\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Manual Testing\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\***SDLC Life Cycle**;

- SDLC means Software Development life cycle.

- there are different Phases in SDLC life cycle.

1) Requirement Gathering

2) Analysis the Req

3) Design the application flow

4) Development

5) Testing

6) Deployment

7)maintanance

**1) Requirement Gathering**

- Busines Analsyst people is responsible to gather the req. from client.

- After gathering requirements BA team will create BRS documents.

**2) Analysis the Req**

-sr BA team will review this requirements.

- they will create one more new documents on the BRS documents.

- BA team will create SRS/FRS/CRS/PRD/CRD/Stories/Epic documents

**3) Design the application flow**

- Sr Developer / Software Design Engineer will design High Level application flow and low level application flow.

**4) Development**

- Front End developer and backend developer they will start writing the coding on the basis of req and design.

**5) Testing**

- Once development is done testing team will test the application.

**6) Deployment**

- DevOps team will deploye this application on the client enviorments.

- Deployement means sprint or release.

**Fish Model**

- Fish Model is nothing but the graphical representation of SDLC life cycle.

Fish Model

- It is graphical representation of Software Development Life cycle.

there are different phases in fish model or SDLC life cyle model

1) Requirement Gathering

2) Analysis the Requirements

3) High Level design Documents

4) Low Level Design Documents

5) Coding

6) White Box Testing

7) Black Box Testing

8) Deployment

9) maintenance

**1) Requirement Gathering**

- Business Analyst (BA) people gather the requirements from client.

- BA people will create BRS (Business Requirement Specifications) documents on the basis of Requirement given by the Client

- BRS documents define the client requirements to be developed as software/application.

- BRS documents is act as bridge between client and technical team (Design Team, Development Team and testing team , DevOps team).

- BRS is formal documents which is created on the client requirements.

e.g.

Client Requirement BRS documents

1) dual SIM 1 Device contains 2 SIM

2) ATM withdrawn Amount/ check Balance

3) Login Functionality username, password, login button./ forgot password link

4) payment using UPI

**2) Analysis the Requirements**

- BA team will analysis the BRS documents and they will create one more detailed documents as SRS Documents

- SRS documents is also known as FRS/CRS/PRD/CRD/Stories/EPIC documents

SRS means system Requirement specification

FRS means Functional Requirement specification

CRS means Client Requirement specification

PRD means Product Requirement documents

CRD means Client Requirement documents

- SRS documents is developed on the basis of BRS documents.

- One BRS documents contains multiple SRS documents created by the BA team.

- this documents is define the client requirements to be used and functional/software to be developed.

BRS Documents SRS documents

1) 1 Device contains 2 SIM 1) we insert same operator sim in one device

2) we insert different operator sim in one device

3) user can not make calls from SIM 1 to SIM 2

4) user can not make calls from SIM 2 to SIM 1

5)user can send message to SIM 1 to SIM 2

6)user can send message to SIM 2 to SIM 1

7) user can not use data at same time in 2 SIM

2) send Email in Gmail 1) compose email

i) enter TO email

ii) Enter CC

ii) Enter BCC

ii) Enter Subject Line

ii) Enter body

ii) change font

ii) attachment

2) reply

i) enter TO email

ii) Enter CC

ii) Enter BCC

ii) Enter body

ii) change font

ii) attachment

3) reply All

4) forward

-

**Do u know the SRS documents?**

yes, without knowing SRS documents how it is possible the write test cases and execute the test cases

What are the different things present inside the SRS documents

1) Functionality Requirements

2) Use cases

3) Functional Flow diagram

4) Screenshot / snapshot

5) High Level Design

1) Functional Requirements:

e.g. send Email in Gmail

1) Compose Email

2) Reply

3) Reply All

4) forward

Application should contains TO text box, CC text box,BCC text box,Subject text box, body text area, change font Drop down, attachment link, etc,,

We can add max 50 Email id in TO text box.

We can add max 50 Email id in TO text box.

we can add attachment upto 25 MB,

User can select multiple font on email body.

User can not change font on email subject line.

2) Use cases

use cases is nothing but the input, output and process.

Input : user have created gmail account, login the gmail application,

Process: user valid To email id, CC, BCC, Subject,Body, font etc..

Output: User should be displayed in Sent functionality

3) Functional Flow diagram

4) Screenshot / snapshot

- the main quetions is there is not start of development phases then how it possible to create screenshot

- to create th screenshot BA team the are using

i) HTML code

ii) Idoc software.

5) High Level Design

they will mention where this requirements we can add.

**Fish Model**

- it is graphical representation of SDLC life cycle

- there are different phased in SDLC life cycle

1) Requirement Gathering

2) Analysis the Requirement

3) Review Requirement

4) High Level Design

5) low level Design

6) Coding

7) White Box Testing

8) black box Testing

9) Deployment

10) Maintenance

-

4) High Level Design

-it is also known as Main functionality Design or External design

- This document define the structure of all possible functionality to be developed as software.

- This document is created by the Senior developer, design Engineer, Project Architect, Solution Engineer.

-To create this document design team they use Enterprise Architect , Microsoft Visio, Draw.io, Graph -VZ tool etc..

- High Level design is part of Functional flow diagram of Whole application

- In high level design, Design just combine the multiple module to make the software.

5) low level Design

- it is also known as internal design . sub functionality design

- it define static logic of each and every sub module.

- Low level design document is detailed design of High Level design documents

6) Coding

- Development they will start writing the front end coding and backend coding on the basis of SRS document and design documents

- Font end Development team use HTML, CSS, Java Script, Angular JS, React JS language to develop the Front End layer/ GUI/UI.

- Backend development team use java, python, PHP, dot net, C, C++, Perl,C#, Ruby, etc to develop the backend layers.

7) White Box Testing

- After developing front end and backend layer developer will perform WBT testing on application to test their own code before sharing to testing team.

- White box Testing is also known as Glass box testing.

- WBT is coding level testing technique used to check correstness and completeness of code/programs.

there are 3 main types of White Box Testing

1) Unit Testing

2) Operational testing

3) Mutation Testing

1) Unit Testing

- unit testing is known as Execution testing.

- in unit testing developer are going to test the smallest part of code.

- there are 3 different technique in Unit testing

1) Basic Path Coverage

- Developer are going to check blockwise execution of code using JUnit/TestNG too.

- e.g. if developer written 1000 lines code then developer will execute first 100 lines of code then 200 lines,300 lines etc.

2) Loop Coverage

- Loop Coverage means termination of loop.

- if use enter invalid credentails then proper error message should be displayed on red color.

3) Program Technique Coverage

- it means Optimization of code.

- Program should be run in such way that it should take less number of memory cycle/ CPU cycle.

2) Operational testing

- in Operational testing, developer are verifing the code in different OS and browsers.

-Developer will verify same code in Mac OS, Window 7,8,10,XP OS, Linux etc..

- Developer will verify same code in different browsers as Chrome, Edge, Safari,Opera etc..

Developer team they use browserStack tool, AWS, dockers etc.

3) Mutation Testing

- Developer are intentionally change the coding logic and and they will verify the logic.

Development team they will share build to testing team.

What are the different Object/Element Present in Web Page or web Application/Software/System.

**different Object/Element Present in Web Page**

1) Text box firstname,middle name,last name, mobile number , username , password, email id text box, debit Card Tex box,CVV, PANCARD,Aadhar Card etc..

What is text Box Bahaviour?

i) Check text box is displayed or not in Web Page

ii) check text box is enabled or not in Web page

iii) Enter the value in text box

iv) clear the text box value

v) capture/copy/cut the text box value

2) Button login button, register button, submit, cancel,save button,delete button,upload, update, clear,reset ,create new account

What is button Bahaviour?

i) check button is displayed or not in web page

ii) check button is enabled or not in web page

iii) click on button

iv) check button background color

v) check button value

3) Radio button

What is Radio button Bahaviour?

i) check radio button is displayed or not in web page

ii) check radio button is enabled or not in web page

iii) check radio button is selected or not in web page

iv) click/select the female radio button

v) if click on male radio button then female radio button is unselected.

vi) user can select only one radio button

vii) check radio button values

4) check box

i) check checkbox is displayed or not in web page

ii) check checkbox is enabled or not in webpage

iii) check checkbox is selected or not in web page

iv) if click on checkbox it become selected state

v) if click on checkbox one more time it become unselected state

vi) we can select multiple checkbox

vii) we can unselect multiple checkbox

viii) we can checkbox values

5) Drop down - date, month, year, state, city, country etc.

i) check drop down is displayed or not in web page

i) check drop down is enabled or not in web page

iii) select the drop down values

vi) check drop down is value is selected or not in web page

v) check specific value is present or not

vi) count how many values in drop down

6) link

i) check link is displayed or not

ii) check link is enabled or not

iii) click on link

iv) capture/cut/copy the link

v) check link color

vi) count total link in web page

vii) check how many invalid link in webpage

7) Image

1) general Image/static image

i) check image is displayed or not

2) Image link

i) check Image link is displayed or not

ii) check Image link is enabled or not

iii) click on Image link

iv) capture/cut/copy the Image link address

v) check Image link color

3) image button

i) check image button is displayed or not in web page

ii) check image button is enabled or not in web page

iii) click on imagebutton

iv) check image button background color

8) mouse Event

i) right click

ii) left click

iii) double click

iv) drag and drop

v) scroll down/up

vi) mouse over

vii) select the text

viii) Zoom in/out

9) keyboard Event

i) Enter text

ii) key up or key down

iii) press any key shift, caplock,fn ,control, backspace, etc..

etc..

10) calender

i) check calender displayed or not

ii) check calender enabled or not

iii) select past/future date from calender

11) video

**Black Box Testing Technique:**  **VVVVVIMMMPPPPPPPPPP**

- Black box Testing is done by the tester.

- Black box testing technique is nothing but """Test theBuild Functionality with respect to client requirement documents""""/SRS/FRS/CRD/PRD/CRD/Stories/Epic.

- Black box testing technique it start after the integration testing.

Black Box Testing Technique:

- BBT Technique is performed by Tester /QA Team.

- Without knowing the internal code we perform BBT technique on UI Layer.

- BBT Testing is technique is used to create the test cases and also we BBT technique in Functional and non Functional testing level.

-**there are different technique in BBT**

1) Equivalence Partitioning

2) Boundary Value analysis

3) Error Guessing

4) Decision Table testing

5) State Transition Tesing

1) Equivalence Partitioning:

- which types value is accepted by Object/Element we have to validate using Equivalence Partitioning:.

- also we can call that Web Object it accept which data types values.

there are 3 main DataTypes

1) Character [Upper case and lower case character]

2) Numbers [real number and decimal number]

3) Special Character.

Req:

Validate the first Name is accept the character values.

Valid Scenario: Upper case and lower cases values

Invalid Scenario: Numbers and Special Character

2) Boundary Value analysis

- we have to validate length or size of web Object/Element.

Client Req.

first name text box it should accept value between and equal to 3 to 12 Characters.

Section 1 Section 2 Section 3

firstname<3 3<=firstname<=12 firstname>12

Invalid scenario valid scenario invalid scenario

minimum : 3

maximum : 12

e.g

minimum-1 invalid

minimum 4 valid

minimum+1 valid

maximum valid

maximum+1 invalid

maximum-1 valid

Ass:

https://prafulp1030.github.io/sporthobby.html

https://prafulp1030.github.io/sporthobby.html

there are different technique in BBT

1) Equivalence Partitioning

2) Boundary Value analysis

3) Error Guessing

4) Decision Table testing

5) State Transition Tesing

3) Error Guessing Testing:

- Error Guessing testing technique Test Engineer use his/her experience/ knowledge about the application functionality to guess the errors.

- Many defect can be found using error guessing techchniques where most of front end developer/ back end developer will do mistake

at the time writing the script / code.

Application: https://prafulp1030.github.io/sporthobby.html

UI/GUI Defect:

1) last Name and state firstname first character should be upper cases.

2) aligement of application object and phyiscal text is incorrect order.

3) Email id text box should be present after last name.

4) age interested checkbox should be above submit button and below pincode and interested text "I" should be Upper case.

5) star is missing for Mandatory fields.

6) About Me Functionality name mismatched.

Functionality

1) First name it accept the numbers,special character

2)last name it accept the numbers,special character

3) email id text box it accept the numbers,special character without domain name

4) age accept long numbers

5) city text box it accept the numbers,special character

6) state text box it accept the numbers,special character

7) pincode text box it accept the 0 times number

8) reset button it not working

9) selected Interests it not showing in result page properly.

https://demoqa.com/automation-practice-form

4) Decision Table testing:

- In decision table testing it helps to tester to write tthe test cases depending on multiple inputs condition and we check the what is expected result /outcome willl get after executing the test cases.

- We write system behavior in table format.

5) State Transition Tesing

- Every software will have various possible states, and state changes from one another based on the user actions / input provide end users.

- state transition testing is helpful for write test cases in order to cover all possible application functionality.

Req. if end user enter 3 times wrong credentatils then block the accounts for next 24 hrs.

**What are the different types of Environment in software?**

there are 5 types of Environment in software.

1) Development Integration Testing (DIT) Environment

- Development they will write code in DIT Env.

- After writing the code they will test this code by using WBT.

- to perform WBT testing development team will use Junit/TestNG testing framework tool.

2) System/software Integration Testing (SIT/ (Quality Analyst)QA) Environment

- Once testing team get the Build from the Development team very first time testing team will perform Smoke testing.

- once smoke testing is done then testing team will perform Functional and Non functional testing.

3) User/Client Acceptance Testing (UAT/CAT) Environment

- once testing is done in SIT Environment, then we push code to the UAT Environment.

- and UAT team will perform Functional and Non functional testing.

4) Pre Production Environment

- once we get build in pre Production Environment and Client will will perform Functional and Non functional testing.

5) Post Production Environment

- Post Production Environment product/software goes on live.

client team will perform Functional and Non functional testing.

Development team will create build using maven or gradle tool.

**Functional Testing Types**

**1) Smoke Testing**  **VVVVVVVVVIMMMMPPPPPPPPPP**

- Smoke Testing is also known as "build verification testing" or "0 Level testing " or "Build acceptance testing" or "tester acceptance testing".

- Smoke testing is also known as sub set of Acceptance Testing.

- Smoke testing is done to make sure build we received from development team is testable or not?

- Smoke testing is done at day 0.

- Smoke testing is done at ""build level"".

- Smoke testing is helps to not waste the time to simply testing whole application when the main functionality dont work or key defects are present .

by using smoke testing we focus on core and main functionality of the application.

- to conduct the smoke we never create the test cases, we just pick the up the test cases from already written test cases.

- if main functionality is not working perfectly then testing team will reject the build,

then willl share feedback comments to the development team regarding same,and development team will team remodify the same build and they will share

that build to the testing team, and again testing will perform smoke testing. and testing team will verify build is testable or not?

- in smoke testing we pick minimal number of test cases that will not take more than is 30 to 45 mins.

**you u logged defect in smoke testing?**

No,we never log defect in smoke testing.

- if we found any critical defect then we reject the build, and if found any minor defect then we accept the build and after accepting this build,

at the time of performing functionality testing we will log defect.

**2) Retesting** **VVVVVVVVVIMMMMPPPPPPPPPP**

- If tester found any defect at the time testing application, then testing team will log the defect to the development team,

and development will fix/solve this defect and again they will share modified code/build to the testing team,

and testing team will perform retesting on modifed build.

- Retesting means re- execution of same build with multiple test data (inputs) to validate the functionality.

- Retesting is performed on failed test cases only.

- by retesting we verify defect is closed or not?

e.g

Req: username text box accept the values between 6 to 30 character.

at the time of testing this application username text box it accept the 3 character..

multiple test data/ input

Username 6 character nikhil@gmail.com

5 character ajita@gmail.com

3 character oma@gmail.com

15 character akshay12341234@gmail.com

30 character akshay12341234akshay12341234@gmail.com

29 character akshay1234234akshay12341234@gmail.com

31 character [akshay1230041234akshay12341234@gmail.com](mailto:akshay1230041234akshay12341234@gmail.com)

Project types

1) Tranditional Project -> development and testing in same company

2) off the Shelf project > development and testing in different company

3) Maintenance project

**Functional Testing types:**

1) Smoke Testing

2) Retesting

3) Regression Testing

4) Sanity Testing

**3) Regression Testing**  **VVVVVVVVVIMMMMMPPPPPPPPPPP**

- If we find any defect during the test execution, we assign this defect to development team, and development team will fix these defect

and again they will share modified build to the testing team, and testing team will verify is this """any side occur or not in existing functionality (software/ application)"" by using regression testing.

- once we add a new functionality in existing module/application then we perform regression testing.

- We Perform regression on pass test cases.

- we perform regression testing by using automation tools.

**diff between Retesting and Regression testing? VVVVVVVVVIMMMMMPPPPPPPPPPP**

**4) Sanity Testing**

- Sanity testing is done on release level to verify the main functionality of application without going deeper.

- Sanity testing is subset of regression testing.

- Due to the time constraints it is impossible to perform regression testing on application then we choose sanity testing to validate

the main and core functionality of the application.

- most of the times we dont get enough time to complete whole testing on application, especially in agile model , we get pressure

from Product Owner (BA) to complete testing in few hours or end of the day in this scenario we choose sanity testing/.

- We dont write test cases for sanity testing, we just pick up the necessary test cases from already written test cases.

- we perform sanity testing if we dont have time to perform regression testing.

**diff between smoke testing and sanity testing VVVVVVVVVVVVVVVVVVVIMMMMMMMMMMPPPPPP**

if attend 10 interviews > 8/9 company

1) diff between Retesting and Regression testing?

2) diff between smoke testing and sanity testing

3) Agile Model

4) defect life cycle

5) Priority and severity

Functional Testing Types

1) Smoke Testing

2) ReTesting

3) Regression Testing

4) Sanity Testing

5) Exploratory Testing

6) Adhoc Testing

7) Monkey testing

8) Risk based testing / priority Based Testing

5) Exploratory Testing:

- Exploring the application and test the application functionality.

- understand the system or application requirements, or refer the existing the test cases or refer existing documentation to test the application.

- When we are not aware about the application functionality and we have to test these application with the help of test data then we perform exploratory testing.

- Exploratory testing is nothing but the level by level functionality coverage.

- Exploratory testing is nothing test the unknown application (Project) step by step.

6) Adhoc Testing

- Adhoc testing is informal testing type with aim to break the system or find the defects in application.

- this testing is unplannned activity.

- Adhoc testing does not follow any process and test case design technique to create the test cases.

- Adhoc testing is done when tester have high knowledge about the application/software/ functionality.,

- Adhoc testing does not require any documentations, planning and process to be followed.

- this testing main use to find the defect in application without any documentations.

- test the known application step by step.

When we execute/perform adhoc testing.

- this testing is performed if we have more time.

- this testing is performed when we have in depth knownledge about the aplication and it time remaining.

7) Monkey testing

- Monkey testing is also called as speed testing or gorilla testing.

- Monkey testing is conducted on application to find the tricky defects.

8) Risk based testing / priority Based Testing

- Identify the critical functionality in the application and test it.

- conducting the testing on application as per the test cases priority wise.

**What u do if we have 100 test cases and need to execute the before EOD/ within the hrs? VVVVVIMMMMPPPPPPP**

- in this scenario we prioritize the test cases and first we execute high priority test cases then if time remaining then we execute the medium priority

test cases then low prority test cases.

- if we have not completed the high priority test cases then we convince to the TL/SM/PM, to extends the release dates.

- if we have completed all high priority test cases but still medium and low priority test cases is remaining then we release this application in next environment, and once start new build then we execute the remainining medium and low priority test cases.

100 Test cases

- 40 High

- 35 medium

- 25 Low

**Functional Testing Types**

1) Smoke Testing

2) ReTesting

3) Regression Testing

4) Sanity Testing

5) Exploratory Testing

6) Adhoc Testing

7) Monkey testing

8) Risk based testing / priority Based Testing

9) Integration Testing

10) user Acceptance Testing

9) Integration Testing

- After completion of White box testing, development team will integrate all module to make the software/application.

- Development team will combine all dependent and independent module to make the software as per the high level design and low level design.

- Integration testing is done by the development team as well as testing team.

- there are 3 approaches in Integration testing

1) Top down approach

2) bottom up approach

3) hybrid up approach.

**1) Top down approach**

- main module is developed by development team and sub module is under constructive then we use top down approach.

- to achieve top down approach developement team write temporary program using stub.

- temporaty stub program written in JSON/XML format

JSON - javaScript Object Notation

XML - Extensive Markup language.

- stub is also called as called program.

**2) bottom up approach**

- sub module is developed by development team and main module is under constructive then we use bottom up approach.

- to achieve bottom up approach developement team write temporary program by using driver.

- temporaty driver program written in JSON/XML format

JSON - javaScript Object Notation

XML - Extensive Markup language.

- driver is also called as calling program.

**3) hybrid approach.**

- it is combination of top down approach and bottom up approach.

- it is also called as sandwich approach.

- to achieve the hybrid approach we use strub and driver.

- stub and driver written in JSON/XML format by development team

**Non Functional Testing:**

validating the various non functional aspects of the application such as user interface, user friendliness, performance of application,

security of the application, compatibility of the aplication, etc it is called as non functional testing..

**Non Functionality Testing types**

1) compatibility Testing

2) Globalization Testing

3) installation Testing

4) uninstallation testing

5) UI/GUI Testing

1) compatibility Testing

- compatibility testing means checking application compatibility with different Operating System, browser and its version and hardware devices.

there are 3 main types on compatibility testing

1) Operating system compatibility testing

2) browser compatibility testing

3) Hardware compatibility testing

1) Operating system compatibility testing

- it checks the software should be compatiable with different Operaing system as like Window (7,8,10,11,XP), and Mac OS and linux os etc..

- to achieve the Operating system compatibility testing we use browser stack tools,

https://live.browserstack.com/

2) browser compatibility testing

- it checks the software/application should be compatiable with different browsers and its version

there are 3 types of browser compatibility testing

1) forward browser compatibility testing

2) backward browser compatibility testing

3) cross browser compatibility testing

1) forward browser compatibility testing

in forward browser compatibility testing we have to verify the behaviour of the system/application with latest version browsers.

2) backward browser compatibility testing

in backward browser compatibility testing we have to verify the behaviour of the system/application with oldest version browsers.

2) cross browser compatibility testing

in cross browser compatibility testing we have to verify the behaviour of the system/application with differents version browsers.

Client Req:

application should be support for minimum 50 version and all latest browser versions.

chrome browser :50 v

chrome browser : 122V

Edge Browser : 50 vs and 122 v

- to achieve the browser compatibility testing we use browser stack tools,

https://live.browserstack.com/

3) Hardware compatibility testing

- we verify application should be supports for different hardware devices.

2) Globalization Testing

it is also called as language compatibility testing.

we will verify weather our application support for different languages or not?

there are 3 types of language

1) Global language

it deals with English

2) Local language

it deals with local languag, Marathi, Gujarati etc..

3) international language

- it deals with france, germany etc..

3) installation Testing

checking if we are able to install the software or not as per the installation documents/user guide documents

4) uninstallation Testing

checking if we are able to ininstall the software or not as per the installation documents/user guide documents

5) UI/GUI Testing

- UI means User Interface testing

- GUI means Graphical user interface

- validating if all interface are professionally designed or not is called as UI/GUI testing.

- we verify all elements should be visible or present in web page.

- Check spelling of the Objects,

- check alignments of the Objects.

- check content displayed in Web page.

- check mandatory fields are highlights or not?

- check brackhground color, font type, font size, etc.

- check default values for objects

Non Functional Testing:

validating the various non functional aspects of the application such as user interface, user friendliness, performance of application,

security of the application, compatibility of the aplication, etc it is called as non functional testing..

**Non Functionality Testing types**

1) compatibility Testing

2) Globalization Testing

3) installation Testing

4) uninstallation testing

5) UI/GUI Testing

6)Usability Testing

7) Load Testing

8) Stress testing

9) Storage Testing

10) Data volumn testing

11) Endurance Testing

12) Authorization

13) Authentication

14) Enryption and descryption

6)Usability Testing

- it is type of software testing which we verify the user friendliness of the applications.

- in Usability Testing, we verify how easily end user able to understand the application flow and application functionality.

Performance testing

7) Load Testing

8) Stress testing

9) Storage Testing

10) Data volumn testing

11) Endurance Testing

Security testing

12) Authorization

13) Authentication

14) Enryption and descryption

Performance testing

- we test application performance by using jmeter or load runner tools.

- it opens the application in headless mode.

- and it return the response code and response line

- there are different response code series present in performance testing tools

1) 100 series : it provide the information about the pages

2) 200 series : it provide the information about the pages and we will it sucessfully web page it open.,

3) 300 series : it navigate from one page to another web page.

4) 400 series : it gives information client side errors.

5) 500 series : it gives information about the server side errors.

7) Load Testing

- Execution of application under client expected configuration and client expected load to eastimatre thee

application functionality performance is called load testing.

Req:

test the application functionality for 1lac users.

8) Stress testing

- by using stress testing we find out the application peak load capacity.

- by using stress testig, we verify how much maximum load can be handle by the application.

e.g

application it handle max load as 120K user loads.

9) Endurance Testing

Endurance Testing we verify how much time application can bear a load.

- We put Maximum load for specific amount of time to check application abnormal behavior.

e.g

application it handle max load as 120K user loads for 24hrs.

10) Storage Testing

- Storage means the capacity

- in storage testing we verify the huge amount of resources to eastimate the application performance.

- we verify the storage limits,

e.g.

application storage limit is 15GB.

11) Data volumn testing

- Data volumn testing means number records available in database.

- find the number of resources stored in databases.

- by using Data volumn testing we eastimate peak limit of data/storage in data volumn testing.

-

12) Authorization

- Authorization we have to verify user is valid or not?

13) Authentication

- Authentication we have to verify is user have specific permission to access the application functionality.

14) Enryption and descryption

- data conversion between client client and server in Enryption and descryption format.

e.g

if enter credentails it convert to the base64 format.

**Software Testing**

there are 2 main types of software testing

1) Static Testing

2) Dynamic Testing

**1) Static Testing**

- it is also known as Verification testing.

-it is also called as Non-Execution Testing beacuse under static testing we are not executing the code.

- Under the static testing we review the different documents as like requirment documents, design documents,

development documents, test plan documents, test scenario documents, test cases documents, automation test script documents,

user manually documents.

- we review this documents by using different review techniques

i) informal review

ii) formal review/ technical review

iii) Walk through review

iv) inspection review

- main objective of static testing to improve the quality of software by finding defects/errors in early stage of develoment life cycle.

- Static testing is more effective than the dyanamic testing

documents Document created by documents review by

1) Requirments documents BA sr BA

2) Design documents Design Engineer Sr Design Engineer

3) Development documents Developer sr developer

4) Test Plan Team Leader Project Manager

5) Test Scenario and test Cases documents test Engineer sr. test Engineer

6) Automation Test Script documents Automation Test Engineer Sr. Automation Test Engineer

7) Help documents/ User Manully docs BA Client

Static testing techniques:

i) informal review

- in informal review, we does not follow any process, procedure to reviews the documents.

- after review is done we provide informal comments on it.

ii) formal review/ technical review

- in Technical review we focus on technical documents, as like test strategy documents, test plan documents, test cases doicments

and web page content/design documents.

iii) Walk through review

- Author of work product/software,he will explain software to his team.

- Pariticipants can ask the questions to the author.

- Walk through review is created by the author.

- KT in the form of Walk through review.

e.g

Full Stack QA.

1) Manual Testing

i) JIRA/HP ALM/ HP QC / Azure Devops tools/bugzilla tool etc...

2) API testing

i) Postman/ SOAP UI/ Karate/ Jmeter/swagger etc..

3) Database Testing

i) SQL server, mySQL, oracle, Mongo DB, AWS Db, etc..

4) UI automation Tesing

i) Selenium, HP UFT/QTP, Karate, TOSCA, etc..

5) API Automation Testing

i) Rest Asssured, POSTMAM, TOSCA, Protector tool, UFT/QTP etc.

6) Database Automation testing

7) version Contol tools

8) Build tools

9) CI/CD tools

iv) inspection review (formal testing)

- inspection review is used to find the errors in documents.

- this meeting is lead by trained moderator/ sr Engineer

- in inspection review where strictkly focus on find the errors in documents.

- inspection review has checklist .

- and we records the erros and we give formal comments on documents.

e.g.

Review the test cases documents

1) self review

2) peer review

3) internal Reviews

4)external review

**2) Dynamic Testing**

- it is also known as validation testing or execution testing.

- We test the application after software/application developments.

- dynamic testing is done by the developer, tester and client with the functional and non functional testing.

- Dynamic testing it takes long time than the static testing.

**there are different technique in dynamic testing**

1) Unit testing

2) Integration testing

3) system testing

4) user acceptance Testing.

1) Unit testing

- unit testing is done by developer.

2) Integration testing

- in Integration testing , developer will integrate the dependent and indepdent to make the software.

and after integrating module testing team will test the modules.

3) system testing

- this testing is also called as end to end testing.

- we test application end to end flow by using system testing.

- we system testing using functional and non functional testing.

4) user acceptance Testing.

- user acceptance Testing is also called as client acceptance testing

- this testing is done by testing team and client team.

**Functional Testing**

1) User Acceptance testing

- User Acceptance testing is also called client/customer Acceptance testing.

- in UAT testing client involvement is compulsory.

- in UAT testing client will focus on user friendliness of the application/software.

- Testing team they will prepare the PPT to guide, how to use the application functionality.

- After completion of Functional and non functional testing testing team focus on UAT testing to collect the feedback from clients team.

- in UAT testing client will focus on UI layer.

- in SIT Env, we focus on UI,API and DB layer/.

there are 2 types of UAT testing

1) alpha testing

2) beta Testing

1) alpha testing

- alpha testing is done by the sr test engineer and it is done in controlled environments

- in alpha testing, testing team is focus on UI,API and DB layer

2) beta Testing

- it is done by client team, and it is done in uncontrolled environment.

- in beta testing,client is focus on UI layer.

**Manual Testing**

**Test Policy documents:**

- Test Policy documents is created by management category people.

- Test Policy documents is a master statement of work to be completed information.

- this documents contains we mention project information as like

i) main functionality in project/ main module information

ii) sub functionality in project / sub modules information

iii) Duration of project means how much time required to completed this project.

iv) cost information means what is cost of project.

v) Resource information means employee information, how many developer or tester we will allocate to develop this project.

vi) Technology information means tools names , language information, hardware , software etc..

- Objective of the test policy documents

1) Test definition

2) Test Process

3) Testing standard

4) Test Measurement and matrices

1) Test definition:

- it is combination of verification and validation

- which type of documents we will verify at the time of developing the application and who will review this documents

- which type of validation technique we will apply to validate the application functionality.

2) Test Process

- Test process is nothing but the proper planning before starting the development or testing. (Test Deliverables)

- In Test Process we mention entry criteria and exist criteria

3) Testing standard

- Which type of testing standard we use at the time of testing the application.

- at the time of testing application tester should find defect in 250 Line of development code.

e.g.

10 functional point/1 defect

4) Test Measurement and matrices

- we measure the process using PCM Technique, TMM technique, QAM technique

- PCM process capability Measurement

- TMM means Test Management Measurement

- QAM means quality Assurance Measurement

Have u created test policy documents in previous company?

No,

**Software Testing Life cycle (STLC life cycle) VVVVVVVVVVVIMMMPPPPPPPPPPPP**

- in STLC life cycle we have different phases as like

1) Test Initiation

2 Test Planning

3) Test Scenario Design

4) Test Case Design

5) Requirement Mapping or traceability matrix documents

6) Test Case Execution

7) Test Closure

1) Test Initiation

- In this stage Testing PM, will concentrate on risk involved in project, scope of the project/release/sprint and requirements of the project

- In this stage Testing PM will create test Strategy documents.

2 Test Planning

- Testing Team Leader will test the tets plan,

- Inside the Test Plan document TL will concentrate on job allocation, resource allocation and estimation.

- job allocation means in terms of What to test?, how to test? , when to test? and who will test?

- estimation means how much time required to complete the task.

3) Test Scenario Design

- Test Scenario Design means what o test.

- Sr test Engineer design the test scenario in Excel sheet and then we import to the JIRA tools using Zephyr plugin.

4) Test Case Design

- testing team create the test cases in Excel sheet then we import the test cases to the JIRA tool using Zephyr plugin

5) Requirement Mapping or traceability matrix documents

- traceability matrix documents is developed by Sr Test Engineer

- in this document we mention mapping of client requirements vs test Scenario vs test cases mapping.

6) Test Case Execution

- Testing team will execute the test cases and at the time of executing the test cases if we found any defect then we log defect to development team, and

development team will resolve this defect and again they will share modified build to the testing team and testing team will perform retesting on same modified

build.

due to change in code we have validate is any side effect occur or not using Regression testing.

- we have to the application functionality till defect is close.

7) Test Closure

- we send daily status report or weekly status reports to the Team leader and project manager

**Test Strategy documents**

- Test Strategy documents is project Level documents

- Test Strategy documents is developed by the project manager

- Test Strategy documents is define the different approach.

1) Scope or Objective or project.

-it define the what is purpose of project and definition of projects.

- Test Definition means which type of testing we are using to the application.

e.g.

functionality , non functionality,security testing, performance testing, automation etc..

- purpose means develop and test the specific domains applications.

2) Cost Analysis

- how much money we are going use on testing.

- Cost defined by the client and management category people.

3) Test Responsibility Matrix

- it is mapping between development stages vs the testing factors

- PM will mention testing name and this responsibility of the jobs

Functional testing - yes

Usability testing yes

database testing yes

API testing Yes

Hardware configuration No

Security testing yes

Performance testing Yes

etc.

Roles and Responsibilities

Team Leader - Test plan is prepared by test Team Leader

-> Test Cases reviews in JIRA

-> Defect tracking in JIRA

Sr test Engineer -> create the Test cases

-> Review the test cases

-> create traceability matrix

> test Case Execution

-> defect log and tracking

Test Engineer > create the Test cases

-> Review the test cases

> test Case Execution

-> defect log and tracking

4) Test Deliverables

-Before starting the testing documents should be completed.

- Test Deliverables means proper planning before starts the testing.

5) communication status reporting.

- PM will specify the required communication between testing team and dev team

e.g.

Jr Tester can meet the Sr. test Engineer daily.

Sr test Engineer meeat TL weekly twice

TL meet PM weekly one time.

if we found any defect then we have send trigger email to development teams or discuss in scrum meeting

6) Defect Tracking and reporting

- we log the defect in JIRA tool and we have track this defect in JIRA tools as will as in Scrum meeting.

- After log the defect, we have to send email to development teams with details information’s.

7) test Automation

- in current project or release can we add the automation scripts.

-if any scenario is repeated again and again then we choose for Automation.

- which type of technology we will use for automation means tools name.

8) Test Measurement and Matrices

- how we are going to measure the testing process

- PCM Process Capability Matrix

- QAM Quality Assurance Matrix

- TMM Test Management Matrix

9) Risk Management

- PM willl analysis the project related risk and non project related risk

10) Training needs

- Required training or KT (Knowledge Transfer) for new joiners

- if we add new technology, we have training session.

**Test Plan documents**

- Test plan is "Project Level documents" and this is documents is created by Test Team Leader or Sr Test Engineer.

- Main objective of Test Plan docs

1) Resource allocation

2) Task allocation

3) Risk analysis

4) Estimation mean time span between testing start date and end dates

there are different types of component present inside the Test Plan documents

1) Test Plan ID

- a unique name or id number

e.g

ReleaseNumber\_ProjectName\_FunctionalityName.

e.g.

R10.0\_SBIBank\_CardFunctionality

2) Iteration

- What new components or functionality we have in current release or sprint.

3) Test Items

- components or Modules to be tested in current release.

- in Test item TL will specify the important Functionality in current release requirements.

- test Item -

1) Domestic Credit Card

2) International Credit Card

4) Feature to be tested

- What feature to be tested in current release.

1) Domestic Credit Card

5) Feature not to be tested

- What feature not to be tested in current release.

1) International Credit Card

6) Test Pass and Fail criteria

- which test cases is pass or fail we have send daily status reports to the TL or PM.

7) Test Environments

- Required hardware or software to the test the applications

e.g.

We have to test same application in different version of same browser.

Application: Browser stack software.

8) Suspension criteria

- if abnormal situation a raised during the test execution as like database connectivity issue.

- then we can stop the testing temporary.

9) Test Deliverables

- Required testing task should be completed within the time.

e.g

Test Plan, Test Scenario design, test cases design, etc.

10) Training needs

- Name of selected TE and training required to team.

- Training on news tools or application / domain training

11) Responsibility

- work allocation to team mates.

Tester Name - Create the test case and execute test casees

Tester name2 review the test case and execute the test cases.

12) Schedule

all testing activity should be completed within the time.

13) Risk ands mitigations

if abnormal situation occur then how to overcome it.

- there are project related risk and non project related risk.

14) Signature and approvals

Signature of PM and PM will review the TL and they will give approval

**Defect Life Cycle/Bug Life cycle: VVVVVVIMMMMPPPPPPPPPPPP**

1) New

When tester find any defect very first time then we status of defect is new.

2) Assigned to:

- Tester Engineer assign the defect to developer.

3) Open

- Development team will open defect and they will start analysing the defect then status of defect is open

4) fix

- Development team accept the defect and they are ready to provide the solution for defect in current release/sprint.

or

- Development team make the necessary changes and they will verify the changes then status is defect is fix

5) Duplicate

- if same defect is logged by another tester then developement team changes the defect status to duplicate.

6) Deferred

- deferred status means development team they have accept the defect but they will resolve these defect in next release or future release.

- Development team they change the defect status to deferred if

i) if defect is found at the end of release and it is minor defect then development team changes the defect status to deferred.

ii) if defect is not releated to the current to current release or sprint.

iii) client is thinking to change the requirement then development team change the status to deferred.

7) Rejected

- if application or software working accoriding to specification but due to some misinterpretation development change the status to rejected.

8) Retest

- when defect is fixed and ready to test it again then testing team will change the defect status to retest.

- retest status testing team will perform retesting on the application or software.

9) Reopen

- if defect still present then testing team reopen the defect and again they will assign the defect to development team.

10) close

- if defect is no longer exist then testing team will change the status to close.

**What u do if defect is rejeced by development team?**  VVVVVIMMMPPPPPP

- first ill talk with developer with strong reason with the help of client requirement documents or stories document.

- if still defect is rejected by development team then ill talk with testing team leader and testing team leader will talk with development team leader.

- if still defect is rejected by developer then illl talk with testing project manager and testing project manager will talk with development project manager and if still defect is rejected by developer.

- then ill talk with BA or Scrum master in scrum meeting (daily standup meeting).

- if still defect is rejected by developer then ill will arrange the meeting the client with permission from project manager.

Scrum meeting VVVVVIMMMPPPP

- Scrum meeting is nothing but the daily stand up meeting.

-in Scrum meeting we discuss different things as like

i) What we did yesterday

ii) What we will do today

iii) any roadblocks, roadblocks means do u face any issue at the time of doing the task.

- Scrum meeting is conducted by scrum master (BA)

- chairperson of Scrum meeting is scrum master.

- timespace of scrum meeting is 15-20 mins per day.

- in my current company we conduct the Scrum meeting at 10.30AM to 10.50AM.

What is daily routine? VVVVVIMMMPPPP

i) Check the email

ii) Reply on important email

iii) to attent the scrum meeting

iv) we will work on task means we write the test cases, we execute the test cases, we review the test cases, we write the automation

test script, we add all files to stagging area, we commit code to local git repository, we push code to the master sub branch,

we merge code to the master branch, we review the test script code etc.

v) we do follow of previous logged defect

What is roles and responsibilities VVVVVIMMMPPPP

i) analysis the SRS documents or stories documents or client requirement documents

ii) Identify the test scenario as per the client requirement documents.

iii) Identify the test scenario for automation testing

iv) write the UI manual test cases in Excel sheet then we import the test cases to the JIRA tool.

v) write the API manual test cases in Postman tool.

vi) write the UI automation test script using selenium WebDriver library in Eclipse IDE.

vii) write the API automation test script using Rest Assured library in Eclipse IDE.

viii) write the DB automation test script using JDBC library in Eclipse IDE.

ix) we execute the manual test cases as well as automation test cases.

x) As automation testing, we have add all files to stagging area, we have commit code to local git repository and then we push code

to the GitHub/Remote sub repository and then we merge code to the master branch in GitHub Repository.

xi) Review the Manual test cases as well as automation test Script.

xii) we send daily status report to the project manager and team leader with time manner.

xiii) we have to attend the scrum meeting

Total Exp is 4 years:

1 year in manual testing and manual database testing

2 year Experience in UI Automation testing

1 year Experience in API Manual and API automation testing and database automation testing

Total Exp is 5 years:

1 year in manual testing and manual database testing

2 year Experience in UI Automation testing

2 year Experience in API Manual and API automation testing and database automation testing

Total Exp is 5 years:

1 year in manual testing and manual database testing

3 year Experience in UI Automation testing

1 year Experience in API Manual and """""API automation testing using Rest Assured""""" and database automation testing

Total Exp is 6 years:

1 year in manual testing and manual database testing

3 year Experience in UI Automation testing

2 year Experience in API Manual and API automation testing and database automation testing

Total Exp is 6 years:

1 year in manual testing and manual database testing

4 year Experience in UI Automation testing

1 year Experience in API Manual and API automation testing and database automation testing

Total Exp is 7 years:

1 year in manual testing and manual database testing

4 year Experience in UI Automation testing

2 year Experience in API Manual and API automation testing and database automation testing

Total Exp is 7 years:

1 year in manual testing and manual database testing

3 year Experience in UI Automation testing

3 year Experience in API Manual and API automation testing and database automation testing

What are the different models in software?

1) Waterfall model -> Standard Release time is 3 Month or 90 day

2) V model -> Standard Release time is 3 Month or 90 day

3) Agile model -> Standard Release time is 1 Month or 30 day but flexible

i) Scrum Agile -

Standard release time is 2 week or 3 week

Scrum ceremonies or meeting

i) Sprint Planning meeting

ii) Scrum meeting or daily standup meeting

iii) Sprint review meeting

iv) Sprint retrospective meeting

JIRA

**Severity of Defect: VVVVVIMMMPPPPPPPP**

- Severity of defect means impact of the defect on the application functionality or software functionality.

- Test Engineer will estimate the impact of defect on application functionality

- defect Severity is categorized in 3 main category

i) High Severity

ii) Medium Severity

iii) Low Severity

e.g

if application is crash

if application main/core functionality is working currectly.

if application is block due to some reason

it will consider as high Severity.

- if application is have some misspelled then low Severity.

- if application issue is look and feel then we can provide the low Severity.

- in application alignments issue then we can provide the low Severity.

**Priority of Defect**  **VVVVVIMMMPPPPPPPP**

- Priority of Defect means important of defect with respect to client requirements.

- Priority of Defect refers to urgency of defect need to be fixed early.

- Priority of Defect is given by development team.

- defect Priority is devided into the 3 main category

i) High Priority

ii) Medium Priority

iii) low Priority

i) High Priority:

High Priority defect must be fixed immediately, beacuse it impact on the application functionality.

ii) Medium Priority

- Medium Priority defect we can fix in current release or next release becuse it will less impact on application.

iii) low Priority

- low Priority defect developer will fix in current release or next release depends on application functionality.

e.g

1) if Login functionality is not working

- High Severity and High Priority

2) in phone Pay application, after sucessful payment it give the red color

- low Severity and High Priority

3) in Banking domain application, transfer functionality is not working.

- High(Medium) Severity and High Priority

4) in ECommerce application, we cannot buy the product

High Severity and High Priority

5) Spelling mistake in application logo

- low Severity and High Priority

6) Spelling mistake in application inner setting functionality

- - low Severity and low Priority

7) In facebook application, forgot account link is disable?

- low Severity and High Priority

8) Application it accept primary and alternative same email id?

- low Severity and High Priority.

9) in application, issue in discount calculation functionality

low Severity and High Priority.

10) in Telecom domain application, we can not the change the postpaid plans.

low Severity and High Priority.

11) application takes longer time to open home page.

- High Severity and High Priority.

12) after entering wrong credentails, still application navigate to home page.

- High Severity and High Priority.

13) We have to create new users in PIM Page Orange HRM application but PIM page link is disable

High Severity and High Priority.

14) After creating new employee in PIM page it generate the duplicate employee id numbers

High Severity and High Priority.

Note:

1) at the time if logging the defect we provide the priority and severity defect in JIRA tool.

2) Defect priority is decided by development team.

3) defect severity is decided by the testing team

4) test Case priority is decided by the testing team.

**Tracebility matrix**

- Tracebility matrix is mapping between the client requirements vs test scenario vs test cases.

- main purpose of Tracebility matrix is we make the sure all the test cases are covered with respect to requiements.]

- Tracebility matrix is prepared by the senior test engineer in excel sheet.

Business Requirements ID Test Scenario Test Case Id number

- **Do u involved in Production issue?**

**or**

**What is production issue? - VVVVIMMMPPP**

- If we found any defects in Production Environment then it is called as Production issue.

- After build or release goes on Production Environment, if any issue/defect occur this would be analyzed by

the development team, testing team, BA team and client team.

- If issue due to test engineer then it is called production issue means if Test Engineer have not tested application

properly or test Engineer have not test application negative scenarios on same requirement, if we found defect on same requirments in production environment then it is called as production issue.

- If issue to client end then it is called as "Request for change" or "change request (CR)" means still client have not given the requirement related defect or issue.

or

- if client have not given requirement to issue then we call issue/defect as change request.

- If we have change request then company will take extra amount from client.

- if any production issue occur then client will apply penalty on company.

**How to resolved the Production issue?**

- Production issue is analysis by the Development team, testing team ,BA team and client team.

- this Analysis is done under the Change quality board (CCB).

- Development team they modified code this is known as "Hot Fix concept".

- After getting build testing team will test the same application functionality at least 2 times.

- We will resolve the production max within 2 or 3 days.

**do u involved issue?**  VVVVIMMMMPPPPP

Yes i involved in production issue and we resolve using "Hot Fix concept".

**What is GraphQL API?**

- GraphQL API is used in mobile based application.

- GraphQL API is uses single endpoints for the multiple purpose as like retrive the entity, create a new entity, update entity and delete the entity.

- GraphQL API has ability to query multiple resources in single resources.

- GraphQL API has more precise control on data, when we retrieve the entity from the server.

What is WebHook API?

- WebHook API is used in real time notifications when specific events in occurs.

- WebHook API mostly used in Social Media applications.

Do u have experience in Mobile Based application testing

No

Do u involved in Red box Testing

No

do u involved in appium Automation testing?

No, appium tool it is used to achieve the mobile Automation Testing

**VVVVIMMMPPPPPP**

1) Diff between Retesting and regression testing

2) Diff between Sanity and Smoke testing

3) Diff between Adhoc testing and Exploratory testing

4) Diff between Priority and severity

5) what is defect/bug life cycle.

6) what is agile and its ceremonies?

7) if any defect is rejected by developer what u will do?

1) diff between DDL and DML

2) diff between delete drop and truncate

3) what is join and its types

4) find 4th max salary

1) diff between API and Web Services

2) diff between SOAP and REST

3) what are different request in API

4) diff between POST and PUT

5) diff between GET and POST

6) what is payload in API

7) what are the different variable in Postman tool

8) what are different challenages we face in API testing

9) what is use of swagger documentation

10) diff between 401 and 403 status code

11) diff between Authorization and Authentication?

12) what is use of 501,502 and 503 status code

13) what is use of 405 status code

14) if we dont add header in request what will happen

15) how to find values from the response payload.

16) how many ways we can send request body or request payload

17) how to upload files in API?

18) diff between OAuth1.0 and OAuth 2.0

19) diff between Query parameter and path parameter

Test Cases Design

-

- Test Cases should be simple

- Easy to understand for everyone

- Test cases should cover all the functionality with respect to client requirements documents

- Test Cases should be precise.

Where we write test cases?

- We write test cases inside the Excel sheet then we import the test cases to the JIRA tool using Zephyr plugins.

- we JIRA tool we install Zephyr plugin.

**Diff between test Scenario and test Cases?**

- Test Scenario define the functionality.

- Test cases define the navigational statement to achieve the functionality.

- One test Scenario contains we have multiple test cases.

- Test Scenario means what to test?

- Test Cases means how to test the functionality or scenario.

-

e.g.

Test Scenario Login Functionality PositiveScenario

Test Cases Enter Correct username

Enter Correct password

Click on login button

User will be home Page

Test Scenario Login Functionality Negative Scenario

Test Cases Enter inorrect username

Enter Correct password

Click on login button

user will be on Error Page

Test Scenario Login Functionality Negative Scenario

Test Cases Enter correct username

Enter incorrect password

Click on login button

user will be on Error Page

Test Scenario Login Functionality Negative Scenario

Test Cases Enter incorrect username

Enter incorrect password

Click on login button

user will be on Error Page

What the component present inside the Test Cases?

1) Test Scenario Name

- Test Scenario means what is exact requirements

- Test Scenario means what ot test?

- Test Scenario is unique names.

e.g

ReleaseName\_applicationName\_FunctionalityName\_Number

e.g.

R10.0\_OrangeHRMApplication\_LoginPageFunctionality\_001

2) Test Cases ID

- Test Cases id always we use unique names or numbers

e.g

ReleaseName\_applicationName\_FunctionalityName\_testCasesNameOrObjectName\_Numbers

3) Test Cases Title

- we mention Test Cases title as like To validate the functionality of "ObjectName".

4) Test Cases Priority

- Test Cases Priority means how much test Cases is important with respect to client Requirements documents

e.g

High

Medium

Low

5) test Case Description

- Test Cases Description in details,

6) Test Cases Reference documents Name

- which documents we followed to write the test cases?

e.g.

R10.0\_OrangeHRM\_SRS\_Documents.docs

7) Test Cases Steps

- What are the different steps we followed to achieve the test cases.

8) PreCondition

-Before testing the functionality what we required is called is precondition or pre- requisites.

9) Test Data

- What test data or inputs we use to achieve the test cases.

10) Expected Result

- What result we will get after executions.

11) Actual Result

- what result actual result we get after the executions.

12) Behavior of Test Cases

- we mention test cases is positive or negative.

13) Database Impact

-we have to mention is it test cases will be stored in database or not?

-

Interview TC:

1)Test Scenario Name

2) Test Cases ID

3) test case title

4) Reference docs.

5) test Cases Steps

6) test Data

7) Expected Result

-

https://demo.guru99.com/telecom/index.html

https://opensource-demo.orangehrmlive.com/web/index.php/auth/login

**Test Cases Reviews:**

- After designing test cases we focus on Test Cases reviews.

- Once we designing test cases then we name as """draft\_version""" of test cases documents.

there are 4 types of reviews of test cases.

1) Self Review

- review done by same testers.

or

- Test Engineer reviews their own test cases

2) Peer Review

- Test cases reviews by testing teammates

3) Internal Reviews

- Test Cases reviews done by Testing Team+ Dev Team + BA Team

4) External Reviews

- External Review done by client or Stake Holders.

- After modification of Test cases documents then we rename documents name as """Issue version"" of Test Cases.

Do i involved in Reviews in Previous company?

-yes I involved in reviews.

- Due to time constraints we only do """internal and external reviews""".

- internal reviews is done by testing teammates and external reviews is done by client or stake holder (Agile Model).

Email Format for internal Test Cases Reviews

To: Testing Team group Emaild id, BA eMail id, Development team group email id

CC: PM(Testing), TL (Email Id), PM(Development team email id)

Subject: Internal Reviews of the R11.0\_ApplicationName\_FunctionalityName\_TestCases\_DraftVersion

Hi Team,

Please find attached test cases of R11.0\_ApplicationName\_FunctionalityName\_TestCases\_DraftVersion.

Please reviews the test cases and please send review comments on/before 17th April 2024, EOD.

If we dont get reviews comments then this will consider as final document of the test cases.

Thanks and Regards,

Tester Name

Software Test Engineer

+91-9922120304

company logo

at the time of sending email we attach the test cases documents in email

We send email in Outlook application

Sr test Engineer they will start the reviews of the test cases

if Senior test Engineer find any defect in application then they share the reviews comments or reply All.

Reply All

Hi Kreeti,

Please find the below reviews comments

Test Case id: R11.0\_OrangeHRMApplication\_PIMPageFunctionlity\_JobDetails\_SubUnit\_TC007

have not written negative scenario test cases or mismatch in test cases steps

Thanks and Regards,

Sr, Tester Name

Senior Test Engineer

+91-9000000000

company logo

Once we done with review then test Engineer check the reviews comments then she/he will modify the test cases

and we will rename test cases documents as issueVersion documents

and again we will share the modified documents to teammates

Reply All

Hi teams,

i have modified the test cases as per the review comments.

please go through it and share the reviews comments

Thanks and Regards,

Tester Name

Software Test Engineer

+91-9922120304

company logo

Once we done with internal reviews then we focus on External reviews.

We send email to client or stake holders

To: Client email Id

CC: Test PM email, Test Team Leader, Testing group emaild Id

Subject: External Reviews of the R11.0\_ApplicationName\_FunctionalityName\_TestCases\_DraftVersion

Hi Peter

Please share your convenitent time, so we can perform the external reviews.

Thanks and Regards,

Tester Name

Software Test Engineer

+91-9922120304

company logo

Once we get reply from client or Stake Holder, then we arrange meeting on mention time and date.

Once meeting is started then we have share your screen and open a modified test cases or Issue version test cases documents

and the we explain one by one test cases to clients

In general time for external reviews is 2 days.

once we get reviews comments then we start executing the test cases.

At the time of testing the application if found any defect then we log defect to development team,

We log the defect in JIRA tool.

What are the different components we fillup at the time of logging the defect.

**What are the different defect component in JIRA tools > VVVVVVVVIMMMMPPPPPPPPPPPPP**

1) Assigned to;

we assign defect to developer.

- if we dont know the developer then we assign defect to dev team leader.

- if we dont know the developer TL then we assign defect to Test team leader.

2) Category

- there are different types of Defect Category

i) Functional

ii) Security

iii) Performance

iv) Automation

- in manual testing we choose functional as category

3) Detected in Release or Version/ Sprint

- On which version we found the defect

R11.0

4) Estimate fix time

- what is estimated time required to resolve the the defect,

5) Planned closing version

- in which version we are planning to close the defect

- if defect type is deferred defect then mention future release names.

deferred defect means we resolve this defect in next release or next of next release or future release.

6) Status of defect VVVVIMMMPPPPP

there are different types of defect status

i) new

ii) Open

iii) fix

iv) duplicate

v) rejected

vi) deferred

vii) Test

viii) reopen

ix) close

7) Browser

on which browser we found the defect

e.g.

Chrome

8) Detected by

we mention tester name

9) Defect detected date

- on which date defect is found

10) Language

- on which language we found the defect.

11) Priority of Defect

How much this defect is important as per the client requirements.

or

How much functionality important as per the client requirements.

we mention different types of Priority of defect

i) High

ii) Medium

iii) low

- Defect Priority is decided by the development teams.

12) Severity of defect.

How much this defect functionality is impacting on application / application application.

there are different types of severity of defect

i) High

ii) Medium

iii) low

- Defect Severity is decided by the testing team

13) Subject line

1 or 2 lines of defect descriptions

14) attach the defect screenshot

|  |  |
| --- | --- |
| Retesting | Regression Testing |
| Retesting is performed on fail test cases | Regression testing perform on pass test cases |
| By using Retesting, we verify original fault is fixed or not? | By regression testing we verify is any side effect occur or not? |
| We perform retesting by using manually | We perform regression testing by using manually as well as automation,. |
| Retesting is performed if any test cases is fail | Regression testing is performed if new functionality is added on existing functionality as well as if we done any changes in code. |
| Retesting it includes the defect verification | Regression does not include defect verification |

|  |  |
| --- | --- |
| Smoke Testing | Sanity testing |
| Smoke testing is done at build level | Sanity testing is done at release level |
| Smoke testing is sub set of Acceptance testing | Sanity testing is sub set of regression testing |
| Smoke testing is done on stable or unstable build | Sanity testing is done on stable build |
| We don’t log the defect in smoke testing | We log defect in sanity testing. |
| If we found any critical defect the we reject the build. | If found any critical defect we log the defect. |
| Smoke testing is done by manually | Sanity testing done by manually and automation |
| Once we get build from development team compulsory we to perform smoke testing | If we don’t have time to perform regression then we choose sanity testing. |
| Smoke testing is documented | Sanity testing is not documents |

|  |  |
| --- | --- |
| **Adhoc testing** | **Exploratory testing** |
| Adhoc testing means test the known application | Exploratory testing means the unknown application |
| Adhoc testing does not follow any documentation and process | Exploratory testing follows documentation and process |
| Adhoc testing is unplanned activity | Exploratory testing is planned activity |
| To perform Adhoc testing tester should high knowledge about the application | To perform Exploratory testing, no need to high knowledge about the application |
| To perform Adhoc testing we require less time. | To perform Exploratory testing, we require more time |
| We aware about the application | We do not aware the application |
| Adhoc testing is informal testing | Exploratory testing is formal testing |

|  |  |
| --- | --- |
| **Stub** | **Driver** |
| We stub in top-down approach | We use driver in bottom-up approach |
| Stub is also called as called program | Driver is also called as calling program |
| In stub main module is developed by developer | In driver sub module developed by developer |
| In stub sub module is absent | In driver, main module is absent |

**Waterfall Model: VVVVVVVVIMMMPPPPP**

- Waterfall model is earliest SDLC approach that is used to develop the software or application.

- Waterfall model is also known as sequentail model or linar sequentail model,

- Waterfall model is simple to understand and easy to use.

- in Waterfall model each phase is completed before starting the next phase.

-in Waterfall model each and every phase is freeze so we cannot go back to previous Phase.

- in Watefall model each and every phase they defined clearly as like requirement gathering, analysis the requirements, design the high level documents,

design the low level coding, coding phase, testing phase, deployment phase and maintenance phase.

- in Requirement Gathering phase BA team will gather the requirments from client and they will create the BRS documents.

and BA Team will analysis the BRS documents and on the basic of BRS documents, again BA will create the SRS/FRS/PRD documents.

- on the basic of SRS documents design team will design the high level and low level design.

- on the basis of design development team will the development the application functionality,

-after the developing the application functionality testing team and client team will test the application functionality.

- Once testing is done then we will deploy application to the production or prod environments.

- and it is maintenance then we will do the maintenance of the application.

- deployment/Release time for waterfall model is 3 months or 90 days.

- compulsory we have to deploye application on 3 months.

if we deploy application after the 3 months then we have to pay the penalty charges to the client team.

- Waterfall model used When

- Requirements are clear and fixed.

- there is no confusion in Requirements

- It is good to use then technology is understable.

- this model is used when project is short and low cost.

- When risk is minimum then we implements the waterfall model.

**Advantages of Waterfall model**

-Simple and easy to understand

- clearly defined the project scope, technology, task and phases.

- Project is small

- Easy to manage the task because each and every phase is freeze, or each and every phase is completed at one time.

- Well understood deliverable and tasks.

- welll define the reviews process

- Result are well documents

**disadvantages of Waterfall model**

- Change requirement are not possible

- We can not use waterfall model for complex project

- High amounts of risk and uncertanity.

- poor model of long and ongoing project.

- it is difficult to measure the process within the stages/phases.

- Not suitable model for changing the requirements

- Testing is phase is not started until the development is completed.

- Each and every phase is freeze, we cannot go previous phase in waterfall

**V Model:**

- V model is also called as Verification and validation.

- V model is also known as Mapping between the verification and validation.

- On left hand side the development teams work and right hand side testing team works.

- In V model Development and testing team work parallely.

- Release time for V Model is 3 Months or 90 days.

- in V Model we map the development phase to the testing phases.

**Business Requirement Specification Vs Acceptance testing.**

- BA Team will gather the requirements from client and they will create the formal BRS documents.

- and Client derive the acceptance of test on the basis of BRS documents, Client will verify the application is developed as per

the Requirement documents are not?.

- Acceptance testing is nothing the UAT testing.

**System/Software Requirement Specification Vs System Testing.**

- BA team will create the SRS documents on the basis of BRS documents.

- Once SRS documents is created then BA team will share the SRS documents to the Design team, Dev team and test teams.

- Testing team test the application main functionality as welll as sub functionality on the basis of SRS documents.

- We refer the SRS documents to create test cases, and to execute the test cases.

**High Level Design Vs Integration Testing.**

- Design Engineer will create the High Level Design of the application on the Requirements documents.

and once they create the HLD documents then development team willl integrate the all high modules to make the software.

- After combining the high level modules, testing will test the the all high modules.

**Low Level design Vs Unit testing.**

- As per the low level design, Development team they are developing the application functionality

and then development team they are performing the unit on the application.

**DisAdv of the V Model**

- it is expensive and costly model

- Poor Model for ongoing and complex project.

- if Requirements is flexible then we can not implements this model.

-This model is not suitable for changing requirements.

- V Model is time consuming, to compare with the Agile Model, release time for V model is 3 months and release time for agile model is 1 months

What is means iterative/increment approach?

What is time box for sprint?

if we cant complete the requirement within time box what you will do?

What are the different types of Agile model

What are the different types of artifact in Agile model

diff between Sprint planning meeting and sprint review meeting?

**Agile Model: VVVVVVIMMMPPPPPPPPPP**

- Agile model is an iterative approach to traditional project management tool typical used in software development life cycle.

- Iterative means we split the development of software into the sequence of repeated cycle using sprint.

- Each sprint is fixed time length is known as time box.

- ""in my current company we follow 2/3 week sprint time box"".

- within the time box we develop small requirement and we deploye this developed requirement to client environments.

- Standard release time for agile model is 1 Month or 30 days but in my current we follow 2 sprint life cycle/release time box.

- but agile time is flexible.

- in agile if any requirements we can not complete within the time box then we add same requirements in next sprint

or we extend the sprint time box.

- As compare with V/Waterfall model, V/waterfall model is known planned driven model and agile model is known as value driven model.

- Value driven model means frequent changes in requirements does not impact on the development, testing and production life cycle.

- in agile model any stages of SDLC life cycle, may requirement come, in agile model we have accept this requirements and we have to

fulfill this requirement within the time without taking any extra charges for small requirements changes.

- in agile model higher priority is to satisfy the stake holder or client team,

- if any high priority requirement comes then we accept the requirements and we fulfil it in current sprint time box.

there are different types of agile model

1) Scrum Agile model

2) Lean Software Agile model

3) Kanban Agile model

4) Crystal clear Agile model

5) Extrame Programming Agile Model

6) Dynamic Software Development Method (DSDM) Agile model

7) Feature Driven Development (FDD) Agile Model

-in my current company we use scrum agile mode.

- in Scrum Agile Model we have 3 main artifact

1) Product Backlog

2) Sprint Backlog

3) Increment definition to done

- in Scrum Agile Model different types of scrum ceremonies/meeting

1) Sprint Growing Meeting

2) Sprint Planning meeting

3) Scrum Meeting /Daily stand up Meeting

4) Sprint Review Meeting

5) Sprint Retrospective Meeting

in Scrum Agile model we have """Scrum Team"", in scrum team contains we have different people as like

1) Product Owner/ Product Manager

2) Scrum Master

3) Development team

4) Testing team

**1) Stake Holder;**

- Stake Holder means client, who will provide the requirements to the Product Owner.

- Product Owner actively engaged with the stake holder to gather requirements and provide the priority for the requirements

**2)Product Owner (PO)**

- PO will gather the requirement from stake holder and provide the priority for the requirements and Product Owner will represent

the business, and PO tells to scrum master or scrum team what is important to deliver in upcoming sprint with the help of Sprint planning meeting.

- This is most important responsbility of the PO.

**3) Product Backlog documents**

- Product backlog document is master list of all requirement documents.

- Product Backlog documents is overall requirment documents.

-Product Backlog documents is also called as dynamic requirement documents because in this product Backog we add the some requirement or may be we remove the some requirement.

- this document is created by the Product Owner.

- Product Owner constantly revisit Product Backlog documents and he will change the requirement priority as per the growing meeting

**4) Sprint Planning Meeting.**

- Sprint Planning Meeting is represented by the Product Owner (PO).

- Sprint Planning meeting is event in scrum that define what can be deploy / deliver as part of the upcoming sprint.

- PO describe the meeting goals, What are the different requirements we can take as part of upcoming sprint.

- Scrum team will analysis the requirements and they will negotiation with PO regarding requirements.

- Sometime PO is forcing to take more requirements in current sprint, once the requirement is anlysised by the scrum team,

Scrum team will removed the some requirements from upcoming sprint.

- Once we accept the requirements then scrum team start work necessary to achieve the next goal.

- We can not conduct the sprint planning meetig without PO and Scrum team.

- Time box for sprint planning meeting is 2 hrs for 2 weeks

**5) Sprint Backlog Documents**

- Sprint Backlog Documents is list of task to be completed in upcoming sprint.

- During the Sprint planning meeting, scrum team already selected the some product backlog items/ requirements/stories/epics.

- in agile we call it is an iteration.

- then Scrum master will the start to achieve the upcoming sprint goal.

- if we develop the goal within the time box then we start working on next sprint.

**6) Stories documents**

- on the basis of Sprint Backlog Documents PO will create detailed requirements as Stories documents.

- Stories / User stories documents is created by Product Owner PO.

- Stories documents is nothing but the SRS/FRS documents.

- one Product Backlog contains we have multiple Sprint Backlog documents

- one Sprint Backlog documents contains we have multiple Stories

- once the sprint is started then Scrum master will allocate task to the scrum team with the help of scrum meeting.

**7) Scrum meeting**

- Scrum meeting is also known as daily standup meeting

- in this meeting we discuss different things as like

i) What we did yesterday?

ii) what we wil do today?

iii) whate the impediments or roadblocks or issue.

- chairperson of scrum meeting is scrum master,

- time box this meeting is 15-20 mins.

- Each and every scrum team members should be involve.

- in my current company, Scrum meeting time is 10.30AM to 10.50AM

- in my current company, Scrum meeting time is 1.30PM to 1.50PM

in my current company, Scrum meeting time is 4.00PM to 4.20PM

- in my current company, Scrum meeting time is 6.30PM to 6.50PM

**8) Sprint Review meeting**

- Chaiperson of this meeting is stake holder or client

- this meeting is attended by the stake holder, PO and Scrum master, developer and tester.

- Testing team give domo to the stake holder on current sprint developed requirements.

- by using sprint review meeting stake holder will determine what is finished and what is not finished.

- Main purpose of Sprint Review meeting, scrum team to show what work is completed within the sprint time box and will compare

communitment given when sprint it started with the help of sprint planning meeting.

- in Sprint Review meeting we discuss main 4 things

i) What work has been done?

We have tested the Login Page, Home Page and PIM Page

ii) What work has not been done?

we have not the completed the Admin Page

iii) what work has been added in the active/ current sprint?

Search employee in PIM Page

iv) What work is removed from the current sprint?

Add More Job details in PIM Page

- Once we done with Sprint Review meeting, Scrum team will ask to the stake holder to review the works/goal/developed requirements/

- Scrum team ask to stake holder to share the review feedback comments regarding the work is completed or not / goal is completed or not?

**Sprint Retrospective Meeting VVVVIMMMMPPPPPP**

- Chairperson of Sprint Retrospective meeting is Product Owner or Scrum master.

- in current company we conduct the Sprint Retrospective meeting after 3/4/5 sprint.

- Sprint Retrospective meeting is known as internal meeting or improvement meeting.

- Main goal of Sprint Retrospective meeting is

i) What went well in last 3/4/5 sprint.

ii) What could be improvement required releated project or tools in each and every sprint.

iii) What commitment we provide to improve in upcoming sprints.

- Time duration of this meeting is 2 hrs for 2 week sprint.

**Diff between EPIC and Stroies(user Stories)**  VVVVIMMMMPPPPPP

**EPIC documents:**

- EPIC documents is main functionality documents.

- EPIC documents is also known as summary documents,epic can be broken down into the multiple user stories.

**Stories/User Stories:**

- Stories is small piece of work that can be done within the time box or within the sprint.

- Stories is also called as SRS/FRS documents.

- Stories is known as sub functionality documents of the EPIC documents.

One EPIC documents contains we have multiple user stories, and 1 User stories broken down into the multiple task.

**What is Burn up and Burn down chart?**  VVVVIMMMMPPPPPP

**Burn up chart:**

- How much work is completed in current sprint or release is called as Burn up chart

**Burn down chart**

- How much work is remaining in current sprint or release is called as burn down chart.

**What is Velocity in agile**

- Measure total Amount of work that team can be deliver during the sprint or release.

**What is Sprint Grooming Meeting?**

- Sprint Grooming Meeting is also called as backlog refinement meeting.

- Sprint Grooming Meeting is process of reviewing, proritizing and updating the product backlog items.

- Product Backlog is list of feature that be completed in upcoming sprint/ release.

- Sprint Grooming Meeting helps to keep manage, clear and relevant item to pick up in next sprint.

- in Sprint Grooming Meeting we shortlist the product backlog items as per the priority wise.

- Sprint Grooming Meeting is conducted between stake holder, Product Owner and Scrum master.

- this meeting conducted before the sprint planning meeting.

- Time box for this meeting is based on how much item is present inside the Product backlogs.

------------------------------------------------------------------------------------------------------------------------------------