WELCOME TO

I- RISE SOFTWARE TRAINING INSTITUTE

'Ambition is the path to success

Persistence is the vehicle you arrive in'

Syllabus-

Manual Testing—

Automation Testing-

- **Core JAVA**
- **Selenium with JAVA**

API Testing-

- API Testing → POSTMAN tool

Database Testing-

- Oracle→ SQL quires

Tools-

- Eclipse
- TestNG
- GitHub
- Jira
- Postman
- Swagger

Project –

2 live project→ different domain

Manual Testing

Manual Testing Syllabus:-

- **Basic of Software Testing**
- **Software development models**
  1. SDLC
  2. Fish model
  3. Waterfall model
  4. V model
  5. **Agile model** (90 % companies using Agile.)

- **Diff types of Testing**
  1. Smoke Testing
  2. Sanity Testing
  3. Functional Testing
  4. Non Functional Testing
  5. Retesting
  6. Regression Testing.

Basic of Software Testing:-

- **What is Software/Application**→ Software is a collection of programs which takes user input and generates desired output.

  Program Languages:-C#, **Java,** Python, Rubby, .net, Php, Perl, etc

- **Types of application:-** 1) **Web-based Application**

  Example:
  - ✓ www.amazon.com,
  - ✓ www.paytm.com
  - ✓ www.gmail.com,
  - ✓ www.majhinaukari.com

  Use : (Enter URL to launch application)

  2) **Desktop based Application**

  Example:
  - ✓ Skype
  - ✓ Photoshop
  - ✓ Media Player

- ✓ Antivirus
- ✓ Microsoft office

Use : (Application build(.exe file) install in system/ computer)

3) **Mobile based Application**

Example:

- ✓ Whatsapp
- ✓ RedBus
- ✓ Zomoto

Use : (Download & Install application from Play Store & IOS store in Mobile.)

Different between Product & Service based company.

Product Based Company	Service Based Company
1. Company which makes the software or application and give it directly to the customers are the product based company	**1.** Company which makes their software or application from other companies and give it for client usage are the Service based company
2. These Companies provided regular updates to client to fix errors.	**2.** These company do not provide direct update to customers, first Customers send query to the Software company & get newer Version
3. Example: - What's app, Amazon, Google, Windows OS.	**3.** Example:-www.SBI.com, www.IRCTC.com, www.EPFO.com.

- **What is Testing**→Testing is a process of checking whether the product works as expected or not.
- **Software Testing**→ Software Testing is a process used to identify the correctness & completeness of functionality/developed software with respect to customer requirements.

Or

- **Software testing** is an activity to check, whether the actual results match with expected results or not.

  There are two ways for QAs to carry out tests:

  →By **manually** executing test cases **(70% manually)**

  →By **automating** test cases using Selenium Web Driver **(30% automation).**

- **Error : Error** is simply a mistake which is done in program.
- **Defect : When actual result** deviates from the **expected result** while testing a software application or product then it results into a defect.

OR

- When Actual result does not the expected result then it is called as defect.

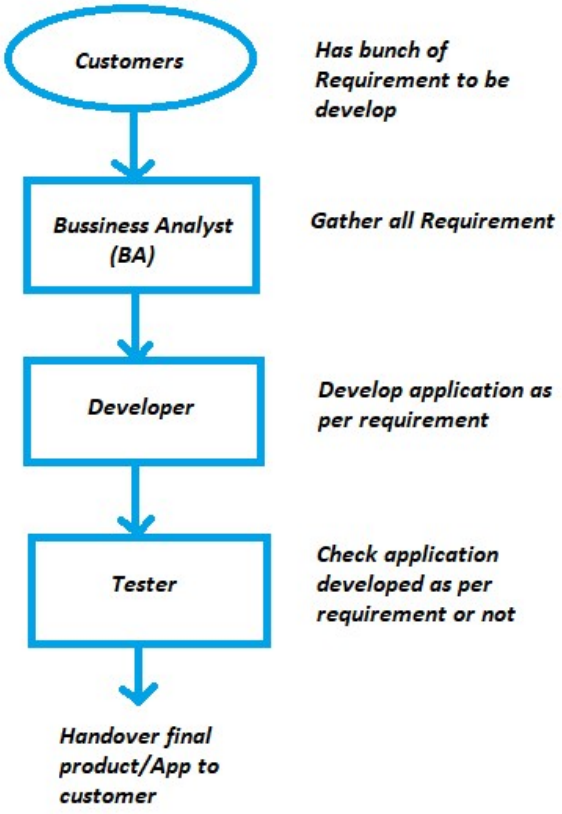
OR

- **Bug** : when developer accept that it is actually a defect at that time it is called as **Bug**.
- **Issue** : Sometimes defect is called as **Issue**.
- **Manual Testing→** Manual Testing is nothing but testing the application/software by running test cases manually by human tester and comparing the actual and expected result.
- **Automation Testing →**Testing an application function or features with the help of an automation tool & test script is called as Automation Testing.
  Different Automation tool**:-** Selenium, Auto IT, QTP(HPQC), Cypress**.**

Why Testing is Important:-

- To identify defects at the development phase
- It is very important to ensure the Quality of the product.
- To build the Customer confidence
- Security: It is the most sensitive benefit of software testing. People are looking for trusted products. It helps in removing risks and problems earlier.
- High-quality product typically has fewer defects and requires lesser maintenance effort, which in turn means reduced costs.
- Note: To deliver Software Defect free.(100 % defect-free not possible).

Work flow and Resources involve in IT Industry/ software Development.



Teams In Software Company:-

1. **Project Team→**

   - To work on **New requirement / Module / Application feature**
   - Different project / software Domain → Banking domain (Ex.www.SBI.com), Telecom domain (Ex.www.JIO.com), E-commerce domain (Ex.www.Amazon.com), Healthcare domain, etc.
   - **Note : We are working in project team.**
   - **Team : BA, Developer, Tester, Designer.**

2. Maintenance / Support team→

- To work on existing application on **Ticket / End user issue / small CR (change request).**
- **Team : BA, Developer, Tester, Designer.**

SQA (Software quality assurance)-

- SQA is communication between customers and BA (Business analysis).
- SQA is done to measures **&** monitor the software development factors like-
  1. Project/customer requirement
  2. Customer Expectations (privacy & performance)
  3. Cost of Project
  4. Time in delivery
  5. Maintenance

1. **Project/customer requirement**:-

   - Communication happens between customers and BA.
   - Process of data gathering.
   - All requirements from customers get documented.
   - Identified the domain of project i.e. requirement belongs to Banking domain, Telecom domain, E-commerce domain.
   - For which purpose customer want software & its objective.

2. **Customer Expectations (privacy & performance) :-**

   - **Privacy: -** for example, in the case of Banking domain software, they want privacy for all the customer's data.

   - **Performance: -** Software should balance load properly.
     It should have to work properly under heavy load.

3. **Cost of Project:-**

- Project cost decided on per hours cost and customer have to pay this.
- Depends on how many persons are involved or how much resources get utilized in the project as well as time to complete the project.

4. **Time in Delivery:-**

- At the time of resource gathering & documentation time to complete a project gets decided.
- If a company exceeds the delivery time then the company has to pay a penalty, that penalty is called Escalation.
- If a delay occurs due to frequent changes in customer requirements then customers have to pay extra charges.

6. **Maintenance:-**

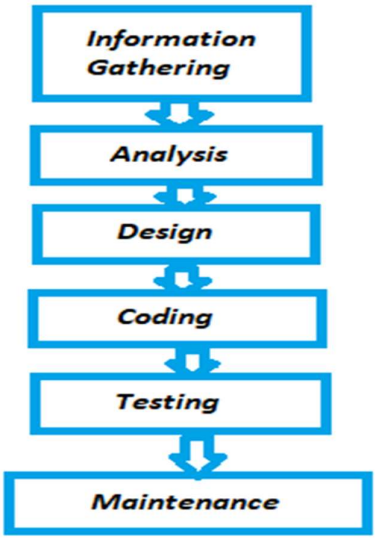
- Maintenance is the part of the service provided by the company after delivery of the project.
- If any problem occurs after delivery of the project then the company has to fix it.
- For some time period as per agreement maintenance is free & after that customer has to pay charges.

SDLC (Software Development Life Cycle) :-

Software development lifecycle (SDLC) is a process used by the software company to design, develop and test high-quality software.

A typical Software development lifecycle (SDLC) consists of the following stages.

- SDLC different stage involved
  1. Information gathering
  2. Analysis
  3. Design
  4. Coding
  5. Testing
  6. Maintenance/Support



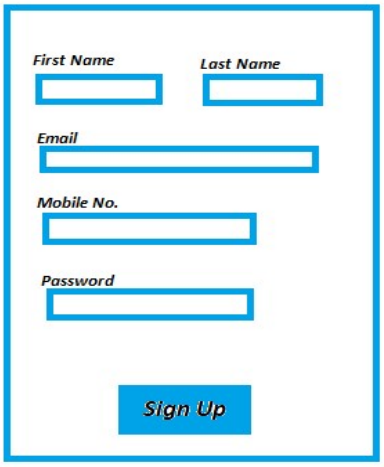
Information/Requirement gathering

- Business Analyst responsible for this requirement/Information gathering by communicating with customers/client
- Gather all the data like what customers want to build, who will be the end-user, what is the objective of the project.
- All the requirement gets documented by the BA which is called as a business requirement specification (BRS).
- BRS is the overall requirement of the project. Ex. For flipkart application →**Sign Up, LogIn, HomePage, Product, Cart, Payment, Thanku Page.**

Analysis

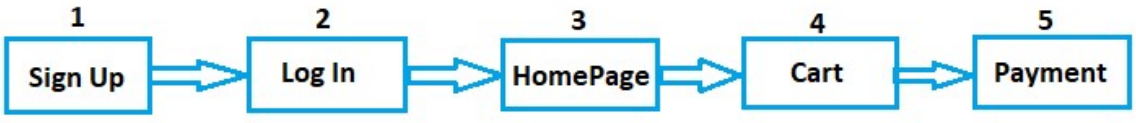
- **B**usiness **A**nalyst responsible for Analysis phase, In Analysis **BA** is preparing the **SRS** (Software requirementspecification**)** documents **from BRS** documents**.**
- **SRS** also called **FRS** (functional requirement specification)/ **CRS** (customer requirement specification).
- SRS is a detailed specification which shows the minor units of Functionality/project.
- SRS is the Functional requirement of the Module. Ex. For sign up page → It includes **First Name, Last Name, Mob.No, E-mail, Gender**.
- When BA prepare SRS he will send**mail to Developers & Tester**
- Developers & tester will **Analysis or understand the SRS.**
- SRS will contain different phases

1. **Function requirement**→
   - Functional requirement means attributes which are requiredto complete a specific function.
   - Let's take an example of Sign Up Page, it contains Fname, Lname, Email, Mobile No, Password.
   - So for First Name functional requirements are :
     - ***a) It should only acceptcharacters.***
     - ***b) It should not acceptnumbers.***
     - ***c) It should not accept spaces.***
     - ***d) It should not accept special symbols.***



   - So like this all the requirements should get fulfill in this phase.

2. **Function flow diagram**→



   - Functional flow diagram means the flow of our task/ proper sequence of tasks.

- This flow shows the relationship between the tasks.
- Relationship of function means dependency of each function.
- Overall this function flow diagram is actually a stepwise representation of software.

3. **Use Case→**
   - Use-case explains the step-by-step procedure of how a particular functionality of the software is used by the end-user.
   - Consider an example of online shopping, Here use-case is different for registered customers, New Customers, Admin & Bank

User/Actor	Register Customers	Non-Register Customers	Admin	Bank
View Product	Yes	Yes	Yes	No
Order	Yes	No	No	No
Payment	Yes	No	No	Yes
Cancel Order	No	No	Yes	No
Confirm Order	No	No	Yes	No
Make Order Report	No	No	Yes	No

Note: Like this use-caseis different forevery Actor/User.

4. **Screenshot / Snapshot**→application image / application without function
    - Snapshot are visualization of functionality before development of Software.
    - Snapshot gives idea to developer that how software suppose to look like.
    - Snapshot are created by the BA by using IRISE software.

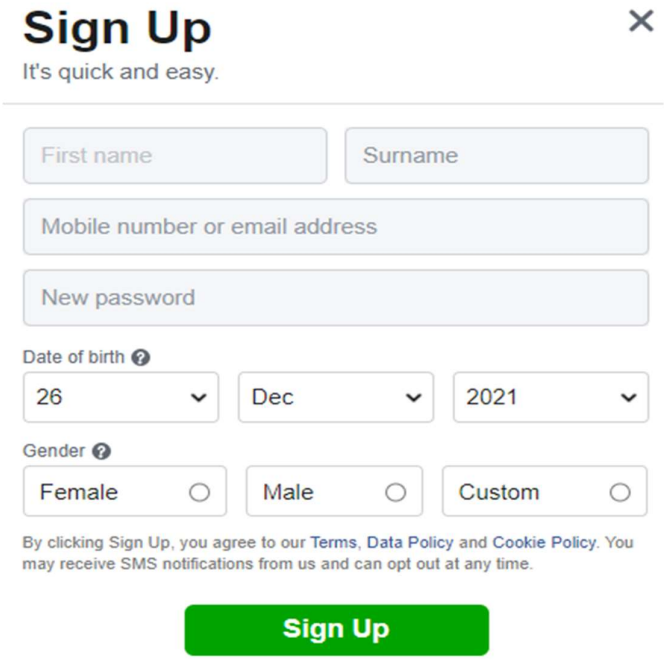


Fig. Facebook Snapshot of Sign Up Page

Design

- The Software design documents are prepared as per requirement specification document.
- This helps define overall system architecture.
- There are two types of Design Document created in this phase

    1.HLD (High Level Design**)**

    2.LLD (Low Level Design**)**

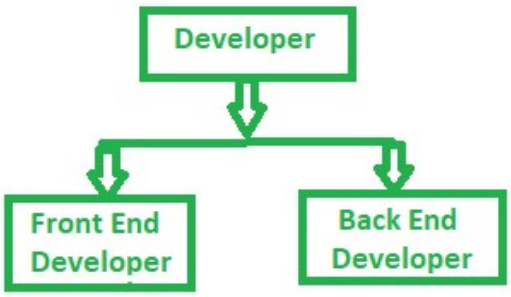
1. **HLD (** High Level Design**)**
    - High Level Design contains design of working of the main module
    - It includes the relation and dependencies of the main module
    - It includes what & how any module do work.

- In case of paytm, **Recharge Prepaid** module, **Pay Electricity Bill**, **DTH** Recharge, Book **GAS Cylinder**.
- EX. HLD→ Pages/Tabs follow (UI/GUI).
- High Level Design is **created by design architecture**.

2. **LLD (**Low-Level Design**)**
    - It includes the design of working of sub modules.
    - After completion of HLD→LLD
    - Low-Level Design includes static logic of every submodule.
    - In case of paytm, Recharge Module is main module & Mobile no, Operator, Amount are the submodules.
    - Ex. LLD→ Paytm – Recharge Module →**Mobile no. enter text box** (accept 10 digit no) OR Circle section/ Operator selection drop down (user will from these drop-down).
    - Low-Level Design is **created by Front End Developer.**

Coding-

- Developers will do coding / write a program to develop software
- There are two types of developer:-



1. **Front End Developer:**

Front End Developer is Responsible to develop.

- Graphical User Interface (GUI / UI)
- Functionalities
- Function Flow

2. **Back End Developer.**

BackEnd Developer is Responsible to develop:

- Data Management
- Data Gathering
- Data Security

Testing-

- Testing is the process of checking the correction & completeness of software.
- This is done to verify whether the application works with respect to customer requirement or not.
- When Tester will get SRS they will prepare TCD (**Test cases design**) as per requirement.
- After completion of coding – **Build/ Application/ Software** , Tester will receive the Build and Tester do the TCE (**Test cases execution**).
- Tester will do Positive as well as Negative Testing.
- During TCE(**Test cases execution**) If Tester found defect then they will Raise / Assign defect to developers to resolve and again developer send build to Tester/QA team for Retest.

Maintenance/Support -

- Maintenance means service provide by the company after delivery of the project.
- If any problem occurs after delivering the project then the company have to fixed it.
- For some time period as per agreement maintenance is free & after that customers have to pay charges.
- End-user →Raise Issue/ defects present in application → Resolved by the maintenance team.

Types of Testing:-

1. White Box Testing (WBT)

2. Black Box Testing (BBT)

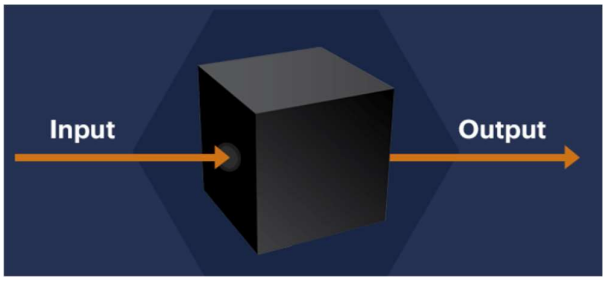
3. Grey Box Testing (GBT)

1. White Box Testing→

- White box testing is also called as Unit Testing / Code Level Testing.
- White box testing is done by the Developer during the development phase.
- When the developer done with his code writing he will check or compile the code and if he found any errors developer have to fixed it.
- Developer cannot send his code to tester without doing white box testing on it.
- Developer checks only positive scenarios.

2. Black Box Testing→

- Black box testing is also called as System and functional testing / System Integration Testing.
- Black box testing is done by the Tester in the Testing phase.
- In Black box testing tester views the software as a black box, ignoring all the internal structure and behavior, their only concern is provide input to system and generate output & check whether it is as per requirement or not.



- During BBT test conditions are created based on software functionality.
- During this testing tester check the overall functionality of the software, this is done to verify whether functionality /application works as per customer requirement or not.
- Tester will do Positive as well as Negative Testing.

Positive Testing→ Testing application by providing the **valid data** set as a input.

Ex. Mobile no. field → tester checks field functionality by entering 10 digit mobile no. whether it is accepted or not.

Negative Testing→Testing application by providing the **invalid data** set as a input.

Ex. Mobile no. field → this field should not accept **9 or 11** digit mobile no. and **(a-z)** characters whether it is accepted or not.

Like this tester checks field functionality by entering invalid data.

Black Box Testing Techniques:

- ✓ **Equivalence Partitioning.**
- ✓ **Boundary Value Analysis.**
- ✓ **Decision Table Testing.**
- ✓ **State Transition Testing.**
- ✓ **Error Guessing.**

3. Grey Box Testing→

- Gray box testing is a combination of WBT & BBT.
- Tester involve in this testing.
- To do grey box testing, the tester should have programming knowledge.
- The role of grey box tester is, whenever final software is handed over to the tester, the tester checks its functionality & if any defects/bugsare found in the output offunction then the tester does not assigndefect/bug to the developer, instead of that tester himself is responsible to solve the or fixed or make changes in code. So knowledge of programming/ coding is required.

Black Box Testing Techniques:

1.Equivalence Partitioning (ECP):

- In Equivalence partitioning is a technique, the input data divided into partitions of valid and invalid values for a particular field
- Here, we have to verify the input data belongs to which data type and class.
- Consider an example for mobile no field which accept the 10 digit mobile no.
- **Example1**. So Equivalence partitioning or Valid and invalid inputs for mobile no field is

Valid I/P	Invalid I/P
0 - 9	a - z A - Z Special Symbol Space

Example 2: For Password(It should accept special character, Upper case letter, lower case letter, digits/ numbers)

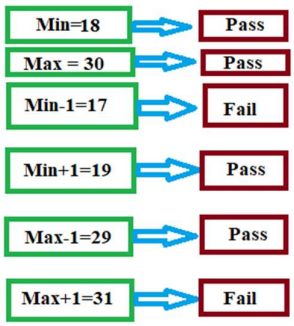
Valid Input	Invalid I/P
Special symbols	**Space**
Digits(Numeric values)	
Special symbols	
(a – z) & (A – Z)	

Example 3: For First Name text Box(It should accept only character **)**

Valid Input	Invalid I/P
A – Z	**Digits**
a – z	**Special sysmbols**
	Space

2.Boundary Value Analysis (BVA):

- Boundary Value Technique is used to test boundary values, boundary values are those that contain the upper and lower limit of a Input.
- It tests, while entering boundary value whether the software is producing correct output or not.
- Example.1: Flipkart has an offer which is valid for customers between the ages of **18** and **30** only. Therefore other values such as 17, 18, 30, or 31 can be tested to check whether the inputs are accepted.



Example2.:In case of What'sapp (Valid range to send photos with others in between(1–30).

So **BVA (**Boundary Value) for this condition is :

Min = 1	**Pass**
Max = 30	**Pass**
Min – 1 = 0	**Fail**
Min + 1 = 2	**Pass**
Max -1 = 29	**Pass**
Max + 1 = 31	**Fail**

3.Error Guessing:

- Error guessing is a technique in which there is no specific method for identifying the error.
- It is based on the experience of the Test Engg, where the tester uses the experience to guess the problematic areas of the software.
- For Example 1: while uploading a document at the of submitting any form it will check with all kind of document type**(.docx, .txt, .pdf, .xls, jpg, png.)** will accept or not or it will throws error like "**Invalid document type, only pdf accepted** ".
- Example 2. while uploading a image (image file must be in less than 30kb),So for size more than 30 will not accept it will throw error message.
- Example 3. In case of whats app if we select photos more than 30 than try to send the it will throw error message like "cannot send photos more than 30 nos"
- Like this tester check different scenarios of application.

4.Decision Table Testing:

- If the results are obtained depending on certain conditions, it comes under Decision table testing.
- For Example 1: when any bank release their Recruitment notification, In that notification, vacancies and fee(Charges) for every category(Caste) is different.
- So our application is able to take decision of fees structure for different categories(Caste).

(CENTRAL BANK) सेंट्रल बँक ऑफ इंडिया मध्ये 214 जागांसाठी भरती [मुदतवाढ]

Central Bank of India Recruitment 2021

Central Bank of India Recruitment 2021, (Central Bank of India Bharti 2021) for 115 214 Specialist Officers Posts. *www.majhinaukri.in/central-bank-of-india-recruitment*

Fee: General/OBC: ₹850/- [SC/ST/PWD: ₹175/-]

Last Date of Online Application: ~~17 December 2021~~ 30 December 2021

Date of Online Examination: 22 January 2022

Official Website: View

Notification: View

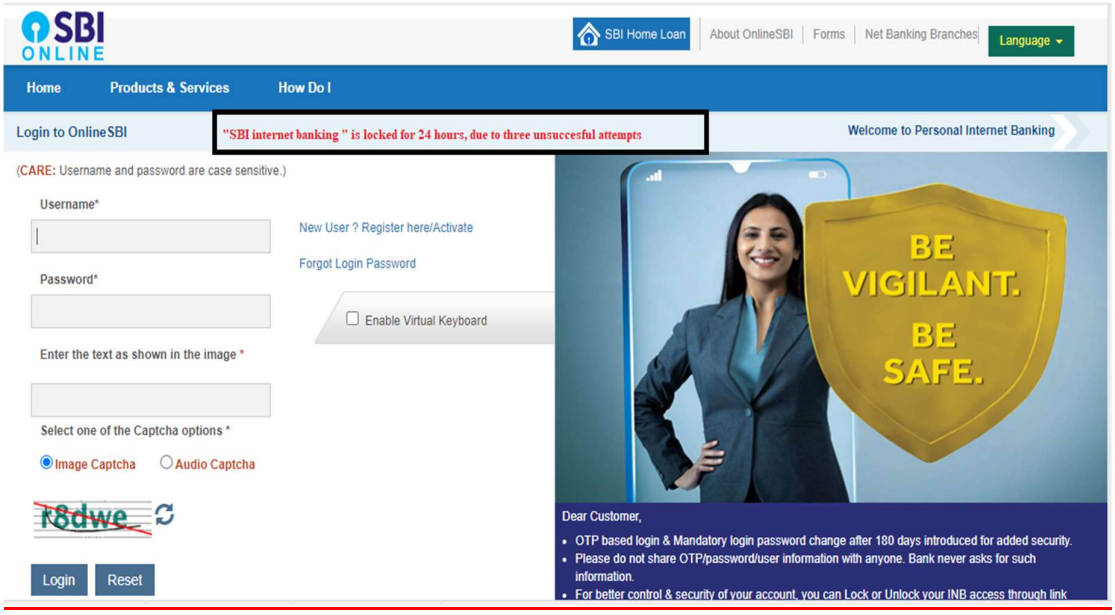
Online Application: Apply Online

Category	Sc	ST	PWD	NT	OBC	Open
Fees	175/-	175/-	175/-	175/-	850/-	850/-

Example 2 : In case of Bus booking app the output i.e (List of buses) is depends on input i.e (Source and Destination), here application is able to take decision to list out the different bus on the selected route

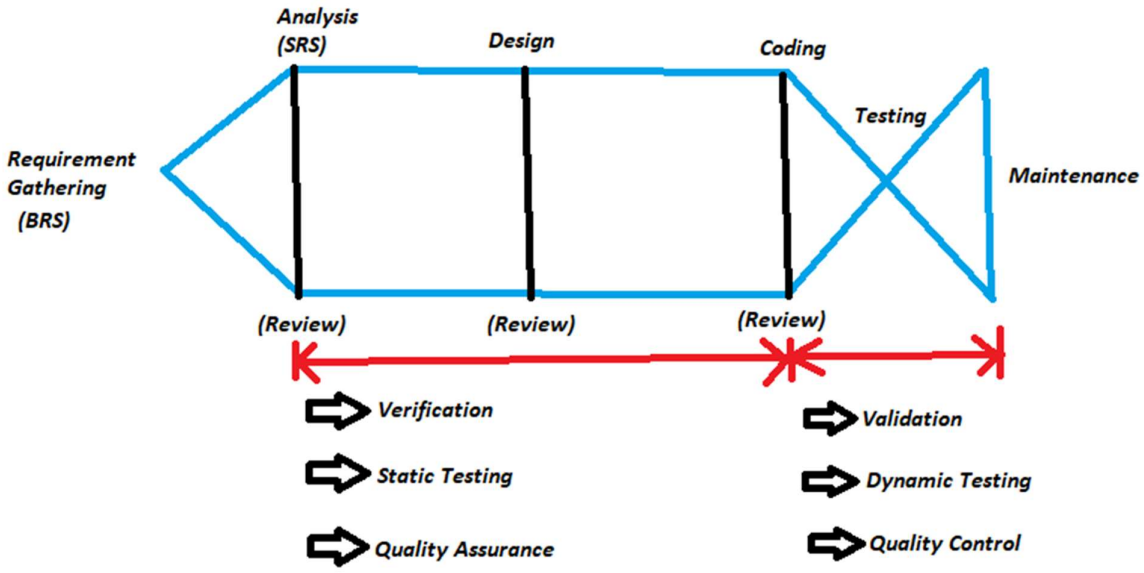
5.State Transition Testing.

- State Transition Technique is used to capture the behavior of the software application when different input values are given to the same function. This applies to those types of applications that provide the specific number of attempts to access the application
- For example if the user wants to login to any application but enters the password wrong, after a particular number of Unsuccessful attempts the account is locked.
- In case of SBI online, After **Three Unsuccessful** attempts the account is locked.



Example 2 : When u fail to unlock your mobile with 5 unsuccessful attempts then your mobile will lock for 72 Seconds like this we have to check whether our application is able to change state from normal to locked.

Fish Model :-



- Fish model is advance version of the SDLC.
- Fish Model is one of the software development methodology which Consist of verification (Review) and Validation (Testing)
- The structure of this model looks like a fish, hence the name shows fish model.
- **Verification** :→
  - Verification is process is also called as **Quality Assurance**(QA)/ **Static Testing.**
  - Static testing is nothing but Review process.
  - In **requirement gathering process**, Business Analyst gathers all Information from customers & BRS document.
  - Then from BRS document BA makes SRS document.
  - Then BA have to check whether it is correct or not, because all the further process are depends on the SRS Document.
  - This **Checking process** is called as **Review process**.
  - Similarly after completing design phase, **design architecture** also have to do **review** process on **Design document.**
  - After coding is over then developer have to Review his code, means he has to check & compile the code and Also find and fix the error. This **code**

Reviewprocess **done by developer** which is called as unit testing or code level testing.

- In the Verification / static testing process responsible authority / person do only positive checking or review.

- **Validation** :→
  - Validation is process is also called as **Quality control**(QC)/ **Dynamic** Testing.
  - Once the coding completed then application will handover to testing team to perform testing on it.
  - In this phase testing team follow SRS document to verify whether it is working fine or not.
  - If anything not working, testing team raises the defect and development team have to fixed it.
  - Dynamic testing focus on quality of the software / product.
  - Whole software functionality get checked in validation process.
  - **Validation** include **Black box testing.**

Que : Difference between Verification & Validation

Verification	Validation
• The verifying process includes checking documents, design, code, and program	• It is a dynamic mechanism of testing and validating the actual product
• It does not involve executing the code	• It always involves executing the code
• They checked Software Specification document. which is called as Review process.	• It checks whether the software meets the requirements and expectations of a customer
• It finds bugs early in the development cycle	• It can find bugs that the verification process cannot catch
• Verification uses methods like reviews, walkthroughs, inspections, and Technical review.	• It uses methods like Black Box Testing and non-functional testing
• It comes before validation	• It comes after verification

Example of verification and validation

Now, let's take an example to explain verification and validation planning:

- In Software Engineering, consider the following specification for verification testing and validation testing,

A clickable button with name Submet

- Verification would check the design doc and correcting the spelling mistake.
- Otherwise, the development team will create a button like

Example of Verification

- So new specification is

A clickable button with name Submit

- Once the code is ready, Validation is done. A Validation test found –

Button NOT Clickable

- Owing to Validation testing, the development team will make the submit button clickable

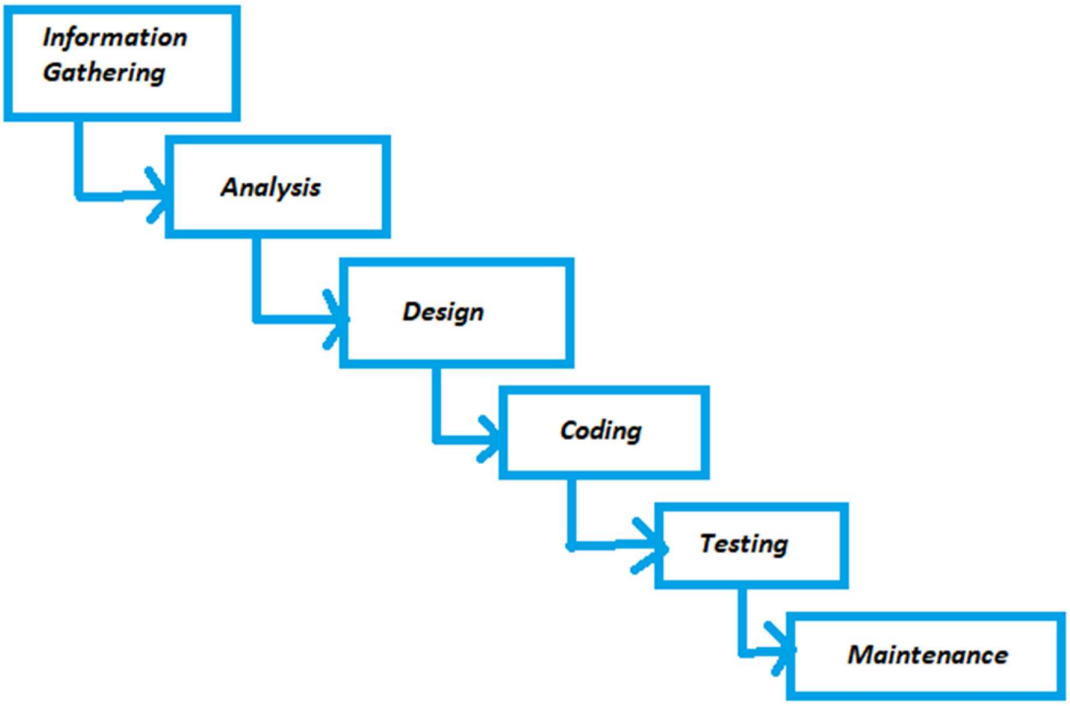
---

Interview Question-

What is software testing and why testing is required ?

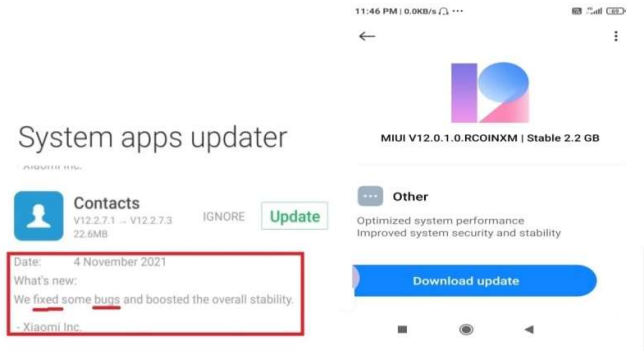
1. What are different factors of SQA ?
2. What is SDLC ?
3. Difference between BRS & SRS
4. What is SRS documents & what it will contains ?
5. What is Use case ?
6. What is difference WBT & BBT ?
7. What is difference Static Testing & Dynamic Testing?
8. What is difference Verification & Validation?
9. What is functional requirement.
10. What is review Process and why we perform review process?
11. What is Positive and negative Testing with example

Waterfall Model :-



Dia. Waterfall model

- Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project.
- In the waterfall approach, the whole process of software development is divided into separate phases.
- Waterfall model is step by step implementation of SDLC / It is sequential model (one after another) of software development.
- In waterfall model whenever first stage over then and then only procedure goes to next stage.
- Customer cannot go back to previous stage to change something.
- In waterfall model, Customer cannot change the requirement in current release.
- After completion of coding then and then only tester will starts testing.
- In waterfall model when tester found any bug then he do not send product / build back to the developer instead of doing that tester log the defect and make a report of that and those defect get fixed in next version of product.
- **Example**:
    - Every mobile phones have Android/IOS operating system
      They release their android version and if some **bugs** Available / occurreds
      Then they **provide Update version** of their **operating system**(OS).



Dia : Example of New Release

- Generally waterfall model use in small scale company and product base company.
- General time duration of the waterfall model is 3 month.

Advantages:-

- Simple and easy to understand and use
- Clearly defined stages.
- Easy to arrange tasks.
- Process and results are well documented.
- Requirements are very well documented, clear and fixed.

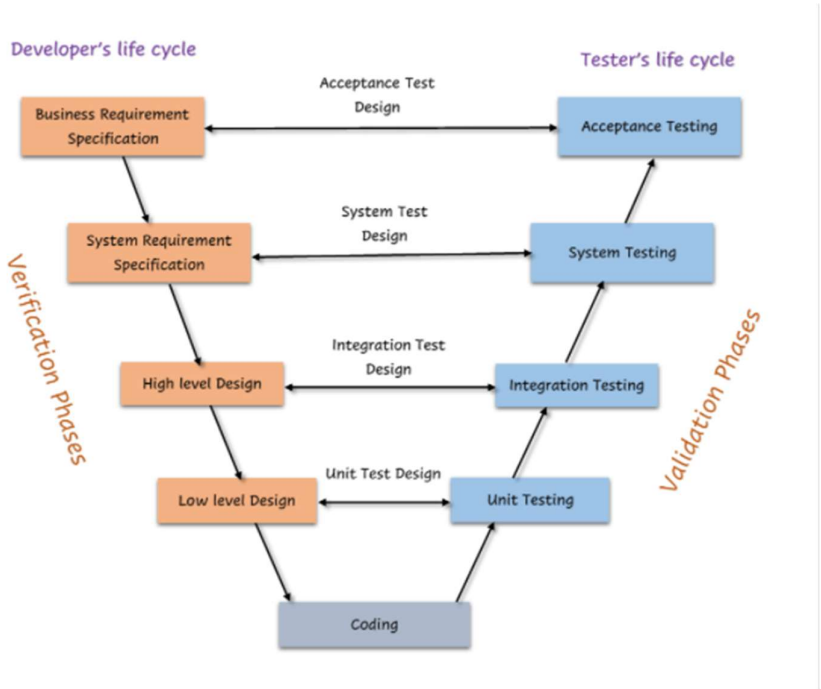
Dis-advantages:-

- Cannot accept changing requirements.
- When first stage completed then and then only process goes to next stage.
- Poor model for long & ongoing project.
- We cannot go back to previous Stage.

Different Phases OR Environment in the software Devevlopment

Development Integration Testing (DIT)	System Integration Testing (SIT)	User Acceptance Testing (UAT)	Real application Handover to Customer (Live application)
person involve-- (Developer)	Person involve-- (Tester)	Person involve-- (Tester & Customer)	Person involve-- (Actual User)

V – Model :



- V-model is an extension of waterfall model which is also called as verification and validation(V & V) model.

- **Verification:** It involves a static analysis method (review) done without executing code. It is the process of evaluation of the product development process to find whether specified requirements meet.
- **Validation:** It involves dynamic analysis method (functional, non-functional), testing is done by executing code. Validation is the process to Checking the software after the completion of the development process to determine whether the software meets the customer expectations and requirements.
- In V- model development stages are mapped with Testing stage, means Testing phases planned in parallel with development phases.
- Each phase of the development has its corresponding testing phase. Both development and Testing activities performs together (Parallely). Unlike of waterfall model, the next starts only when the current phase is completed.
- In v- model **we can go back to previous stage and change the Requirement** but for that **customer have to pay some amount**.
- General duration of V- model is 3 month .
- The various phases of the V-model are as follows:
- In V- model During **Information gathering** is phase, **Parallely User Acceptance Testing** plan is created. The UAT test plan is the strategy that will be used by customer to verify and ensure an application meets its business requirements.
- During **Analysis** is phase, **Parallely system test plan** is created. The test plan focuses software testing activities which meeting the functionality specified in the requirements gathering.
- The **high-level design** (HLD) phase focuses on system architecture and design, **Parallely** An **integration test plan** is created in this phase as well in order to test the pieces of the software systems ability to work together.
- The **low-level design** (LLD) phase is where the actual software components are designed. It defines the actual logic for each and every component of the system, **Parallely Unit Testing plan** is created Unit Testing is performed by the developers on the code written by them.
- **Coding**: This is at the bottom of the V-Shape model. Module design is converted into code by developers, where all coding takes place. Once coding is complete, the path of execution continues up the right side of the V where the test plans developed earlier are now use.

Advantages :

- Testing activities like planning, test designing happens well before coding.
- This saves a lot of time.
- We can change requirement by paying some amount.
- Works well for small plans where requirements are easily understood.

Disadvantage

- If any changes happen in the midway, then the test documents along with the required documents, has to be updated.
- Not a good for a large & complex project.
- Clients can see only final project.

==========

Terms involve in V-model

➔ **DRE (Defect Removal Efficiency):**

- The Defect Removal Efficiency is the process of calculating at which level tester did testing.
- There are two phases in the DRE (A) Defect found by tester (B) Defect found during User Acceptance testing(UAT).

- **DRE** (defect removal efficiency) = **A / (A+B)**

  Ex. = (90/90+10) = 0.9 Good Testing.

  Where,

  A= no. of defects found in BBT/SIT = 90

  B = no. of defects found in UAT = 10

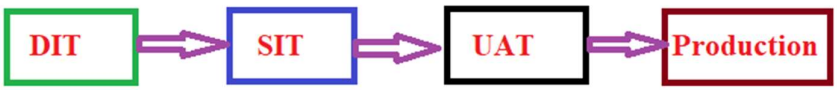
DRE	Remark
0.8 to 1	Good Testing
0.5 to 0.8	Average Testing
Below 0.5	Below Average

➔ RFC (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 SRS document there is section “CR” it is Red color.
- SRS is a PDF format document
- Customer have to pay extra money for that.

==========

➔ Post mortoum Testing:

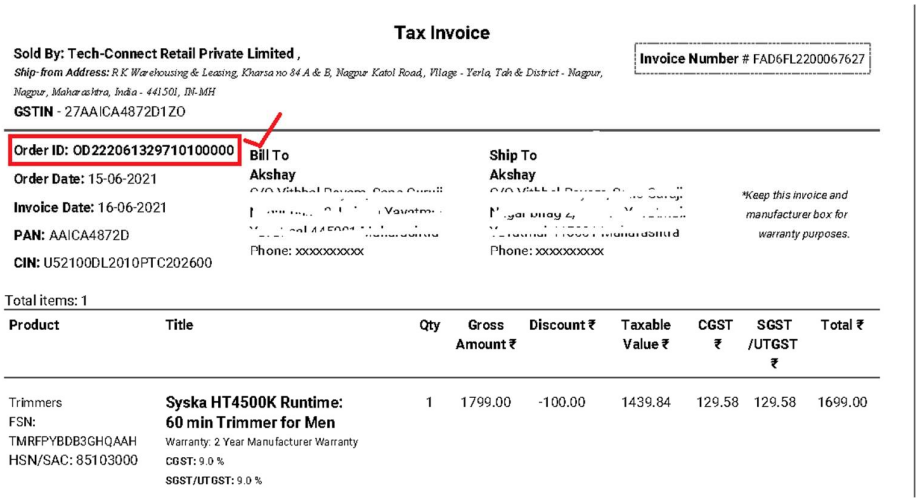


Development Integration Testing (DIT)	System Integration Testing (SIT)	User Acceptance Testing (UAT)	Real application Handover to Customer (Live application)
person involve-- (Developer)	Person involve-- (Tester)	Person involve-- (Tester & Customer)	Person involve-- (Actual User)

- When whole testing done & our product is ready for the production and if product does not produced desired output, then there is problem in system then tester have to test all the functionalities again and White box tester(Developer) should have to check all the modules in details
- Hence **Developer** is **responsible** for **Post mortoum** Testing.

Example:

- Suppose we order one product from amazon, so we get an order ID, this order ID is gets store in OCE(0rder captuation Engine). But there is problem, we get order ID but our order gets fails.



Tax Invoice

Sold By: Tech-Connect Retail Private Limited ,
Ship-from Address: R K Warehousing & Leasing, Kharsa no 84 A & B, Nagpur Katol Road,, Village - Yerla, Teh & District - Nagpur, Nagpur, Maharashtra, India - 441501, IN-MH
GSTIN - 27AAICA4872D1ZO

Invoice Number # FAD6FL2200067627

Order ID: OD222061329710100000
Order Date: 15-06-2021
Invoice Date: 16-06-2021
PAN: AAICA4872D
CIN: U52100DL2010PTC202600

Bill To
Akshay
Phone: xxxxxxxxxx

Ship To
Akshay
Phone: xxxxxxxxxx

**Keep this invoice and manufacturer box for warranty purposes.*

Total items: 1

Product	Title	Qty	Gross Amount ₹	Discount ₹	Taxable Value ₹	CGST ₹	SGST /UTGST ₹	Total ₹
Trimmers FSN: TMRFPYBDB3GHQAAH HSN/SAC: 85103000	**Syska HT4500K Runtime: 60 min Trimmer for Men** Warranty: 2 Year Manufacturer Warranty CGST: 9.0 % SGST/UTGST: 9.0 %	1	1799.00	-100.00	1439.84	129.58	129.58	1699.00

- So there is problem in the application so developer check all the modules in details. That's why we called it as postmortem testing

==========***End of V & V model*** ==========

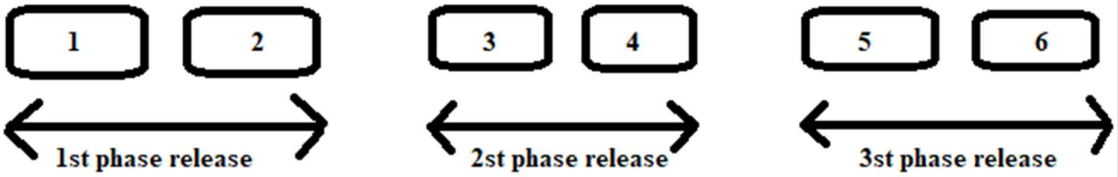
Agile Methodology

SDLC	→	Agile Methodology
Customer	→	Stake Holder
Business Analyst	→	Product owner
BRS (Business Requirement Specification)	→	Product Backlog
SRS (Software Requirement Specification)	→	Sprint Backlog
Functional Requirement in SRS	→	User stories
Release	→	Sprint
Project manager	→	Scrum Master

- Agile is philosophy, it is not plan driven but it is value driven methodology.
- There are 6 methods in agile
  - Xtreme programming
  - LEAN
  - Kanban
  - **Scrum**
  - Dynamic System development method
  - Future driven development method.
- **We are working in Scrum Agile methodology.**

- In v model after requirement get fixed whole team focuses on product development.
- But Agile methodology is the module driven methodology.
- Requirements are changing frequently in the agile methodology , So it is not a plan driven methodology.
- Agile methodology is used in service based company.
- In the agile methodology customer can request for change at any point of development phase or in any phase like: **(DIT → SIT → UAT → Production)**
- This change in the requirement do not affect on development of any other module.

- Customer do not have to pay extra money for changes in requirement and also changing requirement do not increases development time.
- There is **no SRS** in the agile instead of that SRS there is JIRA tool in which any changes are inserted by the product owner & those are visible to the developer & tester.
- Suppose in one project there are 6 modules, so in Agile methodology this project divided in group of module.

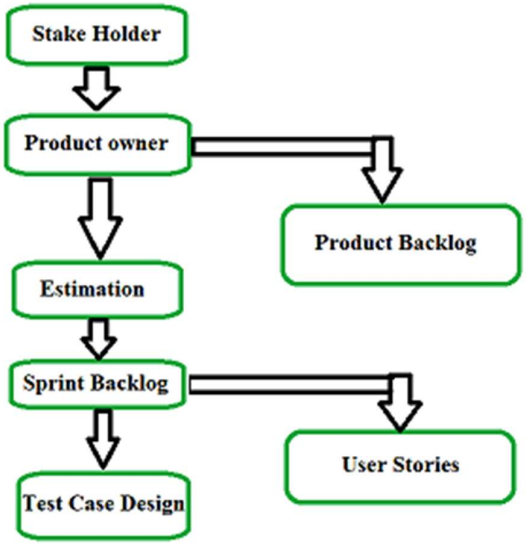


- According to the number of modules to be develop, team formation done by the product owner or Scrum master.

Note : Duration of sprint varies in between (1 - 4 weeks) company to company.

In my current organization the Sprint Duration is 4 weeks i.e 1 month.

Architecture of Agile:



Stake Holder :-

- Stake holder are the customer
- They have bunch of requirements.
- At any stage / phase of development testing, production they can request for changes.
- In agile methodology customer is member of top most body of company.

Product Owner :-

- Product owner gathering requirement from the Stake holder.
- Product owner is team member of Sprint planning meeting.
- Product owner is responsible for creating product backlog.

Product Backlog:-

- Product backlog are the total requirement for the whole project means It include the requirement for all the modules
- A product backlog is a list of items to be done, this items should be broken down into user stories.

Estimation:-

- In agile methodology , the focus is on module based release
- So all the requirement are not useful to develop the specific module
- So to decide specific requirement among all the requirement estimation is done.
- Estimation is also called as sprint planning meeting and members involve in this meeting is :-

  1. Product Owner 2. Development team 3. Testing Team.

- Simply estimation is sorting of requirement from total requirement to develop a specific module.
- Estimation is done based on 3 main factors

  1. Knowledge 2.Efforts 3.Complexity

- Base on these factor we decided number of user story for that sprint.

Sprint Backlog:-

- Sprint backlog is created by product owner.
- Sprint backlog is treated as subset of Product backlog.
- Sprint backlog contains details information of the requirement which are required for the module development in the particular sprint(1 month).
- Sprint Backlog contains the number of user story.

User Story:-

- User story are nothing but the functional requirement for module to be develop in the particular Sprint (1 month)
- User Story Consist of 2 parts
- **Description**: It contains the description that, what user want to do process & what is it desired output.
- **AcceptanceCriteria** : Criteria is nothing but the scenario, If the scenario is true then system generate correct Output otherwise system shows failure.

Example :-

Description: As a user, I want to insert debit card in ATM, so that I can check my account balance .

Acceptance Criteria: If the debit card is valid (Not expired) then it will show u then account balance .

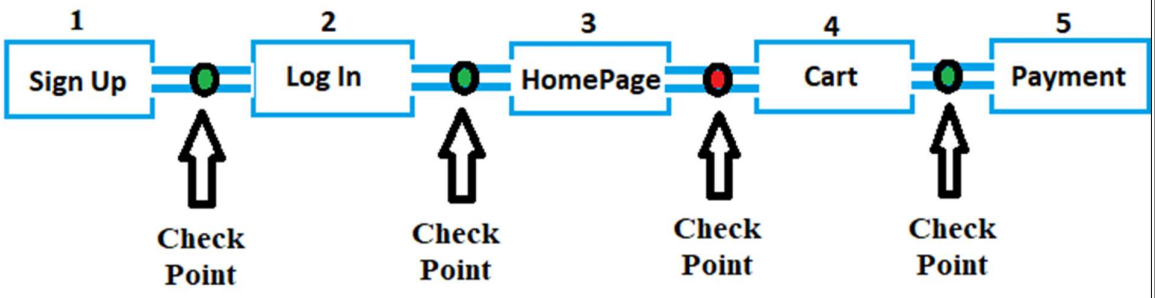
Test Case Design:-

- Tester is responsible for test case design
- Tester will write test cases based on user story i.e based on description and acceptance criteria.

Advantages of Agile methodology

1.Check Points :

- In V module, If at the time production if any defect occurs or application does not provides desired output then developer have to do post mortoum testing.
- But in Agile methodology when any problem / defect occurs at the time of production tester check the check points, so no need do the post mortoum testing
- Consider an example where we developed a module & its sub module.



- If module 1 (i.e Sign Up) is a starting module & module 5 (i.e payment page) is end module, when any defect / error occurs at the time of production then checks the check point between module 1 (i.e Sign Up) & module 2 (i.e Log In) if it is correct.
- Then tester check module 2 (i.e Log In) & module 3(i.e Homepage) if it also correct.
- Then tester check module 3(i.e Homepage) & module 4 (i.e Cart**) if defect occurs in this check point then tester send it to the developer.**
- Developer then click on that check point & code appears and defect code is highlight with red color then developer resolve /fixed the issue.

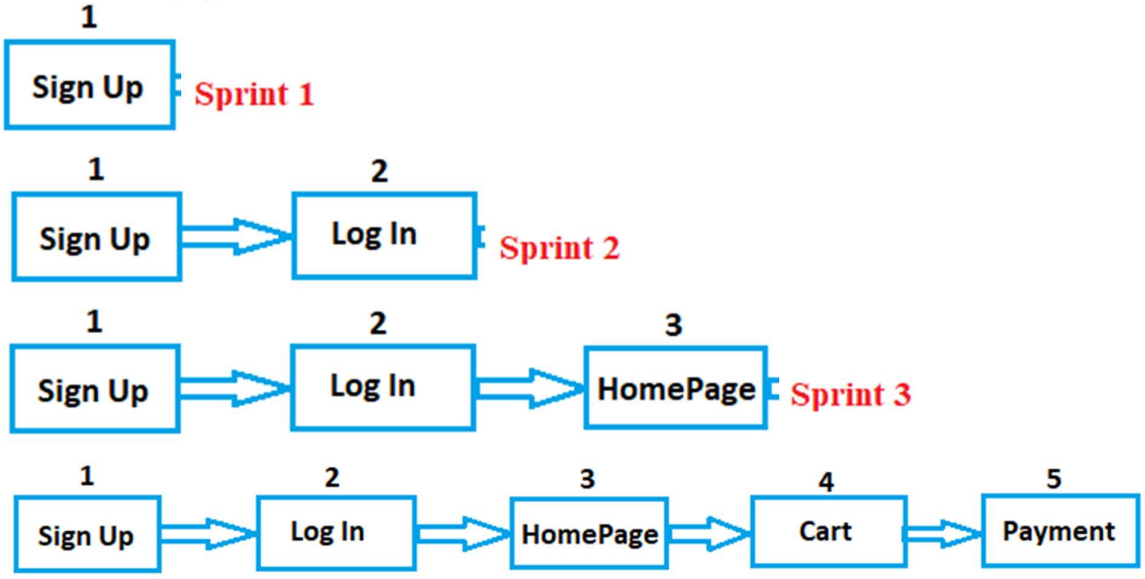
2.Scrum Meeting :

- Scrum meeting is a **daily stand up** or **Daily status call meeting**
- People involved in scrum meeting is
- **1. Scrum master (Chairperson) 2. Testing Team 3. Developer Team.**
- In this meeting Scrum master ask question to the team that
  - **1. What we did yesterday?**
  - **2. What we are going to do today?**
  - **3. What are the roadblocks or issues?**

- General time duration of this meeting is 15 – 30 mins.

3. Implementation of Automation :

- We can implement automation in agile methodology .
- Because in agile we gives sprint wise delivery we deliver the single or individual module in every sprint.



- So tester have to test same module again and again after every integration, which is quite difficult and very much timing consuming with manual testing, So we use/implement Automation Testing.
- Advantages of using Automation:
    1. Less resourses required
    2. Less resourses means less cost required.
    3. High accuracy.i.e no chances of human error.
    4. Required less time to test the application.

4.Sprint wise delivery:

- **In V model** →project duration = 3 months = we delivered 1 release
- **In Agile methodology** →Sprint Duration = 1 month = we delivered 1 or 2 module
- Sprint wise delivery means module wise delivery.

5. Always welcome for changing requirement

6. Faster Feedback from customers

7.Agile is beneficial for project where goal is not defined (Ex. Paytm still is a ongoing project)

8. Customer satisfaction by rapid delivery of useful software.

==========

Disadvantages of Agile methodology

- To implement new module in previously developed software, Tester & Developer should have to knowledge about **Flow of Software**, **different Scenarios**, **Dependancies& Relationship of modules,** if developer and Tester has knowledge of all this then they can work in Agile.
- There is no time for knowledge Transfer(KT) in agile, Becoz release (Sprint) time is only one month.

Meeting in Agile methodology :

1. Grooming Meeting /Session:-

Purpose :

- In this meeting the User story / Requirement get understood, means product owner explains the requirement of project to the team member(Scrum team = Dev & Tester) and team ask their doubts to product owner.
- **When it Conducted** : Before we start of sprint and if any team member has doubt they can arrange grooming meeting.
- **Duration** : 30 min / 1 hr
- **Peopleinvolve** : PO, Dev Team, Testing Team, Designer.

2. Sprint planning Meeting:-

Purpose :

- In this meeting we make sprint goal & plan to achieve it.
- In Sprint planning Meeting Product owner priorities the user story from product backlog and also decided the number of user story for current sprint.
- And based on the knowledge, effort and Complexity we estimate the time required to complete that task.
- Dev and Tester give their time (Estimation) to perfom the task in terms of Story point(1,3,5,7).
- Ex. 1US = 16hr Dev + 8hr Test = 24 hr (3 days)
- **When it Conducted**: At the beginning of a sprint.
- **Duration** : 30 min / 1 hr
- **Peopleinvolve** : PO, Scrum master, Dev Team, Testing Team, Designer

3. Scrum Meeting :

Purpose

- Scrum meeting is a daily stand up or Daily status call meeting
- In this meeting we disscuss on sprint goal progress.
- In this meeting Scrum master ask question to the team that
    - **1. What we did yesterday ?**
    - **2. What we are going to do today ?**
    - **3. What are the roadblocks or issues ?**
- **When it Conducted** **:** Every day (Daily stand up call)
- **Duration** **:** 15 – 30 mins.
- **People involve :** Scrum master (Chairperson), Testing Team, Developer Team**.**

4.Sprint Review Meeting :

Purpose :

- In that meeting the review / Demo of work completed in the sprint are discuss with stakeholder and get immediate feedback from project stakeholders
- Iteration review is a time to showcase the work of the team. Demo will be given by the Tester.
- **When it Conducted** **:** At the end of a sprint
- **Duration** **:** 30 min / 1 hr
- **People involve:** Scrum Master, PO, Stakeholder, Dev Team, Testing Team.

5.Sprint Retrospective Meeting :

Purpose :

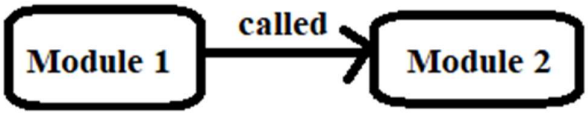
- In this meeting we disscuss what we did good and bad during the sprint.
- Also we discuss what we can do more better or what can be improved for the next sprint.
- **When it Conducted** : At the end of a sprint
- **Duration** : 30 min
- **People involve**: Scrum Master, PO, Dev Team, Testing Team.

Note : Agile meeting also called as **Agile Ceremonies.**

Integration –

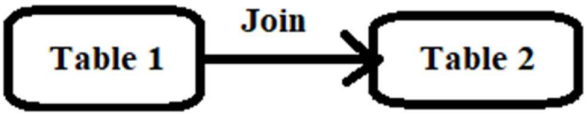
- In DIT environment, developer will work
- When Developer will done coding then they perform the Unit testing on every individual sub module then developer have to do Integration.
- Integration means developer groups the individual modules together & test them
- Integration is the process of mapping new module with old module, for that dev should have knowledge about functionality, Relations, Dependancy of module over each other, then he can do integration.
- Because in Software the output of one module is use as input for another module.
- Integration testing has 2 types
    A. **Frond end Integration** (use Called function):

In the front end integration developer connect modules using “Called ” function.



B. **Back end Integration**(use Join Statement)

In the Back end integration developer connects 2 or more table in database using “Join” Query.

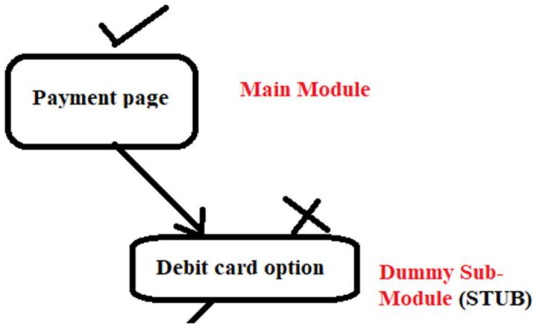


Integration Testing:

- Once developer done with Integration he will do Integration Testing.
- Integration Testing is a process of checking the flow of functionality whenever integration is performed.
- Aim of the Integration Testing is to figure out module function as planned after integration.
- Developer uses testing 3 approaches for Integration Testing
  - Top down approach
  - Bottom up approach
  - Bi-directional / Sandwitch approach

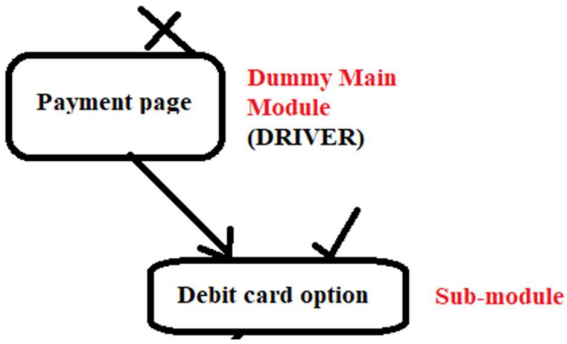
1.Top down approach-

- When Developer have to perform integration testing and if we have main module but we don't have sub-module then Developer used Top down approach
- Developer will prepare the Dummy module of sub-module which is called as **STUB**
- STUB is Dummy program in XML language created by developer.



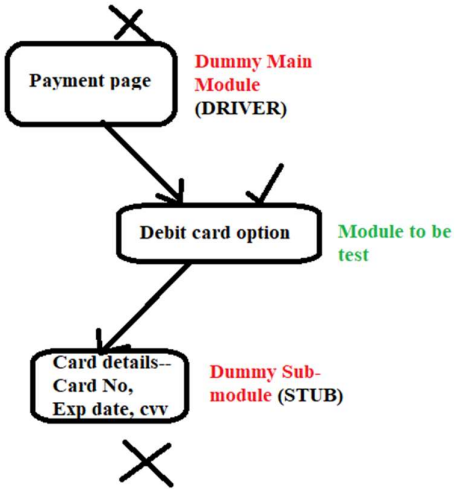
2. Bottom up approach-

- When Developer have to perform integration testing and if we have Sub-module but we don't have main-module then Developer used Bottom up approach
- Developer will prepare the Dummy module of main-module which is called as **DRIVER**
- DRIVER is Dummy program in XML language created by developer.



3. **Bi-directional approach**-

- When Developer have to perform integration testing and if we don't have Main module & Sub-module then Developer used Bi-directional approach.
- In Bi-directional approaches Developer use both the approaches ,For Main module- **DRIVER**& Sub-module-**STUB**.



Smoke Testing:

- When developer done with integration & integration testing then he send code/build to SIT team.
- Once SIT tester receive initial build then the very first testing perform by tester is smoke testing.
- In smoke testing we check the basic and core functionality and ensure that it doesn't crash when it started.
- Means we check whether our build is stable or not or we can say our build is ready for further testing or not.
- During smoke testing tester have to give the proof or evidence of tested functionality
- We do not write any test cases for smoke testing.
- If any defect found during smoke then this defect have to fixed in that day only.
- During smoke testing we have test 5 factors
  - Basic & Core functionality
  - Tab validation
  - Link validation

- Page validation
- GUI validation

Basic & Core functionality :

- **In this Testing, Tester test buttons, Icons from which user proceed for the next page.**
- **There is Submit button at Sign Up page then tester test that submit button whether it is works or not.**

Tab validation :

- **In this Testing, tester check the functionality of Tab's or text boxes in which we enter values.**
- **Whenever we enter any values in tabs / input field by using keyboard we check whether it accepts characters, numbers, and symbols or not.**
- **This functionality of Tab in get validate during Tab Validation.**

Link validation :

- **Link validation is the process in which sequence of interlink pages get tested.**
- **Ex. In Paytm app when user click on Bus Ticket Icon, then Bus Ticket informative page should get open.**
- **Here developer gives link of that page to the "Bus Ticket Icon".**
- **So this link get validate during link validation.**

Page validation :

- **Page Validation means navigation validation,**
- **During page validation we check like, when user click on "Next" or "Back" page should get navigate "forward" and "backward"**
- **This testing is also called as navigation testing.**

GUI validation :

- **In this testing, tester test the interface on which user interact directly.**
- **During GUI validation tester check whether the pages are displayed correctly or not, images should be clear or no, no blur occurs anywhere on the page.**
- **There is no dark colour & have properly aligned modules or not.**
- **This validation of visualization is called as graphical user interface testing.**

==========

Sanity Testing:

- **Sanity testing is advance version of Smoke testing.**
- **It is same like a smoke testing , but the difference is along with smoke testing we do the troubleshooting / Package validation.**

- **Sanity testing = smoke testing + package validation.**

- **Package validation:**

1] In sign Up page we enter user information like (Name, mobile, email, etc.) and when we click on submit button then this information gets store / save in database.

2] When User go to login page and enter valid credentials then if the invalid errors occurs even after submitting valid credentials then Backend Developer have to find from which package this information comes.

3] So backend developer also involved in Sanity validation.

Different Environmentin Software Development

DIT Environment → www.dIT8050v1flipkart.com→WBT, Integration, Integration Testing

SIT Environment →www.SIT8050v1flipkart.com**→ BBT / System & Functional Testing**

UAT Environment →www.UAt8050v1flipkart.com**→get Customer Feedback**

Production Environment →www.flipkart.com**→ Real User uses application**

==========

System Integration Testing (SIT)

- **In DIT phase Coding, WBT,Integration & Integration testing happens.**
- **After DIT, Application deployed to SIT Environment**
- **SIT is also called as System & functional testing, In which we perform Black Box Testing**
- **SIT is process of checking Completeness and correctness of functionality with respect to customer requirement.**

System & Functional testing include:

1.Functional Testing.
2. Non - Functional Testing.
3. Usability Testing
4.Security Testing
5. Performance Testing

1.Functional Testing :

→It is Process of checking the internal functionalities of Application.
→Test cases are executes during this testing.

1. **Behavioral Coverage**
2. **Input domain Coverage**
3. **Error handling Coverage**
4. **Back End Coverage**
5. **Service level Coverage**
6. **Calculation base Coverage.**

1. Behavioral Coverage : In this coverage we check the property and behavior of the object.

Example 1-- CheckBox :

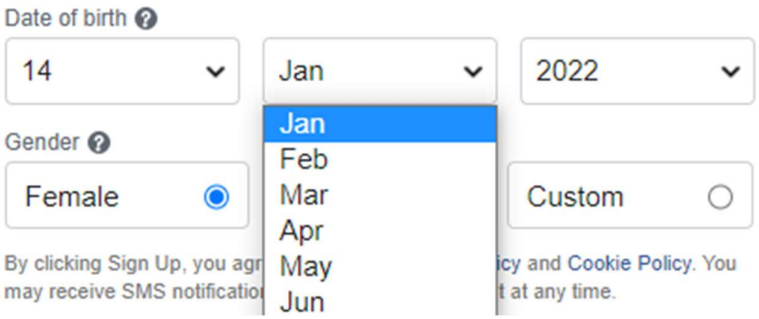
→**Property:**when user click on check box it get tick.

→**Behavioral :**Check and Uncheck.

Example 1-- Dropdown :

→**Property:**when user click on dropdownit should show hidden list.

→**Behavioral :** Show and Hide.

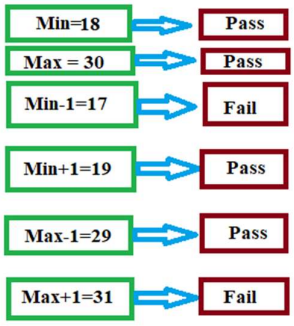


2.Input Domain Coverage :

1. In this coverage we check the type and size of the input.
2. In input domain coverage we have to maintenance
    1. Boundary Value Analysis (BVA)
    2. Equivalence class partition (ECP)

I .Boundary Value Analysis (BVA):

- Boundary Value Technique is used to test boundary values, boundary values are those that contain the upper and lower limit of a Input.
- It tests, while entering boundary value whether the software is producing correct output or not.
- Example.1: Flipkart has an offer which is valid for customers between the ages of **18** and **30** only. Therefore other values such as 17, 18, 30, or 31 can be tested to check whether the inputs are accepted.



Example2.:In case of What'sapp (Valid range to send photos with others in between(1–30).So **BVA (**Boundary Value) for this condition is :

Min = 1	**Pass**
Max = 30	**Pass**
Min – 1 = 0	**Fail**
Min + 1 = 2	**Pass**

Max -1 = 29	Pass
Max + 1 = 31	Fail

II.Equivalence Partitioning(ECP):

- In Equivalence partitioning is a technique, the input data divided into partitions of valid and invalid values for a particular field
- Consider an example for mobile no field which accept the 10 digit mobile no.
- **Example1**.So Equivalence partitioning or Valid and invalid inputs for mobile no field is

Valid I/P	Invalid I/P
0 - 9	a - z A - Z Special Symbol Space

Example 2: For Password(It should accept special character, Upper case letter, lower case letter, digits/ numbers)

Valid Input	Invalid I/P
Special symbols	Space
Digits(Numeric values)	
Special symbols	
(a – z) & (A – Z)	

Example 3: For First Name text Box(It should accept only character**)**

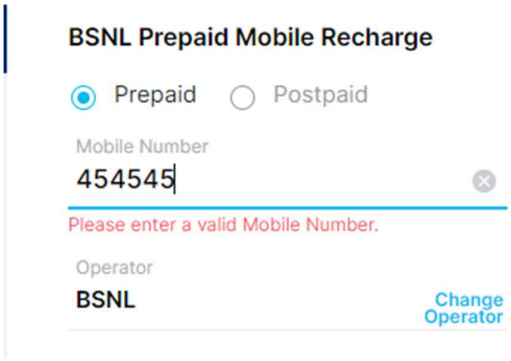
Valid Input	Invalid I/P
A – Z	Digits
a – z	Special sysmbols
	Space

3.Error Handling Coverage :

In this coverage we check whether our system/ application shows error message or not.

Example:

1. If customer requirement is mobile no field should accept 10digit number.
And Developer developed according to it.
2.If user enter 8 digits and click on submit button then system should hightlight the mobile no field with red colour with error message "Please enter 10digit mobile no."
3.So like this we check whether our application is shows error message or not.



4. Back-End Coverage :

1. In backend coverage we checks whether the data entered in front end/ UI from user will be stored in database or not.
2. And we also check whether data is get fetch from database or not

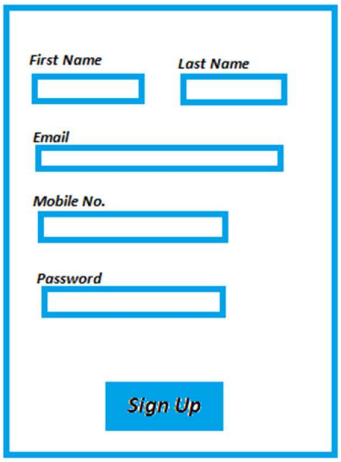


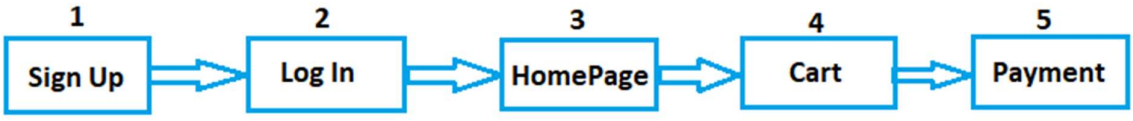
Fig : Font End page

sr.no	First Name	Last Name	Mobile no
1.	Ashish	Patil	9854249620
2.	Pawan	Pawar	9563860114

Fig : Backend (Database)

5.Service Level Coverage :

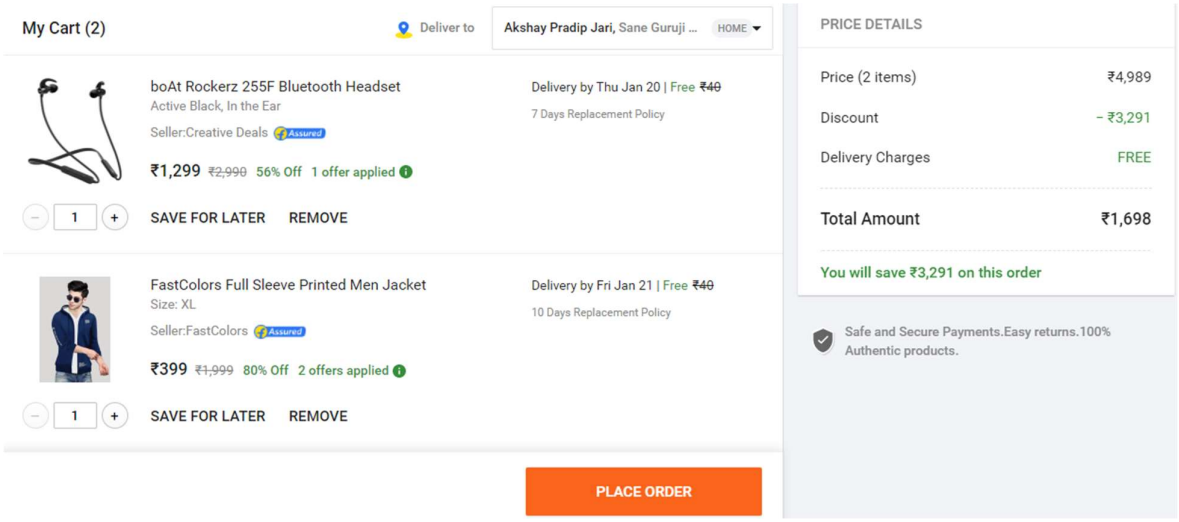
1. In the functional flow diagram BA create sequence of function & modules.
2. So this aspect of sequencetiality of functional modules get tested in service level coverage
3. Simply tester check our application is working as per functional flow diagram or not.



5.Calculation Base Coverage :

1.In Calculation Base Coverage we check whether our application is able to perform different arithematic operations or not like Addition, Substraction, Multiplication, Division.

Example
1.In case of amazon we add one product which is of 1000 rs now total cart amount is 1000 rs and again we add two products which has price of 700 now total cart value is 2400rs and if we remove one product of 700rs then the final card amount is 1700rs
2.Such types of calculation is get check in calculation base coverage.



2.Non - Functional Testing :

➔ Non - Functional Testing is a process of checking external functionality of software

- ➔ External functionality means we check whether our application is running on different **operating system** and different **browser** or not.
- ➔ 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

1.Recovery Testing :

- ➔ Recovery Testing is the process of checking whether system is able to recover from abnormal situation to Normal situation or not.
- ➔ It is also known as Reliability testing.
- ➔ This testing performes in **Service based** and **product based** company.
- ➔ Example :
  - ❖ **Google :**When we are accessing google page & suddenly internet connection lost, then google shows you're offline.

- Whenever internet connection resume page which were accessing is shown by google.
- **Amazon** : When we are buying something from amazon we add item to cart → then we enter address → redirected to Payment page and while doing payment suddenly app crashes, when we reopen the app we have to start buying process from **cart**.

➔ The recovery requirement are given by customer i.e Customer gives requirement that they want recovery from that point or from the start point.

2.Compatibility Testing :

➔ Compatibility Testing is a process of checking whether the build is compatible with user expected platform (User expected platforms means operating system & Browsers.

➔ Compatibility testing has two categories :

- **Forward compatibility testing:**
  - If build / Application is working properly but operating system & Browser is not working properly then it is called forward Compatibility test.
  - We find less no. of errors in this testing.
- **Backword compatibility testing:**
  - If operating system & Browser is working properly but build / Application is not working properly then it is called Backword Compatibility test.
  - We find more no. of errors in this testing.

➔ Compatibility testing has two types :

1. Operating system Compatibility testing (OS).
2. Browser Compatibility test

1. **Operating system Compatibility testing (OS)** :
   - Operating system (OS) Compatibility Testing is a process of checking whether the build / Application is compatible / able to work on different operating systems like **Window, Linux, Mac.**
   - This testing is performed in product based company.(We are not involve in this testing)

2. **Browser Compatibility testing** : It has two types--

   A. Cross Browser Compatibility testing.
   B. Version Comparison Compatibility testing.

A. Cross Browser Compatibility testing :

- Cross Browser Compatibility Testing is a process of checking whether the build / Application is compatible / able to work on different Browsers like **Chrome, IE, Firfox ,Microsoft Edge .**

- This testing is performed in service based company.(We are involve in this testing).

- **Example :**

  i. Test case for (Redbus) with Chrome browser

  1. Open Chrome browser
  2. Enter URL :- www.Redbus.com
  3. Enter Source and Destination location
  4. Select date of journey
  5. Click on search button

  ii. Test case for (Redbus) with Microsoft Edge browser

  1. Open Chrome browser
  2. Enter URL :- www.Redbus.com
  3. Enter Source and Destination location
  4. Select date of journey
     Click on search button

  iii. Test case for (Redbus) with *IE* browser
  iv. Test case for (Redbus) with Firefox browser

B. Version Comparison Compatibility testing.

- It is process of testing in which tester test the build / Application on different version of same browser like **IE 16, IE 17** & Chrome 96.03, Chrome 94.03, Chrome 82.03.

- For this testing tester uses VMware software which has multiple window & we select multiple version of same browser.

3.Configuration Testing :

- ➔ Configuration testing is the process of testing the system with each one of the supported software and hardware configurations.
- ➔ Means we check our application should run on different configuration of hardware & Software or not. Like➔
- ➔ Example :Operating System Configuration - Win XP, Win 7 32 bit/64 bit, Win 8 32 bit/64 bit
  Database Configuration - Oracle, DB2, MySql, MSSQL Server, Sybase
  Browser Configuration - IE 8, IE 9, FF 16.0, Chrome.
- ➔ This testing is performed in **product based company**.(We are not involve in this testing)

4.Inter-System Testing :

- ➔ It is a process of checking whether our application is able to share resources or data with other application or not.
- ➔ Generally banking domain companies uses this testing.
- ➔ This testing performes in Service based and product based company.
- ➔ **Example**: Suppose we have to make payment for Airtel mobile no recharge from Phone-pay. Then Phone-pay fetch information from Airtel, this data sharing get checked in inter-system testing.

5.Installation Testing :

- ➔ It is the process of checking installation of our build with existing software into user expected platform
- ➔ This testing performs in product based company(we are not involve in that).
- ➔ Example : Eclipse IDE, PUPG Games
- ➔ We check that customer has
  1. **Set of execution file(.exe file) which is to be install**.

2. **Easy Interface** :
    a. Interface should be user friendly
    b. Interface should be like that the user can navigate to through application easily.
3. **Disc space**:
    a. Tester check whether Disc space is available at the time of installation.
    b. If there is insufficient disc space then error should be display i.e "Insufficient disc space ".
4. Tester check during installation "**Installation is in progress" window pop- ups** or not.
5. When tester install application it should **create shortcut on desktop or not**
6. **Check Uninstallation**:
    b. In this we check whether our application can uninstall from user system or not.

6. Globalization Testing :

➔ It is process of checking whether our application support different language or not.
➔ It is use in both service base & product base company.
➔ There are three types of globalization testing.

a) **Localization testing**:
- In this testing we are check whether our application support different local language or not like **Marathi**, **Telgu**, **Bengoli**, **Gujrati**.
- **Ex. www.facebook.com**

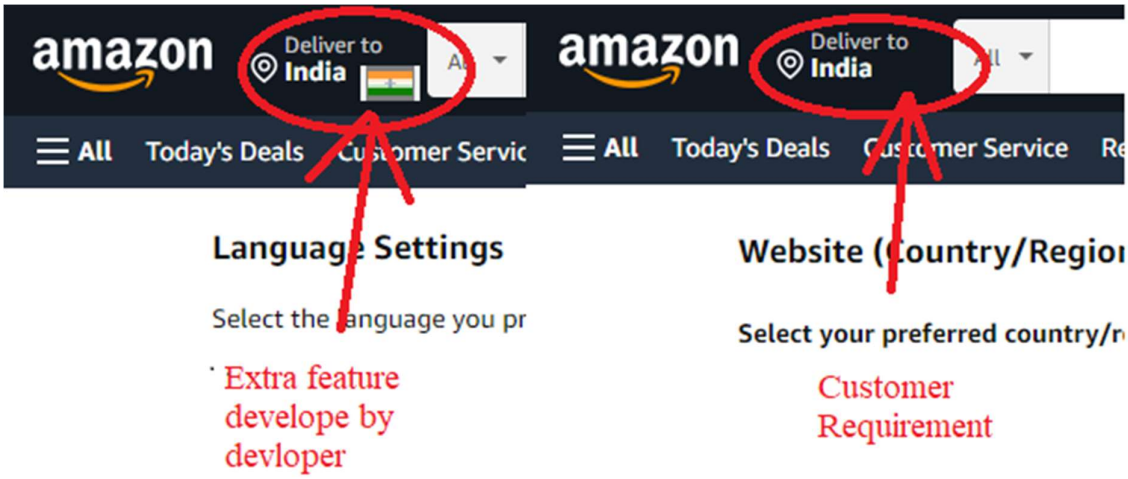
b) **Internationalization / Nationalization Testing**:
- In this testing we are check whether our application support different countries official language or not like **Spanish**, **French**, **Hindi**, **Japanese**, **German**.
- **Ex : www.amazon.com**

c) **Globalization Testing**:-
- In this testing we are check whether our application support **English** language or not.

7. Sanitation Testing :

- ➔ Sanitation testing is the method in which we test or check for extra features which are not mention in the Customer requirement.
- ➔ It is also called as garbage Testing.
- ➔ It is use in both service base & product base company.
- ➔ When tester found any extra feature in the Application / build, Tester log them as a defect & developer have to eliminate that extra feature.

➔ Example:



- ➔ Developer will suggest extra feature to the customer but he has to take a permission from BA.
- ➔ If customer accept that extra features then customer has to pay extra money for that extra feature and Tester do not log it as defect.

3.Usability Testing:

➔ It is process of checking the user friendliness of the application/UI

➔ In this testing we check

- Visual appearance of screen
- Various controls on screen like…Scroll bar, Search bar.
- Help provided by the application to understand the functionality of application to the user (Ex MS word task bar provide suggestion for every icon)
- Example : UserFriendly application →www.Naukari.com
  - Not UserFriendly application →www. majhinaukari.com

➔ It is also called as Accessibility testing

➔ For Usability testing BA makes srs as per user's aspect like design, colour.

➔ For usability testing we use tools

- WAT tool (Web Accessibility Toolbar)
- NVDA tool (NonVisual Desktop Access)

➔ Execution of these test cases happens in WAT.

➔ There are two types of Usability testing

1. GUI testing
2. Manual Support Testing.

1. GUI testing

❖ In this testing we check

- On one click next action should happens immediately.
- For blind person, if blind user click on tab. System should provide voice feedback/ talkback or not.
- This Talkback testing happens in NVDA tool.
- Also we check the speed of processing of our application how quick application respond to the user's action.

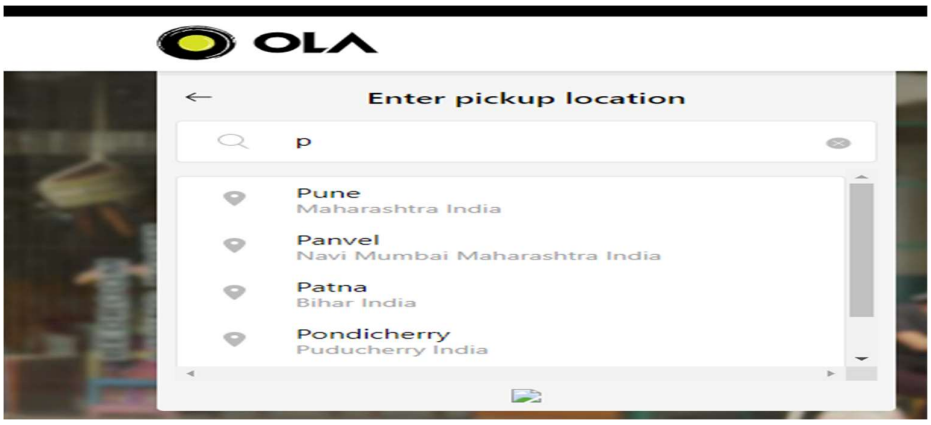
2.Manual Support testing :

❖ It is the process of checking the context sensitiveness to the user's manual input.

❖ In which we check the module or tab respond to user manual action is mention in user's request.

❖ Example: In case of OLA cabs application we have to enter the Pick up and Drop location. If we enter ' P ' in the pick up tab it will shows all the location suggestion starting with 'P'.

❖ It is also called as Regular expression testing.

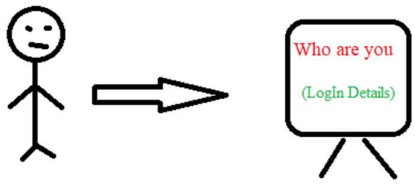


4.Security Testing:

- ➔ Security testing is a testing technique to determine that system protects data and maintains functionality as planned.
- ➔ Security testing is a process of checking privacy related to user operation
- ➔ Basically tester and developer involved in this
- ➔ There are types involve in security test

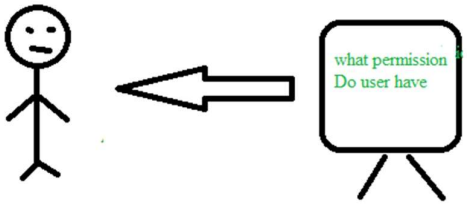
1.**Authorization**:

- ✓ It is process of checking the identity of user for providing access to the system.
- ✓ It determines whether User / person is authorized or not
- ✓ Authorized means registered Person.
- ✓ In Authorization process user's are "**verified**"



2. **Access control(Authentication):**

- ✓ It is process of checking whether authorize person has permission to access specific operation/Functionality.?
- ✓ It determines what permissions do user have?
- ✓ In Access control process user's are "**Validated**"



3. **Encryption and decryption**:

   - ✓ **Encryption**: It is process which converts original / Readable information into unrecognizable form.
   - ✓ **Decryption**: It is process which converts unrecognizable information to Readable information.
   - ✓ This is going to done by developer backend
   - ✓ This Encryption decryption is going to done between client & servers.

4.Performance Testing:

- ✓ Performance testing is a non-functional testing technique performed to determine the system parameters in terms of responsiveness and stability under various workload.
- ✓ Performance testing is process of checking speed of processing of our build.
- ✓ During this Testing we ensure that the application perform according to the customer's Requirement or not by checking the response time and under the specific load condition.
- ✓ It is done by the performance testing Engg.
- ✓ Tester used "**Load Runner tool**" **or " Jmeter tool"** for performance testing.
- ✓ This tool create situation like many user are using the application at a time.
- ✓ We are not involve in Performance testing, we only gives KT to the performance tester.

Performance Testing Techniques:

- **Load testing**- It is the simplest form of testing conducted to understand the behaviour of the system under a specific load. Load testing will result in measuring important business critical operation or functionality and load on the database, application server, etc., are also monitored.

Or

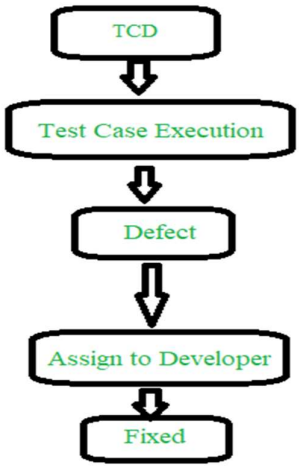
- **Load Testing** is a non-functional software testing process in which the performance of software application is tested under a specific expected load.
- It determines how the software application behaves while being accessed by multiple users simultaneously. The goal of Load Testing is to improve performance and to ensure stability and smooth functioning of software application.

----------*//*----------

- **Stress testing**- It is performed to find the upper limit capacity of the system and also to determine how the system performs if the current load goes well above the expected maximum.

==========

Re-Testing



➔ Re-testing is the method of Re-executing same build with multiple test data

➔ We performed Re-testing 2 times:-

1. **Before we log the defect we do Re-testing:**

- When we are doing testing according to test cases we check with one test data and During execution If we found any defect then this is not a valid / not a good defect .
- So here We have to perform Re-testing Before we log the defect to the Developer by Executing the same build with multiple test data.
- If we got same defect repeatedly on multiple test data then we called it as valid defect.
- So to ensure this is a valid defect we perform Re-testing here.

2. **After developer fixed / solve by the Developer :**

- Once tester found defect he assign it to developer then developer fixed / solve it and again send to the tester for recheck / Re-testing.
- We perform Re-testing after defect fixed by develop to ensure that the defect which were found and assign to developer in earlier build were fixed or not in current (Updated) build with same test cases & same test data.
- We are involve in this testing.

==========

Regression Testing

- Regression testing is the process of testing in which we are testing newly corrected build / module to ensure that they are working well & to check their impact on working of other modules.
- Regression Testing = Retesting + side effect
- Regression tester involve in this testing.(We are involve in this).
- This testing is done to make sure that new changes in code should not broken or should not have side effect on the existing functionalities.
- Regression testing happens at two times
    1. During SIT
    2. After completing SIT & UAT.

- How / When we actually do in regression testing.

    1. **When we add new scenario**:-
        - If any new scenario is get added then instead of doing SIT on that we do regression testing on that.
          Example : Suppose we have to add new scenario like we need add +91 before mobile no. so after adding we do regression testing on that.

    2. **When we have to check all failed test cases**:-
        - When we got newly corrected build we need to ensure that the new build working well or not.
        - For that we have to check /Run those fail test cases, so we perform Regression testing.
        -

    3. **When we have to execute High priority test cases**:-
        - When we move from SIT to UAT we first test only High Priority test cases.
        - High priority test case are those test cases which are related to main functionality.
        - If time permits we can check medium & Low priority test cases.

User Acceptance Testing (UAT)

➔ After Completion of SIT testing then our organization is focus on User Acceptance Testing (UAT).

➔ UAT is a process of collecting feedback from customer.

➔ People involved in UAT is

- ✓ Testing team
- ✓ Development Team
- ✓ Customer (Client).

➔ For UAT they are using Q - messenger for desktop sharing.

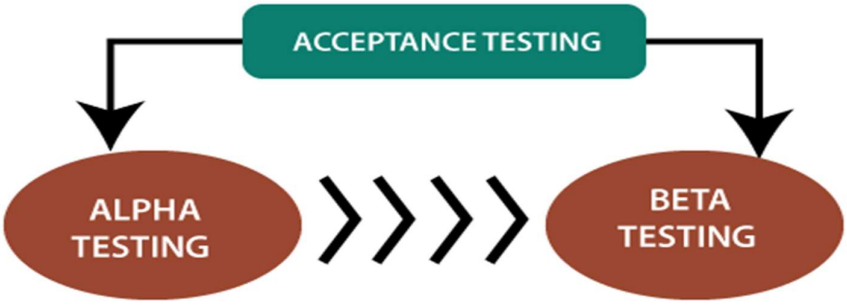
➔ Client decided from total user stories how many user stories they want to test from the tester.

➔ Then Tester check / Execute all the test cases related to that user stories as per customer requirement.

➔ Customer can also change the test data during UAT testing.

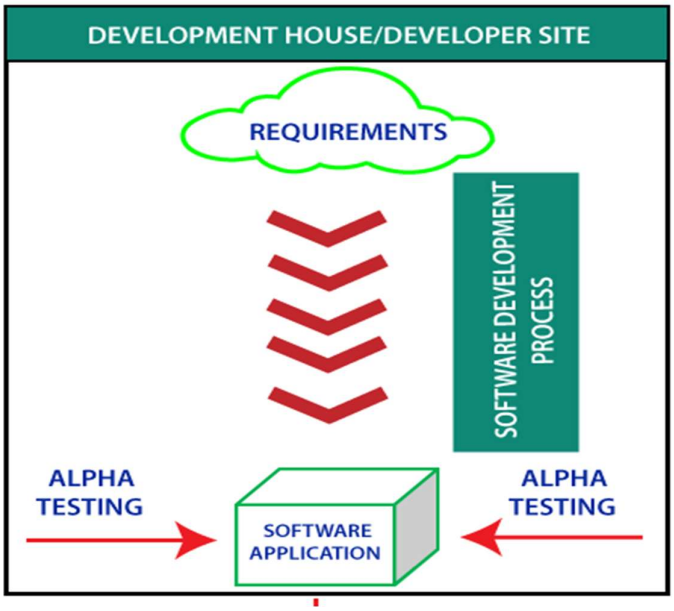
➔ It is Depends on the customer whether they have to send / Deploy application for Production or not.

➔ UAT has two types



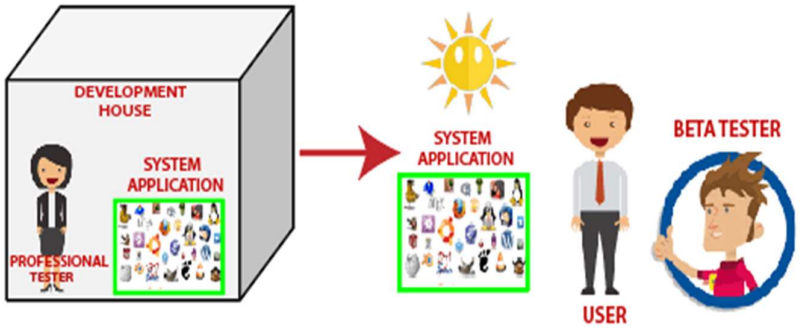
1. Alpha Testing:-

- **Alpha Testing** is a type of acceptance testing; performed to identify all possible issues and bugs before releasing the final product to the end users.
- Alpha testing is carried out by the testers who are internal employees of the organization.
- Alpha testing is conducted in **In-Controlled environment** i.e it is conducted in front of tester, developer & Presence of customers.
- Real customers are involve in this like SBI, HDFC etc.
- Alpha testing is performed in **Service based company.**



2. Beta Testing:-

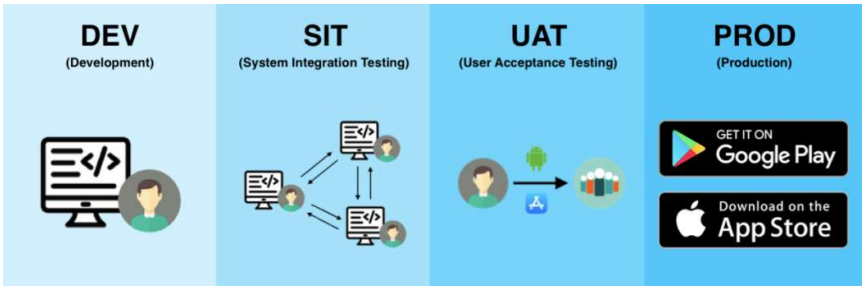
- **Beta Testing** is performed by "**real users**" of the software application in "real environment" and it can be considered as a form of external User Acceptance Testing.
- It is the final test before shipping a product to the customers. Direct feedback from customers is a major advantage of Beta Testing.
- Beta testing is conducts in **uncontrolled Environment**
- Means beta testing is released beta version to a limited audience to check the accessibility, usability, and functionality, and more.
- Internal employees of the organization are not involve in this
- Real user are Microsoft, Whats app, Chrome, PuPG.
- Beta testing is performed in **Product based company**.



BETA TESTING OF THE PRODUCT IN REAL WORLD ENVIRONMENT

==========

Hot Fix Or Production Issue



- Once we done with UAT and customer want to deploy it to production then Regression tester deploy it to the production.
- After application goes to the production, User start using the application
- While using the application, User found any defect in that then it is called as Production issue / Hot Fix.
- When such production issue found then this need to solve / fixed in one day only.
- When user found defect in production then he will contact to customer care.

- Then Customer care Representatives raise a ticket which is assign to Technical Support team (Maintenance Team) then Tester from this team check whether it is actual defect or not , if it is valid / Actual defect then Developer try to solve that issue.
- If Technical Support team not able to solve that defect then they will assign it / transferred it to the project team.
- Then Tester from project team check whether it is actual defect or not then developer from project team solve that issue then tester test it whether defect is get solve/Fixed or not .

Important Definations

Error:- A mistake in program happens due to developer is called as **error**.

Defect:- When tester found this mistake(error) and due to this the application is not working as per the requirement. then it is called as **Defect**

Bug:- When developer accept that it is Actual defect then it is called as **Bug**.

Issue:- When developer facing difficulty to solve bug then it is called as **Issue**.

Like sometimes due to defect has some side effect to other modules then it is called as Issue.

Testing Terminologies

- There are three Testing Terminologies
  - ➔ Monkey Testing
  - ➔ Adhoc Testing.
  - ➔ Exploratory Testing

1. Monkey Testing:

- When we have lots of test cases to execute but don't have time to do execution of that test cases that time we use monkey testing.
- This situation arises when dev find difficulty to solve any defect so in this case Developer takes time to solve this defect, this defect is called as Blocker defect.
- Whenever blocker defect comes we performs monkey testing.
- During this testing we test basic functionality with respect to customer requirement. Means me check/Test only high priority test cases.
- If time permits then we test medium and low priority test cases.
- We concentrate more on positive test cases and less on Negative test cases.

2.Adhoc Testing:

- When we are aware/ Familiar about application but we don't have any test cases and test data then we have to perform the adhoc testing.
- In Adhoc testing, testing performed without proper planning and documentation, it is said to be Adhoc Testing.
- Testing is carried out with the knowledge of the tester about the application and the tester tests randomly without following the specifications/requirements.
- In this case tester test the functionalities based on his previous experience & Knowledge.
- Testing is performed randomly without any test cases in order to break the system.

3.Exploratory Testing :

- When we are not aware about the application but we have test cases & test data then we performed Exploratory testing.
- As we have test cases & test data So we do step by step execution of test cases in exploratory testing.

==========

Priority & Severity

1. Priority:

- Priority is defined as the order in which a defect should be fixed.
- Priority indicates how soon the bug should be fixed.
- Priority is based on customer requirement(Business operation).
- Higher the priority sooner the defect should get fixed.
- Types of priority bugs can be categorized as
    a) High Priority
    b) Medium Priority
    c) Low Priority.

a}High Priority :-- Bug that are related to core functionality of application & that bug which blocks user to do their task

Ex : Not able to login to application

b} Medium Priority:--The bug which allow user to accomplish the task but in incorrect format.

Ex : Search option showing result but in incorrectly.

C} Low Priority:-- A bug which do not harm to system/application.

Ex: A spelling mistake in a page.

==========

2.Severity:

- Severity is the degree of impact that a defect has on the operation of product.
- Severity indicates the seriousness of the defect is affecting on the product functionality.
- Severity is associated with functionality.
- Higher the severity more the functionality is affected by the defect
- Types of Severity bugs can be categorized as
    a) High Severity
    b) Medium Severity
    c) Low Severity

a} High Severity:-- Bug which Breaks the functionality & blocks user to do their task

Ex : When we click on log button application crashes.

b} Medium Severity:--The bug which do not restrict its functionality in any way but application behave like undesirable manner

Ex : If application accept password without special symbol also(I-rise123) but it should only accept (I-rise@123).

C} Low Severity:--A bug which do not harm to application in critical way.

Ex: (1) mistake in Company logo , (2) Login button won't have colour.

Examples of Defect as per Priority&Severity

1. High Priority &Low Severity

- Example 1 : wrong design of company's logo on the website like
- 
- If logo of apple company is design wrong opposite.
- As the logo of company will be reflecting the brand image & objective of company to their customers.
  →As this defect impact on business so its priority is high & it will not break any functionality its severity is low.

Example 3 : spelling mistake in company name.

2. High priority &High Severity

- **Example 1**: Unable to log in the application
- As we know log in page is a entry point for any application, if user is not able to log then he will not able to use of application for his business.
  - ➔ As it effects of business so its priority is High and It also breaks the functionality its Severity is also High.

Example 2: If user not able to Add product into the cart.

Example 3: If user is not able to join google meet.

Example 4: when user click on about us page its shows error.

3. High Severity &Low priority

Example 1: If application crashes on passing very large input to log in for processing which is very rarely done

➔ As its breaks the application functionality so its Severity is high & as it will not impact on business because very rarely user do that, so its priority is Low.

Example 2: If user use the application on outdated browsers and company logo , modules, images etc are load properly.

4. Low Severity &Low priority

Example 1: The paragraph related to the company's description(About us page) on the website consisting of minor spelling mistakes.

→As this defect does effects on business so its priority is low & it does not break any functionality so its severity is low

Example 2- If the font color of any text in paragraph is not as per design.

Example 3- At the bottom of page in flipkart we have “Condition to use” text is not properly aligned.

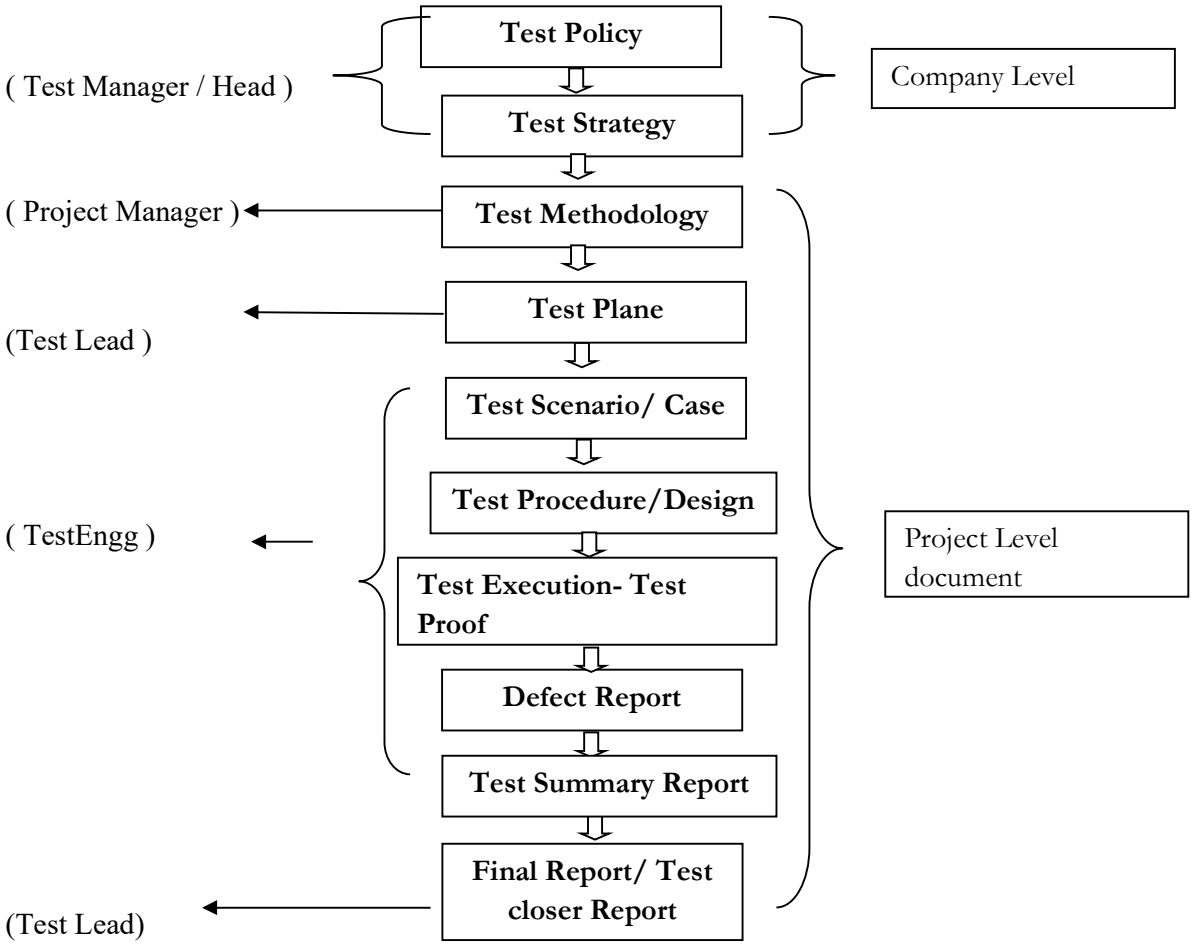
---------- **End Of Manual Part 1**----------

Manual Part 2

Syllabus :

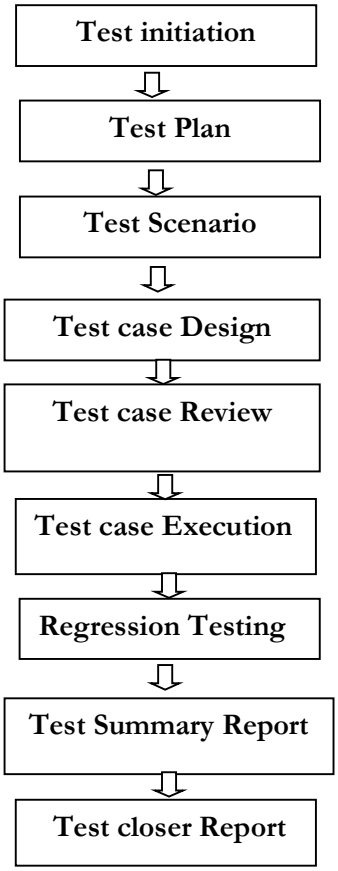
- ➔ Test document Hierarchy
- ➔ STLC
- ➔ Test case review
- ➔ TRM
- ➔ Test Plan
- ➔ Agile Test plan
- ➔ BUG Life cycle
- ➔ RTM
- ➔ Test Scenario & Test cases

Test Document hierarchy



Software Testing Life Cycle (STLC)

- ➔ STLC stand for Software testing life cycle
- ➔ STLC is a sequence of different activities performed by the testing team to ensure the quality of the software / application.
- ➔ Different stages of STLC are –



1.Test initiation:-

- ➔ In my organization, testing process start with Test initiation
- ➔ In Test initiation, PM will prepared TRM documents.
- ➔ TRM completed then sent to Test lead.
  - ➔ Project manager focuses on

- **Requirement of project** :- In this they identified domain of the project like Banking , Telecom, Healthcare,E-commerce domain.

- **Scope of project** :- Check whether the requirement are testable or not & also idenfied the scope of testing. EX. for flipkart application only those testing are selected which are related to e-commerce application.

- **Risk involve in the project** :- Risk involve in project are :-

    - Lack of resources
    - Lack of test data
    - Lack of knowledge

2.Test plan :-

Team Lead is responsible to prepare test plan & he will do team formation during test plan.

➔ Test plan includes : 1. Job allocation

2. Resources allocation

3. Estimation

1. **Job allocation** :-

➔ In scope of project application scenarios are got selected.
➔ Based on scope which types of testing we have to perform get decided
➔ Then which types of jobs are available are get decided which is called as job allocation .

2. **Resource allocation** :-

➔ In job allocation which types of test have to perform get decided
➔ Then in resources allocation who will do that testing is get decided based on that resources are allocated (Hired).

3. **Estimation** :-

➔ In Estimation they decided the time required to complete the testing means they decided the start date and end date of the testing.

3. Test Scenarios:-

- ➔ Test scenarios identified what to test in application
- ➔ It include functionality that can be tested in whole application
- ➔ Test scenarios are derived from requirement / user story
- ➔ We write only positive scenarios.

4. Test Case Design:-

- ➔ Test case indicates What to test and How to test
- ➔ It includes the information like test steps , Input data ,Actual & Expected result.
- ➔ Test cases are derived from Test scenarios
- ➔ We write positive as well as negative test cases.

5. Test case Review :

- ➔ Test case review is done by tester himself and his colleague as well.
- ➔ Tester verifies whether all the requirement are covered or not in test cases.
- ➔ Tester check for spelling & Grammatical mistake, Template format, Language used , any Duplications, wrong navigational steps.
- ➔ It will check these things which affects on the entire test execution process.

6. Test case execution :-

- ➔ After test case scenarios and test case design we execute those test cases.
- ➔ While executing if we found defect then we assign this defect to developer.
- ➔ Once developer fixed this defect then he will send us corrected build.

7.Regression testing :-

- ➔ When we get corrected build then we do regression testing to check failed test cases.
- ➔ We also check newly added scenarios (if any) during regression testing.
- ➔ And also we check high priority test cases, if time permits

  We execute medium & low priority test cases.

8.Test summary report :-

➔ Test Engg is responsible to create test summary report
And tester send it to team leader.

➔ Example:

Sr No.	Total Test	Pass	Fail	No Run	Defect
Daily	30	23	7	0	7
Monthly	300	210	60	30	60

9.Test Closure Report :-

➔ Team leader is responsible to create test closure report

➔ Test closure report include a summary of all test conducts during STLC.

➔ This report is in graphical form.

➔ Once TL create report he will send to project manager.

==========

Test Responsibility matrix

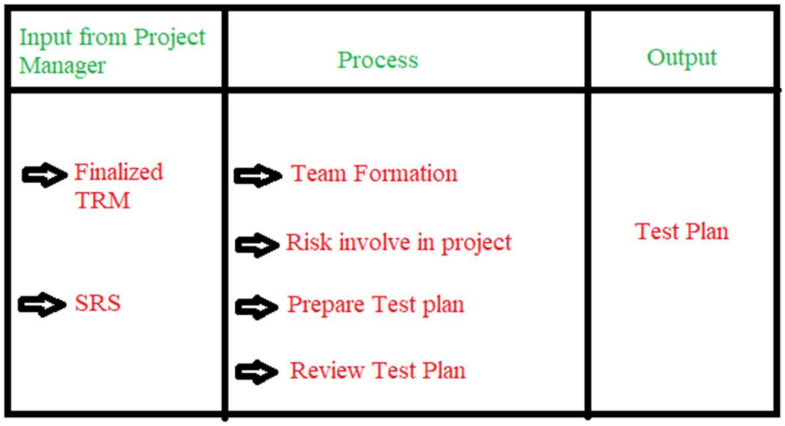
- Test responsibility matrix is mapping between test factors and development stages
- It shows which type of testing technique has to be chosen for corresponding functionality
- Different test factors are –

Test Factors	Type of Testing
1. Authorization ---------->	Security Testing
2. Acces control ---------->	Security Testing
3. Audit trial ---------->	Database Testing
4. correctness ---------->	Functional Testing
5. Continuity of processing----->	Integration Testing
6. Coupling ---------->	Intersystem Testing
7. file integrity ---------->	Recovery Testing
8. Performance ---------->	Performance Testing
9. Portability ---------->	Compatibility Testing
10. Service level ---------->	Service level Testing

==========

Test Plan (SDLC)

- ➔ Test plan is a project level document which is created by Team Lead
- ➔ A test plan is a detailed document which describes software testing areas and activities.
- ➔ Which include objectives, resources, and processes for a specific test of a software.
- ➔ Test plan include **job allocation**, **Resource allocation**, **Estimation**
- ➔ When project manager finalize TRM & SRS then he send it to team lead



Team formation :-

- ➔ **Team formation depends on TRM & SRS doc.**
- ➔ **First they consider which types of jobs are present i.e (Job allocation).**
- ➔ **Then they consider how many resources requires for this job i.e (Resources allocation)**
- ➔ **And time required to perform testing i.e (Estimation)**
- ➔ **Also checks the availability of Test environment, Required Hardware& Software, Checks whether we have required Operating System & Browsers.**

Risk involve in project :-

- Lack of resources
- Lack of test data
- Lack of knowledge

Prepare test plan :-

1. **Test plan ID**
2. **Test Items**
3. **Features to be tested & not to be tested**
4. **Finalized TRM**
5. **Test pass/Fail criteria**
6. **Test Environment**
7. **Test deliverable (** without completing 1st phase we cannot move towards 2nd phase Ex. Without test case design we cannot move to test case Review & execution)
8. **Testing Task (** without completing task assign to resource , do not give allotment of other work)
9. **Staff & training needs**
10. **Responsibility of each resource**
11. **Schedule of each task**
12. **Signature & approval**

Test Plan Review :-

➔ After completing test plan ,T.L send it to Project manager(PM)
➔ PM is responsible for Test plan Review
➔ PM focus on 3 factors

1. BRS base coverage :- Check whether test plan is as per SRS & TRM or not
2. TRM base coverage ;- Check the things mention in test plan is as per TRM or not.
3. Risk base coverage :- Check risk and their solution mentioned or not.
4. After this PM manager gives permission &Approval of finalize Test plan.

What is review &Types of Review

- Once tester done with Test Case Design (TCD) the he will send to his colleague or senior tester for review
- If any correction are found in TCD they mention comment on that test case and tester make changes or modifications
- Review include:
  - → Spelling mistake
  - → Grammatical mistakes
  - → Blocks in Navigational steps
  - → Use any unnecessary words and sentences.
  - → Is there any Missing scenarios
  - → All scenarios are converted into test cases
  - → Duplication in test cases
  - → Test Case able to mapped with requirement
  - → Maximum test coverage

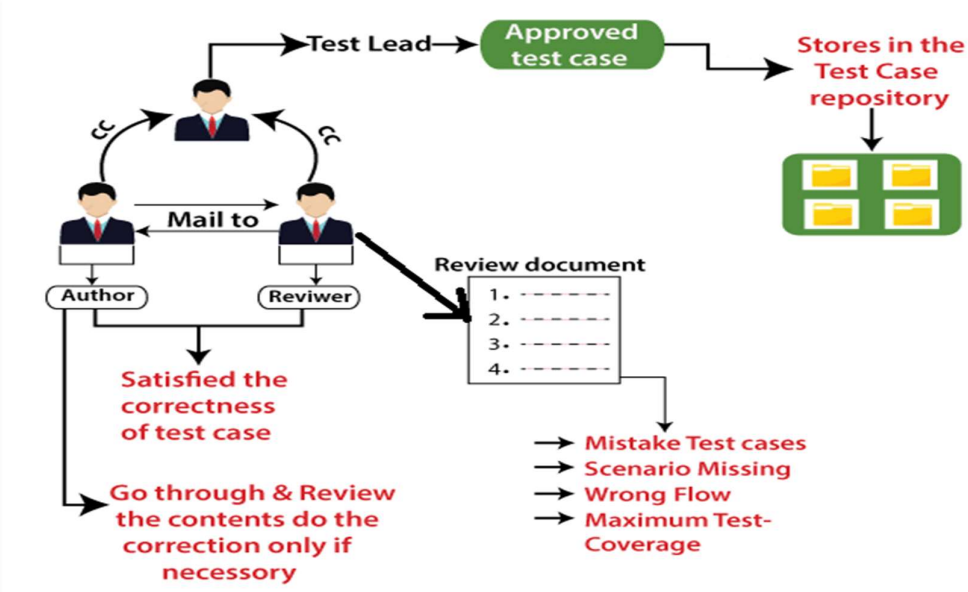
There are different types of review

1.Self Review

2. **Peer Review (In my current organization Peer Review happens)**

3. Internal Review

4. External Review.



1. Self Review;

- It is done by tester himself, who has written the test cases. He can verify whether all the requirement are covered or not by looking into SRS.

2.Peer Review;

- It is done by another tester / your colleague who hasn't written those test cases but familiar with that application or module.
- Note : In my current organization peer review happens

3.Internal Review :

- It is done by your Team Lead or project manager means test cases review by TL or PM.
- In agile product owner reviewed test cases in internal review

4.External Review :

- The review is going to done by the client during UAT.
- External client is going to involved in that.

Requirement Traceability Matrix (RTM)

- ➔ It is mapping between the **prepared test cases and Business requirement** to ensure that we have written all the test cases for each condition.
- ➔ This document is designed to ensure that all the requirement covered in testing phase or not
- ➔ RTM helps us to identified if any requirement is missing while writing test cases.
- ➔ We go for RTM two times before the execution of the test cases and after the execution of test cases.
- ➔ There are two types of RTM
    1. Forward traceability matrix
    2. Backward traceability matrix

1.Forward traceability matrix:-

It is mapping between prepared test cases with requirement(User story) that time is called as Forward traceability matrix.

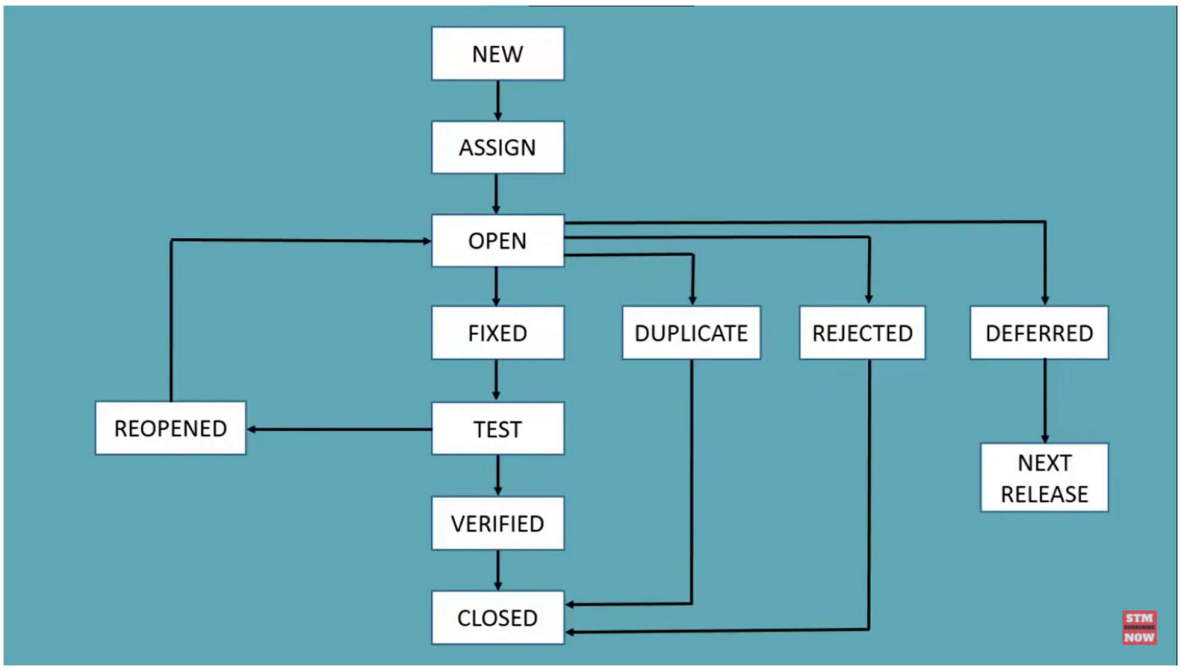
	user story	user story	user story	user story 4
TC_1	N	Y	Y	
TC_2				
TC_3				
TC_4				
TC_5				
TC_6				

2.Backward traceability matrix :-

It is mapping between the defect with the Business requirement that time is called as Backward traceability matrix.

	user story	user story	user story	user story 4
Defect_1				
Defect_2	N	Y	Y	N
Defect_3				
Defect_4				
Defect_5				
Defect_6				

Defect / Bug life Cycle



New: When tester find bug/defect then the status of that defect/bug is "**New"**.

Then tester assign that defect to developer then the status of that defect is "**Open**".

Open: Developer will check/ analysis the defects & then decide whether he has to

- ➔ **Fixed**
- ➔ **Duplicate**
- ➔ **Reject**
- ➔ **Deferred**

Fixed(If the defect is valid)

When developer fixed that defect then he will send build to tester for retesting. Then Tester verified that the defect is fixed / resolved or not

If defect is Fixed → Tester remark it as "**Closed**"

If defect is not Fixed → Tester remark it as "**Re-open**" and again assign it to developer

Duplicate(If this defect already log or found duplicate) :

If developer found the defect is duplicate then remark the status as "**Duplicate**"

Rejected (If the defect is not valid) :

If developer found that it is not a defect / or it a invalid defect then developer marked it as "**Rejected**".

Deferred (fixed in next release)

If developer accept that it is defect but this defect will not affects on the functionality(Low priority defect) So developer mark it as "**Deferred"**

This defects are not fixed in current sprint. These defects status marked as Deferred.

Differed defects will be decided by PM, BA/PO and finally Developer will mark status as Deferred

Re-open:

After performing the retesting on build and he found defect is not fixed then again tester re- assign it to developer that time tester mark it as "**Re-open"**

Closed :

The defect is verified in retest and found that given defect is fixed then tester mark it as "**Closed**" and also if the defect is rejected or found duplicate then its status is **closed**

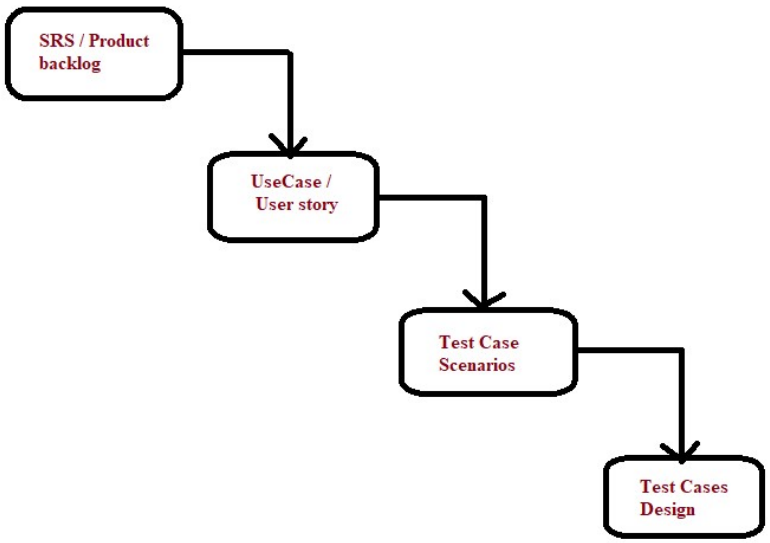
Note : There are different tools are available in market for defect logging like Bugzilla, **Jira**, HPalm, BugHerd, zipboard, mantis bug tracker, rally etc

In my current organization we are using "jira"for defect logging / defect management

Roles & Responsibilities of Tester

- User stories analysis
- Test case scenario identification
- Test case Design
- Test cases Review
- Prepare Requirement traceability matrix (RTM)
- Test cases execution
- Defect log & tracking
- Test summary report
- Meetings

Test Cases Scenarios / Test Case Design



Test scenario-

- Test scenario **derived from Use cases/ User story**
- Test scenario will define **"what to test"**
- Test scenario will **contains multiple test cases**
- Test scenario is nothing but it is **high level test cases**.
- **Ex. US- Login Page**

  Description – Login page for application

  Acceptance criteria –

  1. Login page should accept mobile no
  2. Login page should accept email id
- Ex. **Test scenario-**
  1. Verify login page functionality with mobile no.
  2. Verify login page functionality with email id

Test cases-

- Test cases **derived from Test scenario**
- Test cases will define **"how to test"**
- Test cases will **contains Test steps, Expected result, Actual result, Pass/Fail criteria**
- A test case is nothing but it is **Low level test cases**.

Ex.Test scenario- Verify login page functionality with mobile no.

- **Test cases-**

1. Verify login page by entering valid Airtel mobile no.
2. Verify login page by entering valid BSNL mobile no.
3. Verify login page by entering valid VI mobile no.
4. Verify login page by entering valid MTNL mobile no.
5. Verify login page by entering valid JIO mobile no.
6. Verify login page by entering invalid Airtel/ BSNL/VI/MTNL/JIO mobile no.

Test scenario - Verify login page functionality with email id

- Test cases-
  1. Verify login page by entering valid xyz@gmail.com
  2. Verify login page by entering valid xyz@hotmail.com
  3. Verify login page by entering valid xyz@outlook.com
  4. Verify login page by entering valid xyz@yahoo.com
  5. Verify login page by entering valid xyz@redfmail.com
  6. Verify login page by entering valid xyz@wipro.com
  7. Verify login page by entering invalid xyz@wipro.com/xyz@gmail.cometc

Agile Test Plan / Agile Sprint Plan

- In Agile duration of sprint → **4 weeks(i.e 1 month).**
- Agile Sprint plan is created by Team Lead.
- So Sprint plan for 1 month is →

Week-1 (Monday to Friday)

Grooming meeting

- Product owner(PO) , Dev& Tester arrange a Grooming session, PO Explains the requirement and Scrum team (Dev& Tester) clear his doubts.
- Grooming meeting (Understand / clear doubt US)

Sprint Planning Meeting

- In Sprint planning Meeting Product owner priorities the user story from product backlog (EPIC) and also decided the number of user story for current sprint.

  Sprint Planning Meeting → current sprint = 20 US

Story analysis :

- Tester and Dev analyze / Review the User Story and try to understand the requirement
- If Tester and Dev has any doubts then they ask their doubts to Product owner.

Test case Design:

- Based on user story / requirement tester will write test cases
- Tester write positive as well as negative testing
- Tester mapped test cases with user story.

Week-2 (Monday to Friday)

Test case Review :

Regression Test case design:

Regression Test case Review:

Week-3 (Monday to Friday)

Smoke Testing:

Test case Execution :

Defect Log & fixed

Regression Test case Execution:

Week-4 (Monday to Friday)

UAT :

Defect Report :

Test Summary report

Test Clousure report

Deployment

---

https://sitpaytm.com

https:uat//paytm.com

https://paytm.com

Test cases What-app→

- **Test scenario-**
  1. Verity that chatting message in single chat (person to person)

- **Test cases-**
  1. Verify that by passing text message (Chart, integer, etc)
  2. Verify that single tick marks in single chat
  3. Verify that double tick marks in single chat
  4. Verify that color tick marks in single chat
  5. Verify that time in single chat
  6. Verify that advance option / extra info in single chat
  7. Verify that advance option –message info in single chat
  8. Verify that advance option –reply in single chat
  9. Verify that advance option –forward in single chat
  10. Verify that advance option –forward– single message in single chat
  11. Verify that advance option –forward– double message in single chat
  12. Verify that advance option –forward– more the double message in single chat
  13. Verify that advance option –star message in single chat
  14. Verify that advance option –delete message in single chat
  15. Verify that advance option –delete message as delete for me in single chat
  16. Verify that advance option –delete message as delete for everyone in single chat
  17. Verify that advance option –delete message as cancel in single chat
  18. Verify that alignment for my message/ outgoing message in single chat
  19. Verify that alignment for incoming message in single chat

Test scenario-

2. Verity that chatting message in group chat

Test case-

1. **Similarly for group chat *do it yourself.***

==========

**********We will write Test cases by considering 3 types**********

1. Business logic test cases will considering
2. Input domain test cases
3. GUI/UI bases test cases

Ex. Test cases Pen-

Test scenario-

1. Verify that pen material/body
2. Verify that pen ink
3. Verify that pen types
4. Verify that pen size /color/ length/ dimension/ weight

1.Verify the type of pen- whether it is ball point pen, ink pen or gel pen

2. Verify the outer body of the pen- whether it should be metallic, plastic or any other material as per the specification.

3. Verify that length, breadth and other size specifications of the pen
4. Verify the weight of the pen
5. Verify if the pen is with cap or without cap
6. Verify if the pen has rubber grip or not
7. Verify the color of the ink of the pen
8. Verify the odor of the pen
9. Verify the size of the tip of the pen
10. Verify the company name or logo of the maker is correct and at desired place
11. Verify if the pen is smooth
12. Verify if the pen's ink gets leaked in case it is tilted upside down
13. Verify if the pen's gets leaked at higher altitude
14. Verify the type of surfaces the pen can write at
15. Verify if the text written by pen is erasable or not
16. Verify pen's and its ink condition at extreme temperature is as per the specification
17. Verify the pressure up to which the pen's tip can resist and work correctly
18. Verify the pen is breakable or not at a certain height as the specification
19. Verify text written by pen doesn't get faded before a certain time as per the specification
20. Verify the effect of water, oil and other liquid on the text written by pen
21. Verify the condition of ink after long period of time is as per permissible specification or not
22. Verify the total amount of text that can be written by the pen at one go
23. Verify the pen's ink is waterproof or not
24. Verify if the pen is able to write when used against the gravity- upside down
25. Verify that in case of ink pen, the pen's ink can be refilled again

---

Click on below link to know more about Test Scenarios and Test cases on various pages/functionality.

https://artoftesting.com/chair

==========

Que: What are the different parameters involve in Test case format.

Test cases format-

- In my project, we are writing text cases in excel sheet
- These are field which in present in excel sheet,

1. Test Id
2. Priority
3. Test scenario,
4. Pre-request condition,
5. Test cases,
6. Test step,
7. Excepted result,
8. Actual result,
9. Pass/ Fail criteria
10. Defect Id

==========****End of Manual part 2***==========