# Enhancement of Software Effort Analysis Using Web Based Tool Estimator

Meenakshi , Senthil Kumar.

Dept of Computer Science and Engineering, Valliammai Engineering College,(Affiliated to Anna University)

Kattankulathur, Tamil Nadu.

Dept of Computer Science and Engineering, Valliammai Engineering College,(Affiliated to Anna University)

Kattankulathur, Tamil Nadu.

**Abstract**— Software engineering is the discipline which paves the roadmap for development of software within given schedule and effort and with the desired quality. The initial process with software development includes estimating the size, effort and time required for the development of the software. With the increasing size and complexity of software, software development has become a more tedious process and hence needs to take care of even the simplest activity in the development process. Several techniques are available for effort estimation. However, none of the existing methods guarantees the necessary accuracy. In this paper, we develop the web based tool, based on function point analysis, which shows the influence of quality attributes in estimation and increases the accuracy of effort estimation. By implication, the research suggests that the web based tool is efficient and accurate to make estimators and developers more accruable in the software estimation.

**Keywords**— Software effort estimation, function point analysis, quality factors, accuracy.

## I.INTRODUCTION

Software engineering is the discipline that aggregates the application of scientific and technological knowledge through the medium of sound engineering principles, to the production of computer programs, to the requirements definition, functional specification, design description, program implementation, and test methods that lead up to test the code. Software engineering is about engineering the software development process. It requires highest degree of analyses, hard work and the management of the two. With the increasing size and complexity of software, software development has become a more clamorous process and hence needs to take care of even the simplest activity in the development process. The problems being faced in the software developments are cost overrun, schedule overrun and quality degradation [1].

In the core of these problems lies the problem of poor estimation. Wrong estimation surely results a disaster in the development process. Effective estimation is essential for proper project planning and control and is one of the most critical and challenging task in the development process. Under-estimating a project leads to quality degradation, employee over exploitation and setting short schedule and hence results in missed deadlines. Over-estimating is even worse than the previous condition; allocating more resources to the project and thus increasing the cost of the project without any scope.

Proper planning of the project and tracking the project development is the second essential task for assuring the success of the project. Once the estimates are available the next task is to assign the tasks to individuals [2]. Regular feedback from the development process is helpful in determining the status of the task and the project. Tracking gives opportunity to the project manager to take care of any unexpected situation while development.

As stated earlier, estimation plays the key role in the management of the development process, it is essential that the model or the method being used should be correct and stratified with the most recent data available and if standard parameters are being used in the method then those parameters should be well calibrated with the available data.

## I.    MATERIALS AND METHODS

### A.  The Current Scenario

In the current days, five categories of software estimation techniques are used. The following figure shows five categories of software estimation techniques.
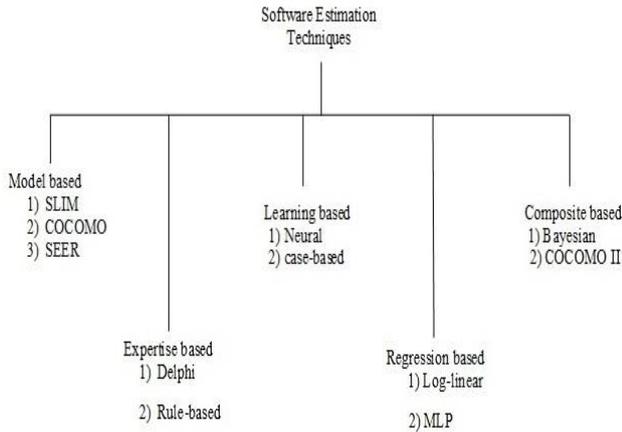
Fig. 1  Software Estimation Techniques

Various estimation techniques have been developed in the past which follows mathematical model for estimation. SLIM (Software life cycle model), COCOMO (Constructive Cost Model), SEER (System Evaluation and Estimation of Resources) are some of the model based techniques for software estimation. Projects' related data is used as input in these techniques and past projects' data is used for calibrating the models. When past projects' data is not available then experts' knowledge is used for estimation. Delphi and Rule-based techniques comes under this category. Delphi technique is based purely on the experts' judgment whereas rule based technique is adopted from the artificial intelligence domain in which a set of rules work together to get the output i.e. the estimates [2].

While planning the project development, the estimates and the productivity of the developers are considered as the baseline and the task is assigned to developers according to their abilities. Developers have to give the information about the work done by them on a daily basis.

The model based technique is based on the parameters of the method being used. These parameters need to be calibrated according to the past data available for different projects in the organization. Calibration has importance because it is going to affect the overall process in future and hence needs great care. The data collected is first checked for consistency, correctness and completeness. And then the approved data is used for calibrating the parameters of the model and new values are assigned to the parameters [4].

*B.   Drawbacks In The Current Scenario*

The existing tools have the following drawbacks in the current scenario [1, 3, and 4].

1)   Tools available for the above activities are isolated to each other i.e. the tools available are either estimation tools or for planning and tracking.

2)   The tools available for planning used to send the information of task assigned to individuals through mails and the information pertinent to the assigned task is kept in some version control system.

3)   Any supporting documents or reports should be available to the person in the organization like SRS

for the project, design specification. Current tools do not have this feature.

4)   During the development, the management needs to keep track of information about the status of project; the tools available do not have such features.

5)   Reports at any stage of development are needed another important feature absent in available tools.

6)   While calibration, past projects' data need to fetched manually.

7)   The method used for calibration of tools does not incorporate the expert's judgment in the resulting parameter values.

*C.   Purpose Of Project Management*

Resources and activities are the key players in any organization for completion of any project. The purpose of project management is to first find out the activities needed to take the project to its end and secondly to allocate resources to these activities in a planned way.

Project management is a vast area which includes all the activities in the above list [5]. The triangle of relationship of the project management is shown in the following figure.
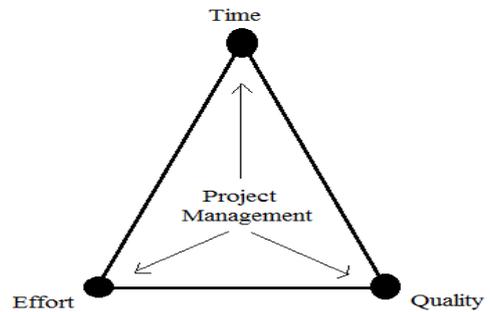


Fig. 2   Triangle Relationship of Project Management

Quality, effort and time are inter-related. If the project demands a higher quality then it is going to use more resources and the effort required will be high and the effect will percolate to time. The first challenge that project management faces is to ensure that the project is delivered within time and budget and with the desired quality. The second challenge is more crucial and grueling one for optimizing the resource requirements. These challenges make the project management a taxing and conspicuous task for any organization.

## II.FUNCTION POINT ANALYSES

As the system grows in size, it is really hard to estimate the size of the software early in the development. Divide and conquer has been the best strategy for tackling bigger problem for decades. Function point analysis, introduced by Allan J Albrecht of IBM in late 1970s, follows the concept of divide and conquer strategy for estimating the size of any software [6].

FPA breaks the system into smaller pieces so that intricacies of the systems become more visible and can be analyzed better. Function point analysis measures size of the

software on the basis of the functionalities to be provided by the software. The method quantifies the functionalities of software by the information provided by the user based on logical design. FPA estimates the size of software in terms of function point counts (FPC) which can be converted into SLOC easily if the equivalent SLOC for unit FPC is available.

### A. Function Point Counting Process

Conceptually function point analysis defines data in two levels; data at motion and data at rest.

Every application has numerous elementary processes which includes various transactions for data movement. It includes transactions bringing data into the application domain and transactions taking data out of the application domain. These are referred as transaction functions. The data maintained by the application or by another application are known as data at rest and referred as data functions.

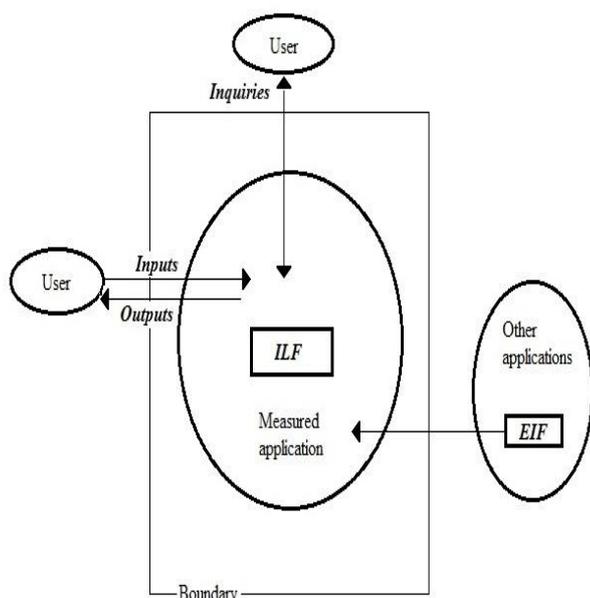Following are the types of data and transaction functions:



Fig. 3 Components in Function Point Counting

### B. Steps For Counting Function Points

Required information for counting is obtained from the software requirement specification. The steps for counting function points are as following [1, 6, and 7]:

1. Identify data functions (External Interface files and Internal Logical Files) and rate them.
2. Identify transaction functions (External Input, External Output and External Inquiry) and determine their complexity.
3. Compute unadjusted function points. Number of EI, EO, EQ, ILF and EIF for each complexity level (Simple, Average and Nominal) is obtained and the corresponding weight for each complexity level is multiplied with the count to finally get the unadjusted function point count. Details of function point count are available in appendices.

4. Determine the ratings of 14 general system characteristics.
5. Calculate value adjustment factor (VAF).

$$VAF = (TDI * 0.01) + 0.65$$

Where, TDI = Total Degree of Influence obtained by multiplying the ratings of general system characteristics.

6. Compute function point counts.

$$FPC = UFP * VAF$$

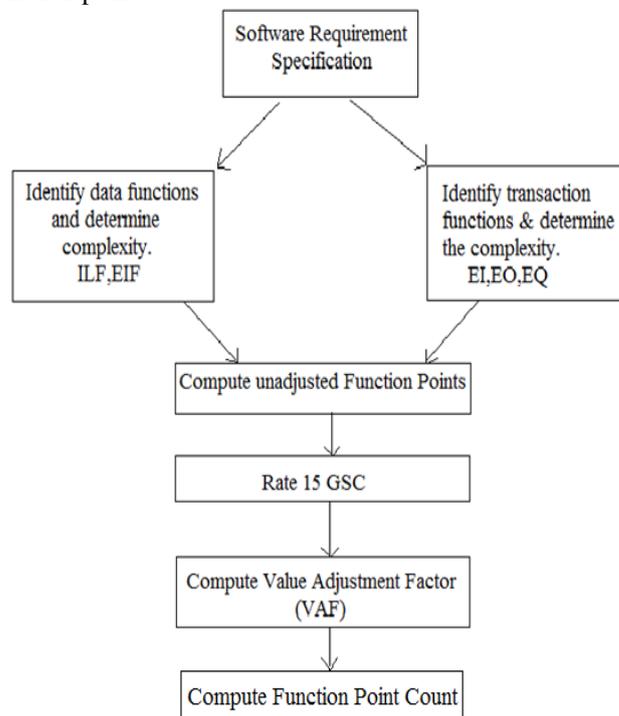The following figure shows the steps for counting function points.



Fig. 4 Steps for Counting Function Points

### C. Proposed Methodology

The major problem in the current scenario is the isolated estimation, planning & tracking and calibration, so the solution would be *Project Management Software* that will combine these activities [8]. The proposed system will first stores the details of the projects, clients and developers which are right now in paper form or if available in electronic form are in isolation to each other. The information about the projects, clients, developers would be available easily. The system will automate the process of the estimation using the functional point model for effort estimation. The system will also help in tracking the status of project by taking daily input from each developer in the organization and will show the status in the form of a Gantt chart.

The system will generate the reports for the projects. While calibrating the model the system will incorporate the experts' judgment in the final values of parameters of the model. The system will give the information about the activities in the organization and the time taken in each activity.

*D. Benefits Of Proposed Solution*
1. Clumsy calculation for estimation is no longer needed.
2. Planning and tracking would rather be a simpler task.
3. Information about the projects, clients and developers are no longer needed to be stored in other forms.
4. Activity details would be available easily.
5. The reports could be generated with a single mouse click.
6. Notification on various conditions can be customized according to the users' choice.
7. Data for calibration would be available in the tool itself and no manual data entry is required for calibration.
8. The calibration would be more accurate and hence the estimation too.
9. With all the information available management will have an edge in improving the conditions in the project development.
10. Solution will be available at low cost.
11. The system could be extended to meet any future requirement easily.

In system analysis more emphasis is given to understanding the details of an existing system or a proposed one and then deciding whether the proposed system is desirable or not and whether the existing system needs improvements. Thus, system analysis is the process of investigating a system, identifying problems, and using the information to recommend improvements to the system. To analyze a system, one has to study the systems in details. System analysis consists of system element, process and technology.

E. *System Overview*

The current scenario includes a beginning estimate of how many hours a project team will need to complete the software project. This estimate is broken down by 4 phases that the software project will go through in its life cycle. The estimator may enter the release number, specification name, work type of that particular specification, its associated business value, risk associated with that specification, and its preset complexity [9, 10]. These values are taken as the preliminary input for the functional point count. The next step includes considering the 14 general system characteristics to calculate the value adjustment factor. These values are multiplied to get the initial estimate.

However, the accuracy of the effort obtained with this method can further be improved by considering the non-functional characteristics. The proposed methodology of effort estimation would take the performance metric into consideration in order to improve the accuracy of the effort estimated using functional point analysis.

This paper initially investigates the structure of the system, which is currently in use, with the objective of identifying the problem and difficulties with the existing system. The major steps involved in this paper included defining the user requirements and studying the present system to verify the problem. The performance expected by the new system was also defined in this paper in order to meet the user requirements. The information gathered from various documents were analyzed and evaluated and the findings reviewed in order to establish specific system objectives.

## III. IMPLEMENTATION

A. *Implementation Description*

Estimation of effort is implemented in the first phase. A diagrammatic representation of how the proposed estimation technique is carried out is shown below:
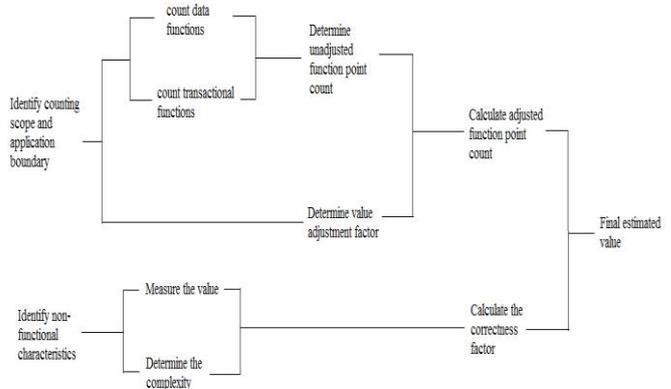


Fig. 5 Effort Estimation

Initially identify the scope of counting functional aspects of a component or a module along with its application boundary. Counting the component's functional characteristics is considered the most important and difficult task in the overall estimation process.

Functional requirements are very detailed and outline exactly what needs to be delivered and would typically be read by business analyst, developers, project managers and testers.

A boundary must be drawn around each system to be measured prior to classifying components. This boundary must be drawn according to the sophisticated user's point of view. In short, the boundary indicates the border between the project or application being measured and the external applications or user domain. Once the border has been established, components can be classified, ranked and tallied.

The next step is to identify a RET, a DET and a FTR. Understanding how to identify DET's and FTR's is critical to distinguish one transaction from another. While in practice understanding the exact number of DET's and FTR's may not impact a function point count, understanding DET's and FTR's can help understand how to count function points for enhancement function point counts.

With a clear understanding of how to classify the obtained DETs and FTRs as one of the EIs, EOs, EQs, EIFs and ILFs, the next step is to determine the business value and its associated preset complexity factor. These two factors greatly influence the function point value. With all the available data, the unadjusted function point value is calculated.

There are external factors which affect the software a lot and also the cost of it. When we submit a function point to a client, the client normally will skip everything and come to GSC first. GSC gives us something called as VAF (Value Adjustment Factor). The performance of the estimated effort is further improved by the proposed correctness factor, which was indicated in Fig. 5. The performance metrics of non-functional characteristic is considered to calculate the correctness factor and its value is calculated by using the formula given below:

$$\text{Correctness factor (CF)} = 0.01 * \left( \sum_{i=1}^{8} V_i * C_i \right)$$

*Where,* $V_i$ = *Value of each performance factor*
$C_i$ = *Complexity factor*

The correctness factor is then added to the adjusted function point to obtain the corrected function point. This value is then used to calculate the effort for the project. The effort can then be used for efficient planning of a software project which will lead to proper utilization of resources and facilitate accurate cost estimate.

The estimated results page displays the function point count, value adjustment factor and the correctness factor obtained from previous pages. These values are then used to calculate the adjusted function point and corrected function point values. These two values are then used to compute the effort estimation with and without correctness factor. These two values are used for comparing the results and determine the accuracy of the estimated results.
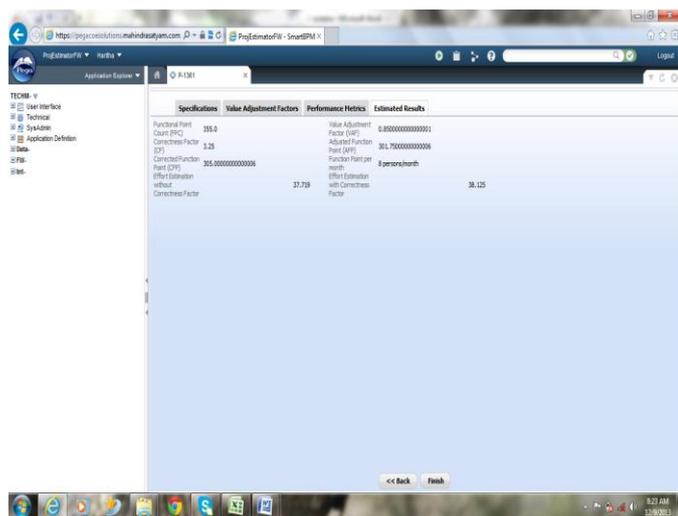


Fig. 6 Screen displaying estimated results

## IV. RESULTS AND DISCUSSIONS

The proposed tool is implemented with a sample data from order fulfillment Telstra BOA sizing project obtained from Tech Mahindra. The estimated results obtained with and without correctness factor is compared with each other. The results are plotted in graph to compare their accuracies. The graph is shown below:
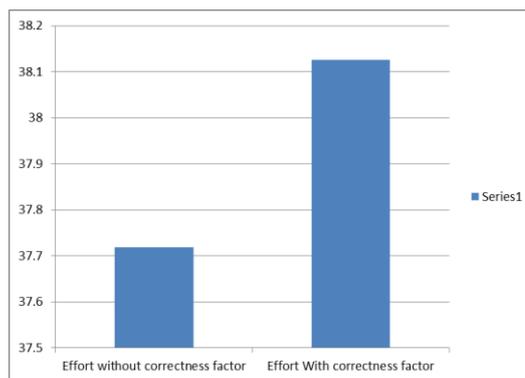


Where,

$Effort_{old}$ = Effort without performance metrics

$Effort_{new}$ = Effort with performance metrics

From the graph it is known that the effort obtained with the influence of performance metrics yields a more accurate result when compared with the traditional effort calculated using adjusted function points. The effort obtained by using the corrected function point also found to be more accurate when compared with the actual effort obtained from the sample project. Hence, it can be practically proved that the proposed methodology yields more accurate results than the traditional methods.

## V. CONCLUSION

The application has been developed with much care that it is free of errors and at the same time it is efficient and less time consuming. The important thing is that the system is robust. Also provision is provided for future developments in the system. A tool for effort estimation is done and the result is compared with traditional effort calculated using function point analysis. The calculated effort using the proposed method is found to obtain a more accurate result when compared with the traditional approaches. This work is very useful to the Time critical and Mission critical project which needs more accuracy and high quality estimation. The estimated result can be used for further planning, tracking and calibrating the project.

## REFERENCE

[1] Luigi Lavazza, Vieri del, Bianco Geng Liu "Analytical Convertibility of Functional Size Measures: a Tool-based Approach" 2012 Joint Conference of the 22nd International Workshop on Software Measurement and the 2012 Seventh International Conference on Software Process and Product Measurement

[2] H. Azath, R.S.D. Wahidabanu "Efficient effort estimation system viz. function points and quality assurance coverage" Published in IET Software, Received on 17th August 2011, Revised on 11th November 2011

[3] Bing Chiang Jeng, Dowming Yeh, Deron Wang, Shu-lan Chu and Chia-mei Chen "A Specific Effort Estimation Method Using Function Point" Journal Of Information Science And Engineering 27, 1363-1376 (2011)

[4] Marcus Meyerh¨ofer, Klaus Meyer-Wegener "Estimating Non-functional Properties of Component-based Software Based on Resource

Consumption" Electronic Notes in Theoretical Computer Science 114 (2005) 25–45

[5] Matijevic .T, Ognjanovic and Sendeji.R."Developing open source application for function point analysis of software Projects", IT'12, 200-204, 2012.

[6] Matijevic .T, Ognjanovic and Sendeji .R.,"Enhancement of software projects' Function point analysis based on conditional non-functional judgements", Mediterranean conference on Embedded computing, 250-254, 2012.

[7] Poonam Kaushal., "Software effort estimation and risk analysis – A survey". Int. J. of Engineering and Innovative Technology, pp18-22, 2012.

[8] Symons,1988,function point analysis: Difficulties and improvements.IEEE Trans.software Eng,14:2-11.

[9] Stirling et al., "Multicriterion decision making with dependent preferences" . IEEE symposium on computational intelligence in multi criteria decision making, pp227- 234,2007.

[10] "PEGA systems", www.pega.com