

Testing Tools

Date	11
Page	10 Student Notebooks

Contents:

Manual Testing

Quality Center

FAQ's for both manual & Quality Testing

QTP

FAQ's in QTP & CV Prep^n

WR

Load Runner

Selenium

SW → A group of statements, set of logic & related data that gives instructions to system what to do & flows to do

→ Based on the usage SW has divided into two types

i) Product SW (sys. SW)

ii) Project SW (appl. SW)

Product

- 1) Product SW is developed for segment of customers
- 2) There is no end of development & testing in product becoz product will be keep on releasing into the market versions by versions so that for every version of the product, we require development & testing
- 3) Client like people are performing testing in the product
- 4) examples of product based companies : Microsoft, google, Dell, hp etc.

Project

- 1) Project SW is developed for single client
- 2) There is an end of development & testing in project becoz we are developing the project for single customer so that whenever we satisfied all the customer requirements we need to stop the development & testing
- 3) Directly client will perform testing in the project
- 4) ex's of project based or services based companies : TCS, Accenture, wipro, infy etc.

→ Prod t based company → A company which is mainly focusing on developing their own products is called product based company.

Services based company → A company which is mainly focusing on developing the individual projects to the individual clients is called services based company.

Different teams in product or project

In every project or prod t development we require two different types of teams.

- i) Development team
- ii) Testing team

Development team → The responsibility of the development team has to develop the application as per the client requirements.

Testing team → The responsibility of the testing team has to test the application to verify whether the development team has developed the application as per the client requirements or not.

Differences b/w development & testing

Development

1) It is respective to the technology: That means as a Java developer we can able to develop ^{only} Java applications.

Testing

1) Testing is irrespective of the technology, that means as a tester we can able to test any appln which has developed in any technology.

- | | |
|--|---|
| 2) Technology might be changing from year to year & regularly | 2) Testing never changes. That means the app that has been developed in any technology requires testing |
| 3) More technical skills are required for development | 3) Less technical skills are required for testing. |
| 4) There is no company in the industry is doing only development | 4) There are some companies are in industry that are doing only testing
<u>ex - AP-Labs, Value-Labs, stag s/w etc.</u> |

Testing → Testing is the verification & validation to ensure to deliver the defect free (problem less) application or product into the market.

S/W Testing → performing testing on a sw application or a product is called sw testing

→ S/w testing has divided into types:

- 1) Manual testing
- 2) Automation testing

Manual Testing → Performing testing on s/w application or product with the human interaction is called manual testing.

Automation testing → Performing testing on s/w application or product with the help of some automation tools is called automation testing.

Note - In both manual & automation testing we are performing the same testing but the way we are performing testing is different in both manual & automation.

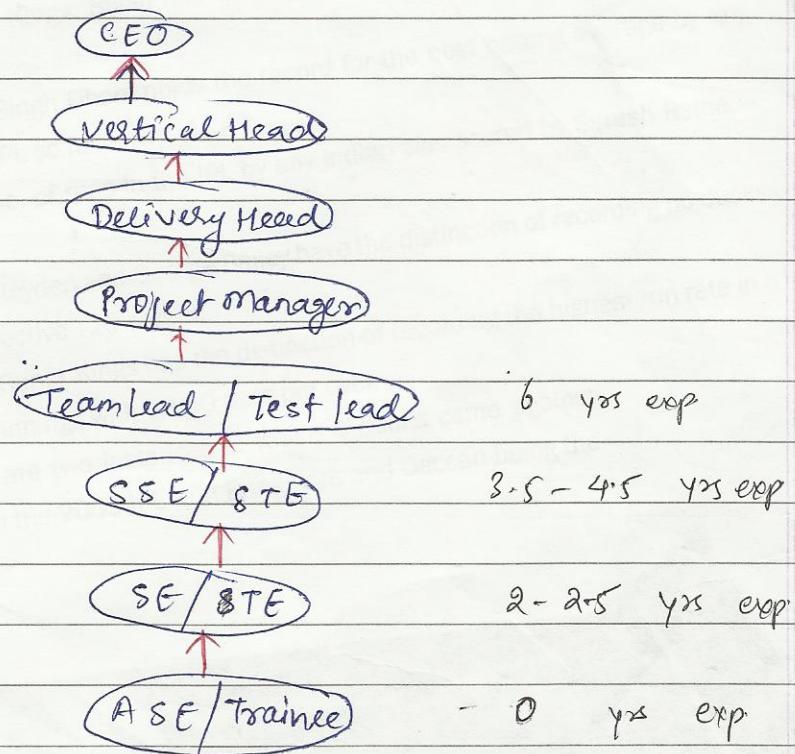
(4)

Client → is the individual or organization who is providing requirements to the company for developing SW.
ex ICICI Bank, Airtel, vodafone, Indian Railways etc.

Company → is the organization who is developing the SW based on the client requirements.
ex TCS, Accenture, Infosys etc

End users → is the person who is using or working on the application finally
ex customer of ICICI bank, customer of Airtel, passenger of Indian Railway etc.

Reporting Hierarchy in the organization → (or) Different designations in the organization →



14/7/12

⑤

Different environment in the projects →

There are 4 different types of environments available in every project

- 1) Development environment
- 2) Testing environment
- 3) UAT (user acceptance testing) environment
- 4) Production environment

Development environment → In this environment development team will be involving to develop the application as per the client requirement. After the completion of development of the application, they will deploy the application into testing environment.

Testing environment → During this environment testing team will be involving to perform the testing on application based on the client requirements. After completion of the testing in testing environment, development team will deploy the application into UAT environment.

UAT environment → During this environment client will be involving to perform testing on the application.

After completion of testing, development team will deploy the application into production environment.

Production environment → Using this environment client or end user will be involving to use the application.

(6)

Environment → It is the location which contains different HW & SW settings.

Note:- In most of the projects, testing team is only performing testing, in UAT environment also.

TESTING TYPES:

1) 'Sanity Testing' (or) 'build acceptance testing' (or)
'build verification testing' → validating the major functionality of the application is known as sanity testing.
once the appln has developed & that has deployed into testing environments we are performing sanity testing. because if the major functionality is working fine then only we will continue further testing on application.

Q: what ?, why ?, when ?, what is no approach ?

Q: what is the approach we are following when we identify defect (problem) during sanity testing.

Ans → If we identify any defect during sanity testing, we need to report that (assign) the defect to immediately to the development team & development team also needs to fix the defect immediately.

Q: what type of defects we are identifying during sanity testing?

- Ans
- 1) Environment issues (appln is not opening)
 - 2) Application crashes (appln is suddenly closing or hanging)
 - 3) Exceptions (java exception error, null pointer exception errors) etc.)

(7)

(8)

2) Usability testing (or) user interface testing (or) graphical user interface (GUI) testing →
 sometimes pronounced as 'guy'

verifying the user friendliness of the application in terms of colours, fonts, logos, images & alignments of the pages etc. is called usability testing.

3) Functionality testing → Validating the overall functionality of appl'n with respect to client business requirements & business requirements is called functionality testing.

ex- Conduct sanity testing, usability testing & functionality testing for the two wheeler

<u>Sanity</u>	<u>usability</u>	<u>functionality</u>
Starting	colour	start
Breaks	logo	Break
Clutch	Comp. name	clutch
Smoothly running	Comfortness	accelerator
		Headlights
		Horn
		AC in car
		Audio set in car
		wiper

Note:- Sanity testing is the part of functionality testing becoz during sanity testing we are validating the major functionality & during functionality testing we are validating the overall functionality that includes the major functionality.

(8)

- 4) Database testing → Validating the backend database w.r.t front end application is called database testing.
During perfo database testing we are performing 3 diff. types of validations.
- i) whatever we are performing in the frontend applⁿ that should reflect in the backend database.
for ex → If we added one user at front end applⁿ, that user details should be adding successfully into backend database.
 - ii) whatever we are performing in the backend database, that should reflect successfully in the ^{frontend} applⁿ.
for ex if we deleted one user at back end database, that user details should be deleted successfully into frontend applⁿ.
 - iii) Verifying whether the applⁿ is successfully fetching or retrieving the data from database or not.

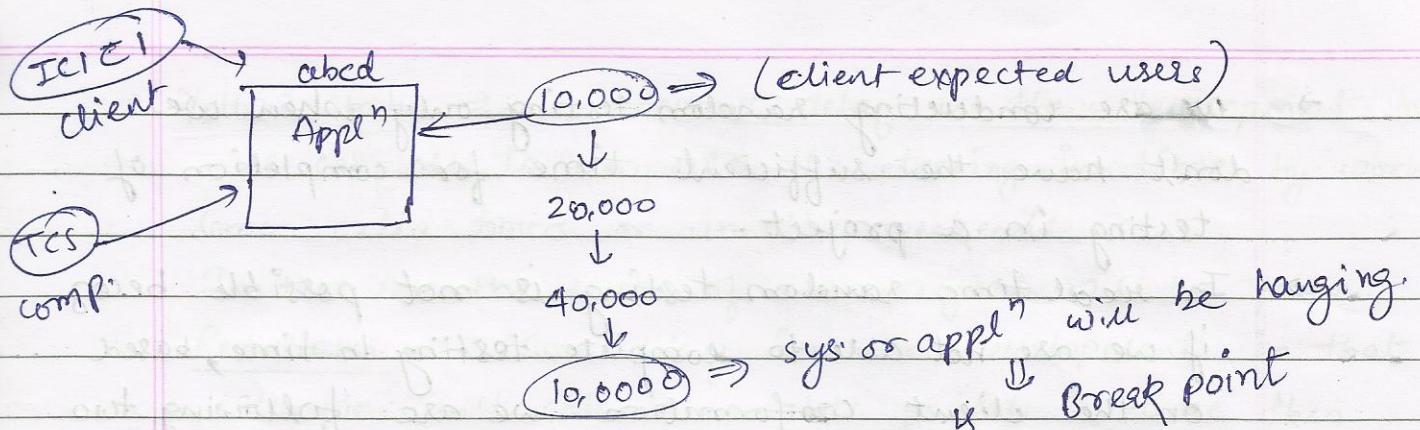
Note:- In most of the projects we are maintaining separate team called database testing team to perform the database testing.

5) Performance testing → Validating the performance of the applⁿ w.r.t load & stress is called performance testing.

6) Load testing → Validating the performance of the applⁿ with client expected users is called load testing.

Stress testing → Validating the performance of applⁿ with Ned no. of client expected users upto the max. level & identifies the break point of the applⁿ is called stress testing.

(6)



Ex - As per the client requirement 10,000 users will be login to the 'abcd' application at any time. After the completion of development of abcd appl^n, we need to validate the performance with 10,000 users. This is called as Load testing.

→ After completion of load testing we need to validate the performance by giving the client expected users from 10,000 to 20,000, 20,000 to 40,000, 40,000 to 80,000 etc. upto the max. level. At one specific ^{pt of} time appl^n will be hanging this is called as stress testing.

At the time of releasing the appl^n to client we need to provide the break pt. info to the client.

Note :- In most of the projects we are maintaining the separate team called performance testing team & we are performing the performance testing using some performance tools like load runner, sick performer etc.

(6) Random testing → (or) Monkey testing -

performing testing on the appl^n (without having sequence) randomly is called random testing.

200 functionality	1, 7, 9, 16, 25 189, 200
-------------------	---

checking only
20 funct.

- We are conducting random testing only when we don't have the sufficient time for completion of testing in a project.
- In real time random testing is not possible becoz if we are not able to complete testing in time, based on the client conformation we are following two approaches
 - i) Partial Release
 - ii) Postponed Release

Partial Release → In partial release we are releasing the completed modules to client in current release & rest of the incomplete modules we are combining with the next release.

Postpone Release → In this approach client will postpone the release date from current date to future date. By that time we have to complete the testing releasing app to the client.

Different status of the project →

There are 3 types of status available in the project

- i) Green
- ii) Yellow
- iii) Red

Green status → If everything is going fine in the project that means development team has completed development in time & testing team is going to complete testing in time. In that case status of project will be green.

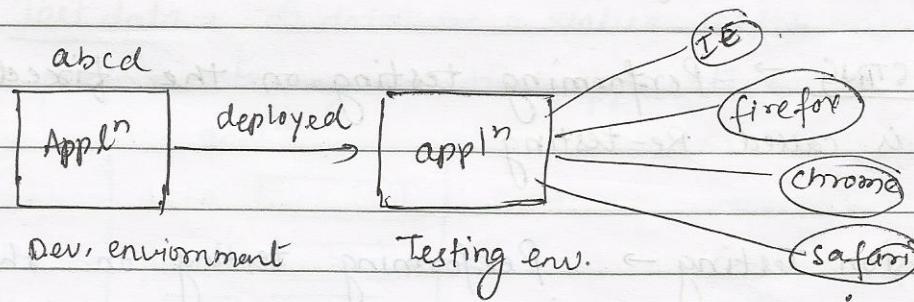
Yellow status → somewhat delay in the development but we can able to complete the testing in time by working some extra hours or in the weekends.

In that case the status of the project will be yellow.

Red status → More delay in the development in that case we can't able to complete project in time then status of the project will be red.

Note :- we are choosing partial release or postpone release only when the project status is red.

* ⑦ COMPATIBILITY TESTING (OR) BROWSER COMPATIBILITY TESTING (OR) CROSS BROWSER TESTING →



Performing testing on the application in diff. browsers like internet explorer, Mozilla firefox, Google chrome, Safari, Opera etc. is called compatibility testing.

→ for ex as per the client requirement abcd appⁿ has to work fine in every browser. After the completion of development of abcd application, we need to open that appⁿ in IE & perform testing, open appⁿ in firefox & performing testing. This is called as compatibility testing.

(12)

(11)

(8) CONFIGURATION TESTING →

Performing testing on the appl'n in diff. configurations of the system like RAM, Hardisk, OS, processor etc. is called Configuration testing.

- for ex as per the client requirement xyz application should work fine in every configuration of the system.
- After completion of development of xyz application we need to open the application in high config'n med. config'n & low config'n systems & performing testing. This is called as configuration testing.

Note:- Compatibility testing is suitable for projects or applications & configuration testing is suitable for proj's

⑨

RE-TESTING → Performing testing on the fixed defect is called Re-testing

18/7/12
⑩

Regression Testing → Performing testing on the modified application is called regression testing.

- During performing testing if we identify any defect we are reporting the defect to the development team. After the development team has fixed the defect we need to verify whether the fixed defect is working properly or not.

This is called as Re-testing.

Once the fixed defect is working fine we need to think, for fixing this defect developer might have changed the related functionality of the defect in the application or fixed defect might have automatically impacting on other related functionalities in the application. That is the reason we are performing

testing on the related functionality of the defect in the application. This is called as Regression testing.

(11) SMOKE TESTING → Validating the major functionality of the application by the development team before deployment of appln in the testing environment is called smoke testing.

Note:- Defnwise only we have a difference b/w sanity & smoke testing but practically both are same. only testing team is validating the major functionality in testing environment. some tester are calling it as sanity testing & some of them are calling it as a smoke testing.

Test data → A data or a value which we are using to perform testing on the application is called test data

19/7/12 ex

Requirement of user	appn	
	userid	
	password	
	OK	Cancel

- 1) OK button should be enabled b/w when user enters userid
- 2) Cancel button should be enabled when user enter password.
- In the above example we will verify whether the OK button or Cancel button are enabled or not only after entering userid & password values so that without entering userid & password values we can't value be able to test OK & cancel buttons.
- How we are getting the test data?
We are getting the test data in a project in three different ways
- i) In some projects directly client is providing the test data

- (ii) In some projects we (testing team) are creating the test data to perform testing on the application
- (iii) In some projects one dedicated team (separate team) called TDM (Test Data management) is creating the test data for all the teams in a project

(12) POSITIVE TESTING → Performing testing on the application with +ve or valid data is called +ve testing.

(13) NEGATIVE TESTING → Performing testing on the application with -ve or invalid data is called -ve testing.

ex Validating the login functionality with valid userid & valid password is called +ve testing.

Validating the login functionality with invalid userid & invalid password, valid userid & invalid pwd, invalid userid & valid pwd, etc. is called -ve testing.

(14) INTERNATIONALIZATION (I18N) TESTING → (OR) GLOBALIZATION TESTING (OR) I18N TESTING

Performing testing on the application in different international languages like French, Spanish, Chinese etc is called Internationalization testing.

(15) Localization testing or L10N testing →
Performing testing on the application in diff. local languages like Hindi, Telugu, Tamil etc is called L10N testing.

Note:- for performing both internationalization & localization testings we should know the specific language.

(16) STATIC TESTING → Performing testing on the application without performing any activity or action is called static testing.

(17) → DYNAMIC TESTING → Performing testing on the application by performing some activity or action is called dynamic testing.

uid	<input type="text" value="nn"/>
pwd	<input type="password"/>
<input type="button" value="signin"/>	

- verifying userid, password, signin buttons are available or not in the application. we need not to perform any activity or action in the application. This is called as static testing.
- for verifying signin button is enabled or not we should enter the userid so that for performing testing we are performing some activity or action on application. This is called as dynamic testing.

(18) SECURITY TESTING →

Validating the security of the application w.r.t authentication & authorization

Authentication → verifying whether the applⁿ is accepting the valid user & rejecting the invalid user or not. This is called as authentication.

Authorization → verifying whether the applⁿ is providing right permissions to the right user or not. This is called as authorization.

For ex - as per the client requirement both valid user & valid admin can able to login to gmail

application. This is called as authentication.

- once they login to the gmail application we need to verify only admin can able to connect to the database. This is called as authorization.

2017/12

(19)

EXPLORATORY TESTING →

- Performing testing on the appl" by learning the domain knowledge is called Exploratory testing.
- We are performing exploratory testing when the resource (employee) is shifting from one domain to another domain.
- we can learn the domain knowledge in the project in three different ways.
 - i) By writing the domain certifications
 - ii) By attending the training programs
 - iii) By getting KT (Knowledge transfer) directly from the client

(20)

α - TESTING → Performing testing on the application directly by the client is called α - testing..

(21)

β - TESTING → Performing testing on the appl" or prod by the client like people is called β - Testing.

Note:-

α - Testing is suitable for projects & β - testing is suitable for prod products.

(22) SANITATION TESTING → This is the experienced based testing. Based on the experience tester will provide some suggestions to the client about the requirements.

Based on the conformation for the client we will continue testing in the project. This is called as sanitation testing.

Note:- This testing is not practically possible in real time because only the client is the final decision maker in the project.

(23) MUTATION TESTING → Performing testing on the applⁿ after the development team has changed the source code intentionally is called mutation testing.

We are performing mutation testing only if we have the sufficient time or extra time for completion of testing in a project.

During mutation testing development team is changing source code either +vely or -vely, if they change the source code +vely we need to verify application should respond +vely & if they change the source code in -ve manner applⁿ should respond negatively.

→ The main intention of conducting mutation testing is to verify whether the applⁿ is responding properly or not as per the changes made in source code.

(24) DEFECT SEEDED → Performing testing on the application after the development team has injected the existing defect into source code is called defect seeding

During mutation testing development team will change the source code either +vely or -vely. But during defect seeding development team will change the source code only -vely.

(25)

PARALLEL TESTING → Performing testing on the application by comparing with similar types of applications in market is called parallel testing.

→ We are conducting parallel testing if we don't have the sufficient requirements to perform testing in the application.

ex Performing testing on the airtel application by comparing with vodafone appln is called parallel testing.

(26)

RECOVERY TESTING → During recovery testing we are verifying how much time the application is taking to comeback from abnormal state to normal state (down status to up status) is called recovery testing.

→ Due to some limitations in the sw, during performing testing appln will be down for some & it will be up automatically after some time. In that case we need to calculate how much time will the appln will be down & that info we need to provide to the client at the time of releasing the application to the client.

(27)

(19)

(20)

INSTALLATION TESTING → During this testing development team will be involving to verify whether the appl'n is properly installing ^{or} deployed into other environments or not.

(28)

ACCESSIBILITY TESTING → During this testing we are verifying whether the application has developed as per the Web standards.

This testing is suitable for only web based applications.

23/7/22

SDLC S/W Development Life Cycle (or sys. development life cycle)

SDLC is the process which we are following to complete the S/W project or product that include both development & testing.

Different phases in SDLC →

- 1) Requirements
- 2) Analysis
- 3) Design
- 4) Coding
- 5) Testing
- 6) Release or Maintenance

1)

Requirements → This is the first phase in SDLC. Once the project has been confirmed b/w client & company, client will directly provide requirements to the company or (Business) analyst. BA team from the company will collect the requirements from the client. whenever the requirement client has provided, those requirements are not in a understandable format becoz always clients will provide the requirements

business prospective

for collecting the requirements from the client, BA team is following some techniques:

1. Questionnaires → In this technique BA team will collect the requirements from the client by asking the questions.
2. KT (Knowledge transfer) → In this technique BA will collect the requirements by getting the KT about the business from the client.
3. Walkthroughs → In this technique BA team will collect the requirements by going through the documentation part which has provided by the client.
4. Inspection → In this technique BA team will directly step in to the client business & observes how client is doing the business manually & collecting the requirement.

Analysis phase → During this phase BA team will be involving to analyse the client requirements & they will design the understandable format of requirement documents called BRS (Business requirement specifications)

document or usecase document & FRS (functional requirement specification) document or Business rules documents & SRS (SIW requirement or sys. req. specification) document.

BRS document → (or) usecase document →

BRS document is ~~form~~ in the format of word & one

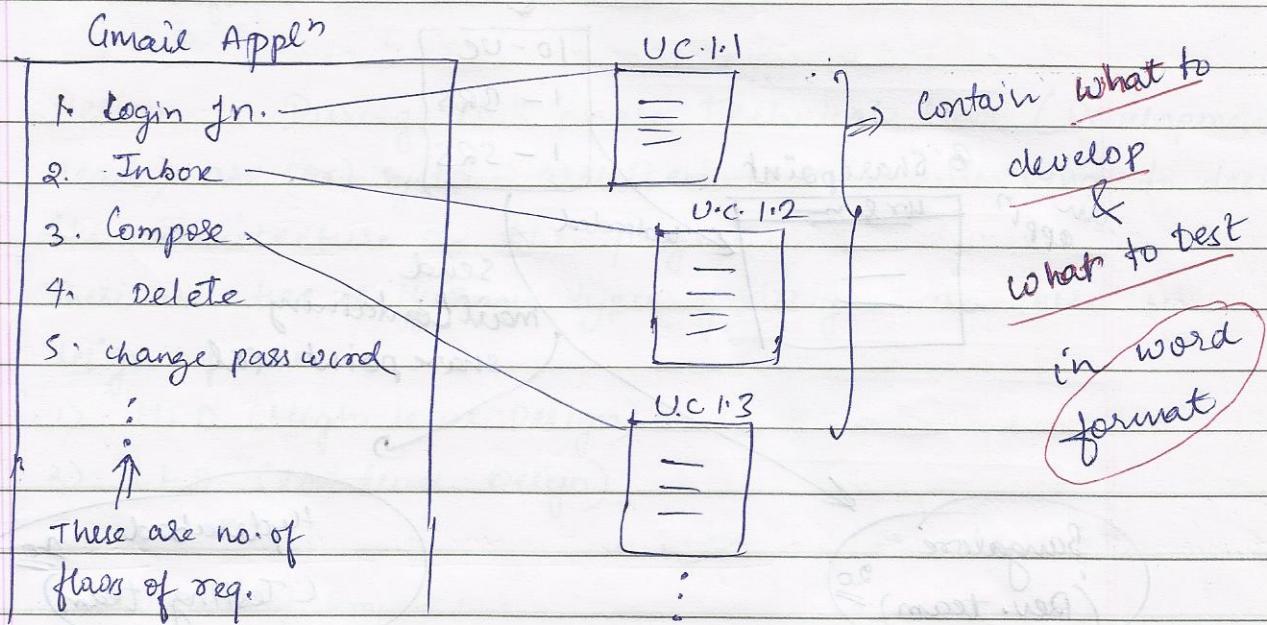
BRS document contains 1 flow of the requirement.

→ BRS doc. defines what to develop & what to test

(21)

in an application

ex



→ No. of use cases in a project is depending on the no. of flows of requirements.

If we have 20 flows of requirements in a project, BA team will design 20 use case documents

FRS or Business rule document → B

FRS or Business rule document defines how to develop & how to test the appn.

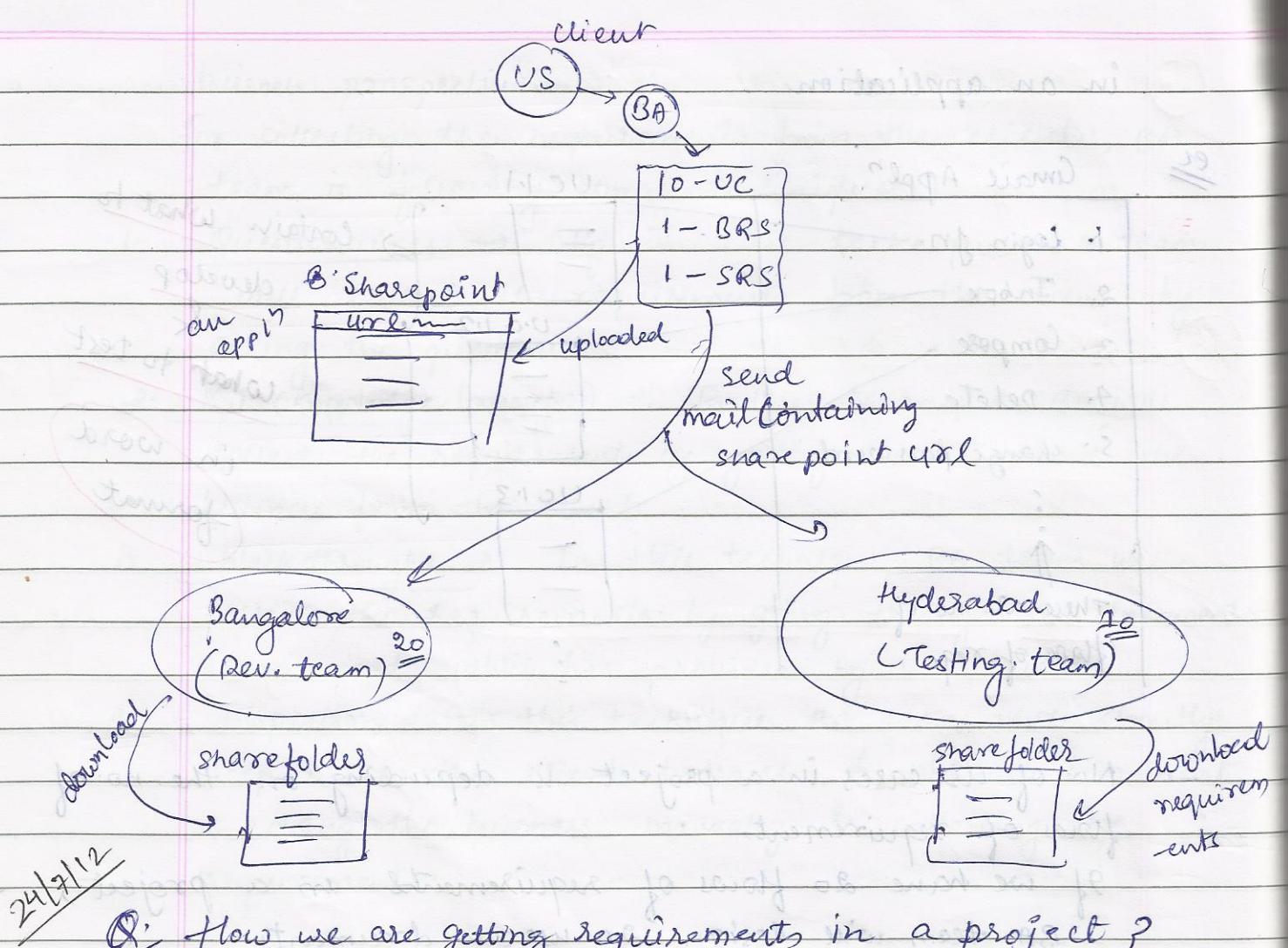
This doc. is in the format of Excel & we are maintaining only one business rules document for entire project.

SRS → This document defines related SW's, HW's, databases which we required to develop the application or test the application in a project.

This formal doc. is also in the format of word & we are maintaining only one document in a project

(22)

(15)



Q: How we are getting requirements in a project?
Ans: we are getting requirements in a project in the form of usecase documents, business rules documents & SRS documents through sharepoint

- Once the BA team has designed three documents, they will upload these documents in sharepoint & send the confirmation mail to both development team & testing team alongwith the location of sharepoint
- once we receive the mail from BA team, test lead will login to the sharepoint and download the documents & keep it in shared folders.

Sharepoint → is the application which we are using to share the infoⁿ across the globe.

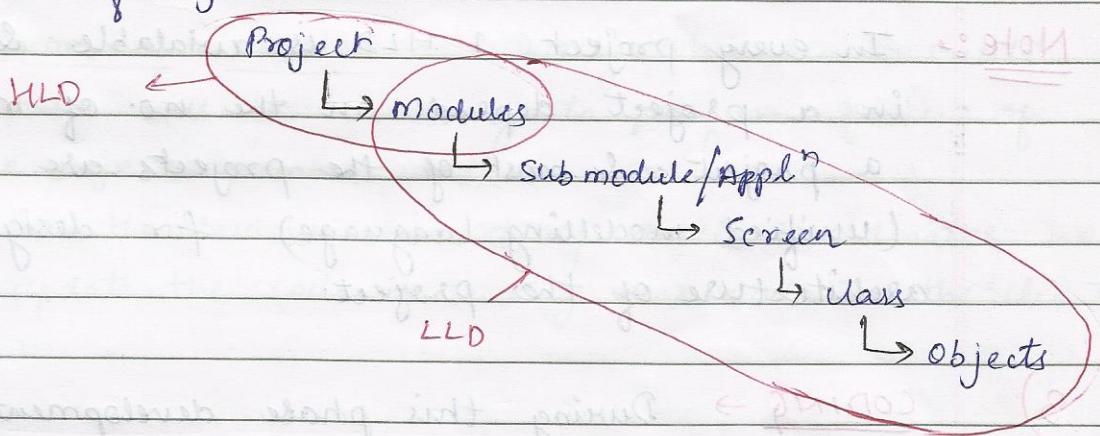
Shared folder → is the folder which we are using to share the infoⁿ within the location.

2) Design → During this phase technical lead (Development lead) and (or) system architect will be involving to design the architecture of the project.

There are two different types of design are there in design phase

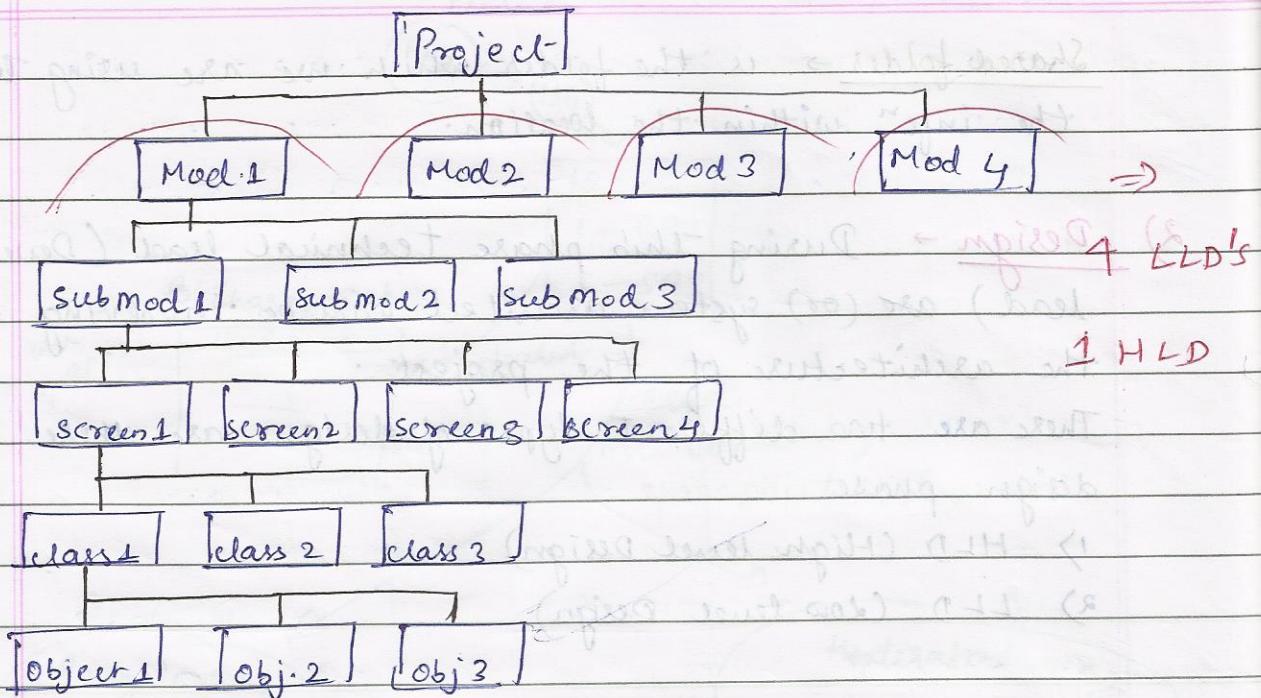
- 1) HLD (High level Design)
- 2) LLD (Low level Design)

Structure of Project:



HLD → defines the overall architecture of the project that includes all the modules

LLD → defines the overall architecture of individual module that includes all the submodules & screens.



Note:- In every project 1 HLD is available & no. of LLD's in a project depends on the no. of modules in a project & most of the projects are using UML (unified modelling language) for designing the architecture of the project.

- 3) CODING → During this phase development team will be involving to write the coding for the functionality of the individual modules
 → After successfully completion of the coding development team will be performing reviews.

- REVIEW → is the process to verify the completeness & correctness of task. As per the CMM (capability maturity model) level 5 standards we have to perform 4 different types of reviews in a project.
 1. Peer review
 2. Lead review
 3. Onsite review Co-ordinator review
 4. Client Review

1. Pear Review → This is the first review in the review process. In this review we will interchange the code among the team members & performs the review. Based on the review comments of reviewer we will update the code & sending that code to the lead review.
2. Lead Review → During this review team lead will be performing review of all the team members task. Based on the review comments of reviewer we will update the code & sending the code to onsite co-ordinator review.
3. Onsite co-ordinator review → During this review onsite co-ordinator will be involving to perform review of all the projects members task. Based on the review comments of onsite co-ordinator we will update the code. & sending that code to the client review.
4. Client review → During this review directly with client or client side technical team will be involving to perform review of all the projects members task. After the successful completion of client review client will sign-off the coding phase.
- 4) TESTING → During this phase testing team will be involving to perform testing on the appl' by executing the test cases. Testing team will perform diff. types of testing like sanity testing, usability testing, functionality testing, security testing etc.