# TEST CASE

Test case is a document that covers all possible scenarios for a specific requirement.

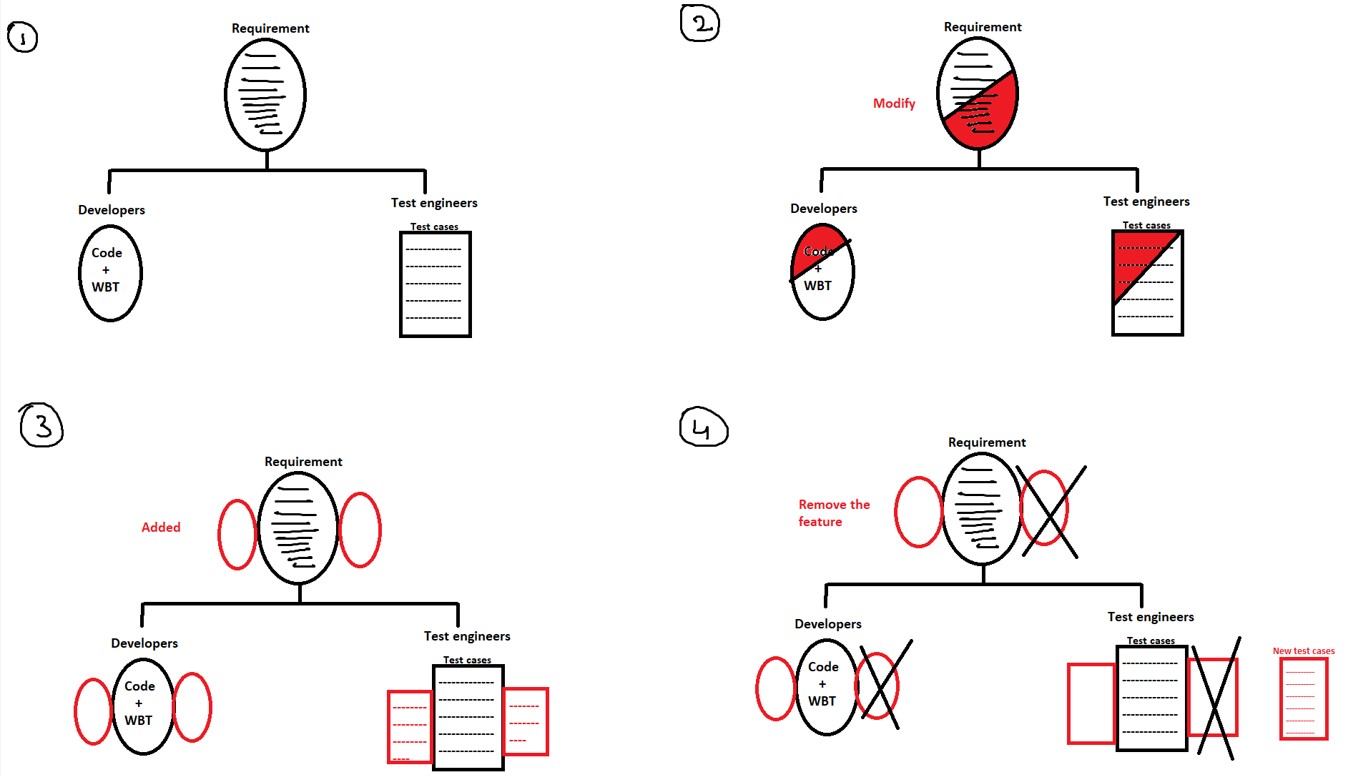
It contains different sections like Step number, Input, Expected result, Actual result, Status, and Comments.

**What is the drawback of seeing the requirements and testing the software?**

1. There will be no consistency in testing if you look into requirements and test the software.
2. Quality of testing varies from person to person
3. Quality of testing depends on the memory power of the test engineer.
4. Quality of testing depends on the mood of the T.E.

**When do we write test case?**

1. When developers are busy in building the product, the testing team will be busy writing the test cases.
2. When the developers will modify or change the feature, parallelly T.E. will modify or change the test cases.
3. When the developers will add the features parallelly test engineers will add new test cases.
4. When the customer is removing the requirement, developers will remove the feature, and parallelly test engineer will remove the test cases to make sure that features are removed from the s/w or not.



**Why do we write Test cases?**

1. We write test cases to have better test coverage.
   1. When the requirement comes in developers are busy building the product same time test engineers are free, so they identify all possible scenarios and document it.
   2. When the build comes we can spend time executing the scenarios, because of this no. of scenarios that you are covering will be more.
2. To have consistency in test execution.
   1. It means if you have documented the scenarios, you can make sure that you are executing all the scenarios in all the test cycles, sprints, or releases.
3. To depend on the process rather than person.
4. To avoid training every new engineer on the product or on the requirement.
5. Test case is the only document that acts like as proof for customer, development team, and also manager that we have covered all possible scenarios.
6. Test case acts like a base document for writing automation scripts, if you refer to the test case and write automation scripts you can ensure the same kind of coverage even in automation.
7. If you documented the test case, no need to remember the scenarios.
8. If you have documented the test cases, test execution happens in a very organized way.
9. If you have documented the test cases, the time taken to execute is much less.

(4-5 points is enough)

**TEST CASE DESIGN TECHNIQUES**

It is a technique which is used while writing test case in order to improve test coverage.

Types of Test case design techniques:

1. Error guessing
2. Equivalence class partitioning
3. Boundary value analysis (BVA)

1. ***Error Guessing:***

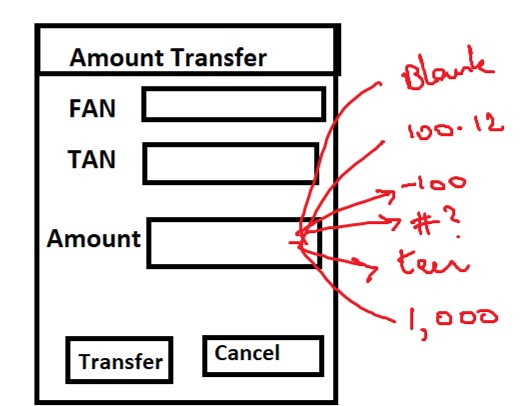
Here we guess all possible errors and we derive the scenarios.

We guess errors based on:

i.Requirement

ii.Experience

iii.Intuition



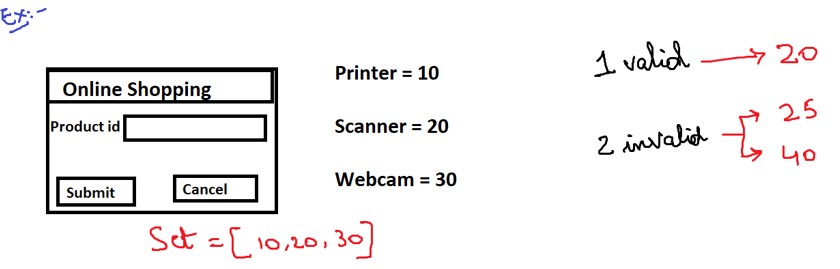
1. ***Equivalence class partition:***

* 1. **Pressman Rule:**

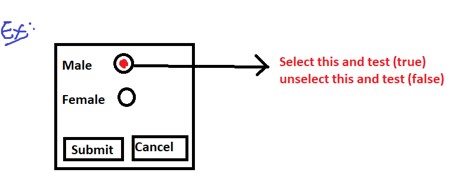
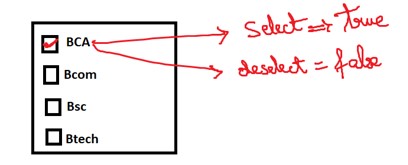
*Rule 1:* If the input is a range of values, then design test case for one valid and two invalid inputs.



*Rule 2:* If the input is in a set of values, then design test case for one valid and two invalid inputs.

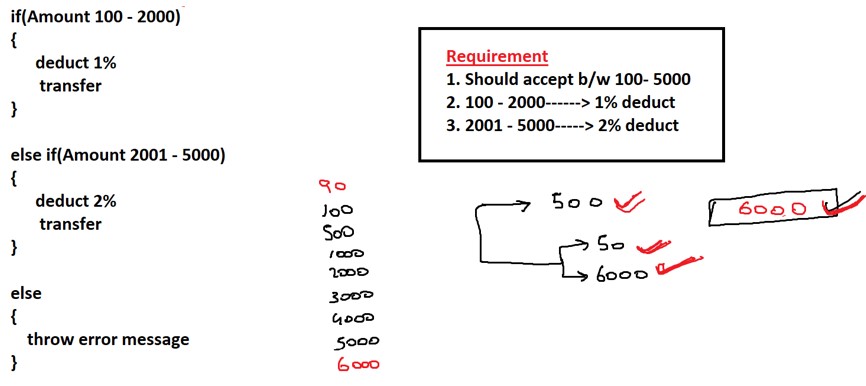


*Rule 3:* If the input is in Boolean, then design the test case for both true and false values.

* 1. **Practice Method:**

If the input is in range of values then divide the range into equivalent parts and test for all the values, make sure that you are testing for at least two invalid values.

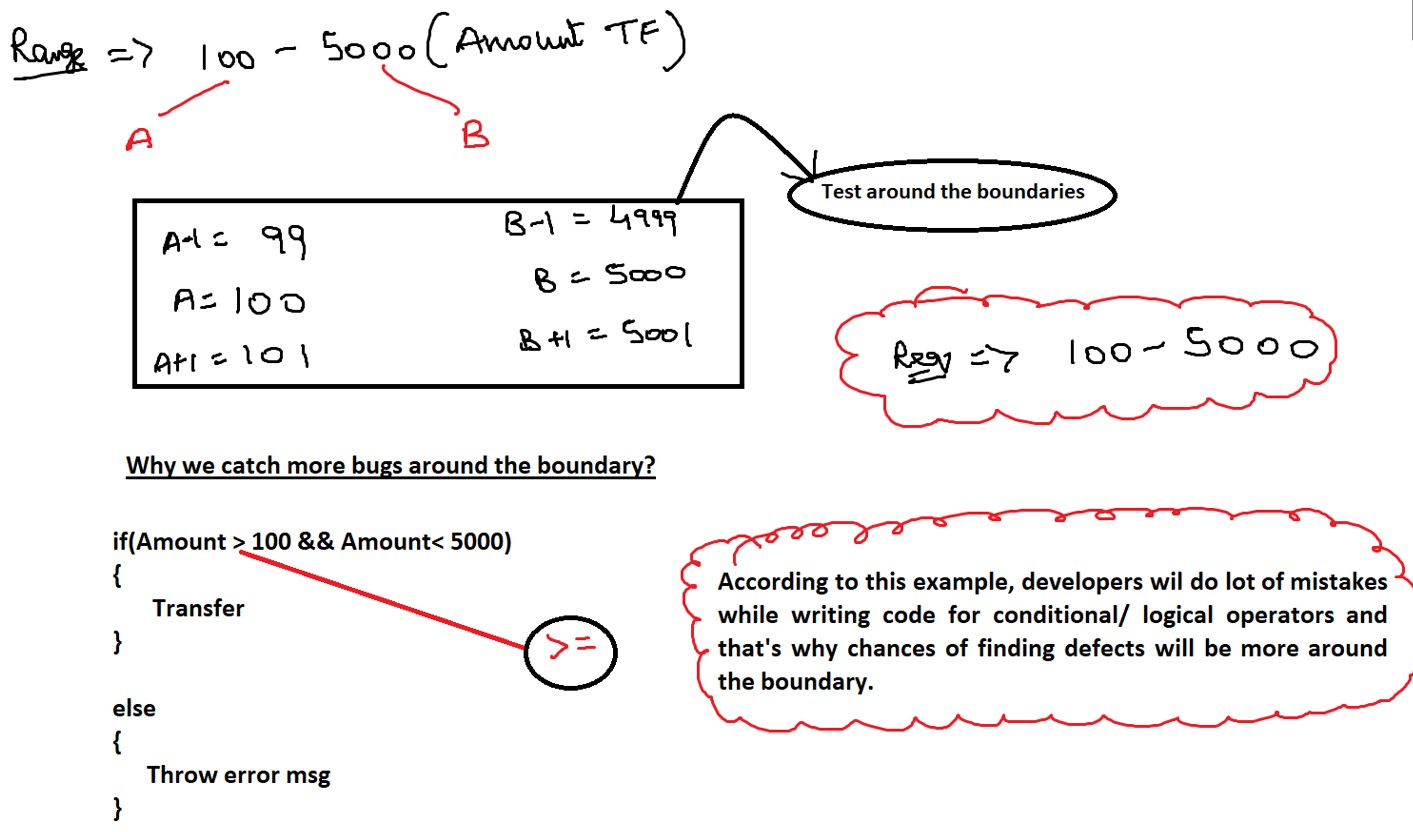


**Note:**

1. If there is a deviation between the range of values then we go got for Practice method.
2. If there is no deviation between the range of values then we go for Pressman rule.
3. By looking into requirements, we will get to know whether there is a deviation or not.

***3. Boundary Value Analysis:***

If the input is range of values b/w A to B then design test case for A, A+1, A-1 and B, B+1, B-1.



**Test case optimization:**

The process of removing the duplicates from the test cases (OR) Removal of repetition of test cases is called test case optimization. We can cover both 1 valid and 2 invalids in BVA itself, so we can skip equivalence partitioning (in only specific cases.

**Difference between test case & test scenarios:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | *Test scenarios* |  | *Test cases* |
|  | It is a high-level document of all the customer business workflow according to the customer's requirement. |  | It is a detailed document of the scenario that helps us to test the application. |
|  | We write Test scenarios by looking into the requirements. |  | We write test cases by looking into both requirement and test scenarios. |
|  | By looking into test scenarios, we can’t test any application until you have good product knowledge. |  | We can test any application by looking at the test case, no matter if you have product knowledge or not. |
|  | Here we mention what to test. |  | Here we mention how to test. |

## TEST CASE TEMPLATE

* Every TE will write test case in Test case Template only
* Test Case can be prepared in “Test Case Management Tool” or “MS Excel”  Test case template is divided into 3 sections:
  1. Header
  2. Body
  3. Footer

**How to fill the header?**

1. *Test case name:*

Format: Projectname\_ModuleName\_scenario

Ex: CB\_AmountTransfer\_integration

1. *Requirement Number:*

BA when he converts CRS to SRS, in SRS for each requirement he will write the requirement no.

Ex: 30.1 Amount Transfer

30.1.1 FAN text field

30.1.2 TAN text field

30.1.3 Amount text field

1. *Test data:*

It is the data written by a TE and has to be done before the test execution. Ex: TE should have UN, PWD, URL, a/c number

1. *Pre-condition:*

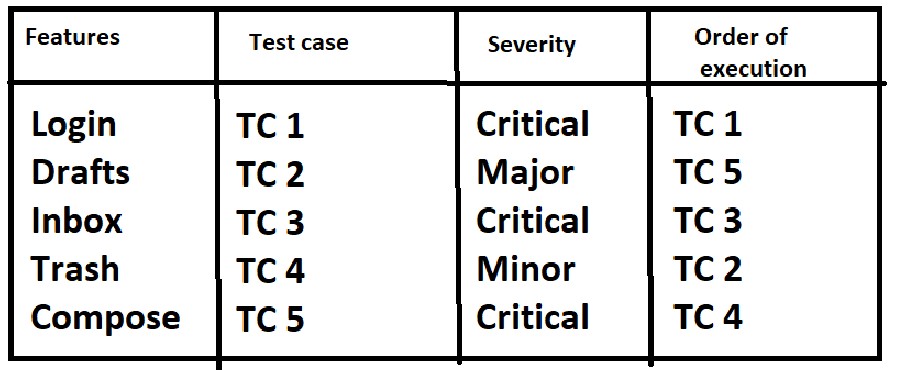
It is a set of actions or settings which should be ready/done by TE before executing the 1 st test case.

Ex: User should have balance in his account.

1. *Severity:*

TE will give severity for every individual test case, based on how important and complex the feature is from customer’s point of view. TE will execute test case based on severity.

There are 3 types of severity for Test cases: Critical/ Major/ Minor



1. *Test case type:*

Here the TE mention what type of test case he is writing.

Ex: Functionality test case, Integration test case, system test case.

1. *Brief Description:*

It describes about the complete test case and the behavior of the test case.

Ex: In the amount transfer module, it should accept only +ve integers.

**How to fill the footer?**

1. Author: Person who writes the test case Ex: Dinga
2. Reviewer: Person who reviews the test case Ex: Dingi
3. Approved by: Person who approves the test case Ex: Test lead
4. Approval date:

Ex: 01-01-2023

**Lessons to remember while writing test cases:**

1. Before we actually write test cases, we should come up with options and select the best option out of it.
2. Always use should be/must be in expected results. Don’t use maybe/ may not be, can be, cannot be.
3. Elaborate only those steps in which you have to focus. Don’t elaborate all the steps.
4. Start writing the test case with navigation steps.
5. Never write hardcore test cases, always write generic test cases.
6. While writing test cases we should imagine/visualize the application.
7. If you organize the scenarios properly, the total no. of steps can be reduced
8. Whatever we have covered in FT, don’t cover the same in Integration test cases.

If something is covered in integration test cases then don’t cover it in the system test cases.