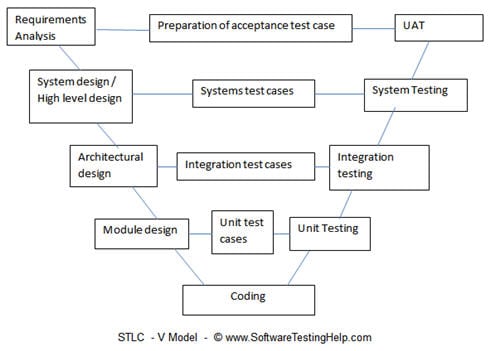
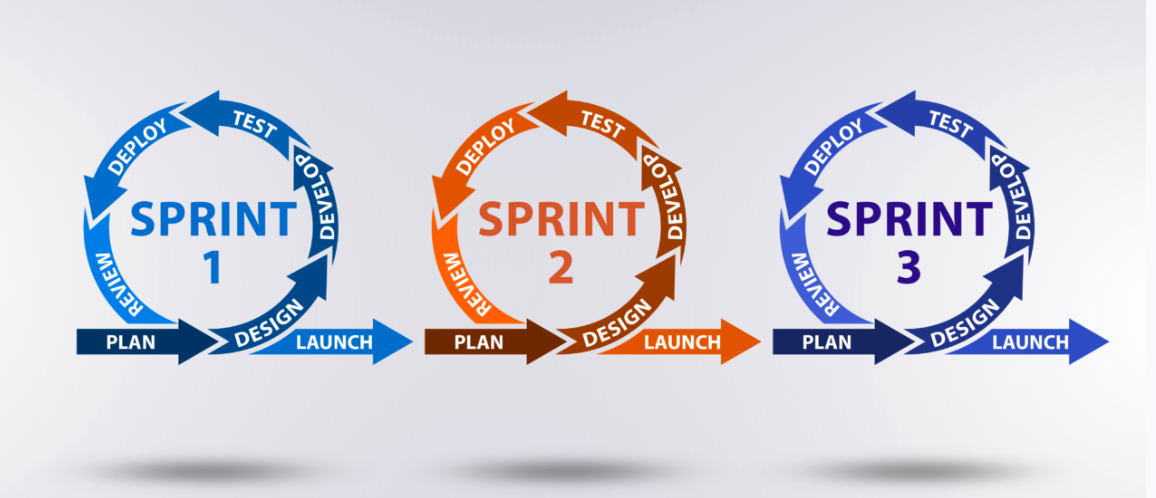
SDLC

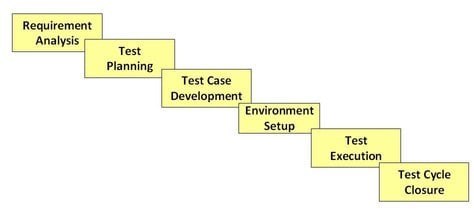




Software Testing

Verification and Validation

STLC



| S.No | Phase Name | Entry Criteria | Activities Performed | Deliverables |
| --- | --- | --- | --- | --- |
| 1 | Requirements | Requirements specification document  Application design document  User acceptance criteria document | Do brainstorming of the requirements. Create a list of requirements and get your doubts clarified.  Understand the feasibility of the requirements whether it is testable or not.  If your project requires automation, do the automation feasibility study. | RUD ( Requirements understanding document.  Testing feasibility report  Automation feasibility report. |
| 2 | Planning | Updated requirements document.  Test feasibility reports “  Automation feasibility report. | Define the scope of the project  Do the risk analysis and prepare the risk mitigation plan.  Perform test estimation.  Determine the overall testing strategy and process.  Identify the tools and resources and check for any training needs.  Identify the environment. | Test Plan document.  Risk mitigation document.  Test estimation document. |
| 3 | Analysis | Updated requirements document  Test Plan document  Risk Document  Test estimation document | Identify the detailed test conditions | Test conditions document. |
| 4 | Design | Updated requirements document  Test conditions document | Detail out the test condition.  Identify the test data  Create the traceability metrics | Detailed test condition document  Requirement traceability metrics  Test coverage metrics |
| 5 | Implementation | Detailed test condition document | Create and review the test cases.  Create and review the automation scripts.  Identify the candidate test cases for regression and automation.  Identify / create the test data  Take sign off of the test cases and scripts. | Test cases  Test scripts  Test data |
| 6 | Execution | Test cases  Test scripts | Execute the test cases  Log bugs / defects in case of discrepancy  Report the status | Test execution report  Defect report  Test log and Defect log  Updated requirement traceability metrics |
| 7 | Conclusion | Updated test cases with results  Test closure conditions | Provide the accurate figures and result of testing  Identify the risks which are mitigated | Updated traceability metrics  Test summary report  Updated risk management report |
| 8 | Closure | Test closure condition  Test summary report | Do the retrospective meting and understand the lessons learnt | Lessons learnt document  Test matrices  Test closure report. |

Test Strategy-

**Test Strategy Template**

**A test strategy is a documented approach that defines the testing methods, domain, environment, configurations, tools, schedules, resource allocations, and staff utilization. It plays a critical role for organizations to ensure the testing process is as effective as possible. A typical test strategy template features the following factors:**

1. **Scope**
2. **Test Approach**
3. **Test Environment**
4. **Testing Tools**
5. **Release Control**
6. **Risk Analysis**
7. **Review and Approvals**
8. **Test Summary**

Test Plan

**Defined by the standard**[**IEEE 829**](https://en.wikipedia.org/wiki/Software_test_documentation)**, a test plan template consists of the following 19 details:**

1. **Test plan identifier**
2. **References**
3. **Introduction**
4. **Test items**
5. **Software risk issues**
6. **Features to be tested**
7. **Features not to be tested**
8. **Approach**
9. **Pass/Fail criteria**
10. **Suspension criteria and resumption requirements**
11. **Test deliverables**
12. **Remaining tests**
13. **Environmental needs**
14. **Staffing and training needs**
15. **Responsibilities**
16. **Schedule**
17. **Planning risks And contingencies**
18. **Approvals**
19. **Glossary**

**Differences Between Test Plan and Test Strategy**

**The difference between test plan and test strategy is that a test plan documents scope, objective, and key elements of software testing, whereas a test strategy defines the techniques and approaches to testing.**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Test Plan** | **Test Strategy** |
| **1** | Test plan is a document that defines scope, objective, approach, and emphasis of a software testing initiative. | Test strategy is a set of guidelines that describe the test design and how to perform testing. |
| **2** | Key elements include- Test plan id, testing features, types & jobs, pass or fail criteria, test deliverables, team responsibilities, release schedule, etc. | Key elements include – scope, formats, processes, tools, reports, client communication, etc. |
| **3** | It describes how to test, when to test, and who will test. | It defines what type of technique to follow and which module to test. |
| **4** | Test plan declares the specification. | Test strategy declares the general approaches to testing. |
| **5** | The test plan may be updated if required. | Test strategies cannot be changed |
| **6** | It determines possible issues and dependencies to identify the risks. | It is a long-term plan of action. You can abstract information that is not project-specific and put it into a test approach |
| **7** | A test plan exists individually. | Test strategy is a section of a test plan. |
| **8** | It is defined at a project level | It is set at the organization level. |
| **9** | Test plan is derived from software requirement specification (SRS). | Test strategy is derived from business requirement specification (BRS). |
| **10** | Test leads or managers prepare test plans. | Project managers or business analysts prepare test strategies. |
| **11** | It is created after requirement sign off. | It is created before the test plan. |

**Wrapping Up**

**When it comes to segregating test strategy vs test plan, many find it quite complicated to identify the key differences between the two. A test strategy is generally a static document and the test plan, on the other hand, specifies what to test, when to test, and how to test.**

Test case design techniques

BVA -

Example: if input box accepts 1 to 1000, then test for 0,1, 2, 999, 1000, 1001

ECP

Example: if input box accepts 1 to 1000, then test for -5, 100, 1010 etc

**CEG-** Cause Effect Graphing

***Situation*:**

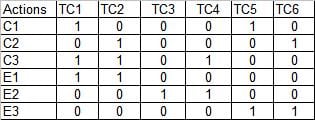
The “Print message” is software that reads two characters and, depending on their values, messages is printed.

* The first character must be an “A” or a “B”.
* The second character must be a digit.
* If the first character is an “A” or “B” and the second character is a digit, then the file must be updated.
* If the first character is incorrect (not an “A” or “B”), the message X must be printed.
* If the second character is incorrect (not a digit), the message Y must be printed.

***Solution***:

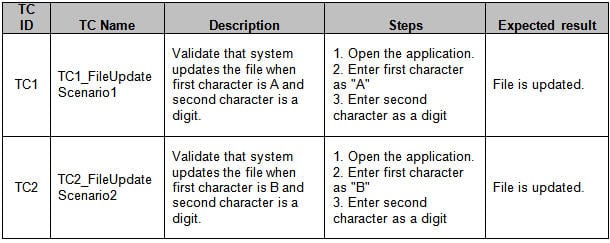
**The Causes of this situation are:**  
C1 – First character is A  
C2 – First character is B  
C3 – the Second character is a digit

**The Effects (results) for this situation are:**  
E1 – Update the file  
E2 – Print message “X”  
E3 – Print message “Y”

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2013/12/Cause-and-effect-graph-testing-12.jpg)

### Writing Test Cases From The Decision Table

**Below is a sample test case for Test Case 1 (TC1) and Test Case 2 (TC2).**

[](https://www.softwaretestinghelp.com/wp-content/qa/uploads/2013/12/Cause-and-effect-graph-testing-13.jpg)

**Error Guessing**

**STT-** State Transition Testing-When testing is required for sequential input combinations

Example- cash withdrawal when there is enough balance(state 1). Cash withdrawal when there is no balance(state 2)

Q cjjjhhggggggfgfffffffgaswasswsujyyjuiyu78ytutrtygrewrtyuiopp;o;ọ,.,m,.,mnbnm,nbvcvbnfgvdcdfcdfcaRegistered user, Guest user buying products.

Home loan application- applied, confirmed, conditional approval, approved.

**Use Case based Test cases**

Use case description, Actors, pre condition , Basic flow, alternate flow, exceptional flow, post condition.

Finally, keep the “**BAOE mantra**”, which means**i) Basic Flow ii)**[**Alternate Flow**](https://study.com/academy/lesson/exception-alternate-flow-in-use-case.html)**iii) Options and iv) Exceptions** for the complete coverage of the functional flow and feature to be tested. Every concept should apply to positive and negative tests.

**For Example,** let us see the simple BAOE approach for the sample login screen above.

* **Basic Flow:** Enter the URL path of the Login in any browser and enter the information required and login to the application.
* **Alternate Flow:** Install the application on a mobile device and enter the information required and log in to the application.
* **Options:** What are the options that are available to come to the same login screen? **For Example,** after logging in to the application, clicking the ‘Logout’ may bring the same screen or if the session timeout or session expired, the user may come to the login screen.
* **Exceptions:** What are the exceptions if my tests are negative? **For Example,** if wrong credentials are entered in the Login screen, whether the user will get an error message or no action associated.

Manual Test case document template

Graphical user interface, application, table, Excel

Description automatically generated

**For Example**, if the test scenario is “Validate the Admin login functionality” – This would yield in 3 test cases (or conditions) – Login (successful), Login-unsuccessful when the incorrect username is entered, Login-unsuccessful when the incorrect password is entered.

https://www.softwaretestinghelp.com/sample-test-cases-testing-web-desktop-applications/

Defect Life Cycle

Severity, Priority

Functonal Testing stages- Unit, component, integration, system, uat, sanity , smoke

Non functional Testing- Performance, security, accessibility, usability, compatibility etc

Testing types

White box, blackbox, alpha, beta, Adhoc, exploratory, Incremental testing, Negative testing

<https://testsigma.com/blog/test-strategy-vs-test-plan/>

<https://www.softwaretestinghelp.com/manual-testing-tutorial-1/>

<https://www.softwaretestinghelp.com/sample-test-cases-testing-web-desktop-applications/>