ISSN : 2494-9150 Vol-02, Issue 05, Aug 2016.

Adaptive Expert Estimation Models for Software Development Methodologies

[1]Manohar K. Kodmelwar, [2]Dr. Shashank D. Joshi, [3]Dr. V. Khanna

[1]Research Scholar, Bharath University, Chennai, India, [2]Professor, Bharathi Vidyapeth, Pune, India.

[3]Head, Bharath University, Chennai, India.

[1]kodmelwarmk@gmail.com, [2]sdj@live.in, [3]drvkannan62@yahoo.com

Abstract - The software estimation techniques are available in used by industries, but still the accuracy of cost & effort estimation is not upto the mark. The software estimation is challenge for industries. The estimation is important to understand the how persons per month require for the development. The estimation before the development is important to avoid the delay, over budget and keep customer satisfied. The industries till now used the various methods like waterfall model, spiral model, V shaped model, currently the industries are using the scrum the agile methodology. This paper will explain the various techniques for software estimation. The model selection for estimation is trial method for industry. This paper will explain the various techniques for software estimation.

Keywords — **TCP (Technical Complexity Factors),UCP (Use Case Point), EAF (Efforts Adjustment Factors) Cost Drivers, COCOMO , FP (Function Point), Efforts Estimation, EF (Experience Factor).LOC(Line of Code)**.

I. INTRODUCTION

It is by survey found that the most of the project overrun the estimated. Therefore is the challenge that to estimate the correct effort and cost required for the development is available. The existing estimation techniques solve the problem of effort & cost required upto some extent. The customer satisfaction about the schedule & cost is critical point. Still there is requirement that the more accurate method should be developed.

The various model like COCOMO, UCP & Function point analysis are available for estimation. In these method various points are applied & the effort are calculated.

II. ESTIMATION TECHNIQUES

The estimation techniques are catgeriosed into non algorithmic & algorithmic based. In non algorithmic based estimation techniques[1][2][3].

1. Expert based judgment : in this technique the expert views are taken . The experts are experienced. The Delphi technique is used for understanding. Where the group of expert opinion is taken by giving the forms. The expert reviews are collected & then group will discuss & finally decisions are taken.

2. Estimating by analogy: In this method the completed project which is similar to the existing project is compared & decision is taken.

3. Top Down Approach: In this the from the estimation are done on the basis of the global properties & then it is divided into the small component

4. Bottom up approach: In this method the cost of each component is estimated & then results are combined to get the overall estimation.

In algorithmic based estimation the mathematical formulas are used to calculate the effort. These mathematical equations are based on research and historical data and use inputs like Lines of Code , defined functions and other cost drivers use of

which programming language, technique for design, the proficiency levels, factor of risk considerations, etc. In the category of algorithmic models like Constructive cost models, bohem's Putnam model, function points based models [17][7].

1. COCOMO Model

This model suggested by Barry Boehm (Boehm, 1981), is the most popular method which is categorized in algorithmic methods. [4][3][16]

A) Simple COCOMO : This model was proposed by the Barry Boehm, which follows following formula:

$$\text{Effort} = a * (\text{K LOC})b.$$

B) Intermediate COCOMO Model : In this method the effort adjustment factor is used in the simple COCOMO effort formula.

$$\text{Effort} = a * (\text{K LOC})b * \text{EAF}.$$

Cost Driver Sample Project Value Description

*DATA Database size, CPLX is a Product complexity, TIME for Execution time constraint, STOR for Main storage constraint., **RUSE** Required reusability, **OCU for** Documentation match to life-cycle needs., **PVOL for** Platform volatility, SCED Scheduling factor, RELY Required reliability, TOOL for Use of software tools , APEX for Application experience., ACAP Analyst capability, PCAP Programmer capability, PLEX for Platform experience., LTEX for Language and tools experience, **PCON** Personnel continuity, **SITE** Multisite development.*

C) COCOMO-II model new Scale factors added for estimation precedentedness (PREC),Development flexibility (FLEX) ,Architecture/ risk resolution (RESL),Team ohesion (TEAM), Process maturity (PMAT, derived from SEI CMM)

NOMINAL PERSON-MONTHS = A*(SIZE)^B

B = 0.91 + 0.01 $\sum$(SCALE FACTOR RATINGS)

Where KLOC are the code size, a & b are the complexity functions in the code.

2. PUTNAM Model

In Putnam model for calculating the estimate for a software task the software equation is solved for *effort*:

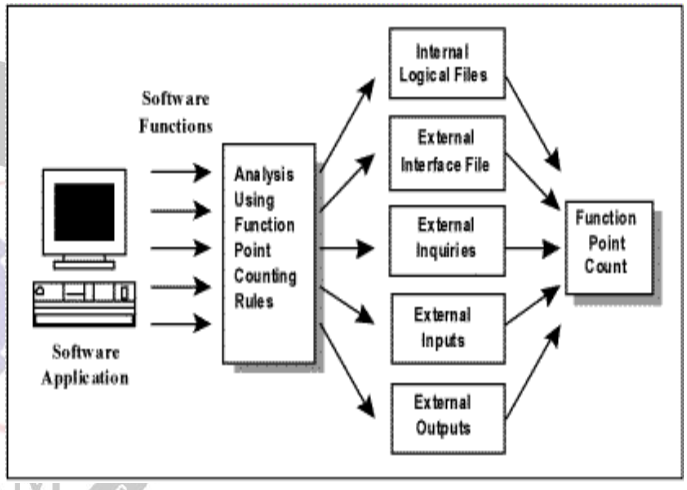
$$\text{Effort} = \left[\frac{\text{Size}}{\text{Productivity} \cdot \text{Time}^{4/3}}\right]^3 \cdot B$$

B= It is a scaling factor and is a function of the project size

3.FUNCTION POINT ANALYSIS

FP defined by Allan Albrecht is a unit of measurement to express the amount software functionality In this how much functionality is given o user on the basis of that the size & complexity of software is calculated.[6][8][18]

It is independent of programming language .It calculate by using five parameters shown in fig-1[20] are first is external inputs, second is external outputs, third is external inquiries, fourth is logic internal files, and fifth is external interfaces are categories into simple, average and complex.



Fg-1 Function Point Model

3. Bayesian Belief Network : This model framework is based on the four basic sub-models, which are used to model quality, effort and schedule information[13].[19]

4. USE CASE Point The UCP technique was invented byin the year 1993 by Gustav Karner, while employed in the Objectory Systems, after some years is is merged into Rational Software and again merged in IBM. The method for predicting the estimation of size for developing a system built on a calculation using following elements:[15]

Unadjusted Use Case Weight (UUCW) – In the UUCW the number of transaction are found & the according to the transaction the weights are assigned. The types of transaction are simple , average & complex.

Unadjusted Actor Weight (UAW) – In this UAW it is categories in the simple, average & complex depending on the number of actors like external interfaces by API, Communication Protocol & GUI. The weights are assigned according to the categories.

Technical Complexity Factor (TCF) – In this TCF the weights are assigned according to the technical factors like distributed system, enduser efficiency, code reusability, easy to use, easy to install, portability, security, access, training, parallel processing ,internal process, response time etc.

Environmental Complexity Factor (ECF) – In this ECF the weights are assigned according to the environmental consideration like team experience, familiarity, stability, part time staff, used programming language, motivation, analysist capability etc.

Comparison Prediction:

Use of agile for project management for cost estimation finds various benefits over traditional model. It lowers the cost, team size & use the iterative & incremental method. For cost estimation in agile find better communication with customer & satisfaction to customer. The modules are developed by priority & in every completion of module the customer is contacted. The new addition in the project is possible at any time. Like traditional Models have less interaction & chances of over budget.

III. CONCLUSION

The estimation of software cost & effort is critical issue. This paper gives the brief idea about the estimating techniques available. The choosing of estimation techniques is depending on the application to developed. The paper will gives idea about the factors affect on the estimation by various point. These consideration are important otherwise the delay in schedule & cost can increase. The overall study indicates that in terms of calculation of software cost estimation of different project the new trends to be developed which can reduce the over budget, delay & increase the features of project. The agile methodology for software development will be better.

ACKNOWLEDGMENT

I wish to express my sincere thanks to Dr. M. Sundararajan, Dean Research, Bharath Institute of Higher Education & Research for providing all necessary facilities.

I wish to place my sincere thanks to Dr. Shashank D. Joshi ,Guide for guiding me during the paper preparation. I place on record my sincere gratitude to Dr. V. Khanna, Head IT, Bharath Institute of Higher Education & Research for continuous encouragement.

I am also thankful to my parents for unending support. I am also thankful to friends & all those who directly or indirectly helped me.

References

[1] Y. F. Li, M. Xie, T. N. Goh, "A Study of Genetic Algorithm for Project Selection for Analogy Based Software Cost Estimation" , IEEE2007.

[2] Khaled Hamdan, Hazem El Khatib, Khaled Shuaib," Practical Software Project Total Cost Estimation Methods", MCIT10, IEEE, 2010.

[3] Chetan Nagar, "Software efforts estimation using use case point approach by increasing technical complexity and experience factors", IJCSE, ISSN:0975-3397, Vol.3 No.10 , Pg No 3337- 3345,October 2011.

[4] Chetan Nagar, Anurag Dixit, "Software efforts and cost estimation with systematic approach",IJETCIS, ISSN:2079-8407, Vol.2 No.7, July 2011.

[5] Chen Qingzhang, Fang Shuojin, Wang Wenfu, "Development of the Decision Support System for Software Project Cost Estimation", World Congress on Software Engineering, IEEE2009.

[6] Yinhuan Zheng, Yilong Zheng, Beizhan Wang, Liang Shi, "Estimation of software projects effort based on function point", 4th International Conference on Computer Science and Education, 2009.

[7] Jairus Hihn, Hamid Habib-agahi, "Cost Estimation of Software Intensive Projects: A Survey of Current Practices", IEEE, 2011.

[8] Yunsik Ahn, Jungseok Suh, Seungryeol Kim, Hyunsoo Kim, "The software maintenance project effort estimation model based on function points", Journal of software maintenance and evolution, 2003.

[9] Jin Yongqin, Li Jun, Lin Jianming, Chen Qingzhang, "Software Project Cost Estimation Based On Groupware", World Congress on Software Engineering, IEEE, 2009.

[10] Pichai Jodpimai, Peraphon Sophatsathit, and Chidchanok Lursin- sap, " Analysis of Effort Estimation based on Software Project Models", IEEE, 2009.

[11] Nancy Merlo Schett, "Seminar on software cost estimation", University of Zurich, Switzerland, 2003.

[12] Hao Wang, Fei Peng, Chao Zhang, Andrej Pietschker, "Software Project Level Estimation Model Framework based on Bayesian Belief Networks", 6th International Conference on Quality Software (QSIC'06), IEEE, 2006.

[13] angyang Yu, Charlottesville, "A BBN Approach to Certifying the Reliability of COTS Software Systems", annual reliability and maintainability symposium, IEEE, 2003.

[14] Ying Wang, Michael Smith, " Release Date Prediction for Telecommunication Software Using Bayesian Belief Networks", E Canadian Conference on Electrical and Computer Engineer ing, IEEE, 2002.

[15] Suresh Nageswaran, "Test Effort Estimation Using Use Case Points", Quality Week , San Francisco, California, USA, June 2001

[16] Kusuma Kumari B.M " Software Cost Estimation Techniques, *International Journal of Emerging Research in Management &Technology ISSN: 2278-9359 (Volume-3, Issue-4)*

[17] Hareton Leung, Zhang Fan, "Software Cost Estimation", Article 2001.

[18] Jyoti G. Borade, Vikas R. Khalkar " Software Project Effort and Cost Estimation Techniques" International Journal of Advanced Research in Computer Science and Software Engineering 3(8), August - 2013, pp. 730-739

[19]http://www.agenarisk.com/newsletters/NS5Articles/Siemens_BNs_software_project_estimation.pdf

[20] www. Softwarems.com

