**Software Testing:**

One of the phase of SDLC(Software development life cycle).

**SDLC:** Planning, Requirements, Design, Develop, Test, Deploy and maintain.

What do they need as an input?

We need requirement doc,

**Waterfall model:** After one phase the other phase starts, he can refer to user stories till he/she gets his work.

**Agile model:** Every duration is in the form of sprints.(2 Weeks a sprint).

Scrum master, scrum meeting/Standup meeting

The number of requirements that can fit in a 2 weeks span.

If the developments are frozen by that team then tester can start testing from acceptance testing.

Unit testing:

Integrated testing:

Acceptance testing:

Quality center: It is a tool where tester raise tickets.

Tickets consists of Who tested? Used case? Errors? ID?

**Requirement tracebility matrix:(RTM)**

It is a matrix format(excel sheet) which will have the mapping between the requirements and testcases.

Multiple testcases together is known as an usecase.

Requirement or user Id is similar

RTM is to track the status of testcases and requirements. It is prepared by testlead / engineer

**Various Testing concepts:**

**Test strategy:** Test approach – Defines how to testing should be carried out:

**Test plan**: A detailed doc which describes the test strategy.

Ex: schedule, Estimatestime), Deliverables(screenshots and scripts needs to be attached if required)

**Test scenario:** It is a group of testcases.

Given a requirement/usecase there can be many possibilities -> Test scenario

When the test scenario is broke up -> Test case

**Test case:** we need to follow a particular flow while executing a situation. This is a xls file

**Test script:** Same as test case but created programmatically.

**SDLC Life cycle:** Planning, Requirements, Design, Develop, Test, Deploy and Maintain.

**STLC Life cycle:** Requirements are frozen and approved, Test planning, RTM, Test strategy, Test plan, Test scenarios, Test case/Scripts, Test execution, Go/No Go(Defect fixing sessions by developers).

**Bug & Defect:**

**Bug:** It is a mistake in the program. It is resolved before acceptance testing. In unit/dev or integrations testing phases the issues can be resolved.

**Defect:** This is a deviation in the requirement. The program is not behaving as per the requirement

Links to testcase documents will be provided in RTM for ease of accessing required data.

Quality analyst: A person in quality center.

Scope: should be well defined.

RTM handler

Automation scope: What can be automated needs to defined.

**Testing types:**

Unit

Dev/Integration/Component/Modules

User acceptance

Pre - production

Production testing

Smoke testing

Sanity testing

Performance testing

Regression testing

Black box

White box

**STLC criteria:**

**Acceptance criteria:** It is the excepted results of the system as per the requirements.

**Epic:** Part/Piece of user story.

**Entry Criteria:** What is needed to start??? Refering requirement doc, RTM, HLD, Test plan and test strategy

**Exit Criteria:** Testcase doc is completed, Pr, supervisor checked, Code checked in, Impact analysis attached, Screenshots or scripts attached.

**Impact Analysis:** Developer testing things.

**Issues:** A risk which will occur is an issue.

**Risks:** Suspecting some problem may occur.

Mention issues and risks in the test plan.

R1 TC1 link to the testcase document.

**Test management:** Using a tool u need to manage the testcases and defects.

**Risk analysis:** Types of rinks can occur

Ex: Resourses, delay and many more

**Defects life cycle:** New defect, assign, in progress, Re-test, close , re open

**QA/QE:** Quality assurance team

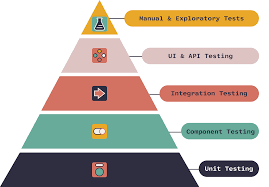
**QE: Quality Engineer(Makes sure to maintain quality)**

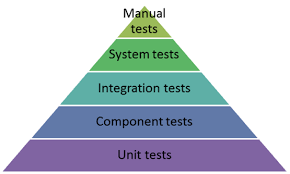
He puts rules to what tools to be used and policing on the quality regulations so that the quality is assured.

**QA: Quality Analyst(Check criteria is met):** He is the one present in quality center.

He/She makes sure that all the work is equally distributed among testers**.**

**TEST PYRAMID:** It is a structure that should be followed to have a quality product which is made faster and more efficient.

****

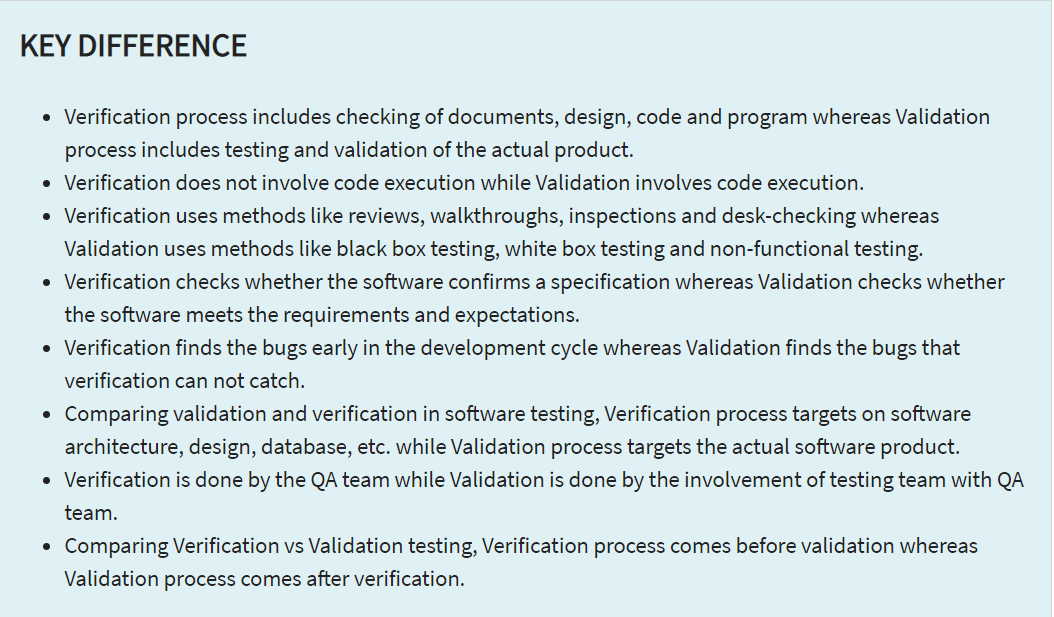


**V & V:**

**Verification and validation(Testing):**

**Verification in Software Testing** is a process of checking documents, design, code, and program in order to check if the software has been built according to the requirements or not. The main goal of verification process is to ensure quality of software application, design, architecture etc. The verification process involves activities like reviews, walk-throughs and inspection.

**Validation in Software Engineering** is a dynamic mechanism of testing and validating if the software product actually meets the exact needs of the customer or not. The process helps to ensure that the software fulfills the desired use in an appropriate environment. The validation process involves activities like unit testing, integration testing, system testing and user acceptance testing.



**User stories:**

**FRD: Functional requirement doc**

**BRD: Business requirement doc**

**HLD: High level Design doc**

**LLD: Low level design doc**

**Smoke testing:** If at all, all the basic functionalities are working fine.

**Sanity testing:** whenever a minor change is made, we check the entire application if its still working fine.

**Functional testing:** To see if all the functionalities working fine or not.

**Performance testing:** How fast the page is rendered. Not more then 3 mins (as specified by client).

**Regression testing:** Any new code has not impacted the existing functionality.

**BlackBox testing:** You don’t care on the internal code, you test the functionalities of the appln.

**WhiteBox testing/open box testing/Code based testing/Glass box testing:** The tester has the visibility on the flow of the data, structure of the code etc is available to the tester.

**Requirement testing, types of it and Static testing**

**Requirement testing** is based on the requirement provided by the client. All my test cases, test scenarios , test data are inclined from Requirements.

Functional(based on the req.) and non functional (performance, system hardware etc) testing:

**Types of Requirement testing:**

Implicit requirements:

Explicit requirement:

Latent requirements:

**Static testing: (done without executing the appln.)**

These dummy features:

1. Review:
2. Static analysis:

**Alpha testing, beta testing**

**Testers**

Alpha testing is performed by **testers who are usually internal employees of the organization**. Beta testing is performed by clients who are not part of the organization. Alpha testing is performed at developer's site. Beta testing is performed at end-user of the product.

**Users**

Who performs beta testing?

Beta Testing is performed by **real users of the software application** in a real environment. Beta testing is one of the type of User Acceptance Testing.

Alpha testing uses both black and white box testing while Beta testing uses only blackbox testing. Alpha testing is done by testers and quality analysts inside the organization whereas Beta testing is done by **real users who will be actually using the software.**