

# How Do I Valudate Test cases for a project?

*Muhammad Mahbub Hussain, Executive Director, CIBL*

We write test cases for a project to ensure that the software that we build is bug-free, efficient and conforms its' respective functional requirements specified. However, how do we establish the conformance and efficiency of the test cases that we write for a specific project? Would we be able to raise a green signal to the fact that the test case package prepared for a specific project is sufficient enough to ensure a quality product?

I have proposed, and at the moment using a simple model for evaluation of test cases prepared for a project at any instance of the project lifecycle.

This model performs analysis on **coverage of test cases** related to 15 Key product attributes/test areas as follows:

1	Documented Business Requirements
2	Documented Data Requirements
3	Documented Functionality Requirements
4	Documented UI/Presentation Requirements
5	Cross-platform Compatibility
6	Internal Jumps and Navigations
7	Application Security
8	Data Integrity
9	Syntactic and Semantic Evaluation of Text
10	UI Integrity
11	Load/Stress
12	Installation
13	Validation
14	Integration
15	Installation Qualification

This model introduces the following set of metrics:

- (1) Test Case Coverage Percentage (TCCP)
- (2) Test Case Coverage Rank (TCCR)
- (3) Average Test Case Coverage Rank (ATCCR)

Performing simple mathematics, the model obtains value of Average Test Case Coverage Rank (ATCCR). Considering your organization/project specific coverage tolerance, you have to set a standard minimum value of for the ATCCR in a scale of 5 for a specific set of test cases.

In my book, if the value of ATCCR is less than my standard minimum for a specific set of test cases for a project, this specific set of test cases is not sufficient for the project of our interest. In such case(s), I review the set and try to figure out deficiencies or discrepancies.

The overall validation process can be presented using the following algorithm:

```
Void PerformTestCasePackageValidation()
{
    IF (ATCCR < Standard Minimum) THEN
        decision = test case package is sufficient ;
        PerformTesting() ;
    ELSE
        PerformReview() ;
        AddNewTestCases() ;
        PerformTestCasePackageValidation();
} // end of PerformTestCasePackageValidation()
```

For bigger projects, I do this analysis multiple times (to complement my practice of incremental test scripting) in pre-planned time intervals, and compare values obtained each time expecting a narrower gap with the standard minimum every time.

I have been using this model very effectively, and most importantly spending minimum amount of time! I would publish the detail of the model in short future.