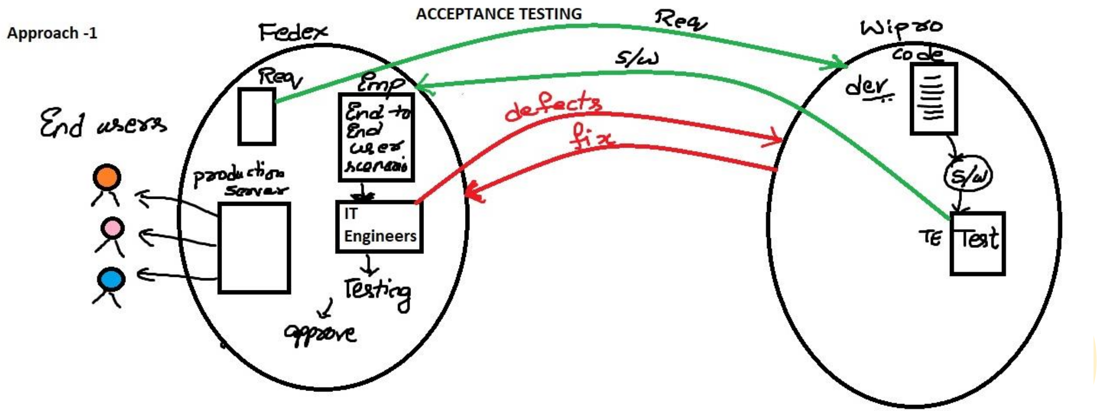
Manual testing

Contents:

ACCEPTANCE TESTING

Approach-1

It is an end to end testing done by the IT engineers who are in the customer's place where in they take all the real time business scenarios and check whether the software is capable of handling it or not.

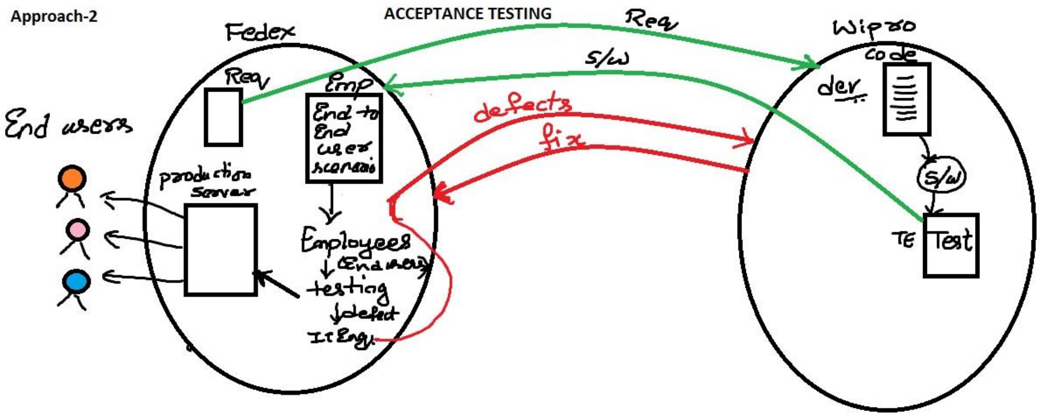


Why do we do acceptance testing?

1) Under business pressure software company might push the software to the customer with critical bugs, to prevent it they do acceptance testing
2) If they use the software with the critical bugs for the business they will undergo severe loss , to avoid that they do acceptance testing
3) Chances are there development team would have misunderstood the requirement and develop wrong features and to find such issues customer will do acceptance testing

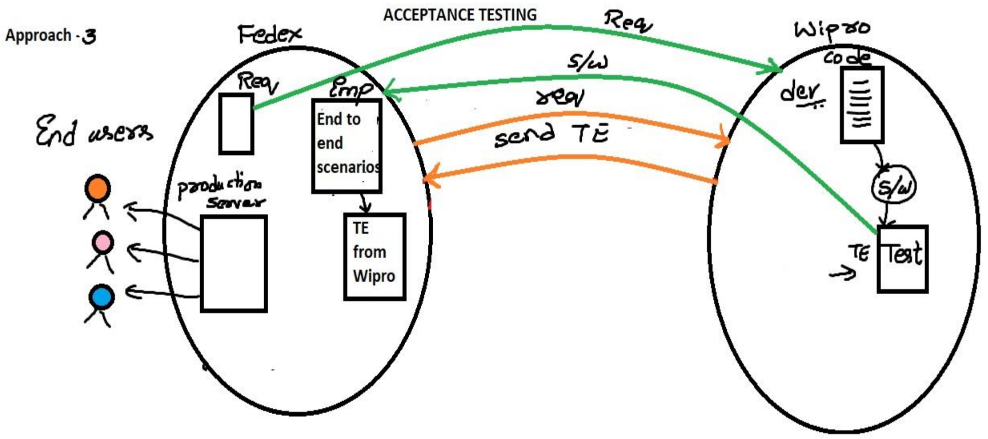
Approach-2

It is an end to end testing done by the end users where in they test the software for business for particular period of time and check whether the software is capable of handling all the real time business scenarios



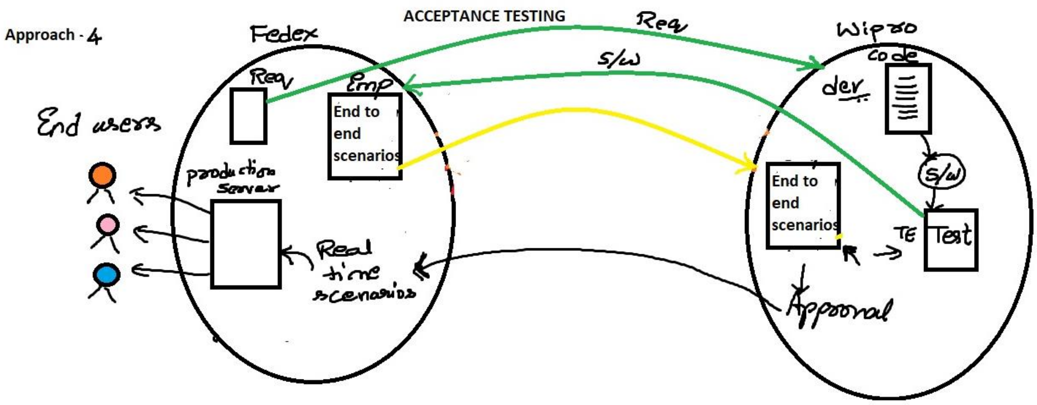
Approach-3

It is an end to end testing done by the test engineers of the company at the customer place looking in to the end to end business scenarios and check whether the software is capable of handling the real time business scenarios.



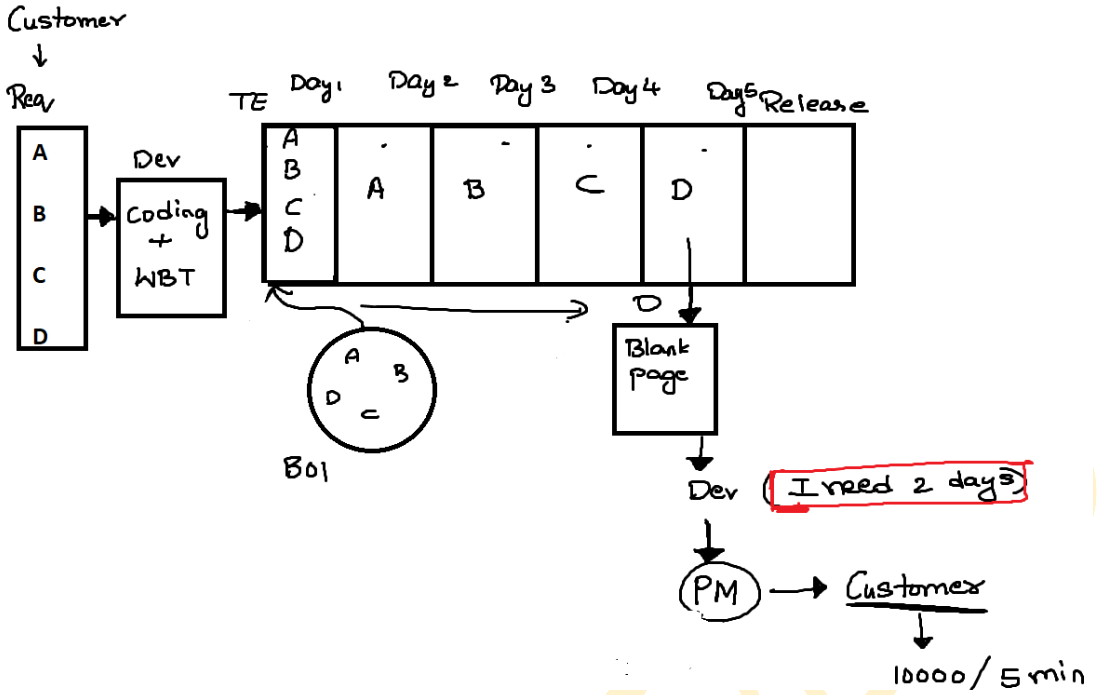
Approach-4

It is an end to end testing done by the test engineer in the company itself where in they refer to the end toe end business scenarios of the customer and check whether the software is capable of handling the real time scenarios or not.



SMOKE TESTING
Sanity Testing/ Build Verification testing

Testing the basic or critical features before doing thorough testing is called as Smoke testing.



Why we do smoke testing?

1) To check whether the product is testable or not. In the beginning itself if we find too many bugs in the product, then it means the product is not eligible for testing. So stop the testing and continue finding more scenarios
2) To check whether the build/ product is installed correctly or not
3) To check if we have received the broken build
4) To complete a health check up of the software ensuring that the build received is correct or not

Note: If these bugs are given to the developers initially itself, then the developers gets sufficient time to fix the bugs, also there will be no delay in delivering the software to the customer

When we do smoke testing?

1) When ever the new build comes, the Test engineer should start with the smoke testing to check all the basic or critical features are working or not
2) Before giving the software to the test engineers, the developers will perform the smoke testing
3) As soon as the release engineers install the build, they will perform the smoke testing to ensure that the build is installed properly or not
4) As soon as the customer receives the software, he will perform smoke testing before acceptance testing

How to do smoke testing?

1) Make a list of all the features in the application
2) Segregate/Divide the features in to two parts
   a) As part of smoke testing
   b) As not a part of smoke testing
3) Pick the important scenarios for the features which are part of smoke testing from the FT/IT/ST scenarios list

Note:

1) We test only the basic or critical features
2) We only perform positive testing (give only valid inputs)
3) We do functionality, integration and system testing or basic features

What is the difference between smoke and sanity testing

Smoke	Sanity
Wide and shallow testing	Deep and narrow testing
We do only positive testing	We do both positive and negative testing
TE, Customer, Developer and Release engineers do smoke testing	Only test engineers do smoke testing
We script the FT, IT and ST scenarios	We do not script any thing

ADHOC TESTING
Monkey Testing/ GorillaTesting

Testing the application randomly without looking in to the requirement is called as adhoc testing

When we do Adhoc testing?

1) When the product is functionally stable then we should do adhoc testing
2) When we are doing smoke testing, we should not do adhoc testing. If we do, we will not be able to test the basic or critical features
3) When ever testing team is free, TE should spend time in doing adhoc testing
4) While the TE is doing FT/IT/ST and if he comes across any adhoc scenarios, he can pause the regular testing and perform the adhoc testing

Why we do adhoc testing?

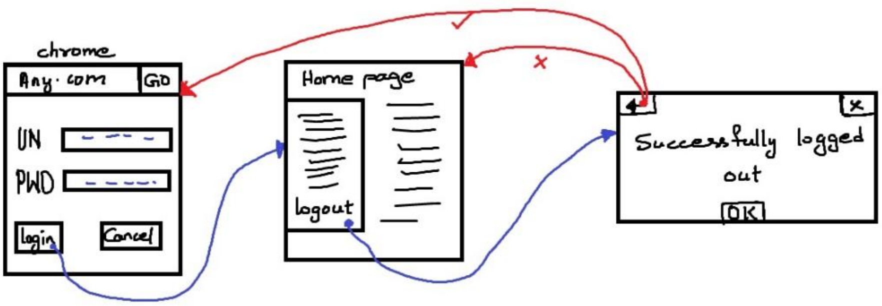
1) When you launch the product to the market, chances are there. End users might use the application randomly, because of which they might get the defects, to avoid that the test engineers should test the application randomly and find all the possible defects
2) If you refer to the requirement and test the application, the number of bugs that you can catch will be less, So think out of the box (out of requirement) and come up with creative scenarios and test the application
3) To some how break the product, we do adhoc testing
4) To some how increase the bug count
5) To improve the test coverage
6) To check whether the product works according to the implicit requirement

Types of Adhoc Testing

1) Pair testing: One developer and one TE together come up with creative scenarios and test the software
2) Buddy Testing: Two TE together come up with creative scenarios and test the software
3) Monkey Testing: Testing the software without applying any logic, randomly

How to do adhoc testing?

1)

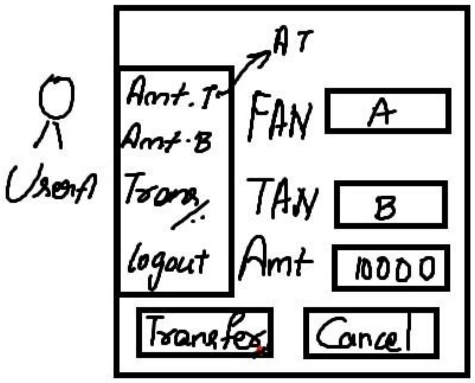


1) Enter the URL
2) ENter the UN and PWD
3) Click on Login button
4) Home page should be displayed
5) Click on Logout
6) Confirmation pop up should appear
7) Click on back button

Expected Results:

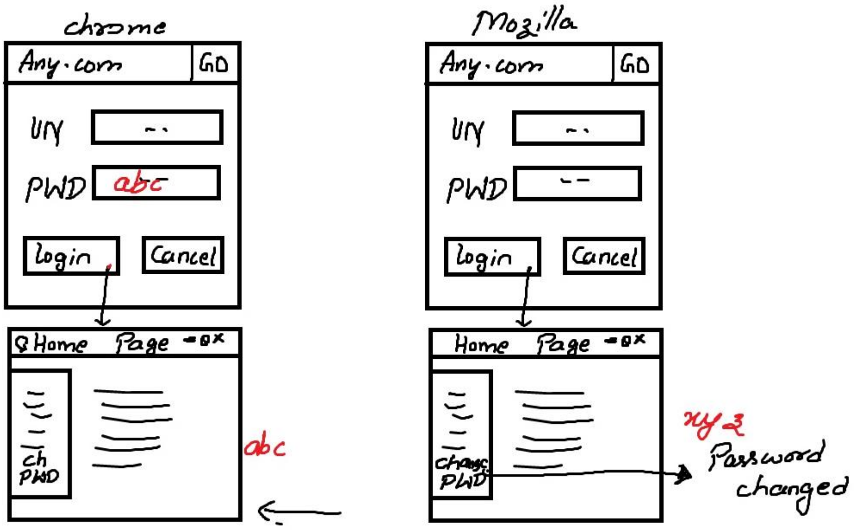
Login page should appear -> Adhoc scenario passed

2)



1) To check that when user clicks on transfer button multiple times, it functions only one time.

3)



1) Open chrome browser
2) Enter the URL
3) ENter the UN and PWD
4) Click on login button
5) Home page should be displayed, minimize
6) Open firefox browser
7) Enter the URL
8) ENter the UN and PWD
9) Click on login button
10) Home page should be displayed
11) Click on change password
12) Change the password
13) Now open chrome browser and refresh

Result:

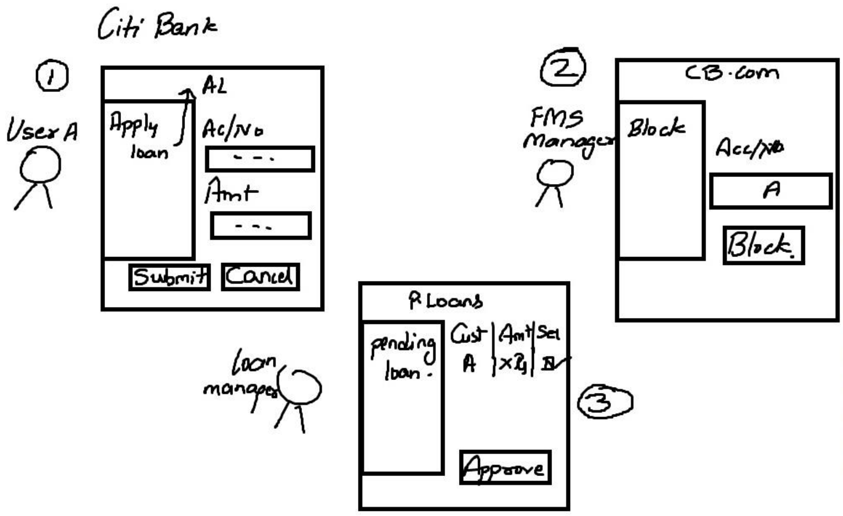
Login page is displayed - Pass

Home page is displayed - Fail

1) Open chrome browser
2) Enter the URL
3) enter the UN and old password

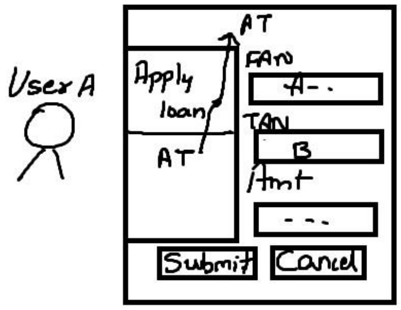
Expected Result: The login should not be successful, error message should appear

4)



1) Login as User A
2) Click on Apply loan, enter the details and click on Submit
3) Login as FMS (Financial Management System) Manager
4) Click on Block, block user A
5) Login as Loan manager
6) CLick on Pending loans
7) Select user A and click on approve
Expected Result: Loan should not be approved and error message should appear.

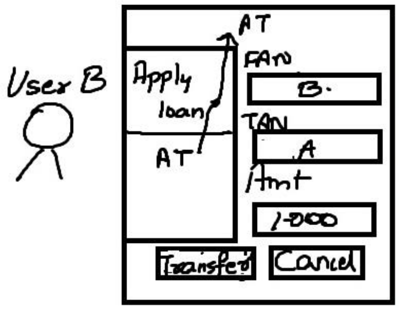
5)



1) Login as User A
2) Click on Amount transfer
3) Login a FMS Manager
4) BLock User A
5) Enter FAN, TAN, and amount
6) Click on transfer
Expected Result: The amount should not be transferred

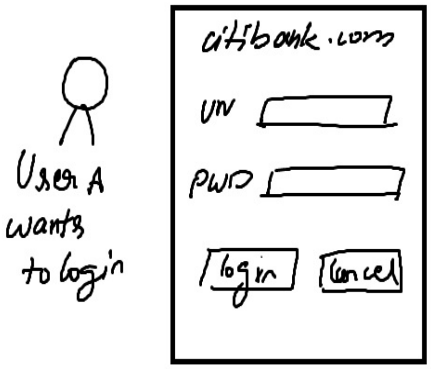
Note: The user A and FMS manager should simultaneously login.

6)



1) Login as User B
2) Click on Amount transfer
3) Login a FMS Manager
4) BLock User A
5) Enter FAN, TAN, and amount
6) Click on transfer
Expected Result: The amount should not be transferred

7)



1) Login a FMS Manager
2) BLock User A
3) Login as User A

Expected Result: User A should not be able to login

PERFORMANCE TESTING

Testing the stability and response time of an application by applying load is called performance testing

Stability:

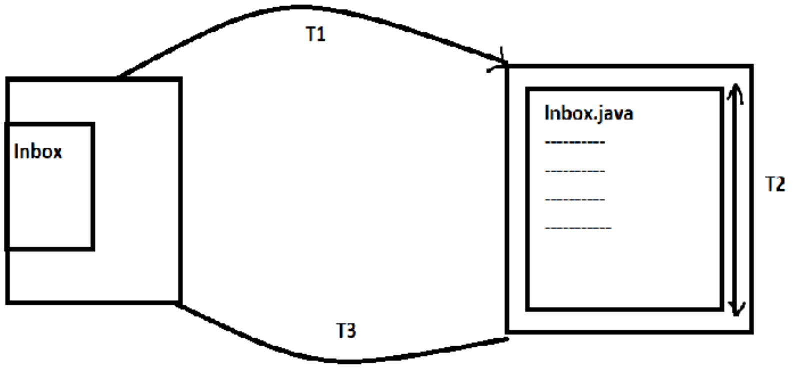
Ability of an application to withstand load

Load:

Defined number of users

Response Time:

It is the time taken to receive request, execute the program an send the response



Response Time = T1+ T2+T3

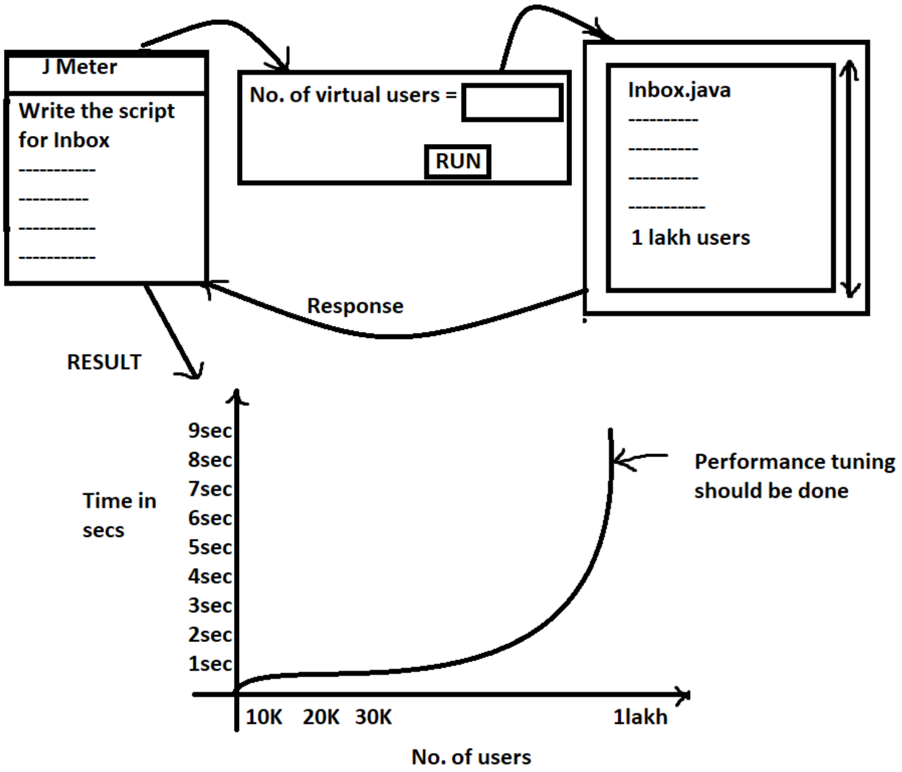
How to do performance testing using performance testing tools?

Tools available for performance testing

1) J meter
2) Noe Load
3) Load Runner
4) Rational Performance tool
5) Silk Performance tool

JMeter Example

a) Take performance testing tool (J-meter), write the scripts
b) Click on run, it will ask you how many users
c) 1lakh requests are fired to the server, now there is a heavy load on the server
d) Server sends the response to the performance testing tool
e) To analyze the response and gives the result in the form of graph
f) Here TE should analyze the result and decide whether test is pass or fail
g) If it is failed, then send it to the developers, they will alter/change the code to improve the performance that is they will do performance tuning
h) Once again TE should run the test scripts
i) Repeat all the steps



Types of Performance testing

There are four types

1) Load Testing
2) Stress Testing
3) Volum Testing
4) Soak Testing

Load Testing

Testing the stability and response time of an application by applying the load less than or equal to of designed number of users

Stress Testing

Testing the stability and response time of an application by applying load more than the designed number of users

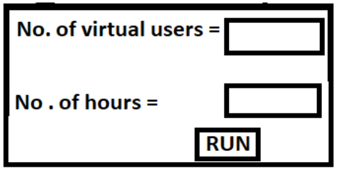
We can check when the application crashes at this point

Volume Testing

Testing the application and response time of an application by transferring huge volume of data

Soak Testing

Testing the stability an response time of an application by applying load continuously for a particular period of time



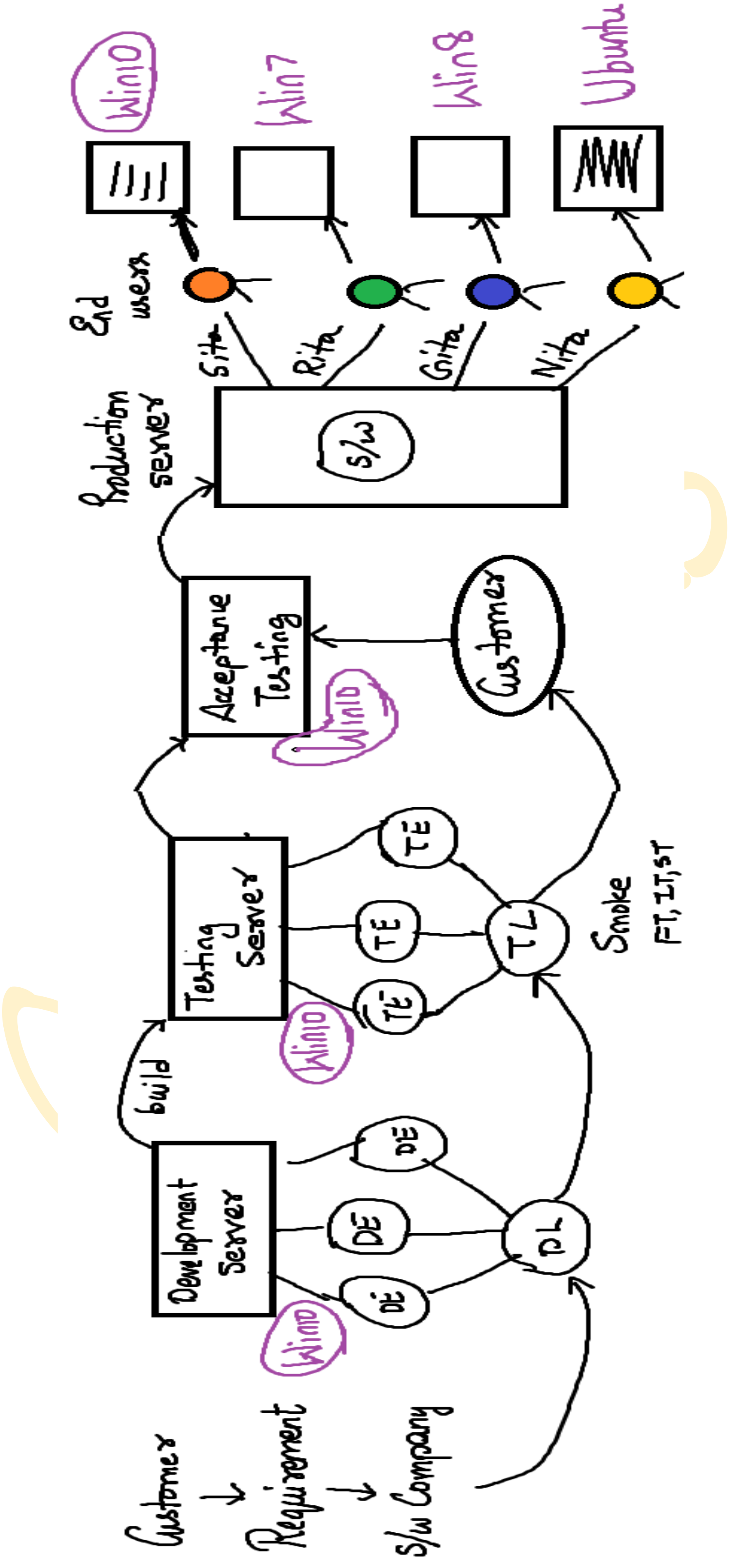
For what kind of applications we should do performance testing?

1) Any application which generates lot of revenue
2) Any application which is used by multiple users

When we do performance testing?

1) When the product becomes functionally stable for some projects
2) From the beginning of SDLC for some projects

COMPATIBILITY TESTING



What is compatibility testing?

Testing the functionality of an application in different hardware and software configuration /platform/ environment is called as compatibility testing

Why we should do compatibility testing?

1. Chances are there developer would have developed the application in one platform and test engineers would have tested application in the same platform, when the product is launched end users might use the application in different platform

Software which works in one platform may not work in other platforms because of which it causes bugs and number of users who use the product is reduced

2. Different hardware and software renders GUI in different ways, we should check whether our application is rendered properly for different combinations

Mother board -> VGA cards -> OS -> Browsers -> Version of browsers

3. Developers would have written common code and claims that it works in all platforms or he would have platform specific code and say it works on some respective platforms. So we have to test in every platform

When to do compatibility testing?

1. Once after the software is tested in the base platform, then only TE will test the software in different platforms by doing compatibility testing
2. Once they are functionally stable

How do they choose the base platform?

1. Based on market research, we can know about the base platform
2. Based on ROI (Return on Investment)
3. Based on company and customer

What is base platform?

1. The base platform for service based companies differ with respect to market.
2. The product based companies have their own platforms Ex: Apple- iPhone

How to do compatibility testing?

1. Web application

- a) Testing the application in different browsers

  Ex: Chrome, firefox, microsoft edge, opera

2. Client Server Application

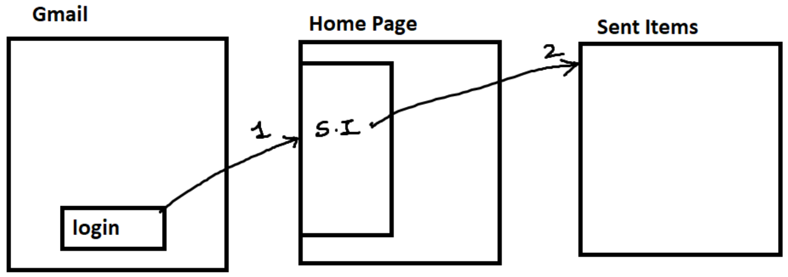
- a) Buy real devices
- b) Rent real devices
- c) Browser stack
- d) Virtualization
  - i. Desktop : VM Ware
  - ii. Mobile:
    - 1. Android : Emulators
    - 2. iOS : Simulators
- e) Crowd testing

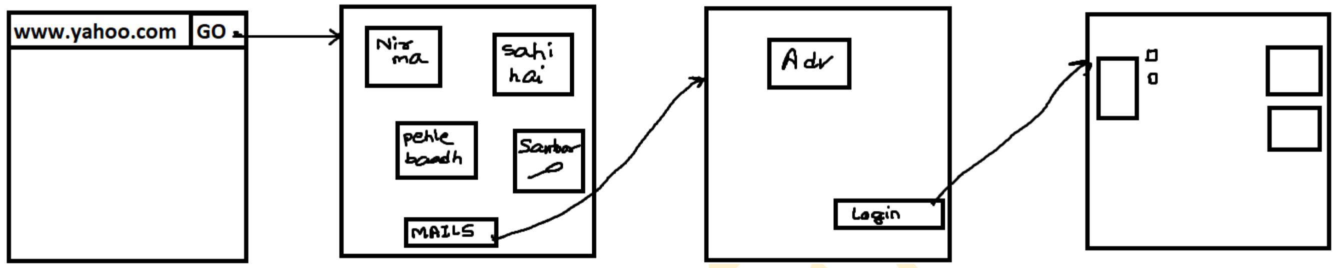
3. Desktop Application (Stand Alone Applications)

Approach to do compatibility testing is same as Client Server applications

USABILITY TESTING

Testing the user friendliness of an application is called as usability testing





We don't do usability testing for all the applications. For what kind of applications we don't perform usability testing

1. Software used for official purposes are bothered only about the functionality
2. Softwares for which training or step by step manual is available

For what kind of applications we do usability testing?

1. Applications which are used by variety of users
2. Applications which generate lots of revenue
3. Applications where we do not provide training or step by step manual
4. Features used by many number of users many times to take less time

How to do usability testing?

1. We will check the look and feel of the software or application is good or not
2. We will check whether the application/ software can be easily understood or not
3. We will check whether the application can be understood and used within less time or not
4. We will check whether the most important features of an application are easily accessible or not

When to do usability testing

1. We start from the beginning for some projects
2. Once the software is functionally stable

Accessibility

(American Disability Act-(508) Testing) Testing the user friendliness of an application or software from the physically challenged person point of view is called Accessibility testing
Ex: Here we check whether red and green coloured objects are present or not

Reliability

Testing the functionality of an application continuously for a particular period of time.
Any application which is used continuously for longer hours
Ex: Autocadd, whatsapp, paint, word pad, instagram

What is the difference between Reliability testing and soak testing?

Soak Testing	Reliability testing
Testing the stability and response time of an application for a particular period of time	Testing the functionality of an application for a particular period of time
We apply load for testing	We do not apply load for testing
Tested using performance tools like J meter, Neo load, Load runner…	Tested using automation tools like Selenium, QTP…

Comparison Testing/ Parallel Testing

Here we compare the newly built application with similar kind of application which is release in the market and we check the weakness, advantages, disadvantages and we make sure that all the features are present in the newly built application
Ex: Amazon with flip kart, Ola with Uber

Yellow Box Testing

Testing the warning messages of an application is called Yellow Box Testing. It is the subset of usability testing.

Recovery testing

Testing how soon the application gets recovered after a crash is called as recovery testing

EXPLORATORY TESTING

What is exploratory testing?

Understand the application, identify all the possible scenarios,document the scenarios and test the application by referring the document is called exploratory testing

OR

Explore the application,understand how each and every feature work and test the application as per your understanding is called as exploratory testing

When to do exploratory testing?

1) When there are no requirements or the requirements are missing, we go for exploratory testing
    a) Why the requirements go missing ?
    -> Big and old projects, requirements go missing for some features
    -> Product based companies where the CEO gives on sot requirement and will be developed
2) When the requirements are there but there is no time to go through

How to do Exploratory testing?

Understand the application ad explore how each and every feature works, and then identify the scenarios and then perform testing

Why we cannot do exploratory testing all the time?

1) Chances are there that the feature might be misunderstood as defect
2) Chances are there that the defect might be misunderstood as feature
3) Chances are there we may never get to know that the feature is really missing
4) It is more time consuming to explore the feature and test than directly looking in to requirement and test

How to overcome the drawbacks?

1) Interact with the Sr. Dev, BA, Product Manager, customer
2) Based on domain knowledge
3) Based on Product knowledge
4) Comparing the application with another similar kind of application
5) Common sense

Do you write exploratory TS and TC?

Yes

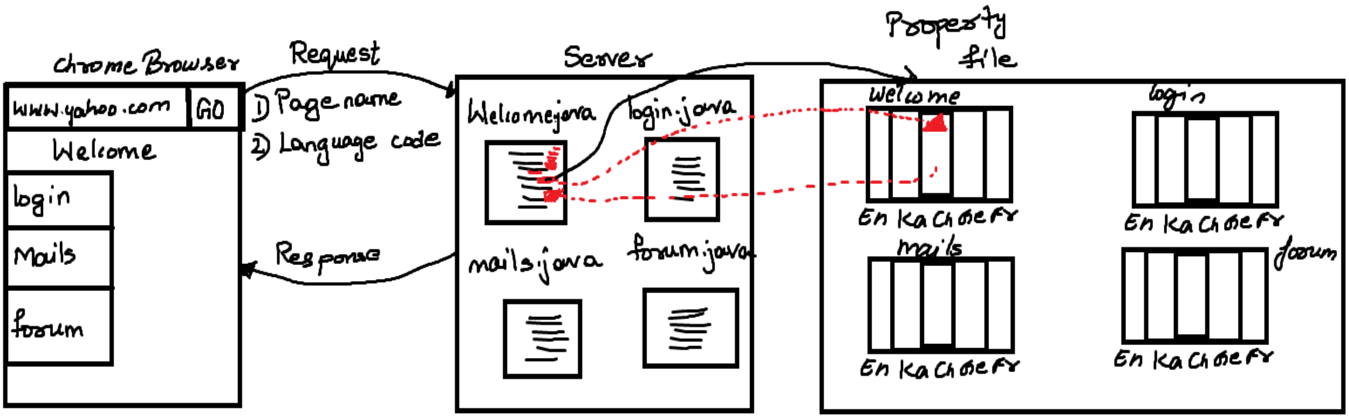
GLOBALIZATION TESTING

Developing the software for multiple languages is called as globalization of the software
Testing the application which is developed for multiple languages is called as globalization testing

Types of Globalization testing

1. Internationalization testing (I18N testing)
2. Localization testing (L10N testing)

How will the developers develop the software for multiple languages?



How to do I18N testing?

1. We will check if the contents are displayed in the right language or not
2. We will check if the right content is displayed in the right place or not
3. We will check whether the features are broken due to language change

The developers would add the suffix and prefix to the translated content to help the TE to test software for multiple languages

Localization testing

We will check if the content is displayed according to the country standards or not
Ex: Date, Time, Pincode, Currency, Image formats