



# **Easy Assessment Technique (EAT)**

**By**

**Manoj Kumar Varatharajan**

© Manoj Kumar Varatharajan, August 2007.

All rights reserved. You may make one attributed copy of this material for your own personal use.

For additional information or assistance please contact Manoj at

[manojkumar.varatharajan@gmail.com](mailto:manojkumar.varatharajan@gmail.com)

**CONTENTS**

Abstract	-----	2
Introduction	-----	2
Objectives	-----	3
Main Aspects	-----	3
Procedure	-----	3
Computational Factors	-----	7
Benefits of Implementation	-----	9
Conclusion	-----	9
General Explanation	-----	9

**Abstract:**

Easy Assessment Technique (EAT) provides an insight for rating an application during its Development as well as Testing phase. It is a simple technique that eases the Developers and Testers to measure the quality of the product at its various levels of Development and Testing, in a normal scale of 1 – 10. This spot out the areas in which the Product or the Application needs concentration, with a complete report. As its denouement, the Developers can rectify the faults and the Testers can focus with a better vision for their next phase of Testing.

**1.0 Introduction:**

Quality is the one of the significant factors that yields the Consumers to choose their product in this technical world, provided if the Product meets their requirements. By taking this into consideration, every Firm releases the Product after ensuring the quality of the product based on their Client's (or Consumer's) Requirements. They undergo various phases of Quality Control techniques adhering to some standard Quality Assurance Processes to achieve the intended Quality. Easy Assessment Technique (EAT) gives some tactics to assess the product under any circumstance in a defined scale, when the product is in the testing phase. This will help the Developers to take some control and recovery measures to ensure the quality of the product developed and the Testers to spot out the erroneous areas with lucid reasons.

## 2.0 Objectives:

The Objectives of this EAT is to

- Effectively calculate and report the stability of any application undergoing the Testing phase using some simple methods.
- Recognize and track the areas that would jeopardize or have any impact over the business requirements by pointing the status of the application.
- Trace the stability of every module of an Application sporadically.

## 3.0 Main Aspects:

The main aspects of this technique are listed below:

- Easy Assessment Technique (EAT) helps to find out the *factors influencing and bothering* the application stability at regular intervals.
- The Reports tend to point out the *fluctuations of Expected and Occurred* in various modules.
- The gap between the Risk and the Safety points can be obtained easily based on the computed Metrics value.
- The stability of the system can be determined graphically by plotting the *Safety Vs Risk*.

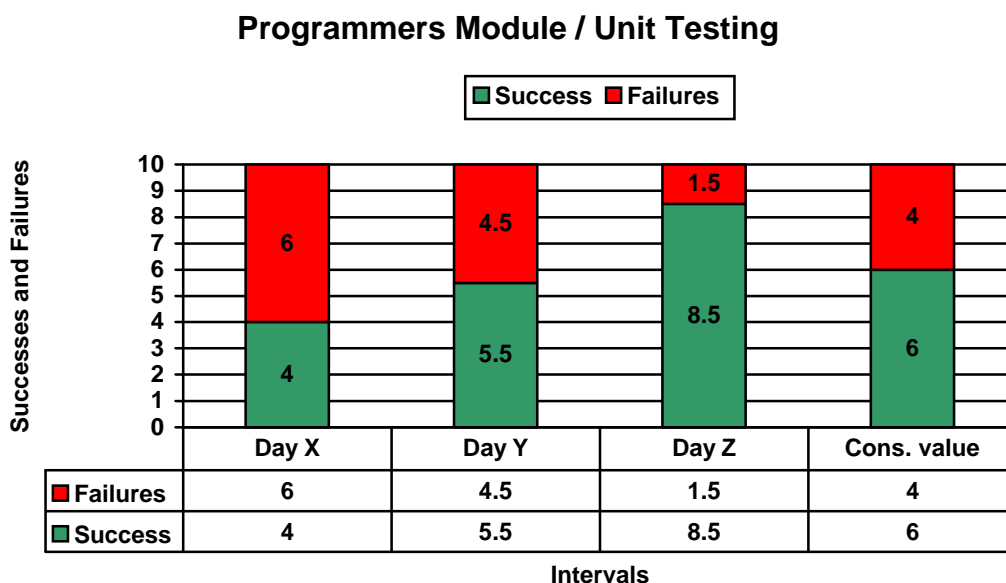
## 4.0 Procedure:

There will not be any literal changes in the Development phase of the Product or Application. But there will be substantial changes in the Testing phase that proportionately impact the Development Phase.

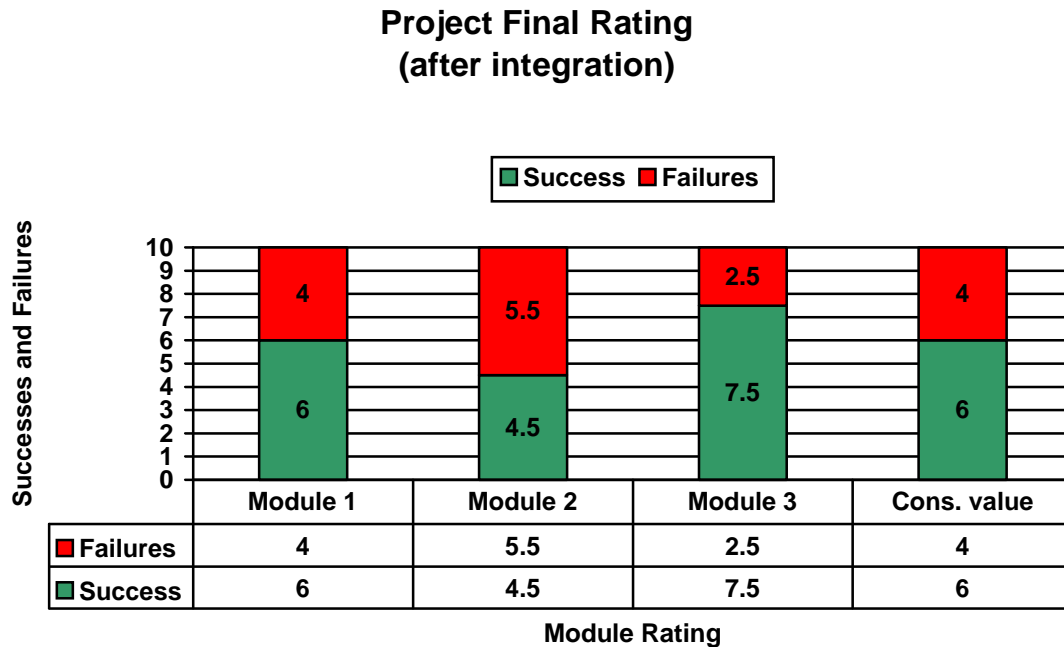
### 4.1 Testing by Programmers:

- The Developer's testing is being called as **Unit Testing / Module Testing / Component Testing**. The Development progresses by modularizing the business requirements based on its functionalities and its mandatory to perform the Unit Testing to mete out the functional requirements. The Unit testing for every unit should be performed for the most possible functional scenarios and the Success Vs Failures results should be recorded everyday (or sporadically).
- Then Successes and Failures need to be plotted graphically in a scale of 1 – 10 for that particular Date / period.
- The same set of procedures need to be repeated in the every module of the Project and the test results need to be plotted for the same, on its date / period.
- At the end of the Project, a Programmer can witness the changes in the quality of the application in the variances of the Successes Vs Failures of the Test Cases (or Scenarios).
- It needs to be repeated for the various modules of the same Project and can arrive at the unique rating of the Product by consolidating the final rating of the modules.

The example for the Testing by Programmers are listed below:



Similarly the graph for the other modules are plotted then the graph for the entire project is arrived as follows:

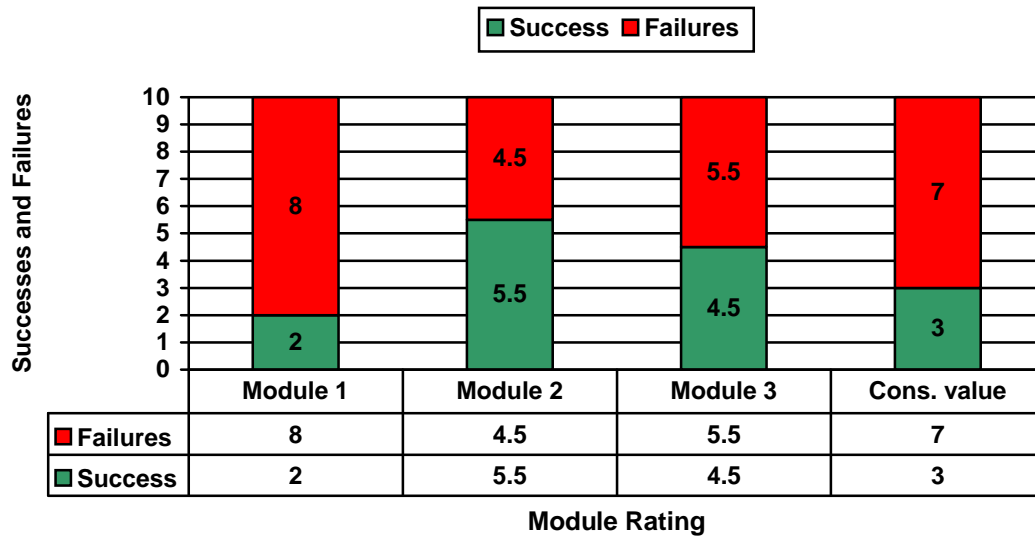


#### 4.2 Application / Product Testing:

- After integrating the modules, the Testing should be performed with most possible scenarios and categorizing the scenarios prior to its modules.
- At the end of every day or sporadically the Success Vs Failure need to be plotted in the graph of 1 – 10 Scale.
- Then consolidation of the final rating of the modules after its first phase of Testing should be derived.
- In this Graph Report, the Risk prone areas and Safety areas need to be shown clearly with required Checkpoints and their impact over the project should also be shown.

The graph after the Integration testing is plotted below:

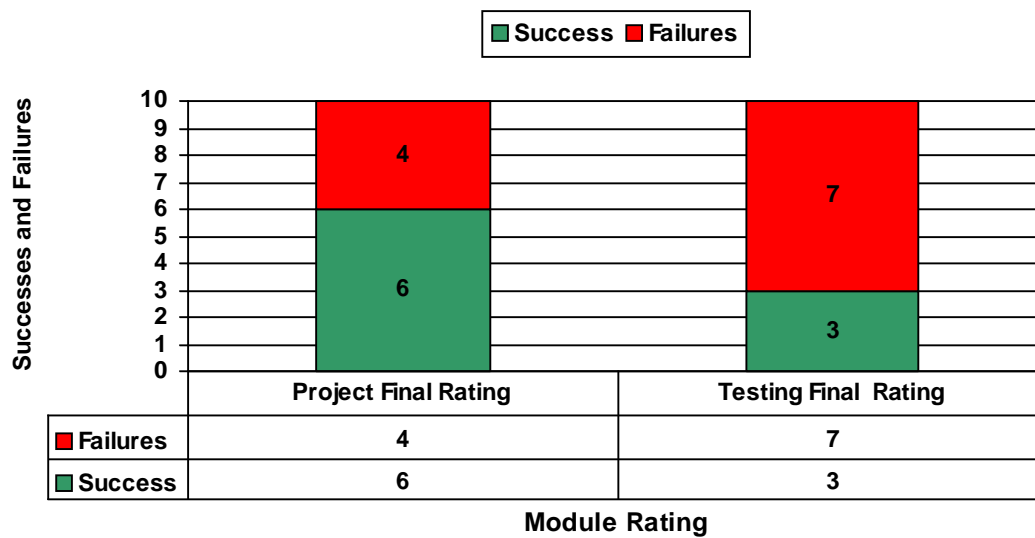
### Testing Final Rating



#### 4.3 Analysis:

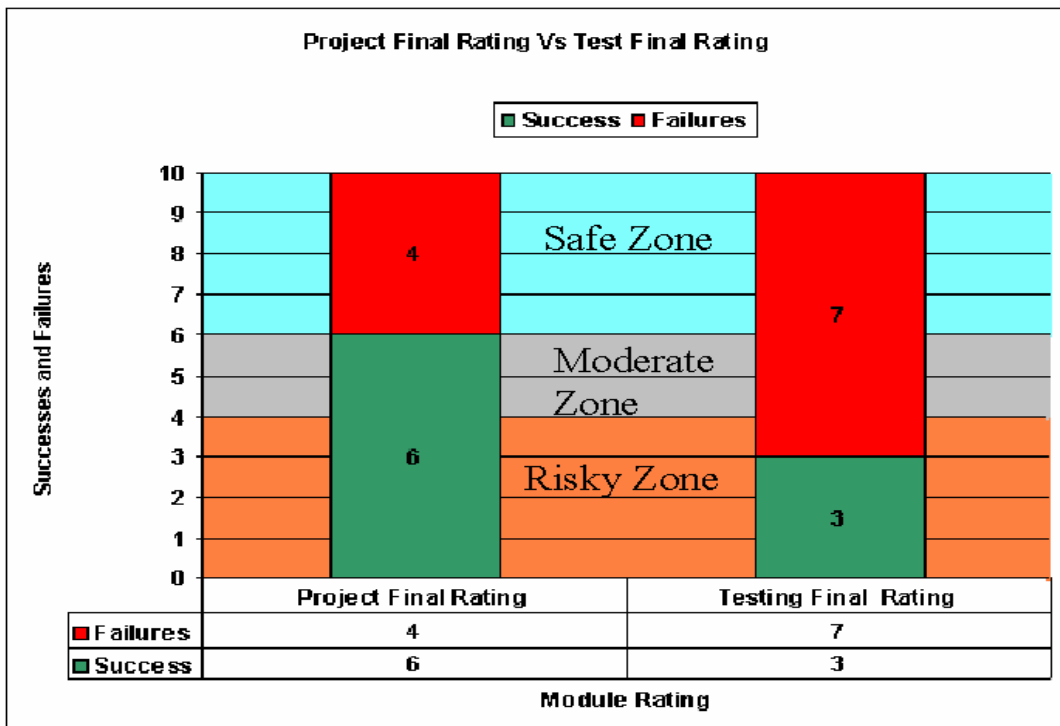
- A complete rating analysis has to be done with the Consolidated reports of the Development and Testing with the Requirements.
- This would help further to decide the number of phases required for Testing to reach the intended end product.

### Project Final Rating Vs Test Final Rating



- Before the inception of Testing phase, it is mandatory to set the Rating level in 1 – 10 (e.g., 7 or 8) Scale that needs to be reached.

From the Analysis-graph the fluctuation (or deviation) of the rating is identified and the further process to reach the desired quality is taken through the regular process. Even in the next phase after the next analysis, the same steps can be followed. This can be done further, by choosing a number between 1 and 10 as its key objective and can arrive with the below mentioned graph.



**5.0 Computational Factors:**

The Test Cases based on the scenario or the criterion is segregated based on the modules and then various aspects are then computed pertaining to those modules. These various ratios listed below helps in calculating various factors to plot a pellucid graph.



- **Test Focus Ratio:** Test Focus Ratio is defined as the ratio of the Number of Test cases covered to the Number of test cases intended to cover.
- **Test Success Ratio:** The Success Ratio is defined as the ratio of the Number of Test cases passed to the Number of Test Cases covered.
- **Defects Ratio:** The Defect Ratio is defined as the ratio of the Number of uncovered Defects to the Number of defects logged.
- **Failure Categorization factor:** The failed test cases may be further divided into Crucial, Major, High, Tolerable, Low, Minor and No Categories. These are being rated based on the Business Requirements and the complete assessment on it.
- **Metrics Report:** Right through the regular practice and the normal Metrics calculation, Risk Areas, Safety areas, System Stability can be easily calculated.

#### 6.0 Benefits of Implementation:

- **Quick Overview:** The EAT envisions the percentage of stability of each module. EAT also proves to be an excellent technique for assessing test coverage.
- **Status Reporting:** This would also help in updating the status of the application or the product under test.
- **System Stability:** The Check-Points between the risk area and the safety area help in deducing the Stability of the system.
- **Focus on module-wise defect fixes:** This helps in point out the module wise fix.
- Helps in Decision-making.
- Enables to rate the product in an easy scale of 1 to 10 at various phases.

**7.0 Conclusion:**

This Technique can be customized into any Project and can ensure the cognizance of the quality of the product in every workaday when it is either in the Development phase or in the Testing phase.

**General Explanation:**

ABC Company is developing an application for XYZ Company. It is mandatory for the ABC Company to deliver the application matching the Business requirements of XYZ Company. So the Application will undergo required testing phases to meet the Business requirements. The Developers will be performing the Unit Testing and the Testers will be performing the Integration, System Testing, and Performance Testing to ensure the quality. EAT helps to differentiate quality of product within and among these testing phases. The EAT should be implemented in the Unit Testing level, then it will help to rate the application in the daily level as a part of Developer's Unit Testing. So when the bugs are being fixed, a consolidated Unit Testing Rating of 1-10 should be made, based on which the decision should be made to move to the next level. Then the Integration Testing needs to be performed. The EAT should be implemented even at this level and should arrive at the test results in a scale of 1-10 with the Success Vs Failures history. Prior to the arrived results, a comparison of Unit Testing (Developers Testing) Vs Integration Testing results should be made and a decision need to be made whether the quality of Product has improved. If anywhere quality is not improved or not to the benchmark set, then a decision should be taken to perform another round of Component Testing to recuperate from the bugs domination. Then another round of Integration Testing should be made and then same steps need to be followed by comparing the Unit Testing Vs Integration Testing results. If they are to the satisfactory level, then it can enter in to the next phase, System Testing. Similarly the Test Results of the System Testing should be compared with the Unit Testing, Integration Testing as Unit Testing Vs Integration Testing Vs System Testing where in by measuring the growth of the application in a scale of 1 - 10 as supported by EAT. The Analysis report would help in deciding the kind of testing that need to be performed to attain

the intended quality. The Status of the application can be reported to the Clients at any circumstance, which will give them some confidence over the Company. Meanwhile if it is followed diligently with an appropriate Test Plan, it would help in attaining the better results.