

SOFTWARE TESTING.

LECTURE - 1

Page No.:

Date: 24/6/19

Yours

1> What is software. ?

Software is collection of specialised programs which takes user input & generate desire o/p.

2> What is Testing. ?

Testing is a process of checking whether the given sw or appln is generating desired o/p.

3> What is software Testing. ?

Process of checking completeness and correctness of software with respect to the client's expect.

4> Resources involve in the software Development.

customer



Business Analyst

(Gather Requirement)



Developer

(Develop application as per requirement)



Tester

(check correctness & completeness of functionality)



Final Product to customer.

→ customer want some product.

→ B.A. gather all the requirements from customer.

→ B.A. then make documents & send it to the developer.

→ Developer then develop the appln as per client's req.

→ Then developed product send to the tester.

→ Tester then test application positively & -vely.

→ Positive testing means suppose there is a Name field. Tester test the field by entering credentials. Name : PRASAD.

This is +ve test by doing it in correct manner.

→ Negative testing is done by Entering wrong credentials in the field. For Name field tester enter Numbers. This Number should Not get accept. This is -ve testing.

5) Software Quality Assurance. (SQA)

→ communication between customer and business analyst is called software quality assurance

→ SQA is done to monitor & measure slw development factors.

→ Factors involve in the SQA.

① To meet the customer requirement.

↳ which type of application customer want

i.e., Banking Domain

Telecom Domain.

↳ For which purpose customer want slw.

② To meet the customer Expectation

↳ Privacy : Privacy include security. Any slw like Banking, gather customer's data which is very sensitive, so client want privacy to all customer's data.

↳ Performance: slw should balance load properly it should have to work properly under heavy load.

③ Costing of Project

↳ Project costing for mNC's are per hour cost. Customer have to pay it. This payment depend upon resource utilization as well as time to complete for project.

④ Timing Delivery

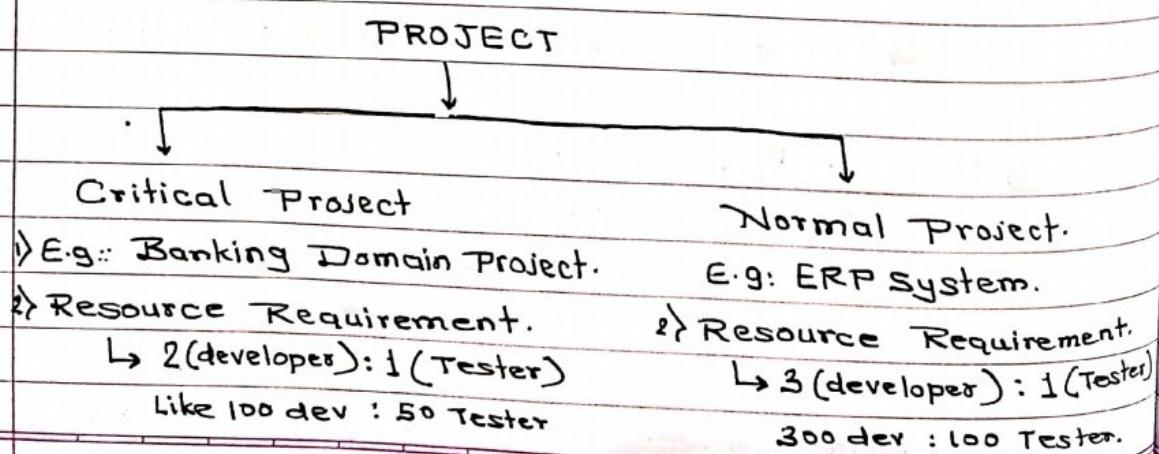
↳ At the time of Resource gathering & documentation time to complete for a project get decided.
 ↳ If company exceeds the delivery time then company have to pay penalty, that penalty is called Escalation.

⑤ maintenance

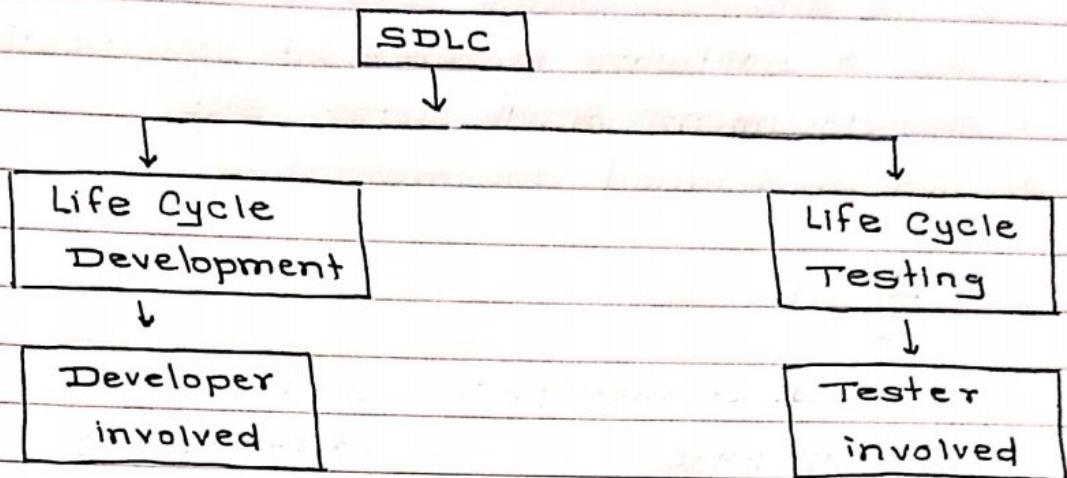
↳ maintenance is the part of service provided by the company after delivery of project.
 ↳ If any problem occur after delivering the proj. then company have to fix it.

6) Projects.

Projects have two categories.



- 7) Software Development life Cycle. (SDLC)
 → SDLC has two types.



→ Stages in SDLC.

① Information Gathering



② Analysis



③ Design



④ Coding (Development)



⑤ Testing



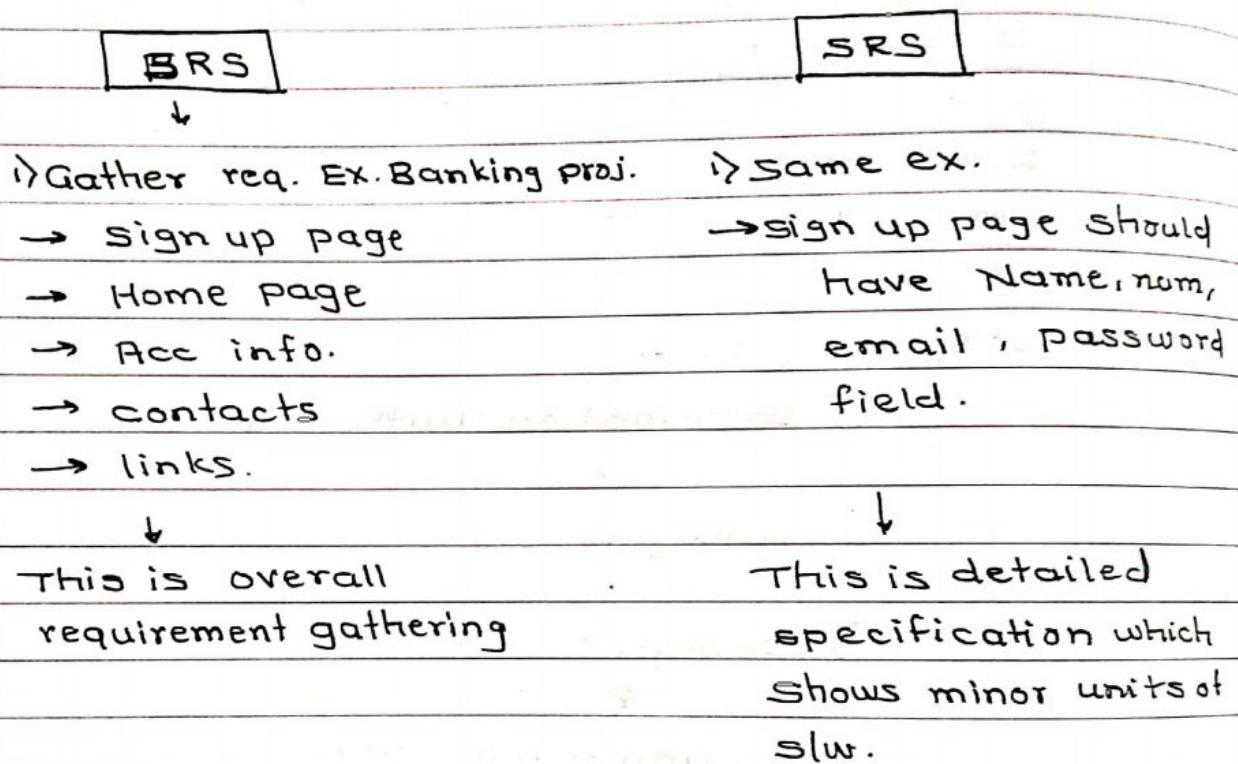
⑥ maintenance.

① Information Gathering

- 1) Business Analyst is responsible for information gathering.
- 2) Info gathering is nothing but requirement gathering from customer.
- 3) Information gathering involve business requirements specification. (BRS).
- 4) BRS is bridge between client & → Developer, Tester.
- 5) Business Analyst prepare BRS documents.

② Analysis

- 1) Business Analyst involve in this process.
- 2) In the analysis phase SRS is made.
- 3) SRS → software requirement specification.
- 4) SRS document made after BRS.
- 5) SRS is detailed documentation.



Lecture

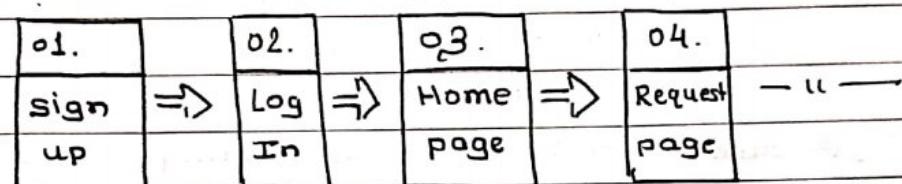
2

25-6-19

- c) SRS - Functional requirement specification.
- From BRS documentation SRS document get generated.
- 8) SRS documentation include:
- (i) Functional flow diagram
 - ↓
 - (ii) Functional Requirement
 - ↓
 - (iii) Use cases
 - ↓
 - (iv) Snap shots.

(i) Functional flow diagrams

- Functional flow diagram means flow of our task.
- This flow shows relationship between the tasks.
- This give proper sequence of task.
- Relationship of function means dependancy of each function.
- Example facebook:



- The functional flow diagram look like this.
- Overall this functional flow diagram is actually a stepwise representation of software.

(ii) Functional Requirement

- Functional Requirement means attributes which are required to complete a specific function.
- Now we have SignUp function.
- For sign up, its requirements are;
- First Name;
- Last Name;
- mobile Number;
- Email ID;
- Password;
- submit button;

- For First Name:
 - ① Name should be in character
 - ② Name do not have numbers.
 - ③ It should not have spaces.
 - ④ It should not have special symbols.

so like this, these all the requirements should get fulfilled in this phase.

(iii) Use Case

- 1) It is the functionality in terms of I/P & O/P.
- 2) Now consider the example of online shopping
- 3) online shopping has users are customers & Bank.
- 4) Admin of slw is company person.
- 5) Now use case for oil shopping is:-

(i)  (actor) = the person/group of people/system who interact with system.

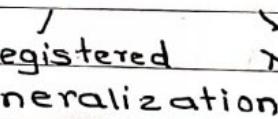
(ii)  (use case) = It is the functionality or operation.

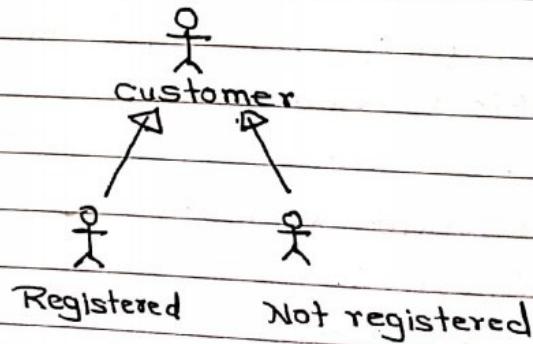
(iii) — = Link.

(iv) → = Generalization.

↳ Now generalization is the part.

↳ Customer can be of two type


↳ so generalization is shown as



(v) ----- = relationship.

↳ Relationship is of two types.

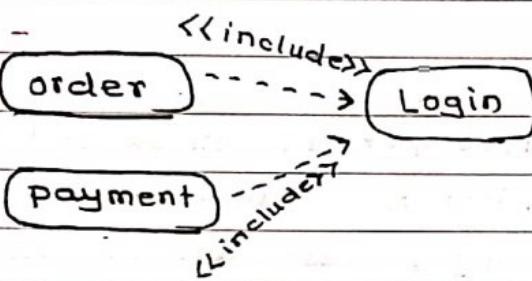
«Include»

«Exclude»

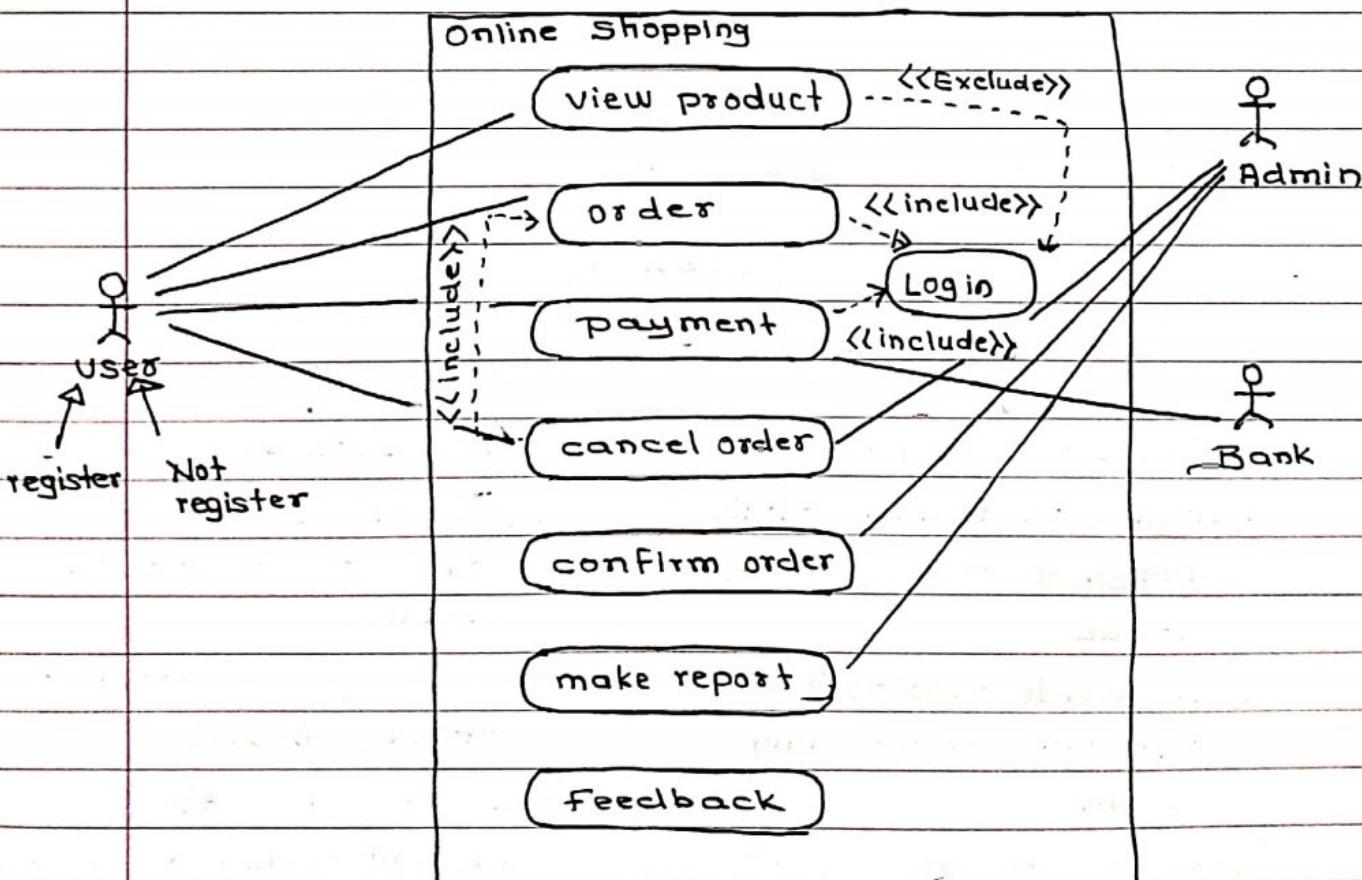
<< include >>

1) In online shopping process customer can not order & pay for things without loged in to the system.

2) So include reln is



→ use case for online shopping

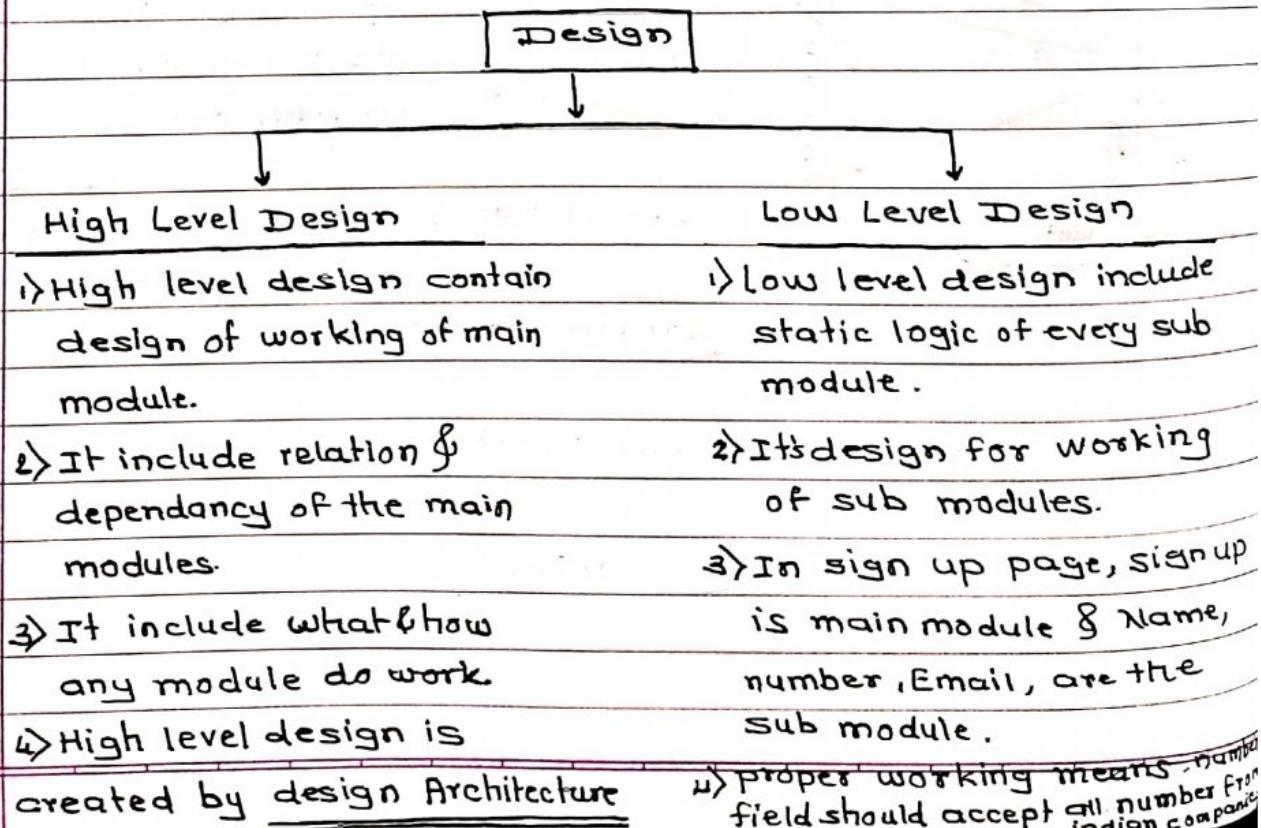


(iv) Snap shot

- 1) Snapshots are visualization of functionalities before development of product.
- 2) Snapshots are created by the Business Analyst.
- 3) Business Analyst create snapshot by using JRIse software.
- 4) Snapshot give idea to developer that how slw suppose to look like.

/* SRS is send to the developer as well as coder */
 /* when coder is developing the code slw tester
 do [Test case design] & [Test case execution Design]
 How to do Testing

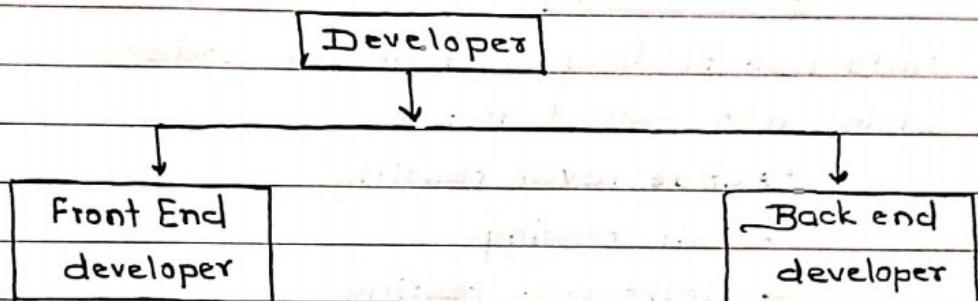
(3) Design



5) Low level design is created by the front end developer.

④ coding

- 1) Coding means programming.
- 2) One line is code.
- 3) multiple lines of code is called program.
- 4) set of programs written by the developer creates software.
- 5) There are two types of developer.

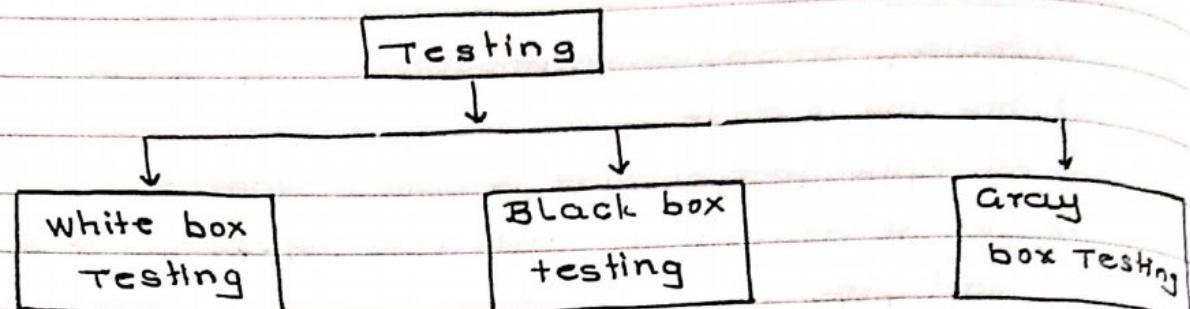


- | | |
|---|---|
| 1) U.I., functionalities, function flows, processes are developed by the front end developer. | 1) Data management
2) Data Gathering
3) Data security
4) Algorithm selection is done by back end developer |
|---|---|

person / developer who can work as front end developer as well as back end developer is called full stack developer

⑤ Testing

→ Testing is the process of checking completeness, correctness of the software.



(i) white box testing

- 1) White box testing is done by Coder.
- 2) It is also called as:
 - ↳ code level testing
 - ↳ Unit testing
 - ↳ clear box testing.
- 3) In the white box testing whenever coder complete his code writing, he checks 'or compile code', then if any bug found, coder have to solve it.
- 4) Coder cannot send code to tested without doing white box testing.
- 5) Coder check or test only positive scenario.
- 6) White box testing has purpose to test correctness and completeness of the program.

(ii) Black box testing

- 1) Black box testing is also known as system & function testing.
- 2) This testing is done by the Tester.
- 3) Overall functionality get checked in this type.
- 4) Tester check internal functionality depend upon external functionality.
- 5) Ex: Tester check whenever data in Sign module got entered & user press sign-up button, this button is process to store entered data. Tester check whether the data is stored correctly or not.
So here internal functionality is storing of data & external functionality is filling up data in fields & submit button's process.
- 6) Tester test the +ve as well as -ve scenario.

+ve scenario

if there is mobile number field, in india mob. num. are of 10 digits, then tester check field functionality by entering 10 digit numbers whether it works or not.

-ve scenario

Let us take same example the number field should not accept 9 digit or less & more than 10 digits, tester check system by entering less than 10 digit & more than 10 digits.

(iii) Gray box testing

- 1) Gray box testing is combination of white box testing and black box testing.
- 2) Tester is involve in this testing.
- 3) To do gray box testing, tester need programming knowledge.
- 4) The role of Gray Box tester is, whenever final sw eis handed over to tester, tester chk its functionality & if any fault occurs in the o/p of function then tester does not

revert system back to developer, instead of that tester himself solve or make changes in the code. So knowledge of coding is required.

⑥ maintenance

- 1) maintenance means provide service after delivery of the project.
- 2) Companies named as Tech Mahindra provide support to customer of vodafone, Jio.
- 3) maintenance involve Non technical support as well as technical support.
- 4) Non technical support is called BPO.
- 5) Technical support is called KPO.

SDLC.

1) Requirement Gathering (B.A. → BRS)



2) Analysis → SRS (B.A.)

- ↳ functional flow diagram
- ↳ functional Requirements
- ↳ Use cases
- ↳ snap shots.

3) Design → High level low level.

- | | |
|----------------|------------------|
| (main module) | (sub module) |
| (design Arch.) | (front end dev.) |

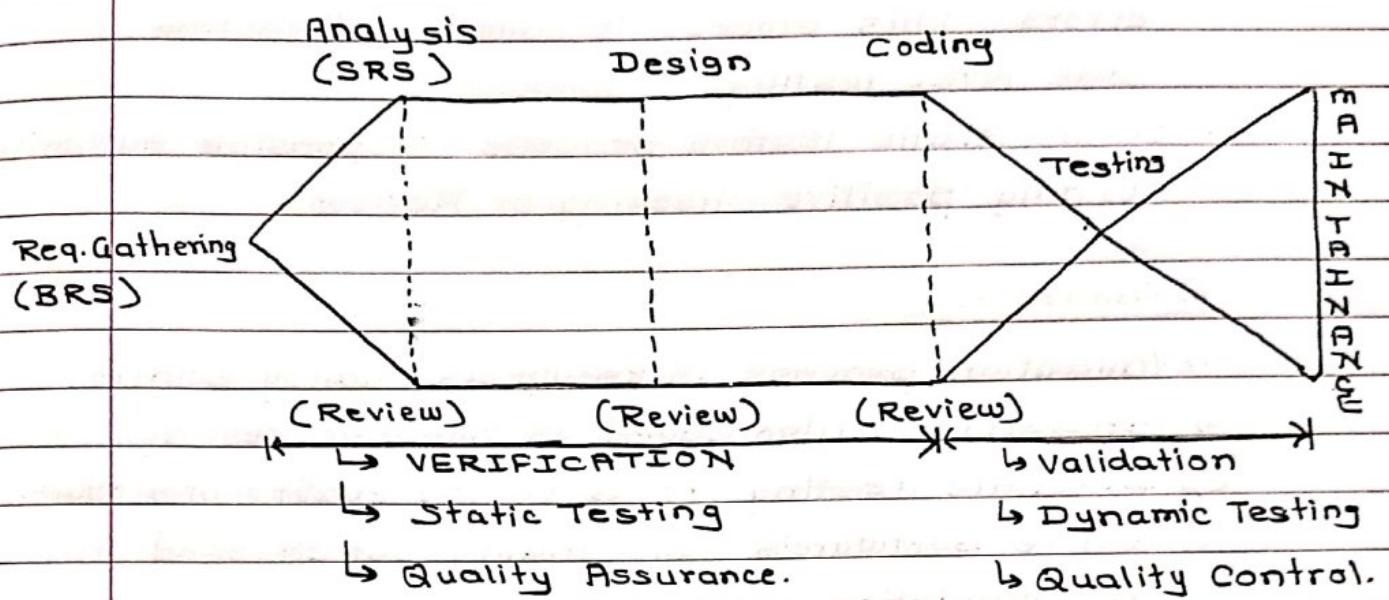


4) coding → set of programs to create s/w.

- | | |
|--------------------|-----------------------------|
| front end dev. | Back end dev. |
| ui, funct., design | data manipulat ⁿ |
| site map. | Security & algo. |

5) Testing : white box, black box, Gray box.

8) Fish Model



- ① Fish model is advance version of the SDLC.
- ② Fish model consist of Verification & Validation.

1) Verification

- 1) Verification process is also called as quality assurance.
- 2) It is also called as static testing.
- 3) Static testing is the review process.
- 4) In the requirement gathering process, Analyst gather all the information from the customer & make BRS documentation.
- 5) Then from BRS document B.A. make SRS document.
- 6) SRS document involve total functionality of project.
- 7) Then B.A. have to check whether it is correct or not. because all further processes are depend on the SRS documents.
- 8) This checking process is called as review process.
- 9) ~~In the~~ design phase after completion, design architect also have to do review process.

10) After coding is over coder have to check all the code, he have to compile code find errors fix errors. This process is called unit testing, or code level testing.

11) In the static testing process responsible Authority do only positive checking or Review.

2) Validation

- 1) Validation process is known as quality control.
- 2) Validation is also called as dynamic testing.
- 3) Dynamic testing focus on the quality of product.
- 4) whole software's functionality get checked in the validation process.
- 5) validation include Black box testing & Gray box testing.

*IMPORTANT QUESTIONS *

Page No.:
Date: YOUVK

* Based on S.T., SDLC, SQA, Test model, etc.

~~Q1~~ What is software Testing?

~~Q2~~ What is SQA ? Features?

~~Q3~~ Why you joined as a tester?

~~Q4~~ Who is good tester?

~~Q5~~ What is software project?

~~Q6~~ What is software dev. life cycle?

~~Q7~~ How many testers are there in your project?

~~Q8~~ How many developers were involve in your project?

~~Q9~~ Which are the generic phases involve in SDLC?

~~Q10~~ Different stages of SDLC?

~~Q11~~ Testing starts before or after coding?

~~Q12~~ What is static testing?

~~Q13~~ What is dynamic testing?

Q.1 What is software testing?

Ans:- Software testing is the process of checking correctness and completeness of the software to ensure its quality and functioning as per the customer's requirement.

Q.2 What is SQA ? Features?

Ans:- 1) Software Quality Assurance

Software Quality Assurance (SQA) is the process of monitoring & measuring the development of SW to ensure quality of product.

SQA has following features:

1) To meet the customer Requirement

↳ process of data Gathering

↳ comm' happen between customer & B.A.

↳ All the requirement from customer get documented

2) To meet customer Expectatn

↳ privacy (security) ↳ performance.

3) cost of the project

- ↳ cost is per hr. cost.
- ↳ If customer changes his req. in the middle of dev. then as per company's policy cost can increase
- ↳ Depends on how many person involved, how much resources get utilised.

4) Timing Delivery

- ↳ Delivery timing is get decided & documented.
- ↳ If any issue occurred by company then company has to pay penality in terms of ~~of~~ some % of total amount of project to customer.
- ↳ If time delay occurs due to frequent changes in the customer's requirement then customer have to pay extra charges.

5) maintenance

- ↳ maintenance happen after actual delivery of the product
- ↳ For some time period as per agreement maintenance is free & after that time customer have to pay charges.

↳ maintenance involve

Non technical
maintenance

BPO

Technical
maintenance.

KPO.

Q.3 Why you joined as a tester?

Ans:- Early era of the market has few companies who were ruling the market by their quality product. Nowdays tremendous competition is in the market & hundred thousands of companies are there who are giving quality product to customers, to ensure quality company needs software testers. for example in last decade of 20th century & early decade of the 21st Century Nokia is the company who ruled market single handedly, but now days companies like apple, Samsung, vivo & many more are compititing with each other to provide quality product to their customers. To focus on quality & functionality they need tester so I choose ~~the role~~ to become a tester after graduation.

Q.4 Who is good Tester?

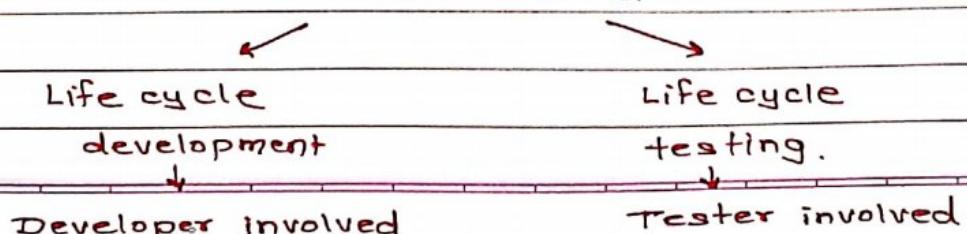
Ans:- The person ~~not~~ / Tester who can understand development of the software correctly from small unit to big module.

Q.5 What is software project?

Ans:- The software related problem solve by the SW engineer by software engineering is called SW project.

Q.6 What is SDLC?.

Ans:- SW dev life cycle is the process which consist of set of particular stages start from requirement gathering to the maintainance of product.
SDLC has two type



For the normal project dev. to tester ratio is
3 : 1.

For the critical project dev. to tester ratio is
2 : 1.

For NASA scientist to tester ratio is
1 : 7

Q.7 How many testers were involved in your project?

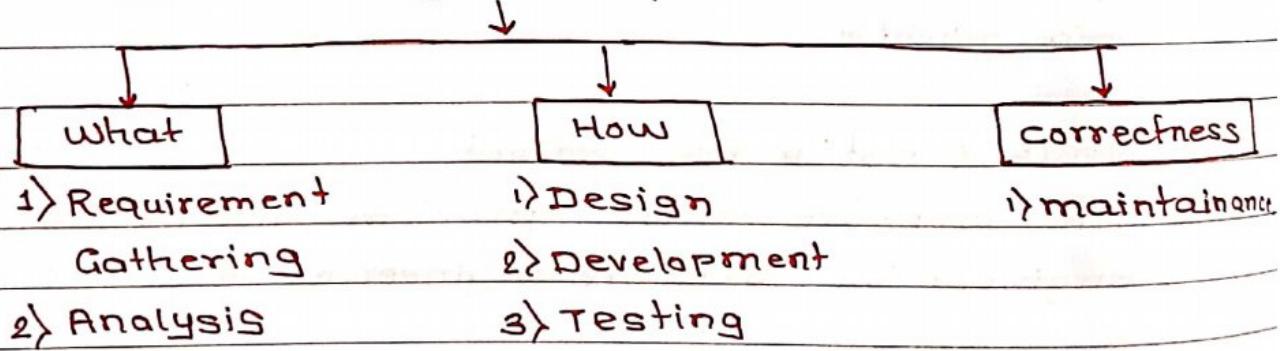
Ans:- 10 to 11 testers were involved.

Q.8 How many developers were involved in your project?

Ans:- Nearly 30 developers were involved.

Q.9 Which are the generic phases involve in the SPLC.

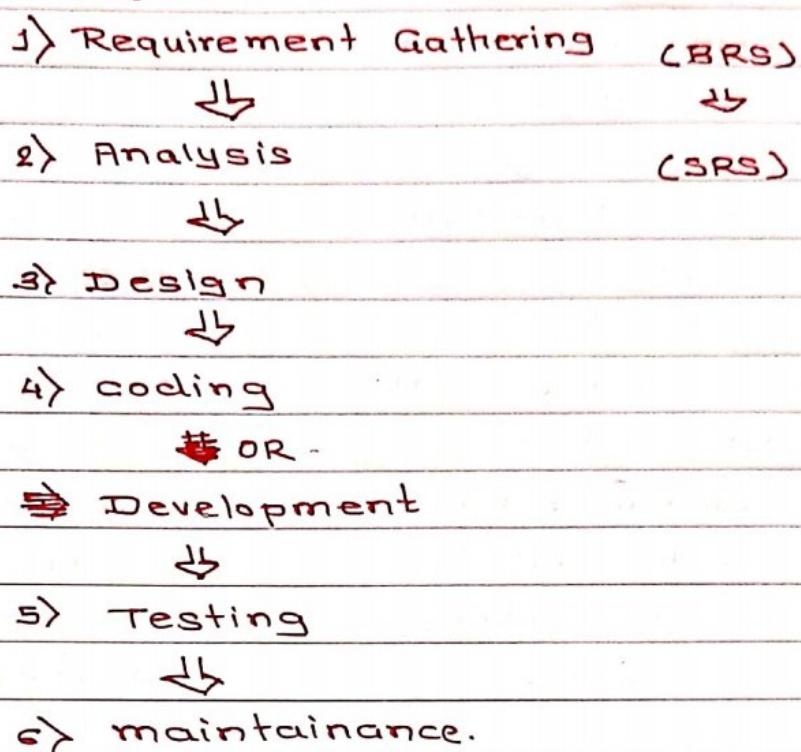
Ans:- There are 3 generic phases.



Generic means one phase then another phase then another phase.

Q.10 Different stages of SDLC

Ans:-



① Requirement Gathering

- ↳ Did by B.A.
- ↳ by communication betn B.A. & customer.
- ↳ All the req. get documented.
- ↳ Domain of project get identified.
- ↳ Doc. of this phase called as "Business requirement specification". (BRS)

② Analysis

- ↳ did by B.A.
- ↳ Analysis is the detail representation in the form of documents.
- ↳ include modules, sub-modules, functionality, desire I/p, desired O/p, relationship betn module
- ↳ phases in SRS :- Functional flow diagram
Functional requirement
Use cases
Snap shots.

3) Design : This phase is the site map for functionality flow, visualization & back end processes it has two types

Low level design

- 1) Design Architect is involved.
- 2) main module design get created.
- 3) Design base upon the dependency of the main modules on each other.

High level design

- 1) Developer (front end developer) is involved.
- 2) sub module functionality design get created.
- 3) Ex: number field in sign up module should accept 10 digit mobile number validated by all Indian telecom companies.

4) Coding : single line is code.

- multiple lines code creates program
- SW is set of programs which generate specific op's on correct I/O.

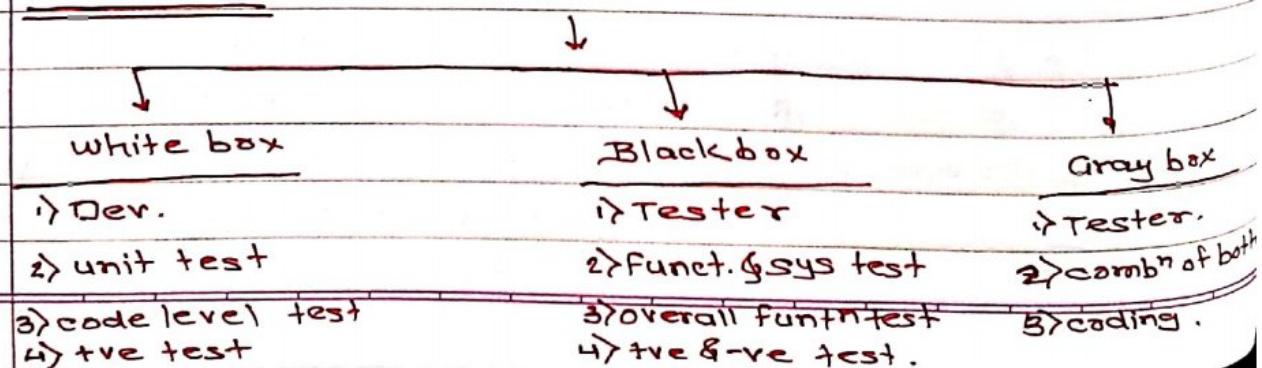
Front End dev.

- 1) UI.
- 2) Functionality.
- 3) Animations.
- 4) pages deve.
- 5) security Algo.

Back end dev.

- 1) Data man.
- 2) Deletion.
- 3) Data analysis
- 4) Data security
- 5) Security

5) Testing



6) maintenance

→ Technical support (KPO)

→ Non technical support (BPO)

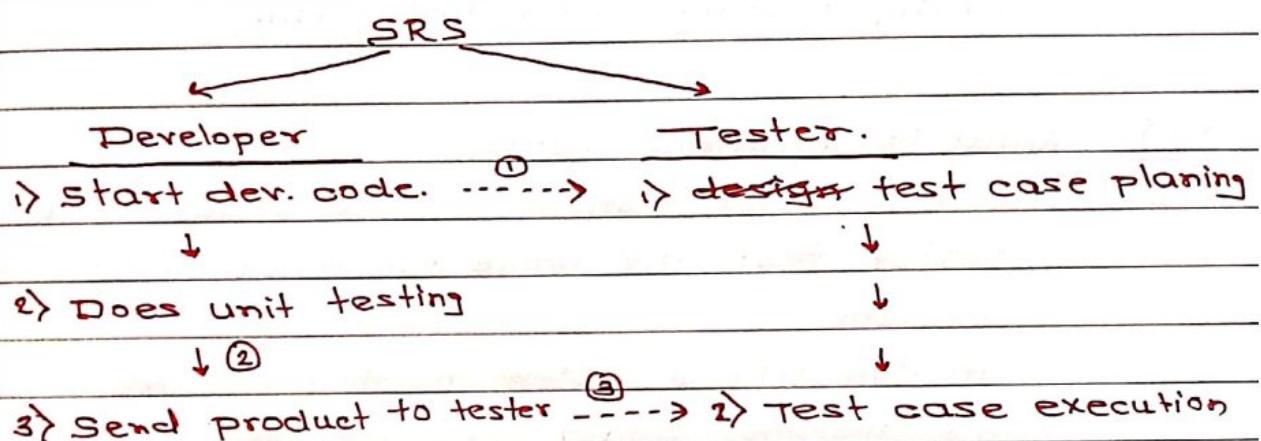
Q.11 Is testing start before coding or after coding.

Ans:- 1) After requirement gathering B.A. make BRS.

2) Based on BRS . B.A. make SRS.

3) SRS gives the overview of the whole system.

4) Then at same time SRS send to the dev & tester.



5) So testing & coding are the parallel processes
start at same time.

Q.12 What is static testing.

Ans:- 1) Static testing is the verification process.

2) Also known as quality assurance.

3) When req. get collected by the B.A. he made BRS.

4) Upon that BRS in Analysis phase SRS is developed.

5) SRS gives detail functionality of the SW.

6) All next processes are depend upon SRS.

7) So B.A. has to review the SRS , if any mistake happened or not.

8) After ensuring that all is ok, then B.A. forwards SRS.

- 9) When design Architect creates design, based on that design dev. develops program.
- 10) If any defect remains in the design then it causes faulty o/p while dev.
- 11) So designer also has to review design.
- 12) Similarly when developer develops code, he has to compile code to find bugs & remove them.
- 13) This process is called as unit test.
- 14) All these review processes are static testing.
- 15) Static testing is positive testing.

Q.13

What is dynamic testing.

- Ans:-
- 1) When coder handover final codes to the tester, tester tests the whole functionality of the system.
 - 2) Functionality & system testing means on desire if system should generate desire o/p.
 - 3) Black box testing + Gray box testing is involved in this test.
 - 4) Positive as well as -ve testing is done by tester.
- positive
-ve.
- 1) Number field should accept only 10 digit which are numbers.
 - 2) Tester test by entering exact 10 numbers.
 - 2) Tester test by entering char, symbols, >10 number less than numbers.
 - 5) Dynamic testing is also called as quality control.

Q.1) Where does the snapshot comes from?

Q.2) What issues you faced during your project? OR
Just mention what issues you faced in your project in excel sheet?

Q.3) What is Review? Do you involved in Review?

Q.4) What is high level design?

Q.5) What is low level design?

Q.6) What is prototype?

Q.7) Did you involved in the prototype?

Q.8) What is coding?

Q.9) Why it is known as white box testing?

Q.10) What is black box testing?

Q.11) What is gray box testing?

Q.12) Did you involved in gray box testing?

Q.13) Explain the review during Analysis, Design & coding.

Q.14) What is prototype?

Q.14) Where does the snapshot come from?

- Ans:-
- 1) Snapshots are created by business analyst.
 - 2) Business analyst create snapshot using HTML code.
 - 3) Now days snapshots are created by the Irise software.
 - 4) Nowdays Irise software has version 8.11.
 - 5) snapshots are the non functional is/w which provide only informative visualization.

Q.15) What issues you face during your project! or Just mention what issues you faced in your project in excel sheets?

- Ans:
- 1) Database Connectivity problem.
 - 2) Build installation problem.
 - 3) Developer not communicating properly.

Q.16) What is Review? Do you involve in the review?

Ans:-

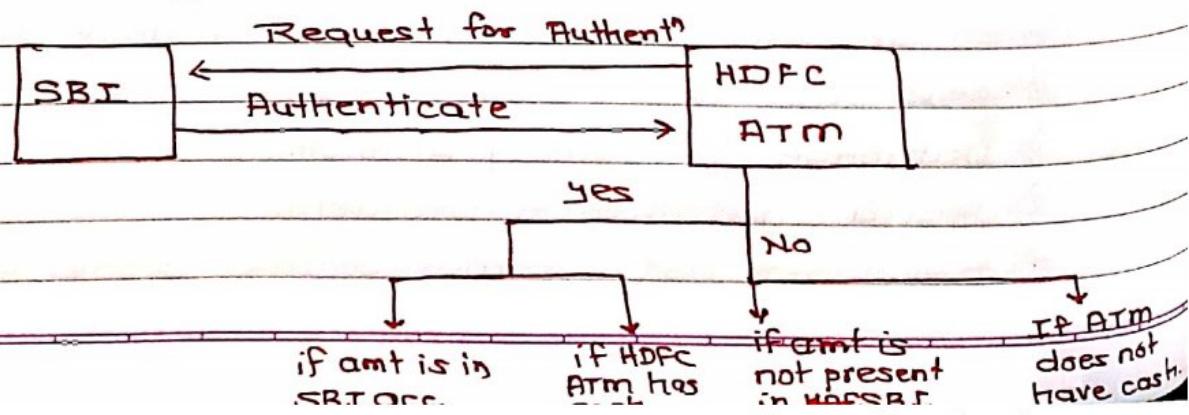
- 1) Review is the process involve in the static test.
- 2) Review process is verification process.
- 3) Review process is done for quality assurance.
- 4) In the verification at time of SRS creation
- B.A. review the SRS documents.
- 5) Design Architect, Sr-dev. review the design s/w.
- 6) After coding, coder review the code to find & solve bugs.
- 7) "Yes Sir I have involved in Test case Review, & defect analysis Review".

Q.17

What is high level design.

Ans:-

- 1) High level design is in term that it is a functional flow of main modules.
- 2) All the connectivities, dependency, relationships as well as flow are design in high level design.
- 3) Creator: Design Architect.
Sr. S/w Developer.
Solution Engineer.
- 4) High level design is called as "functional flow diagm".
- 5) Ex: If a person want to withdraw amt from SBI account using HDFC ATM then high level design is like:



Scanned by CamScanner

Scanned with CamScanner

Q.18) What is Low Level Design.

Ans:-

- 1) LLD is internal design.
- 2) This design state that how ~~function~~ sub module function works.
- 3) LLD include ER diagram
Data flow diagram
Class diagram.
- 4) LLD are created by developer.

Q.19) Did you involve in prototype.

Ans:-

- 1) Yes sir I did.
- 2) During test lead gives Knowledge Transfer to us.
- 3) & we were providing KT to customer.
* Tester give KT to customers*.

Q.21) What is coding.

Ans:-

- 1) coding ~~is~~ means programming.
- 2) Using specific coding language developer write program, to do specific task, * many programs combinely create slw.

Q.22) Why it is known as white box testing.

Ans:-

- 1) white box testing is transparent testing.
- 2) white box testing is aka clear box testing & glass box testing.
- 3) *developer do the white box testing.
- 4) when coding get over for the project. coder review the code.

- 5) In the review process coder compile the code.
- 6) If he found some bug, error then coder resolve that problem.
- 7) It is also called as unit testing.
- 8) Also known as code level testing.
- 9) It is the process of testing correctness & completeness of the program.

Q.23)

What is black box testing?

Ans:-

- 1) Black box testing is functionality & system testing.
- 2) Black box testing done by tester.
- 3) Tester does -ve & +ve testing.
- 4) Give ex of -ve & +ve testing.
- 5) It is the testing in which tester check internal functionality depend upon external functionality.
- 6) Ex: whenever client enter data in any field it should get stored in the db.

Q.24

What is gray box testing?

Ans:-

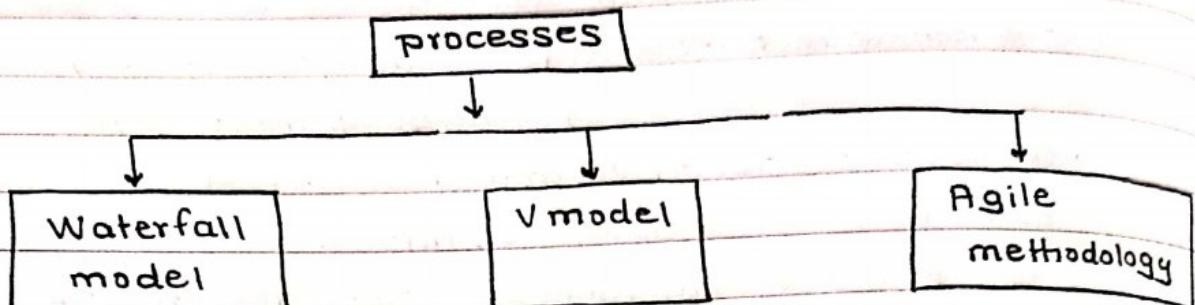
- 1) Gray box testing is combination of both white & black box.
- 2) Tester need some programming knowledge.
- 3) Tester should know internal coding of the system.
- 4) If any function do not generate desired output then tester should able to solve it by changing changes in coding.

Q.25 Did you involved in gray box testing.

Ans:- 1) No sir I did not get chance to become part of gray box testing.

1) Processes for SW development

→ Processes for SW development has 3 types.

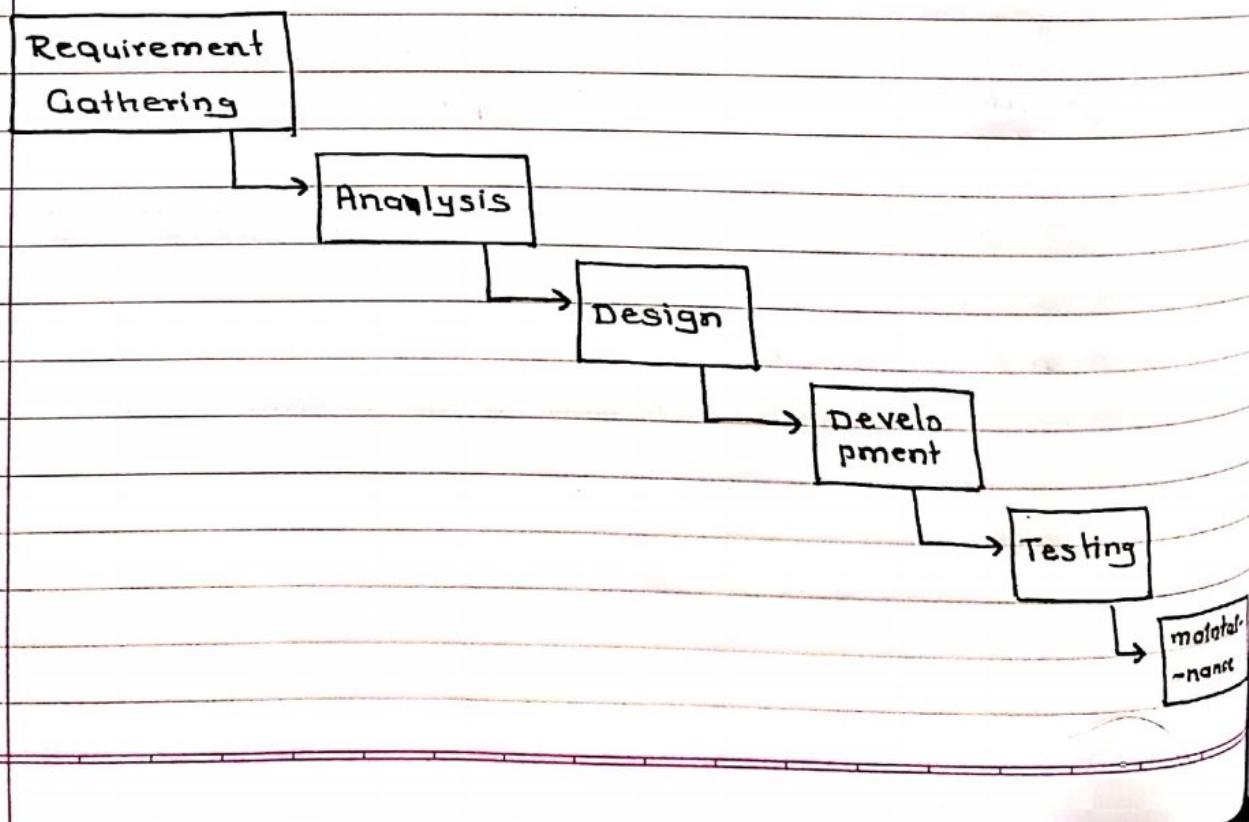


① Once 1st phase over & process goes to next phase then we can not revert back to 1st phase.

② If 1st phase over & changes are occurred in that phase & we can revert back to that process but by paying some Amr.

③ We can revert back & can do any changes in any module & that process does not effect any other module development.

① Waterfall model



- 1) Waterfall model is step by step implementation of the SPLC.
- 2) In the waterfall model whenever 1st stage over then & then, procedure goes to next stage.
- 3) Generally waterfall model use in small scale company and in "product base" company.
- 4) General time duration of the waterfall model is 3 months.
- 5) Product base company Vs. service based company

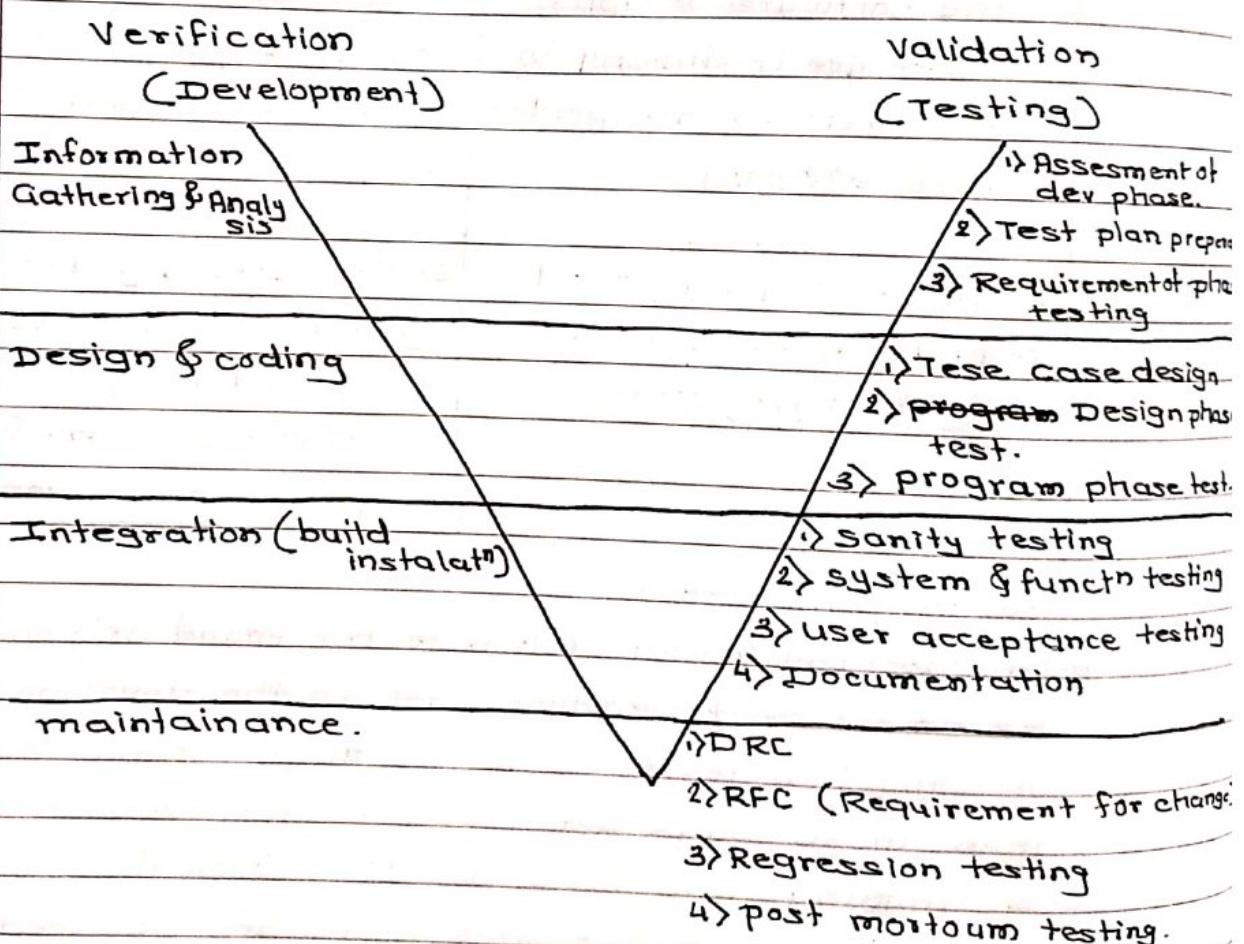
product based company	Service based company
1) company which makes the software or apps & gives give it directly to the customer are product base company.	1) company which makes their SW or app from other companies & give it for client usage are Service provider company
2) These company provide regular updates to the client to fix errors	2) This company do not provide direct update they send query to the product company & get the newer version

6) How they fix errors:

- 1) In waterfall model, when tester found any bug, then he do not send product back to the developer, instead of doing that tester lock the bugs, make report & then those bugs get fixed in next version of product.
- Ex. Samsung have there O.S. UI, when they release their UI version & if some bugs are occurred then they provide next version of their UI.

(2) V-model

- 1) V stands for verification and validation.
- 2) V model has working in which verification & validation are running parallelly.
- 3) V model is that model in which development stages are mapped with testing stages.
- 4) In the V model process, if one stage get completed & second phase is running & changes occurred in 1st phase then we can revert back to the 1st phase by paying some amount.
- 5) V model is use in big organisation.
- 6) Duration for proj dev. according to V model is 3 months.



- ① Information Gathering & Analysis \Rightarrow
- 1) Assessment of dev. phase.
 - 2) Test plan preparation
 - 3) Requirement of phase testing

① Assessment of dev. phase

- ↳ Assessment of dev. phase means to make strategy for testing.
- ↳ This phase decide which methodology is going to be used in our test.
- ↳ Strategy for project dev is also get decided.
- ↳ They plan which methodology for test like Automat testing or manual testing is going to be used.
- ↳ Test responsibility matrix is get finalised (TRM).

② Test plan preparation

- ↳ Now the implementation of TRM is here.
- ↳ Project manager is responsible for TRM implement.
- ↳ Project manager prepare a test team.
- ↳ He distribute work to all team members.
- ↳ Test Estimation is created in this phase.
- ↳ Test Estimation means in how much time test will complete.

③ Requirement of phase testing

- ↳ phase means unit.
- ↳ In this part estimated requirement for phases are created. finalised.
- ↳ Ex. paytm continuously introduce new p modules like donations, so these modules are like new products so for them requirement got estimated.

② Design & coding \Rightarrow

- 1) Design phase testing
- 2) Program phase testing
- 3) Design test cases.

1) Design phase testing

2) program phase testing

- 1) Phase means code testing.
- 2) This code testing starts from small unit of program.
- 3) Developer is involved in both processes.
- 4) Developer is involved because he knows the code & he can solve problem easily.
- 5) This phase is like white box testing.

3) Design test cases

- 1) Test case design goes in two parts.
- 2) one part is +ve scenario testing and second is -ve scenario testing.
- 3) They decide scenario of testing & according to that they will execute test cases in later stages.
- 4) Testers are involved in this.
- 5) This testing is like black box testing.
- 6) From SRS documentation they decide test case scenarios.

Example:- Suppose sign up page is there.

There is sub module named as mobile number.

Then test case scenario are as follow.

\swarrow \uparrow \searrow
+ve test case

-ve test case.

- ① the number field should accept 10 digit number & there datatype is int. Then field should tester design test case as enter 10 digit number.

- ① The number field require 10 digit number & dt. is int then for -ve tester design enter less than 10

③ Integration (Build Installation)

- ⇒
- 1) Design phase Sanity Testing
- 2) System and functionality test
- 3) User acceptance testing
- 4) Test documentation

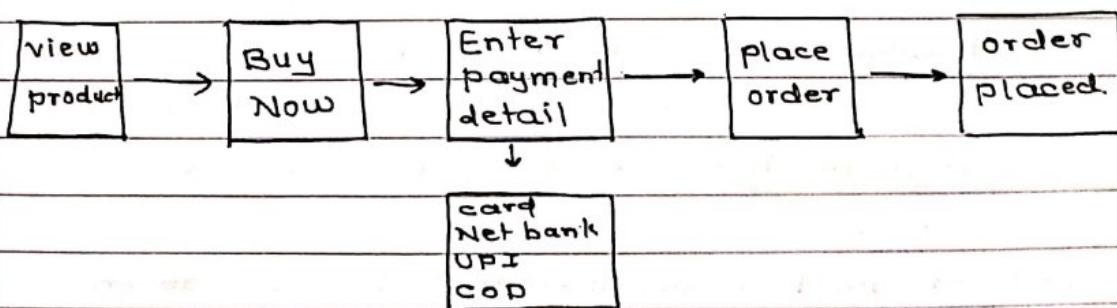
③ Integration (Build installation)

- 1) There are 5 to 6 module development comes under 3 month period of V module.
- 2) Sometimes there is old project & company got requirement of developing new modules, which customer wanted to add in slw (old project)
- 3) Then development team develop that project (modules) they perform WBT on it. When modules are ready they have to add / integrate modules into old slw.

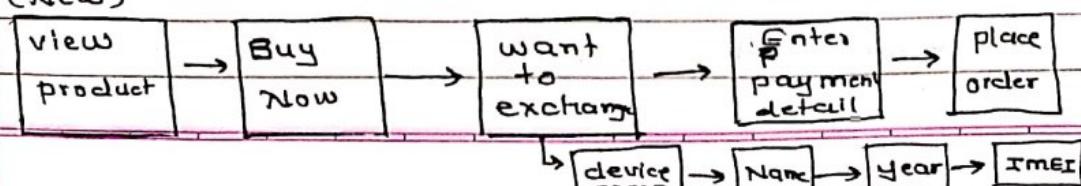
Example:

Paytm slw have payment (Billing) module, they have products which user can buy. Now Paytm wants to add new module like whenever you buy new mobile you can exchange your old mobile. Now this buy-back module has many sub modules like "Select device company", "Select device Name", "Select device mfg year", "Enter IMEI number", base upon this info paytm will show price for old device & Agreement page. And then user can proceed to buy new device.

(Old)



(New)



4) So from above example integration is the process or mapping process of new module in which O/p of newold module work as i/p for newold module & this working should be error free.
Ex. in old version when we select device we can directly purchase it, but now when we enter the old device detail, the price of old device get deducted from final product price.
so here o/p of new module work as i/p for old purchase module.

5) Developers are responsible for integration testing.
6) White box tester does integration testing.

1) Sanity Testing

- 1) Comes under validation process.
- 2) Tester is responsible for sanity testing.
- 3) Sanity testing is core functionality testing.
- 4) Basic functions are get tested in this.
- 5) Only critical errors are get documented (logged)
- 6) Example:-

If there is module of donation, its core functionality is to open donation page when customer hit the donation logo. Tester test same functionality. And the critical error is when after clicking on the donation logo if page does not respond then it is critical error.

- 7) After build & installation first test happens is the sanity testing.
- 8) Without build & installation we can not do sanity testing.

2) Functionality and system testing

- 1) After sanity testing, this testing happens.
- 2) Black box tester is responsible for it.
- 3) This test check all the functionality as per SRS document.
- 4) They test it by using negative testing as well as positive testing.
- 5) Small defect to large defect get documented in this phase.
- 6) Example:

In sanity testing if submit button do not have any color or specific animated styling then also if it is working, it do not recognize as defect but in functionality testing it get termed as defect.

3) User acceptance testing

- 1) After removing defect after system & functionality testing, product now move to the UAT.
- 2) In F&S testing product get validated.
- 3) Then in UAT that product get deployed in user env.
- 4) In user env. product has UI, design, colors, functionality as per user requirement.
- 5) Tester & user both involve in this testing.
- 6) Ex. When new module get added into the paytm the product launch on www.tst05.paytm.com this is the user environment.
- 7) When user validate it as correct then product send to final production.

4) Test documentation

- 1) Test documentation is report of testing.
- 2) Each tester has his own part in whole testing.
- 3) Whenever tester does testing on module, he creates document for testing.
- 4) It include:

Name of module.	Execution	pass	fail

- 5) Tester send test document to the team leader.
- 6) Team leader send document to project manager.
- 7) Project manager send document to customer.

- * we saw first 3 phases of the V model in 4th & 5th lecture Now we will see maintenance phase of V model.

④

maintenance

- Defect removal Efficiency.
- 2) Request for change.
- 3) Regression testing
- 4) post mortum testing

1) Defect Removal Efficiency

- ↳ we have 4 phases of testing Environment.

DIT (Development integration testing)
(person involved : Developer)



SIT (System integration testing)
(person involved: Tester)



UAT (User acceptance testing)



(people involved : customer & Tester)



Release (production)

- ↳ Now the defect removal efficiency is the process of calculating at which level tester did testing.

- ↳ Suppose there is a game having 12 steps, & tester test only 12 then tester test only 50% of total game.

↳ There are two phases in the DRE.

(A) Defect found by tester.

(B) Defect found during user acceptance test.

↳ When tester doing his work he found defects, some defect get solved & some get canceled, so only solved errors/defects get into the consideration.

Ex: suppose there are 100 defects.

Only 90 get solved.

10 defect get cancel.

So we take only 90 defects in consideration.

↳ In user acceptance testing, whenever customer test system in ^{test} environment, he could find defects ~~then~~ defect which get solved take into the consideration.

Ex. suppose at UAT customer found 10 defects.

only ~~10~~ defect get solve.

so we take 10 defect in consideration.

so here A = 90 defect

B = 10 defects.

Formula to calculate DRE = $\frac{A}{A+B} = \frac{90}{90+10} = 0.9$.

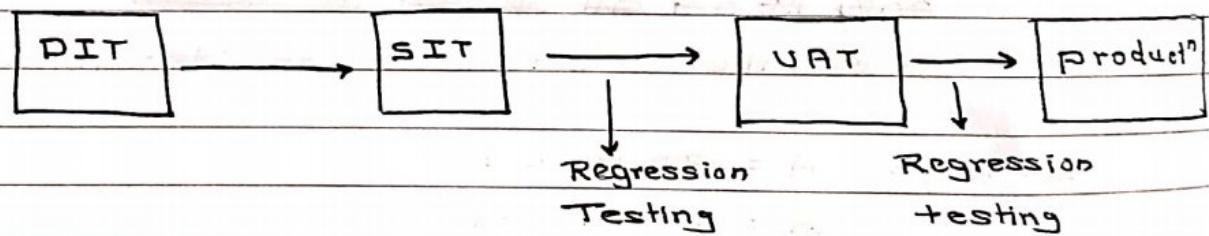
DRE	Remark
0.8 to 1	Good testing
0.5 to 0.8	Avg testing
below 0.5	below Avg

2) Request for change

- ↳ If customer want some changes in his product at the time of release , to do so customer request for change.
- ↳ The changes in the product are mention in the SRS document. At the end of the SRS there is a section "CR" it is in red color.
- ↳ SRS is a PDF format document.
- ↳ Customer have to pay extra money for this.

3) Regression testing

- 1) Regression testing is subset of the sanity testing.
- 2) As we know there are 4 test environments.



3) When tested complete system integration testing he again checks the basic core functionality of system.

- 4) After completion of UAT tester once again chk the basic functionality of the system.
- 5) This basic functionality testing is called as regression testing.

4) Post mortum testing

- 1) Post mortum testing is the work of white box tester.
- 2) When whole testing done & product ready for production, if product does not producing desired o/p then white box tester have to check all the modules in detail.

Ex:

Suppose we order something from ol. store, then we get an order ID, this order id get stored in OCE (Order Captuation Engine). But there is problem we get order ID but our order fails. So there is problem in system. then tester have to test all the functionalities again.

* when interviewer ask explain V model:

Then simply explain V model.

Then ask interviewer should I explain draw diagram.

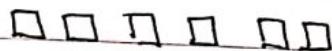
If he ask explain UAT then explain UAT.

He ask that is it correct? then answer obviously this is correct & we are following this in our company.

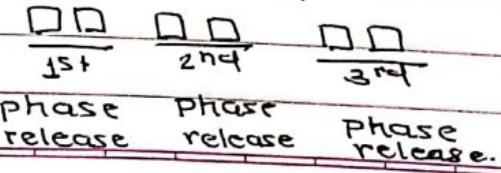
⑨ Agile methodology

- In V model after requirement get fixed then whole team focus on whole product development.
- 3) But in Agile methodology, this methodology is the module driven methodology.
- 3) Requirements are changing frequently in the agile methodology.
- 4) So it is not plan driven methodology.
- 5) Agile methodology is used in service based companies.
- 6) In the ~~service~~ agile methodology customer can request for changing in requirement at any point of dev-phase.
- 7) Customer can request for change in any phase like ~~DI~~ST → SIT → UAT → production.
- 8) This change in the requirement do not affect ~~the~~ development of any other module.
- 9) Customers do not have to pay extra money for changes in the requirement.
- 10) Change in the requirement do not increase the development time.
- 11) Agile method is value driven methodology.
- 12) There is no SRS in the agile, instead of the SRS there is "Rally" tool, in which any changes are inserted by the Product owner & those are visible to the tester & developer.

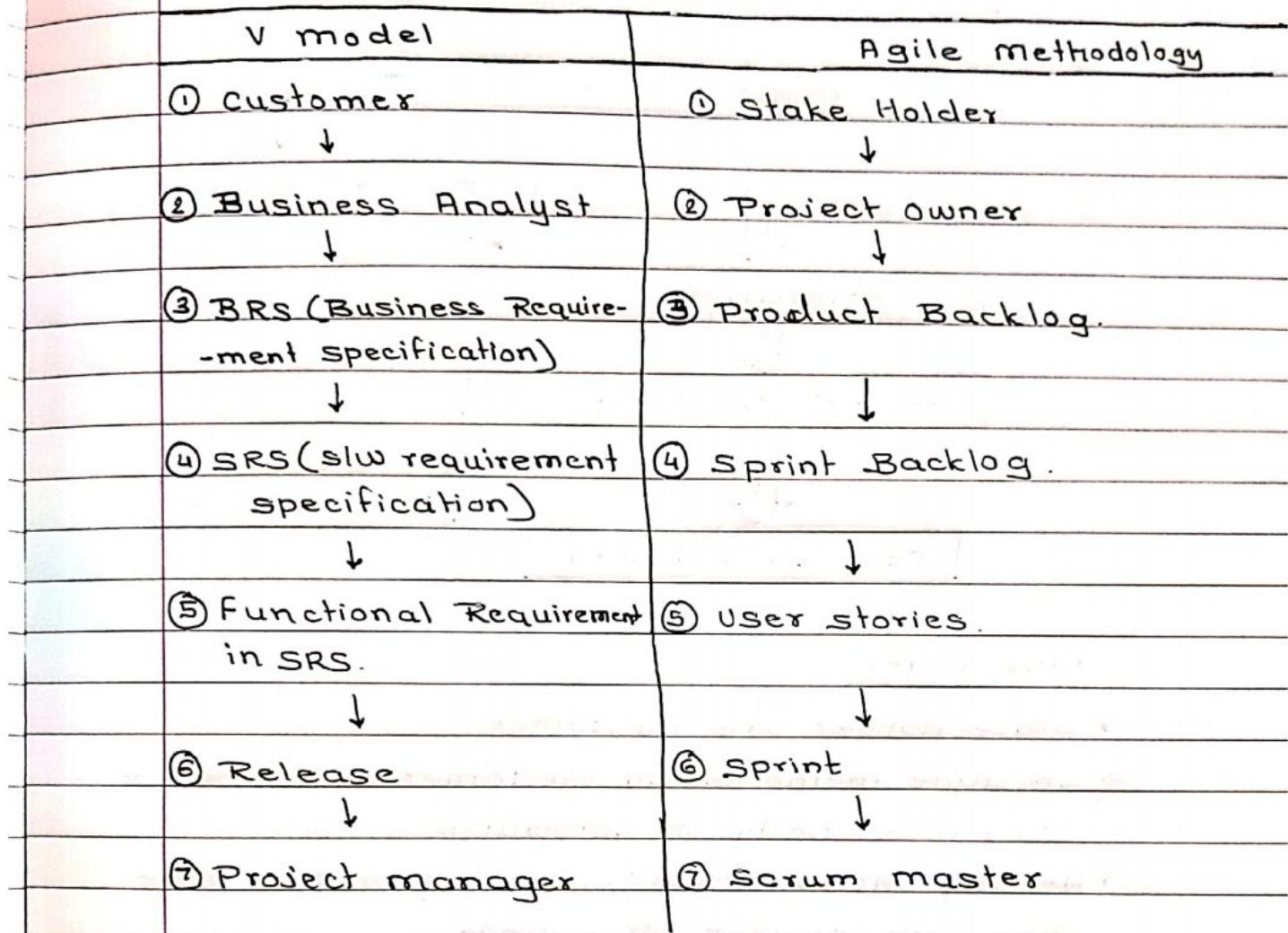
Ex: suppose there is a SW project having 6 modules.



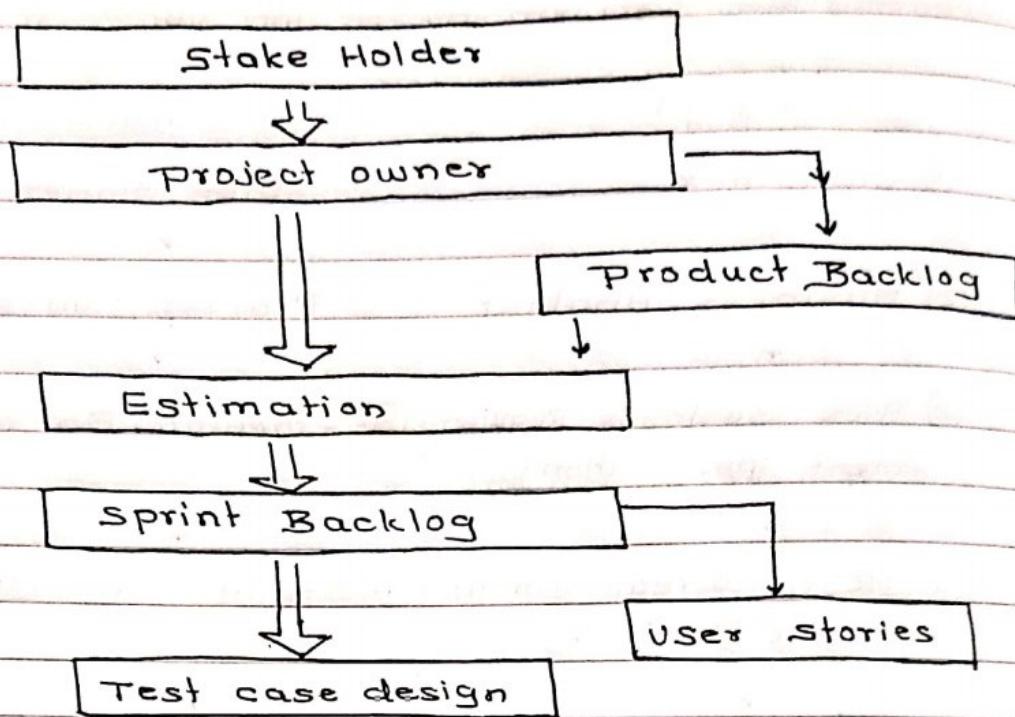
In agile method, this project divided in group of modules



According to the number of modules to be develop, team formation perform by project owner or scrum master.



(4) Architecture of Agile methodology



① Stake holder

- ↳ Stake holder are customer.
- ↳ In agile methodology customer is member of top most body of company.
- ↳ At any phase of dev, testing or production they can request for change.
- ↳ They have bunch of requirements.

② Project owner

- ↳ Project owner gather requirements from the stake holder.
- ↳ Project owner is team member of Sprint planning meeting.
- ↳ Project owner is responsible for creating project backlog.

③ Project Backlog

- ↳ Project Backlog are the total requirements for the whole project.
- ↳ It include requirement for all modules.

④ Estimation

- ↳ In the agile method , the focus is on module base release.
- ↳ So all the requirements are not of any use for specific module.
- ↳ So to decide requirements among the all requirement estimation is done.
- ↳ people involved: Project owner.
Development lead
Test lead.

- ↳ Estimation is also called sprint planning meeting.
- ↳ It is sorting of req. to dev. module.
- ↳ Estimation is the process of how to deal with problem when obstacle comes.

- ↳ There are 3 main factors in the Estimation.

Knowledge	Efforts	complexity
1) whenever team formation done, each member of team should have knowledge about domain of the project.	1) In this Authority decide how much efforts are require for project. 2) They decide how much resources, people required.	1) To do estimation for time, cost, resource, first company mean complexity of the project.
	3) Selection of user stories depend on module.	

5) Sprint Backlog

- ↳ created by the project owner.
- ↳ Sprint backlog contains user story.
- ↳ ↳ Sprint backlog contains detail ^{info} ~~story~~ of the requirements which are required for module dev.

6) User Story

- ↳ User story are the functional requirement for the module development.
- ↳ User stories are get decided in the Estimation phase.

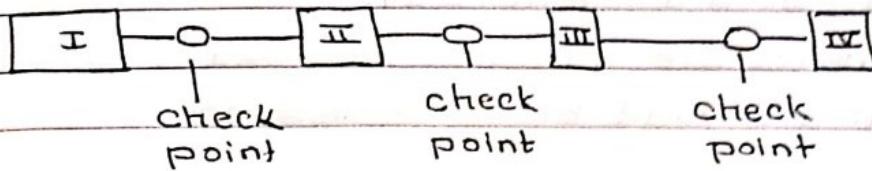
7) Test case

- ↳ Test case are design by testers.
- ↳ Test cases are mapped with user stories.
- ↳ This is done because there should be no point left behind in testing phase, if it is done then it is fine.

⑤ Advantage of Agile method

1) Check Points :-

Consider an example where we have to dev. a module & module has sub modules.



(i) In v model if at the time of production if any defect occur, then developer have to do post mortuum testing.

(ii) But in the agile method when any problem occur at time of production tester check the check point so no post mortuum testing required.

(iii) If I is starting phase & IV is end phase then, at the time of productn error occur, then tester check ckpt betw I & II, if it is correct then he check II & III. if defect occur in this ckpt, then tester send it to developer.

(iv) Developer then click on that ckpt & coding appears, then at the defect, code is highlighted with red color.

(v) To check the ckpt tester use AVAS tool.

2) Scrum meeting

↳ Daily status call meeting.

↳ Time for meeting either (10:30 to 11 or 11 to 11:30)

↳ This meeting daily conducted for 15 to 30 minutes.

↳ People involved : Scrum master (Chair Person)

Testing Team

Development Team

↳ Agenda of this meeting:

"What is the progress of project?"

↳ 3 main questions ask in scrum meeting:

(i) What we did yesterday?

(ii) What we are going to do today?

(iii) What are road block or issues?

(i) What we did yesterday?

↳ It is the report of previous work which was completed by team members whether they are tester or developer.

(ii) What we are going to do today?

↳ It is the work from pending work which team member have to complete by the end of day.

(iii) What are road block or issues?

↳ This include :- Difficulty in executing test cases.
Error or defect in code compilation
Lack of cooperation from team.

↳ General working of agile methodology.

→ Sprint duration 1 month.

Day 1 to Day 15 = test case design

Day 15 to Day 30 = test case execution & Deployment

↳ In day 1 to 15 we can ask difficulty about test case in scrum meeting.

↳ In day 15 to 30 we can ask for difficulty about test case execution in scrum meeting.

3) Implementation of Automation

- ↳ We can implement automation in agile method.
- ↳ Nowdays 70% is manual & 30% is automatn test.
- ↳ selenium is the main tool to do automation.
- ↳ Advantage of automation.
 - ↳ Less Resource requirement.
 - ↳ Less resource means less cost.
 - ↳ High accuracy
 - ↳ less human errors.
 - ↳ Require less time to complete.

4) Sprint wise delivery

- ↳ In V model.
project duration = 3 months.
i.e., 1 release = 3 months
3 months = 5 to 6 modules.
1 year = 4 releases.

↳ In Agile methodology.

- 1 release = 1 month
1 month = 1 to 2 modules
1 year = 12 releases.

↳ Sprint wise delivery is module wise delivery or value driven methodology.

⑥ Disadvantage of agile methodology

To implement new module in previously developed software , tester & developer should have total knowledge about flow of sw , scenarios , dependency & relationship of the modules , if dev & tester have knowledge of all this then they can work in agile.

There is no time for knowledge Transfer (KT) in agile because release (sprint) time is only 1 month.

* Important Questions *

- ~~Q.1.~~ What is the critical functionality of bank appln?
- ~~Q.2.~~ Explain DRE.
- ~~Q.3.~~ In which scenario of DRE, the ratio would be consider as 1.
- ~~Q.4.~~ Is it possible to get zero bug density during maintenance?
- ~~Q.5.~~ What is your release plan for project?
- ~~Q.6.~~ What will you do ^{inv} if the customer tell the changes in requirement during execution or during product?
- ~~Q.7.~~ Why V model is expensive.
- Q.8. Do you involve in agile methodology?
- ~~Q.9.~~ What are the advantages & disadvantages of agile methodology?
- ~~Q.10.~~ Which are the types of agile methodology?
- ~~Q.11.~~ Which tool you are using for agility.
- ~~Q.12.~~ Who is the chair person in Scrum meeting?
- ~~Q.13.~~ What is delivery duration in agile?

Q.1 What is the critical functionality of bank applicn?

- Ans:-
- 1) Security
 - 2) Account
 - 3) Transactn.
 - 4) Performance.

Q.2 Explain Defect Removal Efficiency.

- Ans:-
- 1) DRE is the calculation of how or ^{up to} ~~on~~ which level we do testing.
 - 2) It is the calculation of how good our testing is.
 - 3) Two main factors are require in DRE.
 - (A) Defect solve by tester.
 - (B) Defect found by customer at UAT.

Q.4. Is it possible to get zero bug density during maintenance?

Ans:- Yes, It is difficult but it is possible.

Q.5 What is your release plan for project.

Ans:- 1) In waterfall model:-

1 release = 3 months.

1 year = 4 releases.

2) In V model:

1 release = 3 months.

3 months = 5 to 6 modules.

1 year = 4 release.

3) In Agile methodology.

1 year =

1 release (sprint) = 31 month.

1 month = 1 to 2 modules.

1 year = 12 release.

Q.6 What will you do in V model if customer tell changes in requirement during execution or during product.

Ans:- If this happen in v model then we accept # change in requirement request, but we should take extra money to do changes.

If customer req. For mobile number should be of 10 digits & at time of execution he want that mobile number should contain 91 before it. then we will accept it by charging some extra money.

Q.7 Why V model is Expensive?

Ans:-

In V model development & testing goes/run simultaneously / parallelly so it is costly. & use in large organisation.

Q.8 Do you involved in agile methodology?

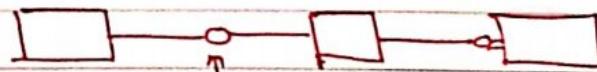
Ans:- Yes I am involved in agile methodology in ~~our~~ organization which I'm working.

Q.9 What are the advantages & disadvantages of agile methodology?

Ans:-

① Advantages

⇒ Check points



check point.

1) Tester check test points.

2) No need for post mortem testing.

3) Tester use AVAS tool.

4) chk only ~~to~~ points betn modules.

5) If problem occurs then forward to dev.

6) Dev. then solve error in code.

2) Scrum meeting

- ↳ Daily progress report.
- ↳ Duration : 15 to 30 minutes.
- ↳ Time : 10:30 to 11 or 11 to 11:30.
- ↳ Agenda: To check project work progress.
- ↳ Question:
 - 1) What you did yesterday?
 - 2) What we are going to do today?
 - 3) What are road block or difficulty?

3) Automation

- ↳ 70% is manual 30% is Automatⁿ.
- ↳ Selenium is main tool.
- ↳ Less resource requirement.
- ↳ Low cost.
- ↳ Less timing.
- ↳ Less human error.
- ↳ High Accuracy.

4) Sprint wise delivery.

- 1 release (sprint) = 1 month
- 1 month = 1 or 2 modules.
- 1 year = 12 sprints.

② Disadvantages

Each & every member should have complete knowledge of previous project like scenario, workflow, functioning, module dependancy, then only they can integrate new module in s/w.

There is no time for KT in agile because there is 1 month duration.

Q.10 Which are the types of agile methodology?

Ans:-

- 1) Xtreme programming
- 2) LEAN
- 3) Kamban
- 4) Scrum
- 5) Dynamic system development method.
- 6) Future-driven development method.

Q.11 Which tool you are using for agility?

Ans:-

"JIRA" is the project management tool ^{we are} using for agile testing.

Q.12 Who is the chair person of scrum meeting?

Ans:-

Scrum master is the chair person for scrum meeting.

Q.13 What is the delivery duration in Agile.

Ans:-

1 release (^{sprint} SPRINT) = 1 month

1 month = 1 to 2 module

1 year = 12 release (Sprint)

→ User Stories

- 1) User stories are the requirements.
- 2) whenever stakeholder gives requirements (Us.) to product owner, those requirements are for whole product.
- 3) In Estimation, at sprint planning meeting members decide which module to develop & those modules requirements.
- 4) Those modules sorted requirements are included in the sprint backlog.
- 5) Those user stories are functional requirements for module to be dev. in 1 month.
- 6) User stories consist of two parts:

← →
Description Acceptance Criteria.

→ Description :- it is the description about what user want to do (process) & what is his desired o/p.

8) Acceptance criteria: These are the scenarios when these scenarios are true then system generate correct o/p otherwise system show failure.

9) Template for description:

As a {user type}, I want to {process} so that {benefit}

customer purchaser	what they want to do	benefit
-----------------------	-------------------------	---------

As a customer, I want to insert debit card in ATM
so that it can open my account

10) ID of user stories started with US001.

Example:

Name of User Story: Credit card payment.

ID: US001

i.e., US001 : Credit Card Payment.

Description: As a customer, I want ability to pay with credit card so that it can confirm my order.

Acceptance Criteria:

1) Discover card

↳ When we enter card number, it should display name of company of card i.e. Rupay master card.

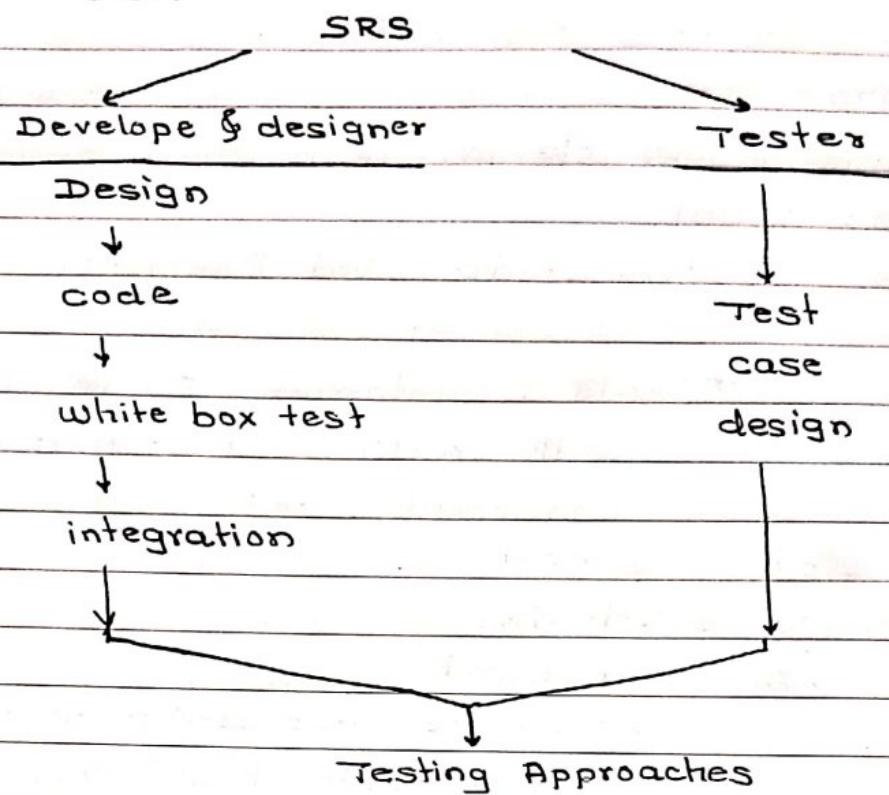
2) Validate Credit card information.

3) Then validate expiry date & CVV.

4) Validate address.

5) Generate Success or failure result.

→ Integration Testing

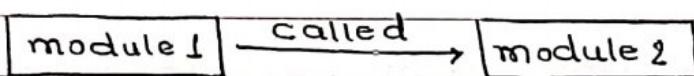


- 1) when SRS get completed & verified, B.A. send SRS to dev,designer & tester.
- 2) when developer done with coding then he do white box testing on it.
- 3) when white box testing over ,then developer have to do integration.
- 4) Integration is the process of mapping new module with the old module.
- 5) Developer should have all the knowledge about functionality, relation, dependancy of module over each other. then he can do integration.

- 6) SIW is the system in which o/p of one module is used as i/p for another module.
- 7) When dev. do integration then he & also do integration testing.
- 8) Integration testing is the process to check correctness & completeness of the flow of functionality whenever integration of module performed.
- 9) Integration is of two type.

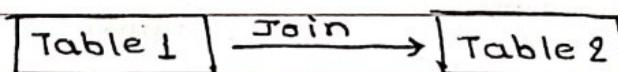
(i) Front End Integration

In the front end integration developer connect modules using "called" function.



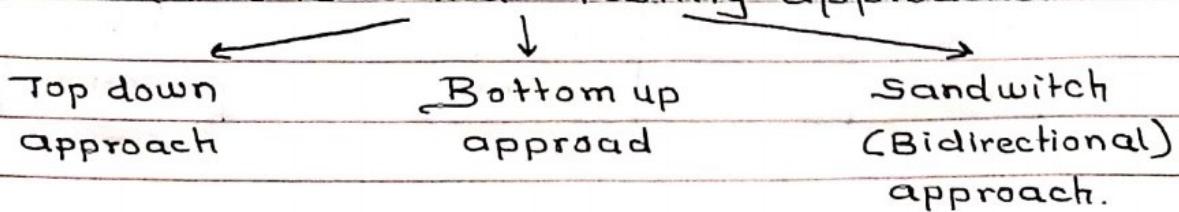
(ii) Back end integration

Back end integration, include connecting two or more tables in database using join query.



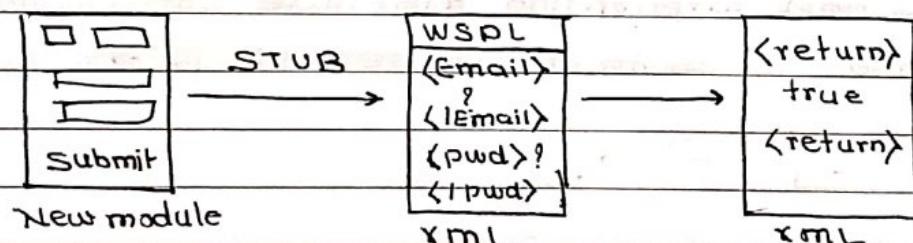
- 10) communication between two module proceed through XML language.

- " whenever integration process end, then testing starts , this is called "testing approaches".

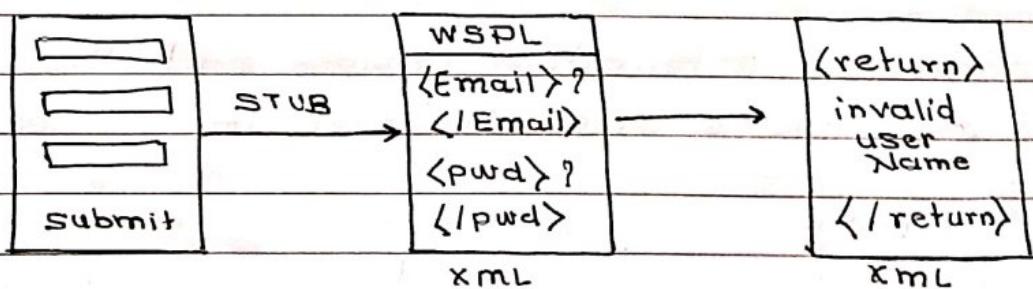


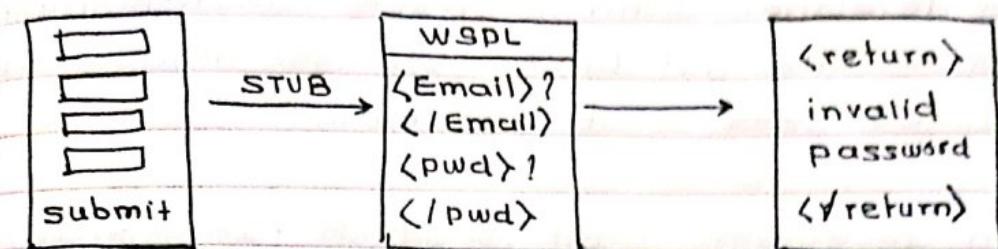
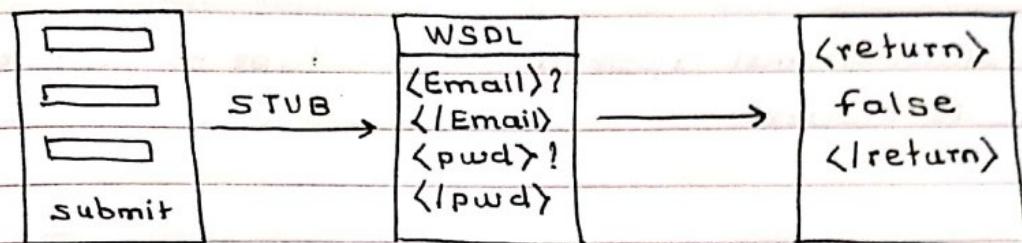
(i) Top down approach

- 1) If we have to do integration testing & we have ~~new~~ developed module but don't have next module from which we can check correctness of new module.
- 2) Then in this case we use dummy module. ^{created from}
- 3) Dummy module is ~~called as~~ STUB.
- 4) STUB is dummy program created by developer.
- 5) STUB is in XML format.
- 6) To check stub we use SOAPUI, simple object access protocol user interface.
- 7) URS for this stub is WSDL (web service description language)
- 8) Request & response is in XML language.
- 9) When we have main module but do not have sub module then For successful integration we use top down.



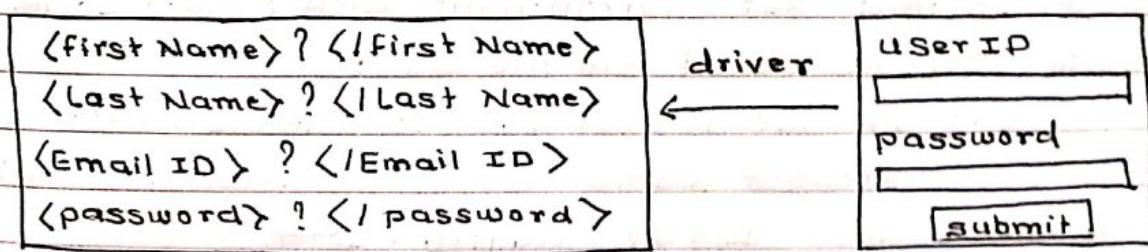
In case of invalid user name.



In case of invalid passwordIn case of user name and password are both invalid.(ii) Bottom up approach

- 1) If we have sub module but do not have main module then in that case we use bottom up approach.
- 2) To check sub module, developer create dummy main module.
- 3) Developer first create program called "Driver".
- 4) These driver programs are in XML language

Example:



(iii) Bidirectional Approach

when developer want to check functionality of main module but don't have sub module then developer uses stub program.

when developer want to check functionality of sub-module but dont have main module then developer uses driver program.

In bidirectional approach both type of testing used by developer.

→ zero level testing:

1) Aim:- To check core | Basic functionality.

2) Also known as:- Sanity Testing

Zero level testing

Build verification testing.

3) People involve: Database Analyst (DBA)
Developer Tester.

4) Working:- whenever integration test ends, zero level testing starts.

(i) Database Analyst team create a test environment

(ii) In that test environment DBA insert code.

(iii) This is called run the build.

(iv) Ex: If our project have to release in 7th month
Project Name is paytm. Then test environment
is tst07.paytm.com.

(v) After creating test environment tester do
testing & aim is that system should generate
desire o/p.

(vi) Test team got a table , in which for which process or op they have to generate answer is mentioned.

(vii) Tester do testing for :

- 1) Basic core functionality
- 2) Tab validation
- 3) Link validation
- 4) Page validation
- 5) GUI validation

1) Basic core functionality

→ In this test, tester test buttons, icons from which user can proceed for next stage, ~~in this~~

Ex: There is a submit button in sign up page then tester test that submit button, whether it works or not.

2) Tab validation

- 1) Tab validation test involve functioning of Tab's
- 2) Tabs are text boxes in which we enter values.
- 3) whenever we enter any value in tab by using on screen keyboard or physical keyboard those char, number, symbols should get entered in tab.
- 4) This functionality of tab verified validate in tab validation.

3) Link validation

- 1) Link validation is the process in which sequence of interlink pages get tested.
- 2) Sequence means if user click on flight icon then flight informative page should open, developer give link of that page to the icon so this link get validate in link validation.

4) Page validation

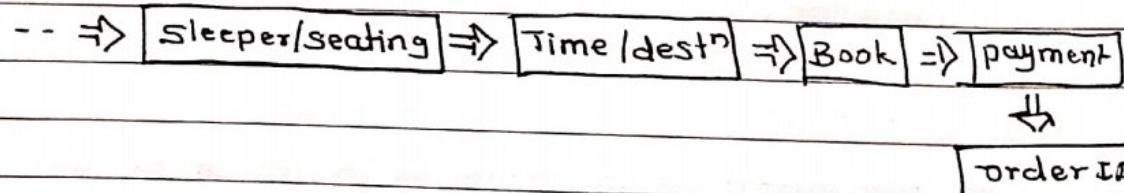
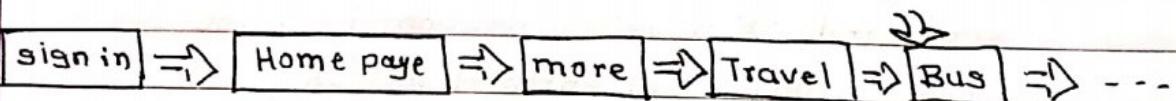
- 1) Page validation means navigation validation.
- 2) In this process when we click on next or back, pages should navigate front & back.
- 3) This testing is called as navigation testing.

5) GUI testing

- 1) This testing test the interface with which user interact directly.
- 2) In this test, tester check whether the pages are displaying correctly or not, image should be clear, No blur to occurred anywhere in page.
- 3) This validation of visualisation is called as graphical user interface testing.

Example

- Suppose we have paytm project.
- We develop module of bus ticket booking.
- We have to integrate it in old modules.
- Then developer integrate it in paytm.



- When this integration over, tester test the integrat'
- This is verification & WBT involve in it.

- Now validation starts.
- To check validation DBA first create test environment.
- In test environment DBA insert code.
- When DBA insert code tester start coding.
- For this module tester get a table in which what he have to check on which device is mentioned.
- To give proof of testing tester have to fill up table & send it to authorities.

Sleeper booking from desktop sleeper booking from mobile	Build Pass or Fail.
Seater booking from desktop seater booking from mobile	Order ID

- Then tester do basic function test, tab validation, Link validation, page validation, GUI testing.
- When this testing complete tester have to fill up table.

sleeper booking from desktop sleeper booking from mobile	Pass Fail	27777 —
seater booking from desktop seater booking from mobile.	Pass Fail	28777 —

- In this case level zero test for mobile fail & for desktop it get pass.
- So system & functionality test for desktop module start & for mobile error get locked.
- This is the critical defect & called as high priority defect.

→ Smoke testing:

- 1) Person involve: front end developer.
- 2) It is advance version of sanity testing.
- 3) Smoke test = sanity test + trouble shooting
smoke test = sanity test + package validation.

4) Package Validation:

- (i) Package is collection of object.
- (ii) If in sign up page we enter some information in tab.
- (iii) When we press submit then information get save in database.
- (iv) When user go on login page & enter credentials then if invalid error occurs even after submitting valid credential then back end dev. have to find from which package this information comes.

5) There are 3 types of projects.

Traditional	off the shelf	maintenance
Dev & test team are from same company so they provide smoke test.	Dev & test team are from diff company so they provide sanity test.	

- ~~Q.1~~ What is sanity testing?
- ~~Q.2~~ What is your first stage of functionality testing.
- ~~Q.3~~ Tell about practical implementation of sanity test.
- ~~Q.4~~ Why it is for 7 days i.e., on friday evening build moves & on mondays we are going to test it?
- ~~Q.5~~ How you are going to verify / identify the build number & where?
- ~~Q.6~~ What is VSS?
- ~~Q.7~~ How you receive the build information?
- ~~Q.8~~ Did you involve in build upgradation activities?
- ~~Q.9~~ What you perform in sanity testing?
- ~~Q.10~~ What which type of defect, tester report in sanity testing?
- ~~Q.11~~ Do you log the defect during Sanity Testing?
- ~~Q.12~~ Do you write test cases during sanity testing?
- ~~Q.13~~ How troubleshoot happens?
- ~~Q.14~~ What is difference between sanity & smoke testing!

Q.1 What is sanity testing?

- Ans:-
- 1) When developer done with integration & integration testing, DBA perform initial build in test env.
 - 2) When we i.e., tester receive initial build we start coding testing.
 - 3) First we test build for correctness and completeness of the basic core functionalities.
 - 4) This testing is called Sanity testing.
 - 5) I mean to say that we check whether the build is stable or not.
 - 6) We check whether the build is ready for test or not.
 - 7) Duration for sanity test is 4 to 8 hrs.

Q.8}

Ans:-

What is your first stage of functionality testing.

- 1) First stage of testing is sanity testing.
- 2) It is process of checking correctness & completeness of the core functionalities.
- 3) we check for stability of build.
- 4) we check that build is ready for testing or not.
- 5) Sanity testing is also known as:-
Zero level testing.
Build verification testing.
Tester acceptance testing.

Q.9}

what you perform in sanity testing?

Ans:-

Sanity testing includes 5 factors.

- 1) core functionality testing
- 2) Link validation/testing
- 3) Page validation /testing
- 4) Tab validation
- 5) GUI validation.

Q.10}

which type of defect , tester report in sanity testing?

Ans:-

- 1) In maximum cases we get environmental defect in sanity testing.
- 2) Environmental defects means system hangout, runtime error, popup or link is not working.
- 3) These defects are called as critical defects or high priority or high severity defect.

Q.11) Do you log defect during Sanity Testing?

- Ans:-
- 1) No, generally we don't log ^{every} defect during sanity testing.
 - 2) We log only critical defect in sanity testing.
 - 3) Critical error are environmental defect i.e., system hangout, runtime error, link or popup not working.
 - 4) We report defect to the developer via Email.

Q.12) Do you write test cases during Sanity Testing?

- Ans:-
- 1) No we don't write test cases during sanity testing.
 - 2) We do not require test cases, we only do basic core functionality testing.
 - 3) We are familiar with core functionality of system.
 - 4) Basically test cases has been developed at initial stage of development.

Q.3) Tell about practical implementation of sanity testing.

Q.4) Why it take 7 days i.e., on Friday evening build moves & on Mondays we are going to test it.

- Ans:-
- 1) It is very complex to move build i.e., from DIT to SIT.
 - 2) 99% on Mondays we are perform sanity testing. 1% if holiday came on Monday then we perform it on Tuesday.
 - 3) Initially project start on Monday, I mean developer start developing code from Monday for build.

- 4) They code it till Friday & on Friday evening they send build to tester i.e., DIT to SIT.
- 5) And from Monday we start testing & find out defects, if any defect occurs then we log defect & communicate with developer about that defect.
- 6) Then developer solve that code till Friday & send new build again on ~~Monday~~ Friday evening.
- 7) Generally sanity testing takes 4 to 8 hrs.

Ex:

- 1) If build no 26 send by dev. to tester.
- 2) Tester starts testing, & if any core functionality error occurs in build then tester report it to the developer.
- 3) For critical error on same day developer fix that defect & on same day he sends that new build to us.
- 4) If any other defect occurs in the build then dev. may take 2 or 3 days. & send new build to dev.
- 5) If tester finds defect in build 26 then he sends it to developer, when dev. sends new build, that build has new number i.e., 27.
- 6) Developer gives number to build for identification.
- 7) On each Friday we ask dev. or DIT, Is build deployed on SIT env ~~or~~ or IS build moved to SIT or not.

Q.5)

How you are going to verify / identify the build number & where?

Ans:- 1) We verify build number from database.

2) we go to db & write query:

select * from table-name desc;

Table name : build_version.

then we write query select * from build-version desc;

Build no	status
27	
26	

when we write query then it shows all build number.

Each build is called consolidate build.

If in 3 month we receive 12 build then master build = consolidate build 1 + --- + 12.

Q.6)

What is VSS.

Ans:- 1) VSS stand for visual source safe.

2) It is product of microsoft.

3) It is unique numbering system for build.

Q.7) How you receive the build?

Ans:-

1) ~~use~~ initially we do not receive build, initially we receive build information.

2) Developer first send mail to tester

Hi XYZ,

we kept build 27 & has link buildno@url, please find it, & start upgrading build to proceed for testing.

on friday 4PM developer send this mail to tester.

3) At 5pm on friday tester write to DBA.

Hi ABC

Plz start build upgradation activity from 26 to 27 & let us know once build upgradation completes.

4) DBA team reply to tester:

Hi XYZ

Build has been upgraded ~~to~~ from 26 to 27. Plz find valid + invalid object as attachment & let us know / inform if any invalid need to be valid. This attachment is in notepad file.

Ex Valid: delete button, cancel button

Invalid: submit.

5) we want that submit button should be valid then we write to DBA about that & they correct that & reply to tester.

6) after build finally upgrade, we start testing from monday 10 A.M.

Q.8) Did you involve in build upgradation activities?

Ans:- No, I never got chance to involve in build verification activity, my DBA team does build upgradation activities.

Q.13) How troubleshoot happens?

Ans:- Suppose there is OK button & when we click on OK button runtime error comes.

Then backend developer check that OK button belongs to which package.

Package is collection of all the interdependent objects which are grouped together & group has given a name.

So smoke test is +sanity + package validation.

Q.14) What is difference between Sanity & smoke testing?

Ans:- In sanity testing we test for:

- 1) core functn test.
- 2) Tab validation
- 3) Link Validation
- 4) Page validation
- 5) GUI validation.

In smoke testing:

smoke test = sanity + package validation / troubleshooting

- Q.15) what is STUB?
Q.16) who does integration testing.
Q.17) what is XML?
Q.18) why we use XML?
Q.19) How you update XML?
Q.20) How to send XML file?
Q.21) Difference between FTP & HTTP.
Q.22) what is web service testing?
Q.23) XML parsing.
Q.24) where you collect the XML?
Q.25) what are the defect you got in your career.?

Q.15 What is STUB.

- Ans:- 1) STUB program is use in Top down approach of integration testing.
 2) when we have main module but dont have sub module then developer write a XML program which is known as STUB.
 3) STUB is also called as "called" program.
 4) Because we call submodule using STUB.

Q.16) who does the integration testing?

- Ans:- Front end dev test functionality according to SRS/Spec
 Back end developer test functionality ~~according~~ db connectivity.

Q.17) what is XML?

- Ans:- 1) XML is Extensive markup lang.
 2) It is understood by every appn in world.
 3) we use notepad++ to update XML.
 4) XML is consist of TAGS & values

<amount> 3000

Q.18) Why we use XML?

Ans:- Because there is no chance like data corruption i.e., space or something missing.

Q.19) How you update XML?

Ans:- we update XML using notepad++.

Q.20) How to send XML file?

Ans:- we send XML file using FTP (file transfer protocol).

Q.21) Difference between FTP & HTTP?

Ans:- FTP = file transfer protocol.
use to transfer file.

HTTP = hypertext transfer protocol.

use to transfer text message.

Q.22) What is web service testing?

Ans:-
1) Data conversation between client & server via request XML & response XML.
2) we have to update XML using notepad++.
3) After updating we brows it via FTP.
4) Then we send file to server via FTP.
5) Once we send we check response from server using in db.

Q.23)

What is XML parsing.

Ans:-

Process of mapping between XML request & XML response is called XML parsing.

Q.24)

Where you collect the XML?

Ans:-

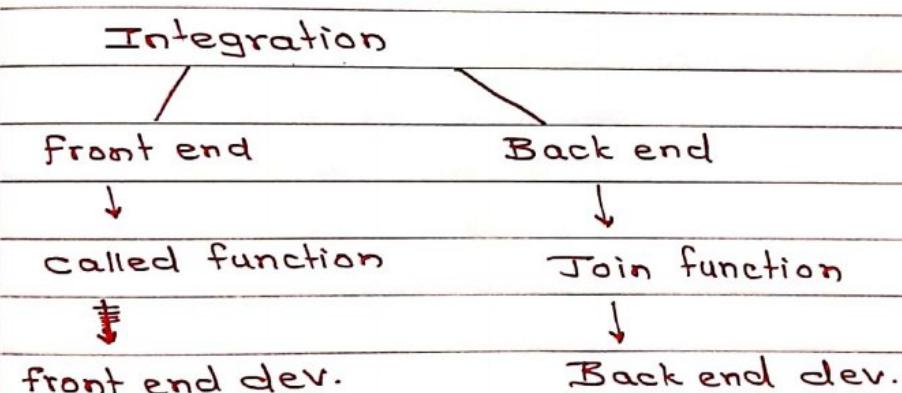
From database.

Q.25)

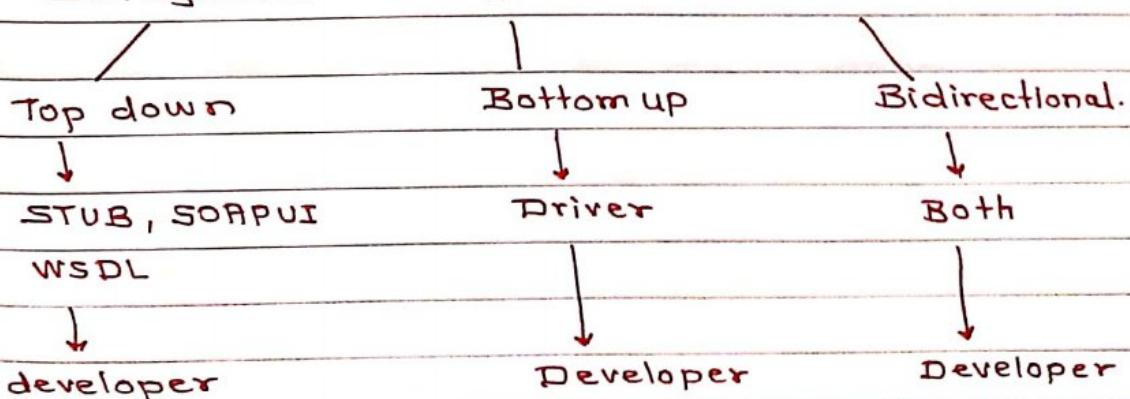
What are the defects you got in your career.

Ans:-

Request XML ↗ Response XML does not match.



Integration testing.



Sanity , O level , build stability , Test acceptance



B.B.T.



core functⁿ

Tab , page , Link , GUI validation.

smoke test



sanity



B.B.T

+

package validatⁿ



Back end developer.

- 1) In DIT phase ~~tester~~ design, code, WBT, integration integration test happens.
- 2) After DIT, sanity test happens.
- 3) After sanity system integration test happens.
- 4) System integration testing also known as system and function testing.
- 5) It is black box testing.
- 6) Tester (BBT) is involve in this.
- 7) SIT is process of checking completeness & correctness of functionality of the system as per SRS or user's requirement.
- 8) It is the process of checking internal functionality depend upon external functionality.

9) System and functional testing includes:

Functional Testing } 95 %
Usability Testing
Security Testing } 5 %
Performance Testing

(i) Functional Testing

- Functional Testing is the process of checking correctness & completeness of the functionality of the build.
- Functr testing is the process of checking internal functionality depend upon external functionality
- Functional testing includes

Functional Testing

Non-Functional
Testing

(ii) Usability Testing

- Usability testing check user friendliness of build.
- It check colors, Fonts, visuality.
- It is known as accessibility testing.
Ex. when in mobile number tab, we enter 2 digits instead of 10 digits then system show message that please enter 10 digits.

(iii) Security testing

- This check privacy of build from user's aspect.

(iv) Performance Testing

- This check speed of processing.
- Ability of system how it handle load.

(1) Functional Testing

- 1) Check completeness & correctness of system's functionality.
- 2) Process of checking internal functionality depend upon external functionality.
- 3) In this process execution of test cases are done.
- 4) It include

Functional Testing

(internal functionality)

Non Functional Testing

(External functionality)

→ Functionality Test

→ we check internal functionalities:

1) It includes :-

- (i) Behavioral coverage
- (ii) IIP domain coverage
- (iii) Error handling coverage
- (iv) Back end coverage
- (v) Service level coverage
- (vi) calculation base coverage

B
I
E
B
S
C

(i) Behavioral Coverage:-

→ In this we check property & behaviour of object.

→ Example: property of text box :- accept user ip
behaviour :- Focus & Non Focus.

property of check box :- do tick when user click.

Behaviour of checkbox :- #check / uncheck.

Property of dropdown :- to show ~~&~~ hidden list on click

Behaviour of dropdown :- ~~&~~ show / hide list.

(ii) IIP domain coverage:-

→ It check type & size of ip.

→ type means datatype of ip.

→ size means if mob num tab is there then size is 10.

→ In ip domain coverage we have to maintain boundary value Analysis & Equivalence class partition

✓
It check size of object

↓
It maintain ~~to~~ data type of object.

→ Example of BVA & ECP.

1) In text box we should accept 4 to 6 char.

BVA	ECP	
	Invalid	Valid
size min = 4 ✓	0-9	a-z
max = 6 ✓	specialsym.	A-Z
mintl = 5 ✓	space(-)	
min-1 = 3 X		
max+1 = 7 X		
max-1 = 5 ✓		

2) mobile number box should accept 10 digits.

BVA	ECP	
	valid	Invalid
size min=10 ✓	0-9	a-z
max=10 ✓		A-Z
min-1=9 X		spec.sym.
mintl=11 X		(-)
max-1=9 X		
maxtl=11 X		

3) Password should accept betw 8 to 14 digit, allow 1 cap char, 1 small char, spec.sym, & digit, No space is allowed.

BVA	ECP	
	valid	Invalid
size min = 8 ✓	0-9	(-)
max = 14 ✓	a-z	
mintl = 9 ✓	A-Z	
min-1 = 7 X	spec.sym	
maxtl = 15 X		

4) check BVA & ECP for cycle stand having 100 cycles.

BVA	ECP	
Size	Valid	Invalid
min = 1	✓	a-z
max = 100	✓	A-Z
min-1 = 0	✗	spcl.sym
min+1 = 2	✓	(-)
max+1 = 101	✗	
max-1 = 99	✓	

5) check for 3 digit number which should not start with 0 & 1.

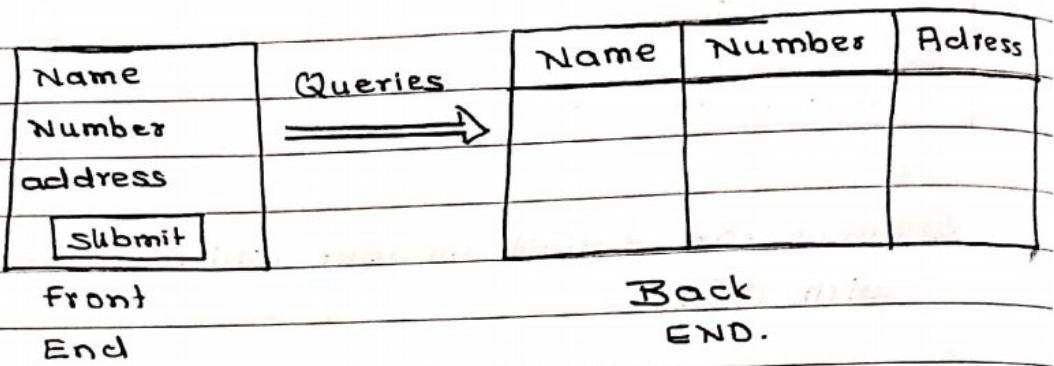
BVA	ECP	
Size	Valid	Invalid
min = 200	✓	a-z
max = 999	✓	A-Z
min-1 = 199	✗	spcl.sym
min+1 = 201	✓	(-)
max -1 = 998	✓	
max+1 = 1000	✗	

(iii) Error handling Coverage.

- 1) Error handling Coverage include checking whether system show error message or not.
- 2) If in customer requirement, mobile number should accept the 10 digits & developer developed build according to it then if customer enter 2 digits & click on OK button then system should highlight text box with red color with error message "Please Enter 10 digit mobile number".
- 3) So this is the process of validation whether system show error message or not.

(iv) Back end coverage

- 1) The back end of any SW system is db.
- 2) In back end coverage dev. check whether the entered information from user get stored in db or not.
- 3) we also check whether data get fetch from db or not.



(v) Service level coverage

- 1) In the functional flow diagram B.A. creates / ment sequence of function & modules.
- 2) This aspect of sequentiality of functional modules get tested in service level coverage.
- 3) They check working of system as per functional flow diagram.

(vi) calculation base coverage

- 1) Calculation base coverage check arithmetic operations.
- 2) Arithmetic operation includes Addition, multi, sub & div.
- 3) Ex. If we add 1 thing to cart & it is of 900Rs then total is 900 & again we add 2 thing which are 50Rs each then total is 1000 & if we remove 1 thing of 50rs then 950 is total. This get checked in this coverage.

(ii) Non-functional Testing

- 1) Process of checking external functionality.
- 2) Ex: check whether appln is running on particular OS or browser.
- 3) It is part of functional Testing & comes under system & functionality testing.
- 4) Non-Functional testing includes:
 - 1) Recovery Testing
 - 2) compatibility Testing
 - 3) Configuration Testing
 - 4) Inter system Testing
 - 5) Installation Testing
 - 6) Globalization Testing
 - 7) Sanitation testing
 - 8) Parallel Testing

1) Recovery Testing

- 1) Aim: Process of checking whether system is able to recover from abnormal situation to normal situation.
- 2) AKA : Reliability testing.
- 3) Comes under function Testing & under the S&F test.
- 4) The recovery requirements are given by customer i.e., customer can give requirement that he wants that system should recover from at that point or from start point.

Example:-

- 1) When we are accessing google's page & suddenly internet connection lost, then google shows that you are offline & whenever connection resume page which we were accessing is shown by google.

2) When we are buying something from Amazon, after entering address we get redirected to payment page & suddenly app crash & when we reopen the app we have to start again from CART.

2) Compatibility Testing

» Aim:- It is the process of checking whether the build is compatible with user expected platform.

2) User expected platforms are O.S. & Browsers.

3) AKA : SIW compatibility testing.

4) Compatibility testing has two categories:-

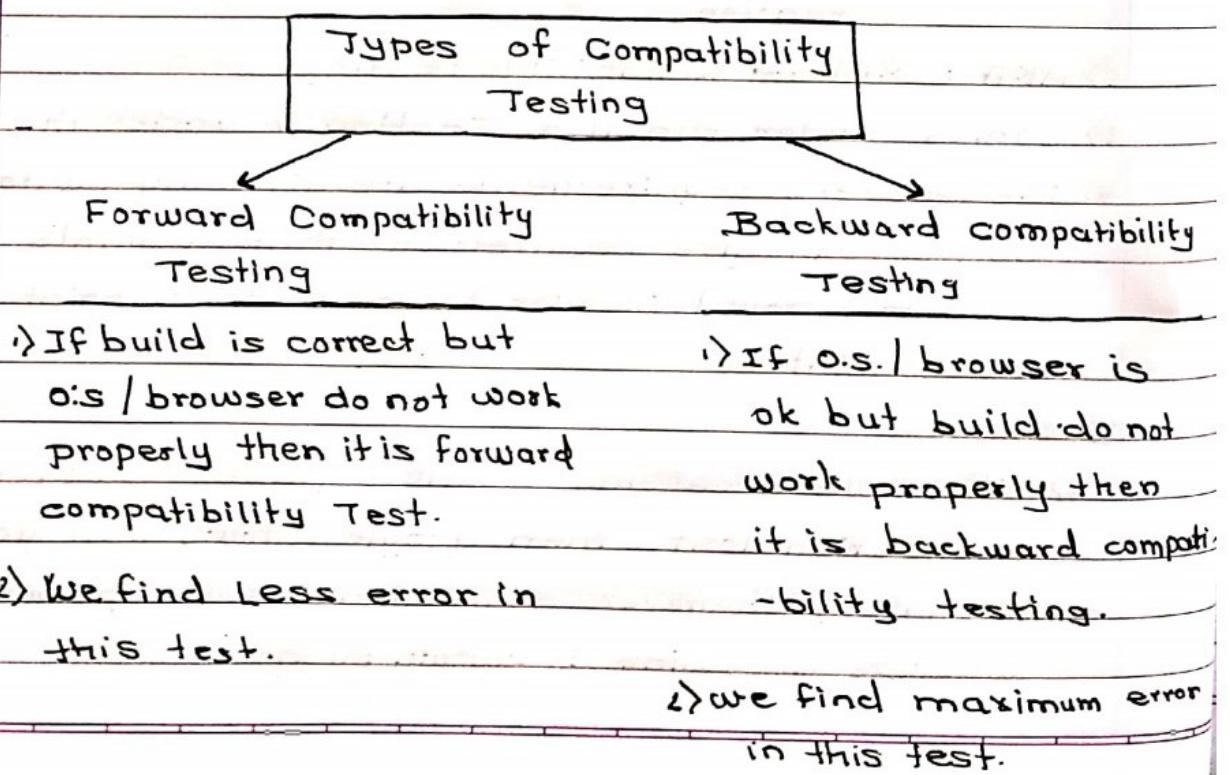
/ \

SIW compatibility (*compatibility testing)	HIW compatibility (configuration testing)
---	--

5) company : service base company.

5) This test falls under non functional testing.

6) Compatibility test have two types:-



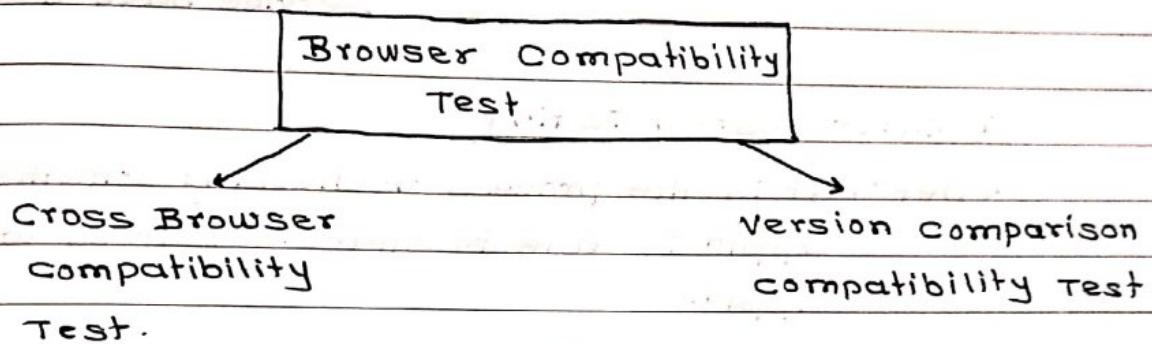
7) Compatibility test includes:-

1) O.S. compatibility testing

2) Browser compatibility testing.

8) Generally we involve in the browser compatibility testing.

9) Browser compatibility test.



1) It is the process in which tester test build on diff diff browsers like Internet Explorer, Firefox, chrome.

2) It is the process in which tester test build on diff diff version of same browser.
i.e., I.E. 16
I.E. 17.

2) To check version of browser Tester install diff diff version in vmWare.

3) Configuration Testing

- 1) Defn: It is the hardware compatibility testing.
- 2) It is the process of checking whether our appln work on diff diff Hardware or not.
- 3) It is the process of checking whether our appln support diff diff appln or not.
- 4) Company: It is used in product base company.
- 5) people involve:- Tester (Not we, we are involve in service base company)

2) Inter system Testing

- 1) Defn: It is the process of checking whether our appln is able to share resources with other appln.

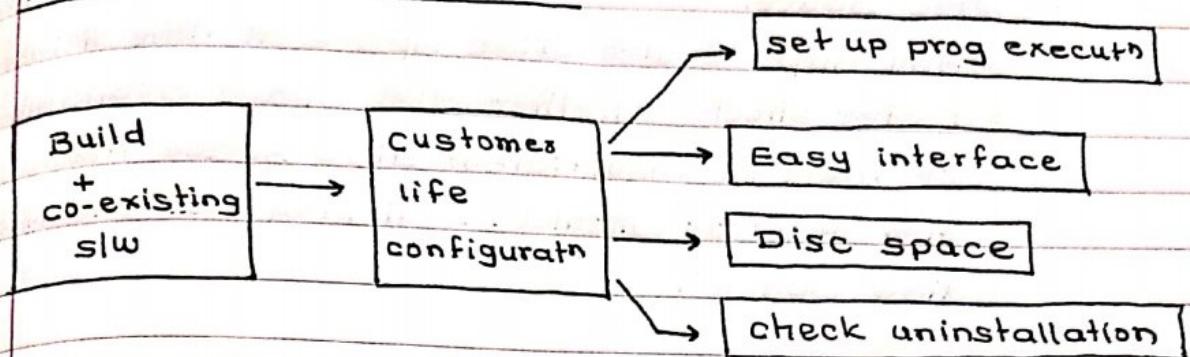
OR

It is the process of checking whether our appln shares data with other appln or not.

- 2) This data communication /communication happen through XML.
- 3) Company: service base & product base.
- 4) People involve: we are involve.
- 5) Generally banking domain companies uses this testing.

Ex: Suppose we have to make payment for JIO app number from paytm. Then paytm fetch info from JIO app, this data sharing get checked in inter system testing.

5) Installation Testing



1) Defn: It is the process of checking installation of our build with existing slw into customer life configuration.

OR

It is the process of checking installation of our build with existing slw into user expected platform.

2) company : Product base company.

3) people involve: Tester (we are not involve).

4) Use in desktop compatibility.

5) when tester install app it should create shortcut on desktop.

1) set up program execution:-

→ check whether package have all files.

→ check whether all drivers present or not.

→ check installation of EXE file.

8) Easy Interface

→ Interface should be user friendly.

→ Interface should be like that, the user can navigate through app easily.

3) Disc Space.

- Any app require disc space at time of instl'.
- tester check whether disc space available.
- If there is insufficient disc space then app sh error should be display i.e., "Insufficient disc space".

4) Check Uninstalation

- check whether appl' can uninstall from system or not.

5) Globalization Testing

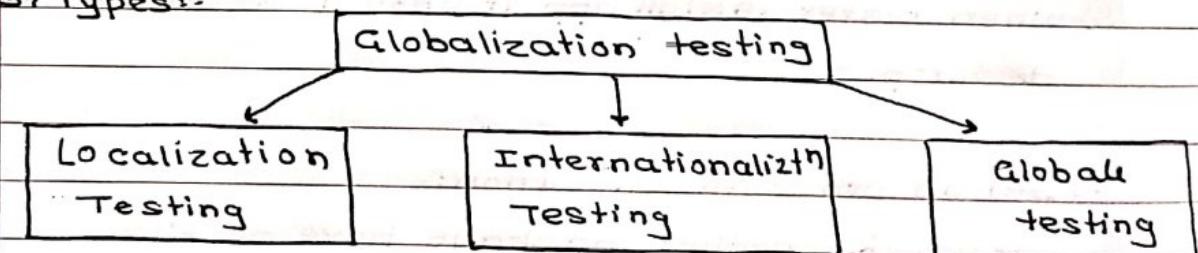
1) Defn: It is the process of checking whether application support diff diff languages or not.

2) Company :- Service base & product base.

3) People involve :- Tester (we are)

4) AKA : multilingualistic feature checking.

5) Types:-



check whether appl' support local lang. like marathi, telugu.

check whether appl' support official lang. of countries.

check whether appl' support English lang.

6) whenever user change language , lang. should get change but numbers should be in english.

7) Sanitation Testing

7) Definition:- Sanit^{aⁿ} testing is the method in which we test / check for extra features which are not mentioned in the customer requirement

1) AKA: Garbage Testing.

2) It is part of Non functional testing.

3) It is use in Product base as well as service base companies.

4) we (testers) are involve in this.

5) When we found any extra feature in the product/ build we log them as a defect & developer have to eliminate that extra feature.

6) Example for extra feature:

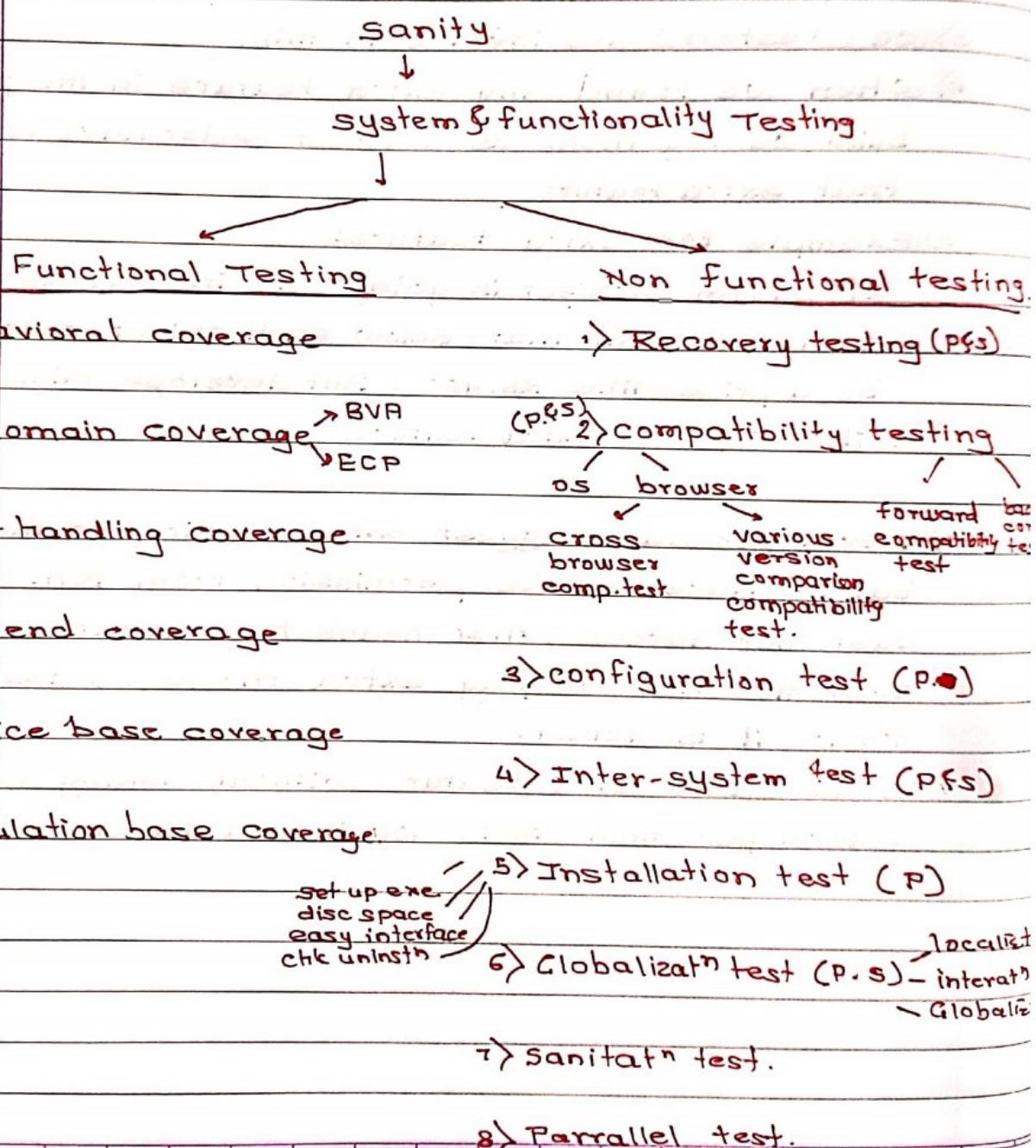
If paytm project is going on, in customer req. customer want that below each object there should be a price like Rs. 400 . But developer did Rs 400 ₹ . This "₹" is extra feature.

7) Developer can suggest extra features to the customer but he have to take permission from B.A. If then customer agrees , that means for that extra feature customer have to pay extra money. & tester do not log it as defect.

8) If dev. add extra feature without asking customer & developer then tester log it as a defect.

8) Parallel testing

- 1) definition: It is the process of checking our product with other product.
- 2) AKA: comparison testing.
- 3) Use in product base company.
- 4) Testers are involve in this test, I'm not involved in this test because I'm working in service base company.



(ii) Usability Testing

1> Definition:- It is the process of checking user friendliness of our build / screen.

2> AKA: Accessibility testing

3> It is part of BBT / system & funct. Testing.

4> For functional testing we make test cases & Execute them for functional testing.

5> For usability testing B.A. make test cases as per user's aspect. like design, colors.

6> To do testing we get 2 folders:-

1> Functional Testing.

2> Accessibility Testing.

7> To do usability testing we use following tools:-

WAT - web accessing toolbar

~~NVDA~~

8> In usability testing we do testing on new modules only.

9> Example for user friendliness:-

In yahoo mail we check there is signup tab at upper corner & whole page is filled with ads. so it is not user friendly.

In gmail sign up & sign in is at middle of page & no ads so it is user friendly.

10> In usability we check for system should take less number of events to complete task or easy validation.

Example:- If user click on submit button then app should immediately open next page quickly.

ii) For accessibility testing B.A. write test cases like:-

Signup	sign in
Name	<input type="text"/>
Surname	<input type="text"/>
mob Num	<input type="text"/>
Pwd	<input type="text"/>
<input type="button" value="submit"/>	

when we click on signup button it should open signup, if we click on S.I. then it should open sign in page. when we click on Name, Name tab should focus.

iii) Execution of these test cases ~~doe~~ happen in WAT.

iv) WAT shows database tables.

v) Two types of usability testing:

GUI

Testing

manual

support testing.

1 GUI Testing

→ Easy of use

- ↳ on one click next action should happen immediately
- ↳ For blind person, if he/she click on any tab, system should give voice feedback.
- ↳ This talk back testing happen in NVDA tool.

→ Speed of processing

- ↳ How quick ^{appin} user respond to user's action.

⑥ manual Support Testing

- 1) Definition:- it is the process of checking context sensitiveness to the user's manual input.
- 2) AKA: Regular Expression Testing.
- 3) which module or Tabs respond to user's manual action is as mentioned in user's request.

Example:-

if there is Uber App in which we have to enter source and destination.

Then when we click P in source tab, it shows all the location suggestion starting with P.

- 4) Regular expression are, sometime more than one object have same logical name or regular expression physical description. This is called regular expression.

3) Security Testing

- 1) Definition: Security Testing is a process of checking privacy related to user operation.

- 2) People Involve:- Tester, developer.

3) Types:-

1) Authorisation:-

- ↳ Process of checking whether person is authorised or not.

- ↳ Authorised Person is registered person.

2) Access control:-

- ↳ Process of checking whether authorised person has permission to access specific operation.

3) Encryption & Decryption:-

4) Performance Testing

- 1) Performance Testing is the process of checking speed of processing of our build.
- 2) People involve :- Performance test engineer.
- 3) Tool use :- load runner tool.
This tool create simulation like many users are using system at a time.
- 4) We are not involve in performance testing, we only give K.T. to performance tester.

1) Re-Testing

1) Definition:- Re-testing is the method of Re-executing same build / application with multiple test cases data.

2) People involve: Regression Tester (we are involve)

3) Re testing happen 2 times:-

1) Before we log defect we do re-testing

2) After developer solve defect we do re-testing.

4) Re-testing is important because if we doing test according to test case and we check it using only one test data, then that should be bad defect.

5) A good defect is that defect which occurs repeatedly on multiple test data.

6) When we check system for first time then we do retesting to ensure that this defect is good defect.

7) When we do retesting when def solve defect we check that system is working well for that test case & on same test data.

8) Process:- we do retesting & we got a defect.



we log defect & give remark to that test case as "fail"



we got ID to that defect.



developer solve that defect.



we again re-test system to ensure that it works well.

2) Regression Testing

↳ definition:- Regression testing is the process in which we are testing ~~systems~~ ^{newly corrected} module to ensure that they are working well & to check their impact on working of other modules.

3) People involve:- Regression Tester (we are)

4) Regression testing happen at two times:
during system Integration testing.
After completing SIT & UAT.

5) Aim of regression testing is to check whether newly corrected system is working well & impact of this system on other modules.

6) what we actually do in regression testing

(i) If any new scenario get added then instead of doing SIT on that we do regression testing.

↳ Scenario means to add +91 before mobile number is scenario , instead of doing SIT on that module we do regression testing on that.

(ii) Check all the fail test cases

↳ we have to ensure that newly ~~add~~ corrected system is working well . for that we have to run those fail test cases , so in regression test we check

III High priority test cases

- ↳ when we move from SIT to UAT we first test only high priority test cases.
- ↳ High priority test cases are those test cases which are related to main functionality.
- ↳ we can check medium & low priority test cases if we have time ^{limit} left.

→ Extra Points ←

1) How many test teams are involve in your project?

Ans:- There are 3 test teams in our project.

1st SIT team

2nd UAT team

3rd Regression team.

2) In which test scenarios we are involved?

Ans:- 1) Test case design

2) Test case execution

3) Log defects

4) SIT

5) Re-test.

6) Regression

7) UAT

3) User Acceptance Testing (UAT)

- 1) Definition :- UAT is the process of collecting feedback from customer.
- 2) People involved :- Test Team
Development Team
Customer.
- 3) AKA: End to end testing.
- 4) Tools used:
 - 1) Q-messenger for desktop sharing.
 - 2) AVAS & Bharat mantri to search sessions.
 - 3) JEERA for ticketing & internal communication.

5) main points:-

- 1) UAT starts after SIT.
- 2) UAT starts after retesting & Regression testing.
- 3)



- 4) User decides from total user stories, how many user stories he want to test from tester.
- 5) Tester then check all test cases related to that user stories.
- 6) Customer can change test data.
- 7) and it is on customer, whether to send system for production or not.

6) Process

① Test team share desktop betn them, user and developer using Q messenger



② Test team execute test cases (scenario) related to customer selected user stories. and tester enter test them by entering test data selected by him and (or) test data given by customer.



③ Test team then do whole functionality checking like front end test , back end test.



④ When test team complete one test case then before proceeding towards the next test case they have to clear cache , coockies , history.



⑤ When tester completes one test case, tool gives a session ID, Session ID contain all the action perform at that session. Tester have to copy that session ID.

* If tester doesn't get session ID, then they contact to DBA (env. team) for ^{log file} session ID, they send request by generating ticket in JEEERA tool. Then DBA (env team) give tester a log file. *



⑥ After copying session ID, tester have to search according to session ID in "Bharat mantri" tool or in "AVAS" tool. whenever search get completed tester will get a log file.

* log file is .txt format file.

Log file has front end operation & Back end operations.

* This log file contains HTML coding *.



⑦ Tester send this log file to dev. team. using Q messenger.



⑧ Then front end dev. check whether front end match with back end db or not. both are mention in log file. ~~Front end~~ dev have to just check own code i.e., i/p & o/p.



⑨ After front end, back end dev. check database using queries. & fetch data to see whether really it is stored or not.



⑩ When all this process get successful result, then system is ready for production. But to whether send it to production or not is get decided by the customer.

Example : Test paytm.com in chrome.

1) we share our desktop with customer & dev. using Q-messenger.



2) Then we run tst07.paytm.com in chrome.



3) we check test cases , selected by user in chrome.



4) After completing we collect session ID from chrome.

→ in case if we did not get ID , then through JEERA we raise ticket for log file, then DBA team gives us log file.



5) Then we enter ID in "AVAS" or "Bharat mantri." & those tool give us log file.



6) Then we share log file with dev. through Q.mesenger.



7) Dev. then chk log file , whether front & back functionality is matching or not.

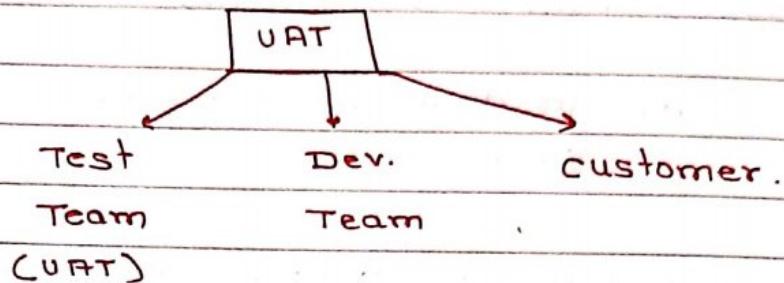


8) Then back end dev. send query & see db.



9) when all test cases get pass , then build can proceed to production if customer wants.

* Important Points.



- cust. decide which story to execute.
- cust. decide whether build go to productn or not.
- Tools used: Q-messenger

↳ To share desktop

↳ To create groups

↳ To ~~creat~~ send files

AVAS → To search session ID

Bharat mantri → To search session ID.

JEERRA → internal comm betn test,
dev, dba team by rising
ticket.

→ log file is in .txt form.

→ log file contains html code.

→ main defect we got are like:

1) when we enter 1@gmail.com it gets stored.

2) when we enter 2@gmail.com then in db it only shows 2@gmail.com this type of defect we get most of time.

→ when we got defect we log them, get defect ID then dev. solve it, tester & regression tester do retest & regression test on them & after that ~~==~~ UAT starts.

→ User acceptance testing has 2 ~~per~~ types.

① Alpha Testing

② Beta Testing

① Alpha Testing

- 1) Alpha testing happen for web base applications.
- 2) Alpha testing conducted in controlled environment.
i.e., it is conducted in front of tested, dev & customer.
- 3) Developer and tester are more involved in the alpha testing.
- 4) Real customers are involved like HDFC, IDBI, ICCI.

② Beta testing

- 1) Beta testing happen in product base company.
- 2) Beta testing conducted in uncontrolled environment.
uncontrolled environment means within organisation, tester do testing.
- 3) Developer and tester are less involved.
- 4) customers are microsoft, Rupay, master card.

- Important Points. -

classmate

Date _____
Page _____

- 1) Error :- mistake in the program is called as error.
- 2) Defect :- When tester found mistake ~~or~~ i.e., error then it is called as defect.
- 3) Bug :- when developer accept that it is actual ~~bug~~ ^{defect} then it is called as Bug.
- 4) Issue :- when developer found difficulty to solve bug then it is called as Issue.
- 5) Do you involve in UAT?
Ans:- Yes, I was involve in UAT but not in same project for diff project I involved in UAT where I conduct client interaction.
- 6) We can involve in only one test team for same project like when we are in SIT then we can't be a part of Regression & UAT.
- 7) In B Testing when developer & Tester complete their work then product is send to the diff. diff & tester to collect feedback. The product send to testers ~~is~~ is called as Beta Version.

1) HOT FIX or Production Issue.

1) Definition -- when after production customer found any defect in the product, then that is called as Hot fix or production issue.

2) Tools used :- HPLM (for tester & dev communication)
JEERA (for internal commn in org.)

3) Points:-

- 1) After UAT regression test happens.
- 2) After regression production starts.
- 3) End user involve in production.
- 4) Regression test engineer deploy product (www.page.com)
- 5) After deployment customer use it.
- 6) When customer find defect then it is called as ~~HOTFIX~~ or production issue.
- 7) When customer found defect then he contact to customer care.
- 8) Then customer care people raise a ticket to assign problem to tech support.
- 9) When customer care people can not solve it then they raise ticket to transfer it to product team in which dev. & tester involved.
- 10) Then B.A. assign it to the team which includes dev & tester, that team is called as hot fix team.
- 11) Then dev solve that issue & after dev solve issue we start testing.
- 12) Dev & tester communicate through HPLM.
- 13) Internal commn betn c.c., tech support, B.A. is through JEERA.

→ Testing Terminologies.

- 1) Monkey Testing (Speed Testing)
- 2) Exploratory Testing.
- 3) Ad-hoc testing.

1) Monkey Testing (Speed Testing)

- When we have lot of test cases to execute but don't have enough time to do execution of test cases, that time we use monkey testing.
- 2) This situation arises when dev. find difficulty to solve bug, then that time dev. take extra time then that bug is called blocker defect.
- 3) When blocker defect comes, that time we do monkey testing.
- 4) We test basic functionality with respect to the customer's requirement.
- 5) We check only high priority test cases.
- 6) If time limit permits then we check medium & low priority test cases.
- 7) We concentrate more on +ve test cases & less on -ve test cases.

2) Exploratory Testing

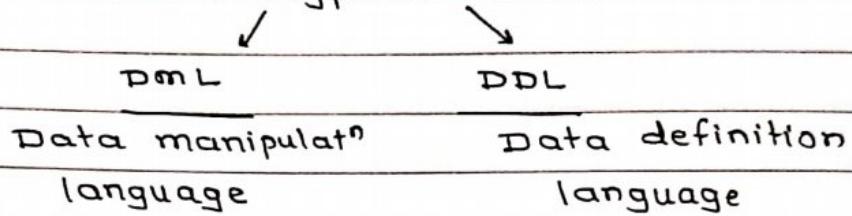
- 1) When we are not aware of the test application but we have test case, test case data, SRS that time we conduct Exploratory testing.
- 2) In exploratory testing we have test case, data, SRS so we do step by step execution of test cases.

3) Ad-hoc Testing

When we are aware of Application, we have test cases but do not have test case data, that time we do Ad-hoc testing.

SQL.

- 1) In SQL we can create db, alter drop db.
- 2) Every appn need db.
- 3) To search / fetch data we use SQL.
- 4) SQL has two types.



- 5) database contains multiple tables.

- 6) Table

	F.N	
	L.N	
	mob	
	Email	
	city	
	submit	

sign-up

F.N.	L.N.	mob	Email	city
Rahul	Gandhi	1111	1@gmail.com	pune
Arvind	Kejriwal	2222	2@gmail.com	mumbai
Anna	Hajare	3333	3@gmail.com	pune
Sharad	Pawar	4444	4@gmail.com	Baramati
Sonia	Gandhi	5555	5@gmail.com	pune

7) DML

Data manipulation language use to perform.

select , delete , update , insert into .

8) DDL

Data definition language use to perform ,

create db | table ,

alter db | table ,

Drop db | table ,

truncate db | table .

1) Select Query

It is a sql statement use to fetch data | column's data or whole table data from database | table .

Syntax .

→ To fetch whole contains

— Select * from table-name ;

Ex. Select * from signup

It shows whole table data .

2) To fetch particular column data

select column-name from table-name ;

Ex. select F.N , L.N from signup ;

F.N.	L.N.
Rahul	Gandhi
Arvind	Keshriwal
Anna	Hajare
Sharad	Pawar
Sonu	- JI

2) Distinct

It is an SQL statement used to show unique values from particular column.

Syntax

`Select distinct column-name from table-name;`

Example:

`Select distinct L.N. from signup;`

O/P

L.N.
Gandhi
Kesriwal.
Hajare
Pawar

`Select count (distinct) L.N. from signup;`

O/P

4

3) Order by

It is SQL statement used to make order of data in ascending & descending form, from selected column.

Syntax

1) For Ascending

`Select * from table-name order by column-name;`

Example: select * from sign-up order by F.N.;

o/p	F.N.
	Anna
	Arvind
	Rahul
	Sharad
	Soniya

2) descending

select * from table-name order by column-name desc;

Example: select * from signup order by F.N. desc;

F.N.
Soniya
Sharad
Rahul
Arvind
Anna

4) TOP

It is a SQL statement used to select top values.

Syntax:-

Select top value from table-name;

Example:-

Select top 3 from signup;

F.N.	L.N	mobNo	Email	city
Rahul	Gandhi	1111	1@gmail.com	Pune
Arvind	Kejriwal	2222	2@gmail.com	Mumbai
Anna	Hajare	3333	3@gmail.com	Pune

5) where clause

It is a SQL statement used to extract those records which fulfil condition.

Syntax:-

Select * from table-name where column-name = <> !:
value;

Example:-

1) Select * from sign-up where city = 'pune' ;

F.N.	L.N.	mob.no	Email	city
Rahul	Gandhi	1111	1@gmail.com	pune
Anna	Hajare	3333	3@gmail	pune
Soniya	Gandhi	5555	5@gmail.com	pune

2) select Email from signup where city = 'pune' ;

Email	city
1@gmail.com	pune
3@gmail.com	pune
5@gmail.com	pune

6) AND

It is an SQL statement used to select which select 2 records when both satisfy given condition.

Syntax:

```
Select * from table name where column = value  
AND column = value;
```

Example:-

```
Select * from signup where F.N = 'Rahul' And  
L.N = 'Gandhi';
```

F.N.	L.N.	mob	Email	city
Rahul	Gandhi	1111	1@gmail.com	pune

```
Select F.N., L.N., mob from signup where  
F.N. = 'Rahul' AND L.N. = 'Gandhi';
```

F.N.	L.N.	mob.
Rahul	Gandhi	1111

→ OR

It is a SQL statement used to select records when both conditions are true or one condition is true.

Syntax:

Select * from table-name where column = value or column
= value;

Example:-

Select * from table-name signup where F.N = 'Rahul' or
L.N = 'Gandhi';

F.N.	L.N.	mob	Email	city
Rahul	Gandhi	1111	1@gmail.com	Pune
Soniya	Gandhi	5555	5@gmail.com	Pune

Select count (F.N., L.N.) from table name where
F.N = 'Rahul' or L.N = 'Gandhi';

Q1P = 2

8) Like

It is SQL statement used to search specific condition data in column.

Syntax:-

1) For ending with specific alphabet.

Select * from table-name where ~~F.N.~~ column like '% alphabet';

2) For starting with specific alphabet.

Select * from table-name where column like 'alphabet%';

3) For starting with alphabets.

Select * from table-name where column like '[ABC]%' ;

4) For ending with alphabets

Select * from table-name where column like '%[ABC]';

Example:-

Select ~~F.N.~~ from signup where F.N. Like 'R%' ;
L.N.

F.N.	L.N.
Rahul	Gandhi

Select L.N. from signup where F.N. Like '%.nd';

F.N.	L.N.
Arvind	Kejriwal

Select L.N. from signup where F.N. Like 'Rati%' ;

F.N.	L.N.
Rahul	Gandhi

9) Insert into

It is a SQL statement which is used to insert records in a table.

Syntax:-

INSERT INTO Table-name (column1, column2, column3)
VALUE (values , value2 , value3);

To insert value in all columns.

INSERT INTO Table-name **value** (value1, value2, values);

Example

INSERT INTO signup **value** ('~~Bear~~' , 'Grills' , 6666 ,
'6@gmail.com' , 'Nashik');

o/p

F.N.	L.N.	mob	Email	city
# - 11 -				
Bear	Grills	6666	6@gmail.com	Nashik

10) update

It is a SQL statement used to update / change records or used to update old records with new records.

Syntax:-

```
update tablename set column=value, column=value  
where column = value ;
```

Example

```
update signup set mob = 7777 , Email = '7@gmail.com'  
where name = 'bear' ;
```

O/P

F.N.	L.N	mob	Email	city
bear	Grills	7777	7@gmail.com	Nashik

ii) Delete

It is a SQL command used to delete specific record or all records from table.

Syntax:-

1) To delete all records:-

`delete from table-name;`

2) To delete specific record:-

`delete from table-name where column-name=value;`

Example

1) `delete from signup;`

→ it will delete all records

2) `delete from signup where F.N. = 'Rahul';`

→ It will delete a row containing F.N. Rahul.

delete	Drop	Truncate
1) one by one deletion of data	1) Table with structure get drop	1) Structure remain but data get deleted.

Roll back
is possible.

18) Wild Card

It is a SQL statement use to find data in column.

Syntax :

1) Ending with alphabet.

Select * from table-name where column like '-alphabet'

2) Starting with alphabet.

Select * from table-name where column like 'alphabet_'

Example :-

Select * from signup where F.N. like 'R----';

F.N.	L.N.	mob.	Email	city
Rahul	Gandhi	1111	1@gmail.com	pune

Select * from signup where F.N. like '----l';

F.N.	L.N.	mob	Email	city
Rahul	Gandhi	1111	1@gmail.com	pune

13) Select Into

It is SQL statement use to select data from table & copy data in another table.

Syntax:-

`Select * into new-table-name from old-table-name;`

Example:-

`Select * into login from signup;`

→ It copy all data from signup to login.

Syntax:-

`Select * into newtableName [IN newdbName] from
old-table-name;`

Example:-

`Select * into login [IN Prasad] from signup;`

14} IN

It is a SQL statement use to select those records which specify in query.

Syntax:-

```
select * from table-name where column-name
IN (value, value ,value);
```

Example:-

```
Select * from Signup where F.N. IN ('Rahul', 'Soniya');
```

F.N.	L.N.	mob	Email	City
Rahul	Gandhi	1111	1@gmail.com	Pune
Soniya	Gandhi	6666	6@gmail.com	Pune.

15} Between

It is a SQL statement use to select value betn range which specified.

Syntax:-

```
select * from table-name where column-name
between value AND value ;
```

Example:-

```
Select * from Signup where Email between
'1@gmail.com' AND '4@gmail.com';
```

F.N.	L.N.	mob	Email	city
Rahul	Gandhi	1111	1@gmail.com	Pune
Arvind	Kedriwal	2222	2@gmail.com	Mumbai
Anna	Hajare	3333	3@gmail.com	Pune
Sharad	Pawar	4444	4@gmail.com	Baramati

16) Alias

It is a SQL statement use to change column or table name temporarily , without changing in database.

Syntax:-

```
Select old-column-name as new-column-name from  
old table-name as New-table-name ;
```

Example

```
Select F.N as Name from signup as registration;
```

17) Union

It is a SQL statement use to show same named column name as one from diff tables.

```
Select common-column-name from table name union  
Select common-column-name from table name ;
```

This will show only unique value.

To Show all unique , duplicate value we use union all.

```
Select common-column-name from table_name union all  
Select common-column-name from table_name ;
```

18) CREATE TABLE

It is a SQL statement used to create a new table in database.

Syntax

```
CREATE TABLE Table-name (  
    column-name datatype NULL/NOT-NULL,  
    column-name data-type NULL/NOT.NULL,  
    ;  
);
```

Example

```
CREATE TABLE SIGN UP (  
    ID int(100) NOT NULL,  
    Name varchar(255) NOT NULL,  
    Salary int(100) NOT NULL,  
    Grade varchar(255) NOT NULL);
```

Output

ID	Name	Salary	Grade
----	------	--------	-------

Q) Constraints

SQL constraints are used to specify rules for data in a table.

Constraints are

- i) NOT NULL
- ii) NULL
- iii) UNIQUE KEY
- iv) PRIMARY KEY
- v) Foreign key
- vi) Default
- vii) Check

① NOT NULL / NULL

```
CREATE TABLE SIGNUP (
    ID      int (100) NOT NULL,
    Name    varchar (256) NOT NULL,
    Salary  int (100) NULL,
    GRADE   varchar (256) NULL );
```

Output

ID	NAME	Salary	Grade
----	------	--------	-------

(ii) UNIQUE

UNIQUE is a SQL constraint which ensures that all the values in a column are unique / different.

Syntax

```
CREATE TABLE TABLE-Name (
    column-name datatype NULL/NOTNULL UNIQUE,
    column-name datatype NULL/NOTNULL );
```

Example

```
CREATE Table signup (
    ID int(100) NOTNULL UNIQUE,
    Name varchar(256) NOTNULL ,
    Salary int(100) NOTNULL );
```

(iv) PRIMARY

PRIMARY KEY is SQL constraint which shows all unique values & PRIMARY key can not be NULL.

Syntax

```
CREATE TABLE Table-Name (
    ID Datatype NULL/NOTNULL ,
    PRIMARY KEY (column-name));
```

Example

```
CREATE TABLE sign-up (
    ID int(100) NOTNULL ,
    Name varchar(256) NOTNULL,
    PRIMARY KEY (ID));
```

(v) Foreign Key

A foreign key is SQL constraint used to link two tables together.

Syntax

```
CREATE TABLE Table-Name (
    Column-name datatype NOT NULL / NULL,
    column-Name datatype NOT NULL / NULL,
    column-Name datatype NOT NULL / NULL,
    Foreign Key (column-Name));
```

Example

```
CREATE Table Signup (
    ID int(100) NOT NULL NOT NULL,
    Name varchar(256) NOT NULL,
    Salary int(100) NOT NULL,
    Foreign Key (Name));
```

(vi) Check

Check is SQL constraint used to limit value entries in a particular column.

Syntax

```
CREATE TABLE Table-Name(
    column datatype NOT NULL / NULL,
    column datatype NOT NULL / NULL,
    check (column >= ! = value));
```

Example

```
CREATE TABLE Signup (
    ID int(100) NOT NULL,
    AGE int(100) NOT NULL,
    check (AGE >= 18));
```

(vii) Default

Default is SQL constraint use to set default value in particular column, when ~~the~~ No value is entered in column.

Syntax

```
CREATE Table table-name (  
column-name datatype NOT NULL Default 'value');
```

Example

```
CREATE Table signup (  
ID int (100) NOT NULL,  
Name varchar (256) NOT NULL Default 'Enter-Name',  
Salary int (100) NOT NULL Default 'Enter-Salary',  
PRIMARY KEY (ID));
```

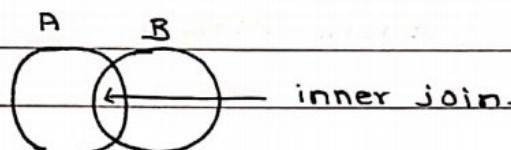
14) SQL Joins

Joins are SQL statements used to join two or more tables based on primary key and foreign key.

1) Types of join

- 1) inner join
- 2) Left join
- 3) Right join
- 4) Full join

1) Inner Join



Inner join is a SQL statement which is used to show records in columns which are related to common values.

Person

P-ID	Name	city	Education
1	Prasad	Pune	B.E. .m.tech
2	Gaurav	Nashik	B.E. .
3	Taral	Umrao	SSC

orders

Order ID	Order No	RID
1	12345	1
2	56789	2
3	98765	1
4	44444	2
5	72727	1 4
6	55555	

Syntax

Select TableName . ColumnName , TableName . ColumnName
From Table1Name INNER JOIN Table2Name
on Table1 . commonColumnName = Table2 . commonColumnName ;

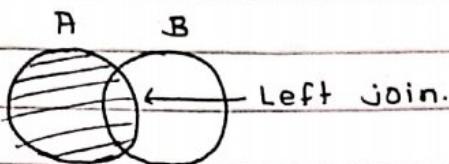
Example

Select person . Name , person . city , orders . orderID ,
orders . OrderNo
from Person Inner Join Orders
on Person . P-ID = orders . PID ;

Output

orderID	Name	city	order ID	orderNo
1	Prasad	Pune	1	12345
	Prasad	Pune	3	97625
	prasad	Pune	5	72727
	Gaurav	Nashik	2	56789
	Gaurav	Nashik	4	44444

ii) Left Join



Left join is a SQL statement which writes all column from left & common/matching column from right table.

↓
all columns from left include those column which do not have common records with rig table.

Syntax

```
Select Table1Name · column Name , Table2 Name · column Name  
from Table1 Name Left Join Table2 Name  
on Table1 Name · common column Name = Table2 Name ·  
common column Name;
```

Example

```
Select Person · Name, Person · city, Person · Education  
Orders · order-ID, orders · order No  
from Person left join Orders  
on Person · P-ID = Orders · P-ID ;
```

Output

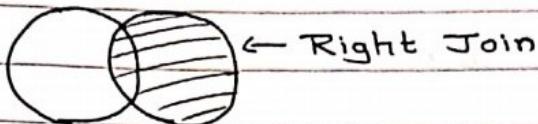
Name	City	Education	order-ID	order-No
Prasad	Pune	B.E M.Tech	1	12345
Prasad	pune	M.Tech	3	97625
Prasad	Pune	M.Tech	5	72727
Gaurav	Nashik	B.E.	2	56789
Gaurav	Nashik	B.E.	4	44444
Taral	umrana	SSC	-	

wrote all the values from left table

wrote only common values

iii) Right Join

A B.



Right join is a SQL statement which writes the all values from right table and only common values from left table.

Syntax

```
select table1 Name . column Name , Table2 Name . column Name
from Table1 Name Right Join Table2 Name
on Table1 Name . common column Name = Table2 Name . common
column . Name ;
```

Example

```
Select person . Name , person . city , person . Education
orders . order - ID , orders . order NO
from person right Join orders
on person . P - ID = orders . P - ID ;
```

Output

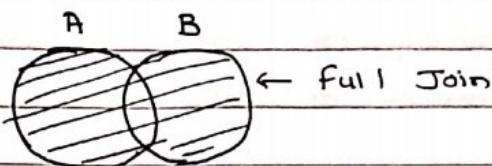
Name	city	Education	order ID	order No
prasad	Pune	m.Tech	1	12345
prasad	Pune	m.Tech	3	97625
prasad	Pune	m.Tech	5	78727
Gaurav	Nashik	B.E.	2	56789
Gaurav	Nashik	B.E.	4	44444
			6	66666

write ~~at~~ common value

from left

write all the value

from right.

(iv) Full Join

Full Join is a SQL statement which writes all the columns from left and right.

Syntax

```
select table1Name.columnName , Table2Name.columnName
from Table1Name Right Join Table2Name
on Table1Name.common columnName
= Table2Name.common columnName;
```

Example

```
Select person.Name , person.city , orders.order-ID ,
orders.order-No
From person Full Join orders
on person.P-ID = orders.P-ID ;
```

Output

Name	city	order-ID	order-No
Prasad	Pune	1	12345
Prasad	Pune	3	97625
Prasad	Pune	5	72727
Gaurav	Nashik	2	56789
Gaurav	Nashik	4	44444
Taral	umrana	6	66666

write all value
from left table

write all value from
right table.

15) Aggregate Function

(1) Alter

Alter is a aggregate function use to add, drop, modify columns in table.

→ 1) Add New Column

Syntax

Alter table table-Name

Add column-Name datatype;

Example

Alter table Signup

Add address varchar(256);

→ 2) Drop Column

Syntax

Alter table table-Name

Drop column column-Name;

Example

Alter table Signup

Drop address column address;

→ 3) modify column

Syntax

Alter table table-Name

Alter column column-name datatype;

Example

Alter table Signup

Alter column •Grade varchar(256);

* (ii) min()

min() is a SQL statement which is used to select minimum value from column.

Syntax

select min (column-name) from table-name ;

* (iii) max()

max() is a SQL statement used to select maximum value from column.

Syntax

select max (column-name) from table-name ;

* (iv) count()

count() is a SQL statement used to select count of values present in columns (No. of rows)

Syntax

select count (column-name) from table-name ;

* (v) Avg()

Avg() is a SQL function used to set/show average of values present in particular column.

Syntax

select column (column-name) from table-name ;

* sum

sum() is SQL aggregate function use to make sum of values in column.

Syntax

Select sum (column-name) from table-name;

* First()

First() is SQL aggregate function use to select first value in column.

Syntax

Select first (column-name) from table-name;

* Last()

Last () is SQL aggregate function use to select last value from particular column.

Syntax

Select last (column-name) from table-name;

16) Group By

It is a SQL statement use to group the values & use with combination of aggregate function.

Syntax

```
Select agg.function (column-name) from table name  
group by column-name;
```

```
Select sum(salary) from signup group by ID;
```

17) Having

It is a SQL statement use to select records which satisfy the given condition & it is use with aggregate function.

Syntax

```
Select aggfunction (column-name) from table-name  
Group by column name Having agg funct (columnname)  
>= != value;
```

```
Select sum(salary) from table signup group by  
ID having sum(salary) >= 25,000;
```

18) To show 2nd max salary

Select max (salary) from T.N. where

Salary NOT IN (select max (salary) from tableName)

19) Case statement

Case st. are SQL statement use to write value for other & instead of value present in table.

Syntax:-

```
Select case column-name  
when column-name = value then value .  
when column-name = value then value  
Else value END as column-name  
from tableName.
```

Example

Gender	write female instead of male
male	male instead of female
female	

Select case gender

when gender = 'male' then 'female'

when gender = 'female' then 'male'

Else 'other' End as Gender

from Signup.

city.

write maharashtra instead of Nashik, pune,

write Gujarat instead of Ahemadabad, Gandhinagar.

Select case city

when city in ('Nashik', 'pune') then 'maharashtra'

when city in ('Ahmedabad', 'Gandhinagar') then Gujarat

Else 'India' End as city

from signup.

20) View

View is SQL statement which consist of virtual table base on result set of SQL statement.

Syntax to create view

CREATE VIEW view-name AS (SQL QUERY);

Syntax to show view

Select * from [view-name];

CREATE VIEW Grade AS (Select count (distinct) from
signup);

Select * from [Grade];

1) Priority and Severity

- 1) At the time logging defect we have to decide priority & severity of it.

2) Priority

Priority is importance of defect with respect to customer's requirement.

3) Severity

Severity is seriousness of defect with respect to functionality.

- 4) Without solving these defect we can not move to the next stage.

Example

- 1) If paytm app do not have symbol then.

High Priority

Low Severity → bcz it do not affect working

- 2) If submit button do not work.

High priority

High severity

- 3) If submit button work properly but color is diff.

High Priority

Low severity → bcz it do not affect working