**SOFTWARE TESTING LIFE CYCLE**

The Software Testing Life Cycle (STLC) is a structured approach to testing software systems. It encompasses a series of well-defined phases that ensure thorough and efficient testing throughout the software development process. By following the STLC, software teams can identify and address issues early on, ultimately delivering high-quality, reliable software to end-users.

Software testing life cycle has the following phases:

• Requirement Analysis

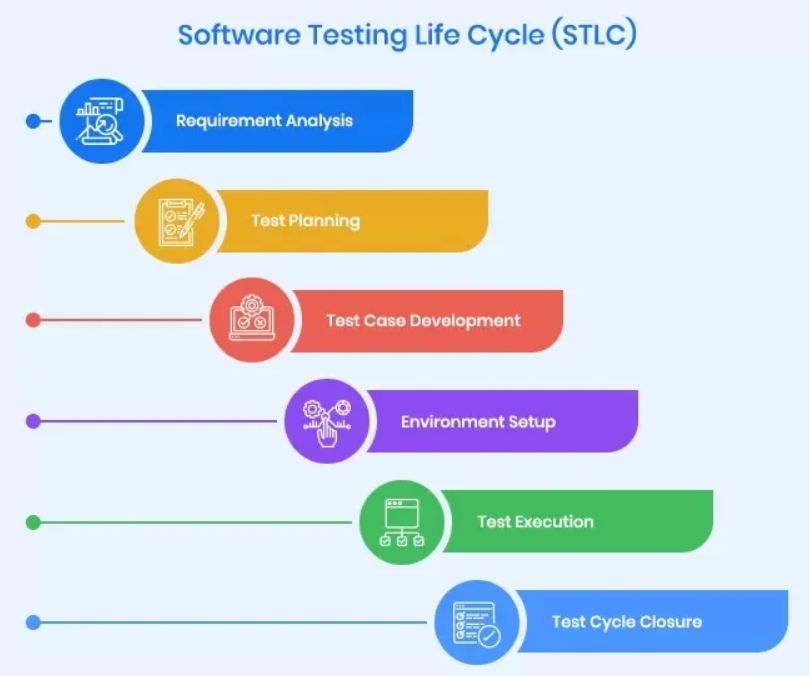
• Test Planning

• Test Case Development

• Test Environment Setup

• Test Execution

• Test Closure.



**Requirement Analysis:**

 **Objective:** Understand and analyze testing requirements.

 **Activities:**

* Review and analyze requirements documents.
* Identify testable requirements.
* Discuss ambiguities with stakeholders.

 **Deliverables:** Requirement Traceability Matrix (RTM), Feasibility Report.

**Test Planning:**

**Objective:** Define the scope, objectives, and logistics of the testing process.

 **Activities:**

* Develop the test strategy.
* Identify test objectives.
* Plan resource allocation.
* Define test environment requirements.
* Schedule testing activities.

 **Deliverables:** Test Plan Document, Effort Estimation Report, Test Schedule.

### **Test Case Development**

* + **Objective:** Create detailed test cases and prepare test data.
  + **Activities:**
    - Design test cases based on requirements.
    - Prepare test data.
    - Review and baseline test cases and test data.
  + **Deliverables:** Test Cases, Test Scripts, Test Data

### **Test Environment Setup**

* + **Objective:** Prepare the necessary test environment.
  + **Activities:**
    - Set up hardware and software configurations.
    - Install necessary applications.
    - Configure test environment as per requirements.
    - Validate the environment setup.
  + **Deliverables:** Test Environment ready for execution, Environment Setup Checklist.

### **Test Execution**

* + **Objective:** Execute test cases and log defects.
  + **Activities:**
    - Execute test cases.
    - Document test results.
    - Log defects for any failed test cases.
    - Retest fixed defects.
    - Track and manage defects to closure.
  + **Deliverables:** Test Execution Reports, Defect Logs, Retest Reports.

### **Test Cycle Closure**

* + **Objective:** Conclude testing activities and evaluate the results.
  + **Activities:**
    - Evaluate exit criteria based on test completion.
    - Document learnings and best practices.
    - Prepare test summary report.
    - Conduct retrospective meetings to discuss improvements.
  + **Deliverables:** Test Summary Report, Metrics and Analysis, Closure Reports.

**RTM: (Requirement Traceability Matrix)**

It is used to track the requirements and to check the current project requirements are met . This document will map with Test scenarios, test cases, test plan documents

A table with a number of items

Description automatically generated with medium confidence

With RTM captures that all requirements proposed by the client .

Forward Traceability : if Requirements map with the testcases.

A diagram of a procedure

Description automatically generated

Backward traceability: If Testcases mapped with the Requirements.

A diagram of a procedure

Description automatically generated

**Bi-directional traceability:** It has a reference from requirement to testcases and vice versa.

Here all the test cases can be tracked with requirements and each and every requirement specified has accurate and valid Test cases for them.

A close-up of a sign

Description automatically generated

**Test Scenarios:**

A test scenario is “what to be tested”. It is make sure of end to end functioning of application/software

is working as per expected and meet business process flow. In scenario testing, testers puts themselves

in the end user shoes and determines the real time scenarios, with the help of stakeholders, clients and

developers and etc.. to create test scenarios.

Test scenarios can have multiple test cases, while testing the software, first we need to analyze the test

scenario’s and based on it, prepare test cases. Test scenarios are mainly focus on the functionality of the

application rather than input data.

**Test Cases:**

A test case is “How to be tested”, and it refers to detailed conditions to be executed to ensure that the

software is functioning properly or not. Test cases are derived from test scenario and it includes test

input, procedure, expected result, actual result and status of the functionality. Test cases can be either

positive or negative test cases.

Ex:

Test Scenario: Validate the login page

Test Case 1: Enter a valid username and password

Test Case 2: Reset your password

Test Case 3: Enter invalid credentials

**Difference between Test Case and Test Scenario:**

|  |  |
| --- | --- |
| **Test Case** | **Test Scenario** |
| Testcase: How to test | Test Scenario: What to test |
| Test cases consist of test case name, ID,  Procedure, Expected result and etc. | Test scenario consists of detailed procedure, thus  we can conclude that a test scenario has many test  cases. |
| Test Cases are low level activity and thus derived  from test scenarios | Test scenario are high level activity. |
| A test case means detailed documentation, which  is used to executing. | Test scenario deals with analyses requirement in  detail |
| A test case is used for freshers to understand the  requirements and functionality of the application  in detail. | A good test scenario is used to reduce the  complexity and repeatability of the product |
| Preparing test cases takes more time because in  test cases it include how to test, and what to test | Test scenario are not detailed enough, it might  take some time to discuss and understand. |