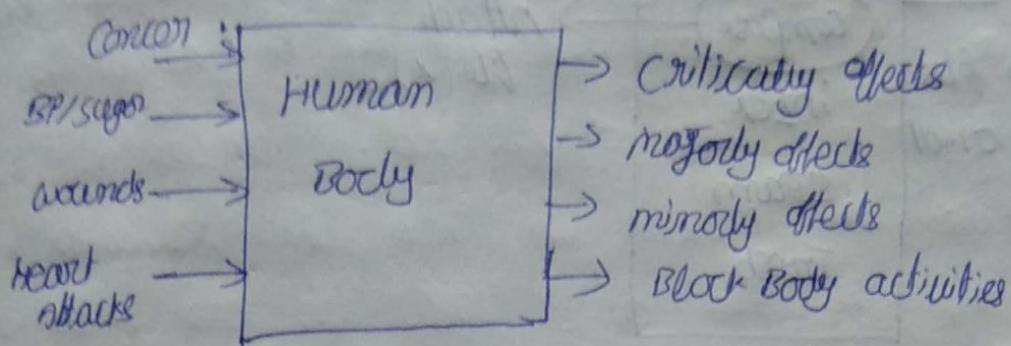


Error guessing

- Error guessing is done by highly experienced test engineers (nearly 20 years experience)

Severity or the impact of bug on any software

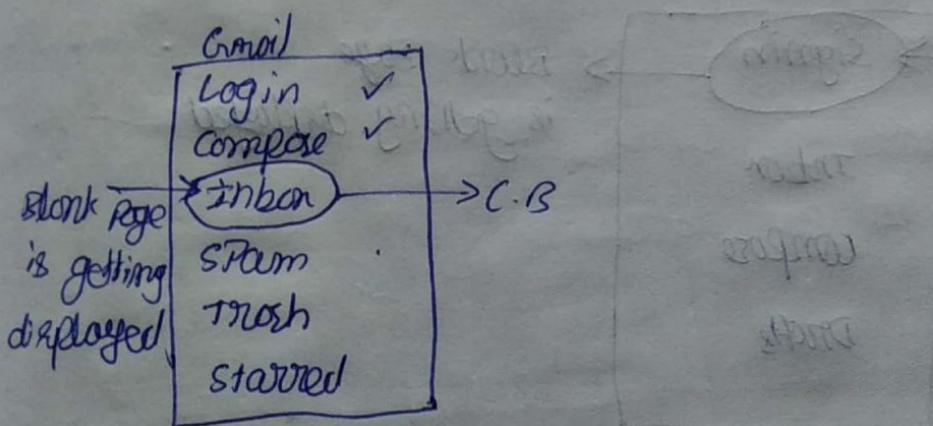
(a) application is called as severity.



Type of severity

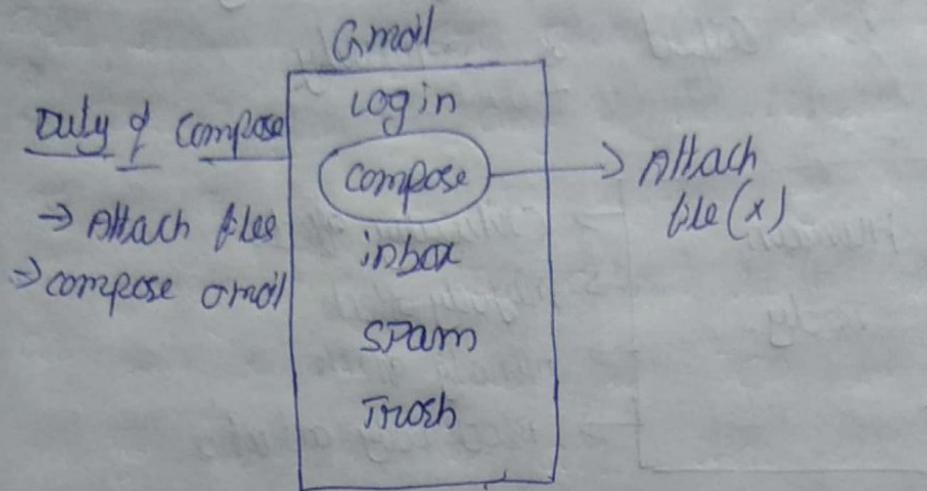
- 1) critical bug
- 2) major bug
- 3) minor bug
- 4) blocker bug

1) Critical Bug



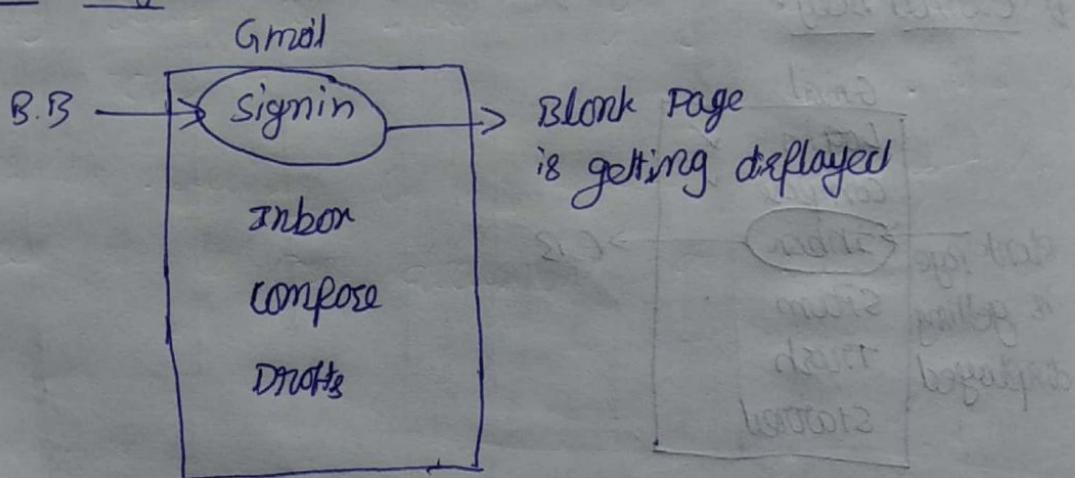
which ever the bug that breaks down the operation of the important features present in the software (a) application (b) build so that the test engineer will not be able to use or test some of the ~~features~~ other functionality present in the software (c) application.

2) Mayer Bug



which ever the bug that breaks down one of the important feature present in a software, so that the test engineer will be having afterwards solution to continue his further testing activities.

3) Blocker Bug

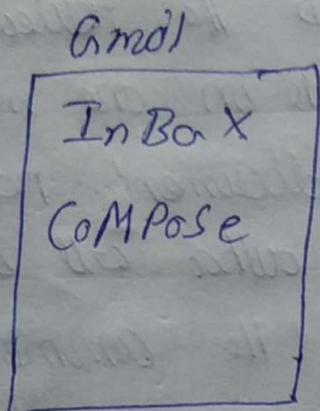


which ever the bug that shuts down (a) breaks down and important feature present in the software (a) application, so that the test engineer will not be having any solution to continue his further testing activities.

ii) minor bug

→ minor bug can be identified by following factors.

- 1) spelling mistakes
- 2) change in font size, name & style
- 3) company logo not found

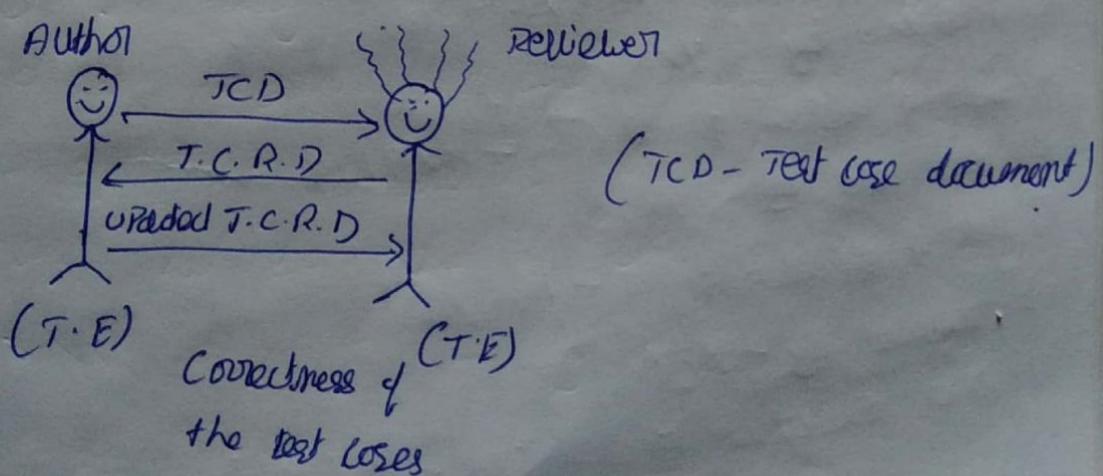


Priority

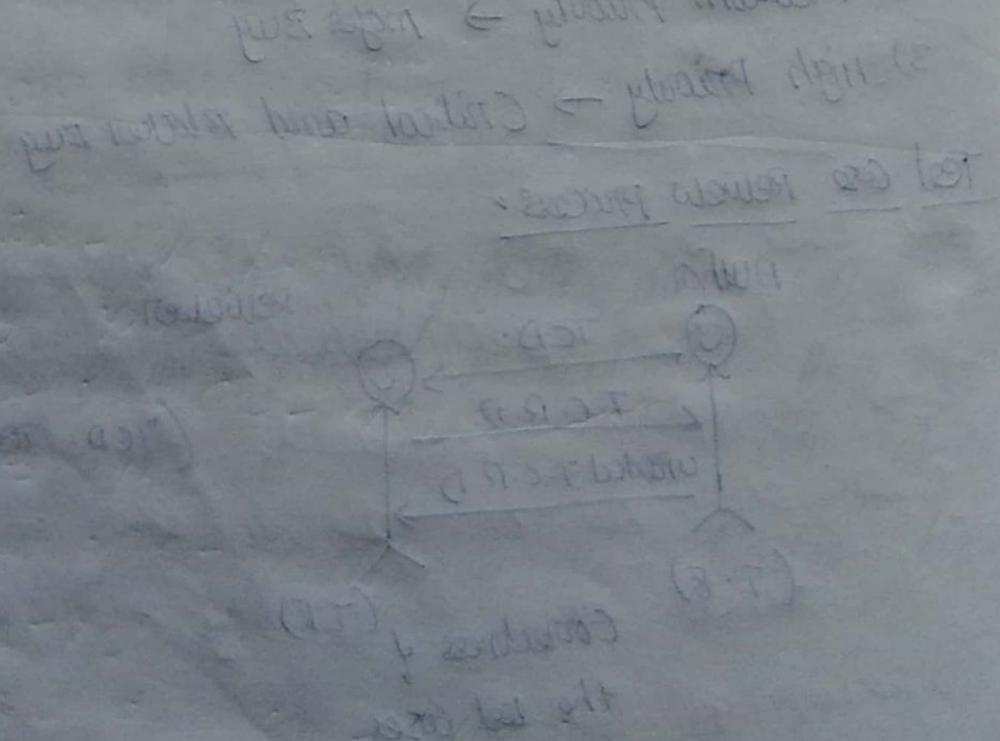
The order in which the bugs are to be fixed by the developer is called as priority.

- 1) Low Priority → minor bug
- 2) medium priority → major bug
- 3) High Priority → Critical and blocker bug.

Test case Review process

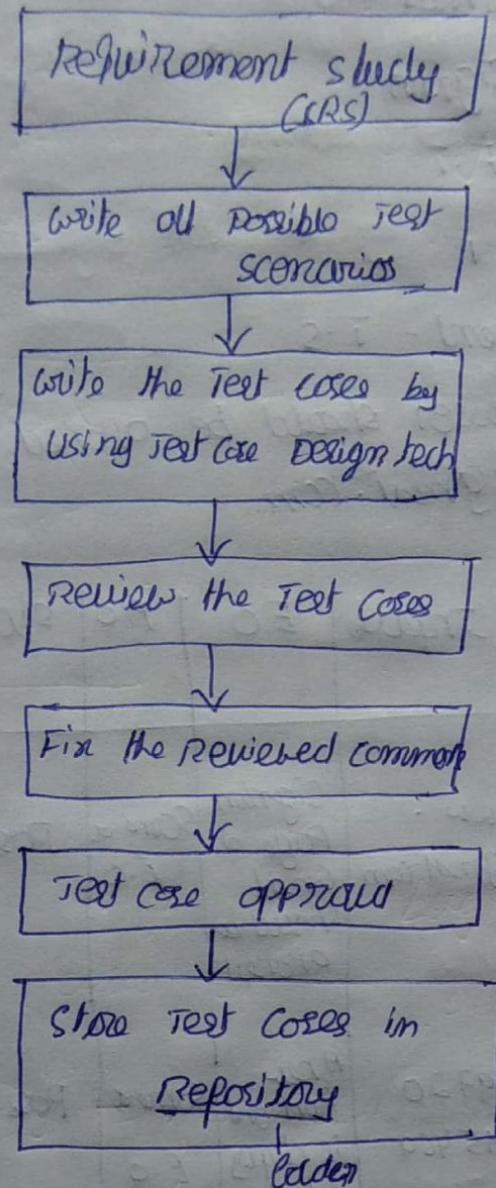


- Author will be writing the test cases in test case document, that testcase document will be send to the reviewer. now the duty of the reviewer is to check whether all the testcases returned by the author in test case document is correct (or) ~~not~~ not. In case if there are some mistakes present in TCD the reviewer will list out their mistakes and update in one more document called as testcase review document. now TCRD will be send to the author now author will be fixing those comments, and after fixing the comments the updated TCRD will be sent to reviewer. and again reviewer will check the test cases are correct or not. if any errors are there is going to exist those errors in TCRD and he send to the author. This process will keep on repeating until reviewer finds all the test cases in test case document are correct.



Test Case Execution

Test steps involved before test case execution



Go through the procedure section and inputs section in the testcase document, now apply those inputs on the software. This process is only called as Test case execution.

Case 1 & then there are no errors.

- 1) Project name - Gmail
- 2) module name - Signin
- 3) Test case type - F.T
- 4) Requirement no - 1
- 5) Reference document - T.S
- 6) Pre-Cond - Browser should be opened
- 7) Testdata - www.gmail.com.

similarly

Test case ID	Procedure	Inputs	E.C	A.O	Status	Ticket no/ defect no	Comments
000001	Open the browser type URL present in testdata and click enter	www.gmail.com	Signin page of Gmail should be opened	Same as E.C	Poss	NA	NP
000002	Enter valid phone no in e-mail Phone no T.F. click enter	99720 45469	A new front end is displayed after user can either hit password	Same as E.C	Poss	NA	NP

- 1) Author - XYZ
- 2) Doc of Test Cases - 16/09/2019
- 3) Reviewed by - XYZ
- 4) Date of Review of Test Cases - 16/9/19
- 5) Appraised by - ZYX
- 6) Date of approval of TC - 16/9/19
- 7) Executed by - XYZ
- 8) D.O Execution of Test Cases - 16/09/19

C-2 When there are some errors

- 1) project name - Gmail
- 2) module name - inbox, compose
- 3) Test case type -
- 4) requirement no -
- 5) Reference document -
- 6) Pre-Cond- Browser should be opened
- 7) Test data - Abh.

Test Case ID	Procedure	Inputs	E O	A O	Status	Seriousity	Tickets/defect no	Comment
000001	open the browser type url Present in Test data and click enter	www. Gmail.com	Sign in Page & email should be displayed	Blank Page is displayed	Fail	Blocker bug	defect 001	NA
000002	click on inbox link	NA	Inbox page should be displayed	Blank page is displayed	Fail	Critical bug	defect 002	NA

- 1) Author -
- 2) Doc of test case
- 3) Reviewed by
- 4) Date of Review of Test Cases
- 5) Approved by - 2yx
- 6) Date of approval of T.C
- 7) Executed by
- 8) D.O Execution of Test Cases

Defect Tracking

The process of identifying the defects, managing the defects and updating the defects to the developer is only called as defect tracking.

* Defects can be reported by the test engineer to the developer in two ways

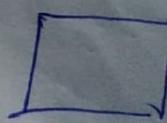
- 1) By using defect report
- 2) By using Bug Tracking tool.

→ mantis

→ bugzilla

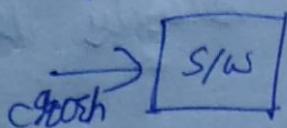
→ Jira

Defect report

- 1) Project name - Gmail
- 2) module name - Signin
- 3) Ref no -
- 4) Defect no - Defect 000001
- 5) Summary - Signin page is not getting displayed.
- 6) Description - Open the browser, type URL present in Test data, click enter
- 7) issue - Blank page is getting displayed.
- 8) severity - blocker bug
- 9) Priority - High
- 10) Assigned to - Dev's name
- 11) Screen shot - 

Recovery Testing

introducing the crash into the software and checking how much time this software will take to recover from the crash is only called as recovery testing.



S-1 / Procedure: open multiple applications in multiple tabs of the chrome browser, suddenly disconnect the power supply that is given to the desktop. After some time, provide the power supply back.

Eo: Recovery message should be displayed "Restore previous sessions".

S-2 / Procedure: open multiple applications in multiple tabs of the chrome browser, suddenly disconnect internet that is given to desktop. After some time again connect the internet.

Eo: Recovery message should be displayed "Restore previous sessions"

Installation Testing

it is a type of testing which is performed in order to check whether the software is working as expected after the installation.

→ it is also called as implementation testing

Security Testing - It is a type of testing where the test engineer will check whether the entire ~~software~~ application is secured or not.

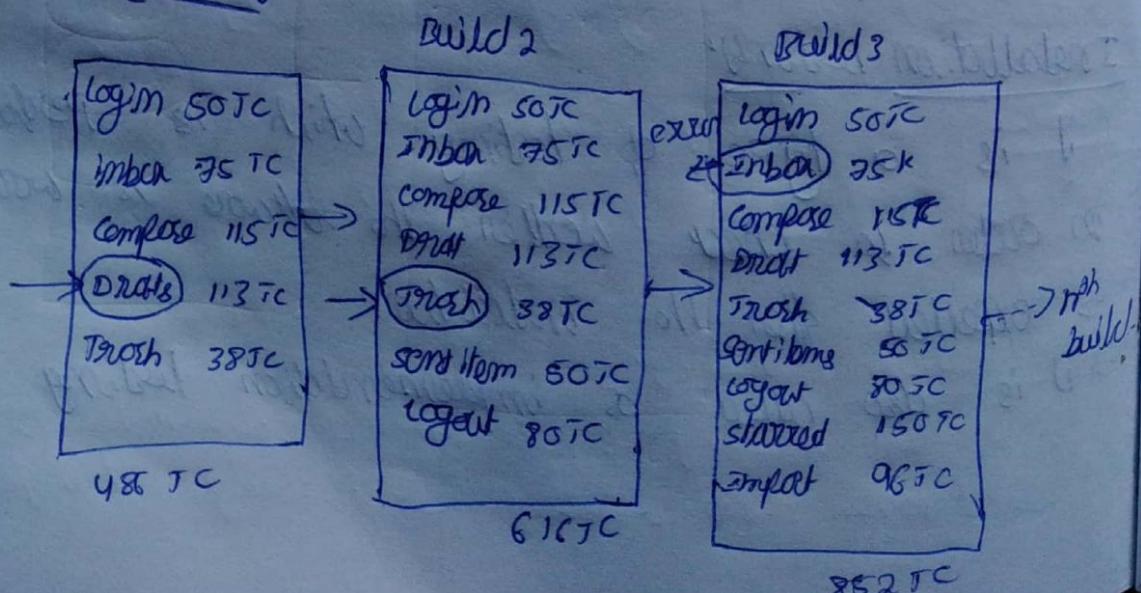
→ ~~below~~ some of the factors the test engineer will consider while performing security testing in any application (or) software.

1) Confidentiality - No data available (or) present in the software or application should be confidential to the relevant users. This can be achieved by performing security testing on the software.

2) Availability - the data regards to respective software or application should be available to the users through out the day (or) atleast during the working hours.

3) Integrity - the data that is stored in the application (or) software should be correct. This is taken care by performing security testing.

Regression Testing



Executing (a) testing we come all functionalities again and again across multiple builds so that when ever the developer adds new functionalities to every build, there are chances that old functionalities might get effected. This is only called as regression testing.

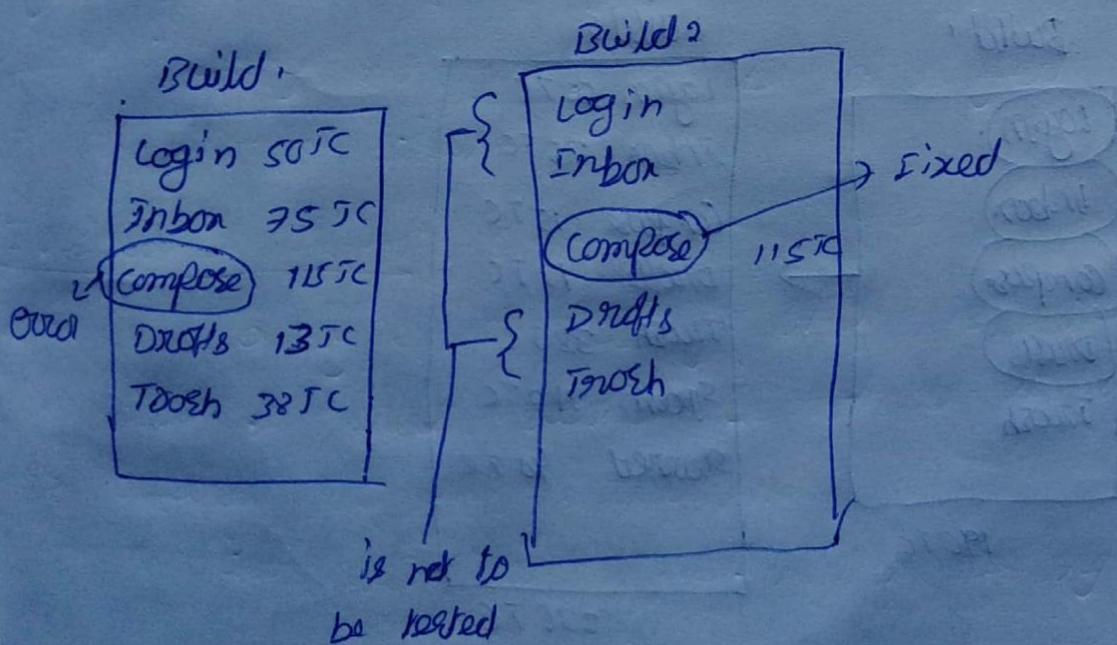
* note Regression testing is performed using automated tools

Types of Regression testing

- 1) unit regression testing
- 2) Regional Regression testing
- 3) Full Regression testing.

1) unit regression testing

Performing regression testing by considering a single unit that has changes in it in the next build. is called as unit regression testing.



→ only composed will be tested.

2) Regional Regression testing

Performing regression testing on the functionality.

- (a) Feature which has changes in it and related functionalities is called as regional regression testing.

Build 1

login	50 TC
inbox	75 TC
Compose	15 TC
Draft	13 TC
trash	38 TC

login
Inbox
Compose
Draft
trash
Starred
Important

related functions need to be tested

3) Full regression testing

Performing regression testing on the entire build.

- is called as full regression testing.

Build 1

login
inbox
Compose
Drafts
trash

196 TC

Build 2

login	50 TC
inbox	75 TC
Compose	15 TC
Drafts	17 TC
trash	38 TC
spam	110 TC
Starred	40 TC

336 TC

login	50 TC
inbox	75 TC
Compose	15 TC
Drafts	17 TC
trash	38 TC
spam	110 TC
Starred	40 TC

Functional requirements

Any requirements that tell you about the working of each and every functionalities present in the software (or) application are called as functional requirements.

Functional testing

Testing performed on the functional requirements is called as functional testing.

Non Functional requirements

Any requirements that tells you about the quality attribute with regards to any software (or) application are called as non functional requirements.

Non Functional testing

Testing performed on the non functional requirements is called as non-functional testing

Exploratory Testing

It is a type of software testing where the test engineer will explore the functionality first in any software or application and then perform testing.

Static testing

Testing which is performed without the execution of my code or programme is called static testing.

* checking whether SRS is properly converted
SRS are not., checking

Dynamic testing

Testing is performed after the execution of code
or programs.