DigitalOcean Droplets
Virtual machines with industry leading price-performance value.
Try it for free

Test Case

TEST CASE Fundamentals

DEFINITION

A test case is a set of conditions or variables under which a tester will determine whether a system under test satisfies requirements or works correctly.

The process of developing test cases can also help find problems in the requirements or design of an application.

TEST CASE TEMPLATE

A test case can have the following elements. Note, however, that normally a test management tool is used by companies and the format is determined by the tool used.

Test Suite ID	The ID of the test suite to which this test case belongs.
Test Case ID	The ID of the test case.
Test Case Summary	The summary / objective of the test case.
Related Requirement	The ID of the requirement this test case relates/traces to.
Prerequisites	Any prerequisites or preconditions that must be fulfilled prior to executing the test.
Test Procedure	Step-by-step procedure to execute the test.
Test Data	The test data, or links to the test data, that are to be used while conducting the test.
Expected Result	The expected result of the test.
Actual Result	The actual result of the test; to be filled after executing the test.
Status	Pass or Fail. Other statuses can be 'Not Executed' if testing is not performed and 'Blocked' if testing is blocked.
Remarks	Any comments on the test case or test execution.
Created By	The name of the author of the test case.
Date of Creation	The date of creation of the test case.
Executed By	The name of the person who executed the test.
Date of Execution	The date of execution of the test.
Test Environment	The environment (Hardware/Software/Network) in which the test was executed.

TEST CASE EXAMPLE / TEST CASE SAMPLE

Test Suite ID	TS001
Test Case ID	TC001
Test Case Summary	To verify that clicking the Generate Coin button generates coins.
Related Requirement	RS001
Prerequisites	1. User is authorized. 2. Coin balance is available.
Test Procedure	1. Select the coin denomination in the Denomination field. 2. Enter the number of coins in the Quantity field. 3. Click Generate Coin.
Test Data	1. Denominations: 0.05, 0.10, 0.25, 0.50, 1, 2, 5 2. Quantities: 0, 1, 5, 10, 20

Expected Result	1. Coin of the specified denomination should be produced if the specified Quantity is valid (1, 5) 2. A message 'Please enter a valid quantity between 1 and 10' should be displayed if the specified quantity is invalid.
Actual Result	1. If the specified quantity is valid, the result is as expected. 2. If the specified quantity is invalid, nothing happens; the expected message is not displayed
Status	Fail
Remarks	This is a sample test case.
Created By	John Doe
Date of Creation	01/14/2020
Executed By	Jane Roe
Date of Execution	02/16/2020
Test Environment	• OS: Windows Y • Browser: Chrome N

WRITING GOOD TEST CASES

- As far as possible, write test cases in such a way that you test only one thing at a time. Do not overlap or complicate test cases. Attempt to make your test cases 'atomic'.
- Ensure that all positive scenarios and negative scenarios are covered.
- Language:
    - Write in simple and easy to understand language.
    - Use active voice: Do this, do that.
    - Use exact and consistent names (of forms, fields, etc).
- Characteristics of a good test case:
    - *Accurate*: Exacts the purpose.
    - *Economical*: No unnecessary steps or words.
    - *Traceable*: Capable of being traced to requirements.
    - *Repeatable*: Can be used to perform the test over and over.
    - *Reusable*: Can be reused if necessary.

Share 1 G+ Share Tweet • Stur

□ Software Testing Myths (http://softwaretestingfundamentals.com/software-testing-myths/)

Agile Testing □ (http://softwaretestingfundamentals.com/agile-testing/)

Search...

Enter your email address:

□

Subscribe

If you do not find the verification email, check your spam folder

Delivered by FeedBurner (https://feedburner.google.com)

Categories

- □ Artifacts (http://softwaretestingfundamentals.com/category/artifacts/)
- □ Basics (http://softwaretestingfundamentals.com/category/basics/)
- □ Career (http://softwaretestingfundamentals.com/category/career/)
- □ Defects (http://softwaretestingfundamentals.com/category/defects/)
- □ Levels (http://softwaretestingfundamentals.com/category/levels/)
- □ Methods (http://softwaretestingfundamentals.com/category/methods/)
- □ Metrics (http://softwaretestingfundamentals.com/category/metrics/)
- □ Resources (http://softwaretestingfundamentals.com/category/resources/)
- □ Types (http://softwaretestingfundamentals.com/category/types/)

STF

Disclaimer (http://softwaretestingfundamentals.com/disclaimer/)

Privacy Policy (http://softwaretestingfundamentals.com/privacy-policy/)

Contact (http://softwaretestingfundamentals.com/contact/)

Social

Facebook Page (https://www.facebook.com/softwareTestingFundamentals/)

Facebook Group (https://www.facebook.com/groups/259049327448031/)

LinkedIn Group (https://www.linkedin.com/groups/4538644)

Twitter (https://twitter.com/stestingf)

□

Ponder

Weinberg's Second Law: If builders built buildings the way programmers wrote programs, then the first woodpecker that came along would have destroyed civilization.

Copyleft - 2018 STF | Theme by Colorlib (http://colorlib.com/wp/) Powered by WordPress (http://wordpress.org/)