

How to Write Test Cases: Sample Template with Examples

What is a Test Case?

A Test Case is defined as a set of actions executed to verify a particular feature or functionality of the software application. A test case is an indispensable component of the Software Testing LifeCycle that helps validate the AUT (Application Under Test).

Test Scenario Vs Test Case

Test scenarios are rather vague and cover a wide range of possibilities. Testing is all about being very specific.

For a [Test Scenario](#): Check Login Functionality there many possible test cases are:

- Test Case 1: Check results on entering valid User Id & Password
- Test Case 2: Check results on entering Invalid User ID & Password
- Test Case 3: Check response when a User ID is Empty & Login Button is pressed, and many more

This is nothing but a Test Case.

Click [here](#) if the video is not accessible

How to Create a Test Case

Let's create a Test Case for the scenario: Check Login Functionality

Step 1) A simple test case for the scenario would be

Test Case #	Test Case Description
1	Check response when valid email and password is entered

Step 2) In order to execute the test case, you would need Test Data. Adding it below

Test Case #	Test Case Description	Test Data
1	Check response when valid email and password is entered	Email: guru99@email.com Password: INf9^Oti7^2h

Identifying test data can be time-consuming and may sometimes require creating test data afresh. The reason it needs to be documented.

Step 3) In order to execute a test case, a tester needs to perform a specific set of actions on the AUT. This is documented as below:

Test Case #	Test Case Description	Test Steps	Test Data

1	Check response when valid email and password is entered	1) Enter Email Address 2) Enter Password 3) Click Sign in	Email: guru99@email.com Password: lNf9^Oti7^2h
---	---	---	---

Many times the Test Steps are not simple as above, hence they need documentation. Also, the author of the test case may leave the organization or go on a vacation or is sick and off duty or is very busy with other critical tasks. A recently hire may be asked to execute the test case. Documented steps will help him and also facilitate reviews by other stakeholders.

Step 4) The goal of test cases is to check behavior the AUT for an expected result. This needs to be documented as below

Test Case #	Test Case Description	Test Data	Expected Result
1	Check response when valid email and password is entered	Email: guru99@email.com Password: lNf9^Oti7^2h	Login should be successful

During test execution time, the tester will check expected results against actual results and assign a pass or fail status

Test Case #	Test Case Description	Test Data	Expected Result	Actual Result	Pass/Fail
1	Check response when valid email and password is entered	Email: guru99@email.com Password: lNf9^Oti7^2h	Login should be successful	Login was successful	Pass

Step 5) That apart your test case -may have a field like, Pre - Condition which specifies things that must in place before the test can run. For our test case, a pre-condition would be to have a browser installed to have access to the site under test.

A test case may also include Post - Conditions which specifies anything that applies after the test case completes. For our test case, a postcondition would be time & date of login is stored in the database

The format of Standard Test Cases

Below is a format of a standard login Test case

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
TU01	Check Customer Login with valid Data	1. Go to site http://demo.guru99.com 2. Enter UserId 3. Enter Password 4. Click Submit	Userid = guru99 Password = pass99	User should Login into an application	As Expected	Pass
TU02	Check Customer Login with invalid Data	1. Go to site http://demo.guru99.com 2. Enter UserId 3. Enter Password 4. Click Submit	Userid = guru99 Password = glass99	User should not Login into an application	As Expected	Pass

This entire table may be created in Word, Excel or any other Test management tool. That's all to Test Case Design

While drafting a test case to include the following information

- The description of what requirement is being tested
- The explanation of how the system will be tested
- The test setup like a version of an application under test, software, data files, operating system, hardware, security access, physical or logical date, time of day, prerequisites such as other tests and any other setup information pertinent to the requirements being tested
- Inputs and outputs or actions and expected results
- Any proofs or attachments
- Use active case language
- Test Case should not be more than 15 steps

- An automated test script is commented with inputs, purpose and expected results
- The setup offers an alternative to pre-requisite tests
- With other tests, it should be an incorrect business scenario order

Best Practice for writing good Test Case Example.

1. Test Cases need to be simple and transparent:

Create test cases that are as simple as possible. They must be clear and concise as the author of the test case may not execute them.

Use assertive language like go to the home page, enter data, click on this and so on. This makes the understanding the test steps easy and tests execution faster.

2. Create Test Case with End User in Mind

The ultimate goal of any software project is to create test cases that meet customer requirements and is easy to use and operate. A tester must create test cases keeping in mind the end user perspective

3. Avoid test case repetition.

Do not repeat test cases. If a test case is needed for executing some other test case, call the test case by its test case id in the pre-condition column

4. Do not Assume

Do not assume functionality and features of your software application while preparing test case. Stick to the Specification Documents.

5. Ensure 100% Coverage

Make sure you write test cases to check all software requirements mentioned in the specification document. Use [Traceability Matrix](#) to ensure no functions/conditions is left untested.

6. Test Cases must be identifiable.

Name the test case id such that they are identified easily while tracking defects or identifying a software requirement at a later stage.

7. Implement Testing Techniques

It's not possible to check every possible condition in your software application. Software Testing techniques help you select a few test cases with the maximum possibility of finding a defect.

- **Boundary Value Analysis (BVA):** As the name suggests it's the technique that defines the testing of boundaries for a specified range of values.
- **Equivalence Partition (EP):** This technique partitions the range into equal parts/groups that tend to have the same behavior.
- **State Transition Technique:** This method is used when software behavior changes from one state to another following particular action.
- **Error Guessing Technique:** This is guessing/anticipating the error that may arise while doing manual testing. This is not a formal method and takes advantages of a tester's experience with the application