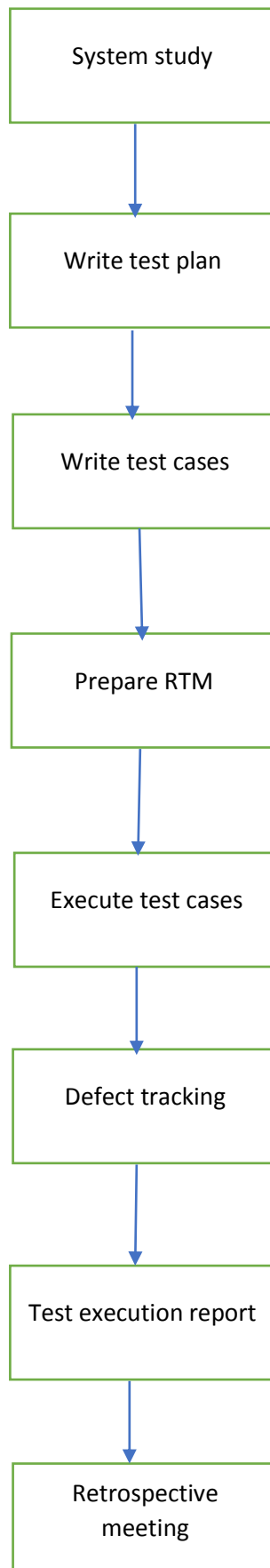


SOFTWARE TESTING LIFE CYCLE (STLC)

It is the procedure to test the software application. It has a different stages and phases like



System study: It is going through the requirement given by the customer and understands how the system works.

Write test plan: it is a document which is prepared for feature to do the testing activities.

It is done by the test lead or test manager in the testing field. Because the plans were done by experienced people.

Write test cases: it is a step by step procedure to perform the testing on the application. It is done by the test engineer. Once we go through the requirement, we identify the scenarios and then converted into the test cases. To write the test case we need requirement and test case template or tools (qc/alm, JIRA)

Prepare RTM:(requirement tractability matrix): it is a document which is prepared to check whether every requirement has at least one test cases.

To prepare RTM we need both requirement and test cases.

Execute test cases: once the requirement is given to the test engineer. He writes test cases for the application. After the developer gives the developed application then the test cases are compared with expected result and actual result. If the expected result and actual result are same then the status is pass, if the expected result and actual result are not same status will be fail. This is called execute test cases. So, to execute test cases we need test cases and software application. This is where the exactly the software is been tested. This is the most important phases of software testing life cycle.

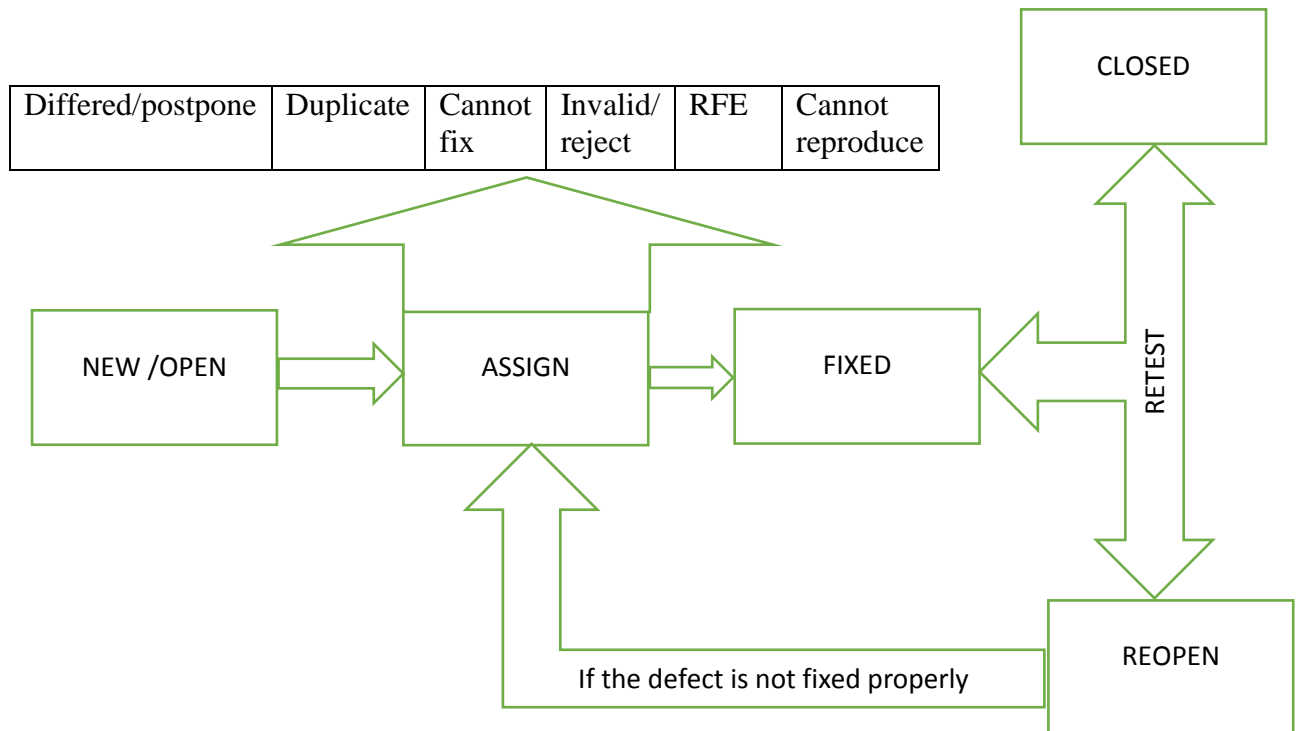
Defect tracking: while executing the test cases we may come across the defects. These defects are raised to the developers and they accept defects and starts fixing it. This is called defect tracking. It is done by using tool called “Bugzilla”.

Test execution report: once we execute the test cases. we should prepare one document called test execution report. This document tells about how many test cases are pass and how many test cases are fail.

Till here customer involvement will be there according to the customer this is the end of the project.

Retrospective meeting: in this meeting all the team members will gather together like PM, TL, DL, TL DEV, BA, TM and they discuss about the good and improvements needed once the project is done. This will be very useful for next release or next project.

DEFECT LIFE CYCLE OR BUG LIFE CYCLE



When we are executing the test cases if the expected result and actual result are not same we come across the defect, then the defect are raised to the developer then the status will be **new/open**. And the new defect is **assigned** to the developer or development lead. The developer will reproduce the defect and accepts it. Then he start fixing the defect in the development server. When he fixes the defect he will change the status as **fixed**. The defect which is fixed is installed in testing server. The test engineer start **retesting** the defect fixed in the testing server. If the defect is properly fixed then the status will be **closed**. If the status is not properly fixed then again, the defect is **reopened** to the developer.

What is differed/postpone?

Whenever test engineer raises the defect the developer accepts the defect but he will not fix the defect. He will give the status as differed/postpone.

Whenever there is a major defect or minor defect. Developer have less time to fix the major defect. In this case the developer will fix the major defect and for minor defect he will assign status as differed or postpone.

What is duplicate status?

Whenever the test engineer finds the defects and he did not raise it. If another test engineer finds the same defect and he raise it unknowingly. Then this status is known as duplicate.

To avoid this when the test engineer finds the defects, he should raise it immediately so that there will not be duplicate defects

Can't be reproduce

When the test engineer finds the defects but developer cannot be able to reproduce that defect. So he will change the status as “cannot be reproduce”.

Ex: Mobile hang .. etc.

Reasons for can't be reproduce

1. Installation problem
2. Improper defect report.
3. Inconsistent bug.

Can't be fixed

Whenever the developer is unable to fix the defect that is raised by the test engineer, then he changes the status as “can't be fixed”. And it is a void defect but he can't fix the defect.

Even though the developer does not fix the defect, it should not affect the business. If the technology doesn't support to fix the defect.

Invalid/reject

Whenever the test engineer raises the defects but developer will not accept the defects and he change the status as invalid/reject.

Reasons for Invalid

Due to the misunderstanding of the requirement.

1. If test engineers misunderstand requirement

New/open → assign → invalid → closed

2. If developer misunderstand the requirement

New/open → assign → invalid → reopen → fixed → closed

Request for enhancement (RFE)

Whenever the test engineer raises the defect which is not given in the requirement so the developer will take it as suggestion. In this case he will change the status as RFE.

SEVERITY

Severity will tell how much that defect effect to the customer business is known as severity

Types of severity

1. Blocker
2. Critical
3. Major
4. Minor
5. Trivial: this defect is negligible.

PRIORITY

Priority says which defect has to be fixed first by the developer. For every defect we have to set priority.

Different types of priority

1. High
2. Medium
3. Low

Who will set severity and priority?

- Test engineer will set severity and priority.
- Developer or managers can discuss about this if they want to change it they have to provide proper justification.
- Priority is very much important for the developer to decide which defect has to be fixed first
- If priority is not their developer may fix the easy defects and they leave all the important defects
- As a project point of view the important part of defects has to be fix earlier.to manage these things we have severity and priority for each and every defect.

TEST CASE TEMPLATE

Test case name						
Project name						
Module name						
Requirement number						
Test data						
preconditions						
Test case type						
priority						
Test case environment						
Test case description						
Test steps						
Sl.no	description	Input	Expected result	Actual result	Status	comments

Author	
Reviewer	
Approved by	
Approved date	

Header Part:

Test data: it is the data we need to execute the scenarios.

Pre conditions: it is the action which we should do before executing the scenarios.

Test case type: in this section we should write which type of testing we are going to use.

Priority: assigning the priority to each and every test cases will help us in prioritizing the defects

Test case description: summery of test cases.

DEFECT REPORT

- When ever the test engineer finds the defect it has to be raised or logged or reported to developer. For this we have to create a defect report.
- When we create the defect report in tools like BUGZILLA, or by using the excel file or word documents

Template for defect report

Defect ID	This will be auto generated if we use the tools
Build no	It is the number of build where defect has found
Test case no	It is a number of the test case where we found this defect
Status	
severity	
Priority	
Test environment	
Module no	
Reported by	
Brief description	
Test data	

Steps To Reproduce:

--

Expected result	
Actual result	
Attachment of screen shot	

ACCEPTANCE TESTING

It is testing which is mainly done to check the business scenarios of the application which is done by the customer.

(or)

It is an end to end testing which is done by the customer to ensure the application is fit for business scenarios or not.

Why acceptance testing is done?

- This is mainly done to get a confidence for customer before he releases the product to the end users.
- Because of business pressures, software company might be releasing the project to the customer with some defects. To ensure this customer will do acceptance testing.
- If the product is released to the end user without checking the business scenarios, it will affect the customer business. To avoid this acceptance testing has to be done.

Approaches of acceptance testing.

- BA or IT employees of customer will do acceptance testing at customer place.
- We may forget some of the business scenarios to test and those scenarios would be tested by the customer

Note:

If the customer finds the defect during the acceptance testing. That is a negative path to the test engineers. So before finding the any defect we should think all those business scenarios and finds defects.

Alpha Testing and Beta Testing are types of **Acceptance Testing**.

Difference between Alpha and Beta Testing

Alpha Testing	Beta Testing
Alpha Testing is performed by tester who are usually an internal employee of an organization.	Beta Testing is performed by the client or end users who are not an employee of an organization
Alpha Testing is performed by the developer site	Beta Testing is performed at the client location or end user of product
Reliability and Security Testing are not performed in depth alpha testing	Reliability and Security and Robustness are checked during beta testing
Alpha testing involved both black box and white box testing techniques	Beta testing typically uses black box testing Techniques
Alpha testing requires lab environment or testing environment	Beta testing doesn't require lab environment or testing environment because software is available for end users and is said to be real time environment
Long execution cycle may require for alpha	Only few weeks execution are required for

testing	beta testing
Critical issues or fix are addressed by developers immediately in alpha testing	Most of the issues or feedback are collected from beta testing will be implemented in feature version of the product
Alpha testing ensures the quality of the product before moving to the beta testing	Beta testing also concentrate on the quality of the product but gathers user impact on the product and ensures that the product is ready for real time environment.
Virtual environment	Real time environment
Off shore	On site
Alpha testing is done under controlled environment	Beta testing is done under uncontrolled environment

Approach no 2: employees or test engineer of customer will do acceptance testing at customer place.

Approach no 3: test engineers of software company will do acceptance testing at customer place.

Approach no 4: test engineer or BA of software company will do acceptance testing under the control of customer at developer place.

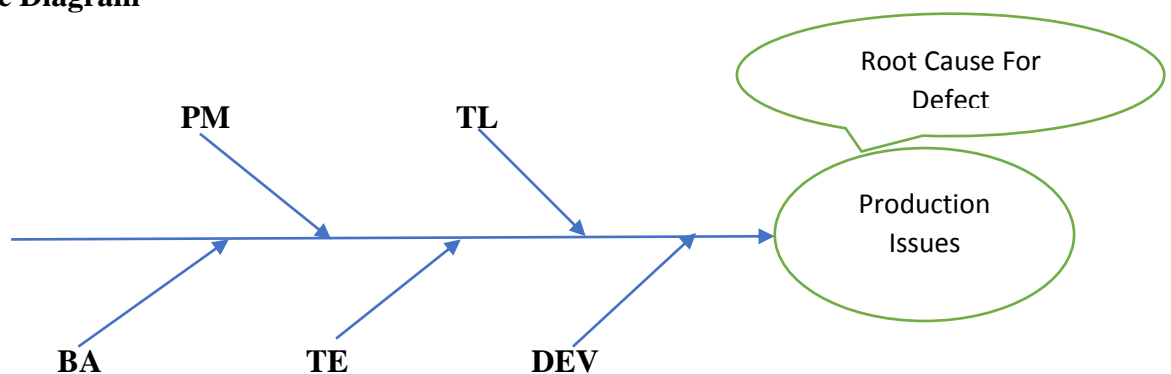
Hot fix / incident management system

When the software company releases the product to customer if the customer finds any defects in the software. The customer will send the application back to the software company, and software company should fix the defect as early as possible and give back to the customer. This situation is called hotfix.

If any problem is find by the customer or end user in the production server, it will be raised as the incident to the software company. Every incident is raised in the form of ticket and every ticket will have a priority.

The commitment between software company and customer is called service level agreement.

Fish Bone Diagram



Whenever there is a production issue all the team members will gather together, and they discuss about reason for production issues that is documented in the form of fish bone diagram. The main purpose of this is to find the root cause of the production house.

SHORT TERM RELEASE/INTERIM RELEASE

Between two planned release of the application some times the customer gives a unplanned release which is done in short duration. It is called short term release.

REAL TIME PROJECT (OR) HOW ARE THE SERVERS USED

Usually we have 3 servers like development server, testing/ QA server, production server. Customer gives the requirement then the developer start writing the source code for developing the application in the local system, this source code will be saved in the repository(github,vlt.cvt), there white box testing will be done by developer. The source code will be compiled and compressed, then we will be getting a file called build. All this will be happened in developer server.

The build has to be installed from development server to testing server. It will be done by the installation team or testing team.

Depending upon project to project, once build is installed to the testing server testing team will perform different type of testing depending upon the content of the build ex. For new features, FT, IT, will be done for old features regression testing. for defect fixes retesting will be done.

While doing the testing we find the defects that will be raised to the developers. Development team will reproduce the defect and they will fix it in the local system and it is saved in the repository again. Same process will be repeated for every build, until we get the final stable build.

Once the testing is completed, we will give the application to the customer for acceptance testing. Customer will do acceptance testing in the testing server or customer will have their own server called user acceptance testing server (UAT Server).

Once everything is fine application is released to end users that is for production server. This is called production release.

What is release?

Starting from gathering the requirement, developing the application, testing the application, finally releasing it to the end users is called release. That is called production release.

What is a build?

The compiled and compressed format of source code is called build. A build contains different formats like compress and archive.

Compress Format:

1. Zip
2. Multiple files can be made in single file.
3. Size of the file will be reduced.

Archive Format:

1. JAR (Java Archive), WAR (Web Archive), TAR (Tape Archive).
2. Multiple files are made in a single file.
3. Size of the file will be almost same.

What does build contain?

A build contains

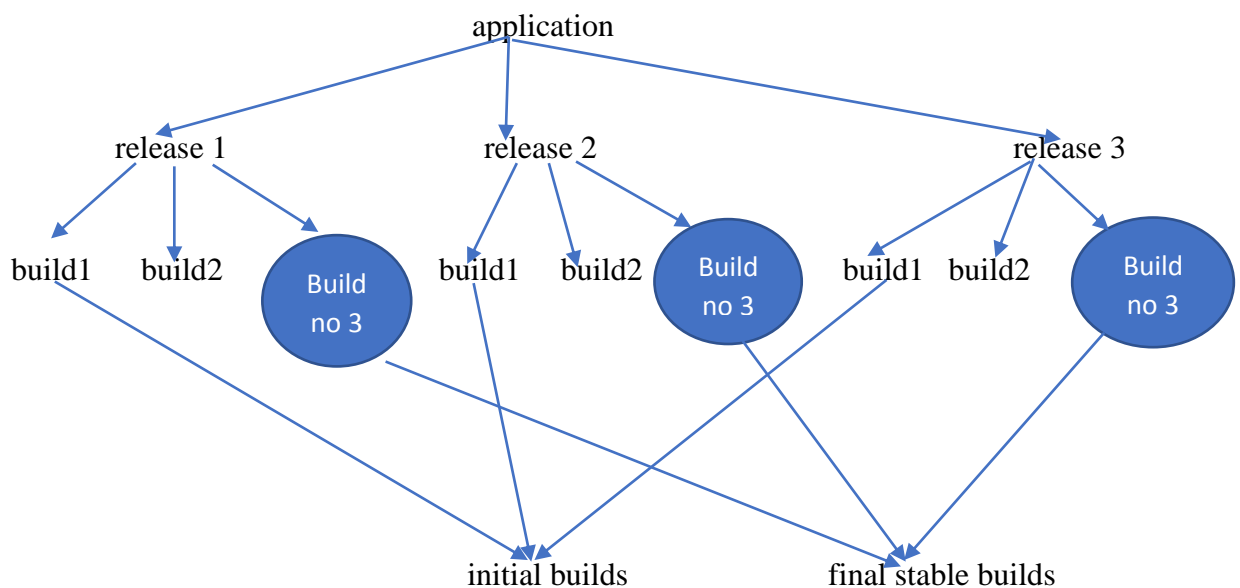
1. New features
2. Old features
3. Defect fixes

It depends upon the contents which developer will add inside an build, and it varies from build to build.

What is test cycle?

It is effort or time spent to test the application once the build is given. The duration of each test cycle can be days, weeks, months, depending upon the build.

Retesting And Regression Testing With Respect To Build



- Application will be having multiple release.
- Release can have multiple builds.
- The final stable build will be released to production server.
- The first release of first build1 we cannot do retesting and regression testing, for other builds we can do retesting and retesting based on the situations.