Asian Journal of Multidisciplinary Studies

ISSN: 2321-8819 (Online)
2348-7186 (Print)
Impact Factor: 1.498
Vol.4, Issue 4, March 2016

A structured approach to estimate efforts for small sized software projects

Suvarna R. Jagtap
Assistant Professor
Department of Computer Science
Bharati Vidyapeeth Deemed University
Yashwantrao Mohite College of Arts, Science and Commerce, Pune

Abstract

When it comes to estimate the size of the project people always find it difficult to estimate the size of the project that is how 'big' or 'small' it is. Usually software project sizes are generally categorized as small, medium and big depending on the source lines of codes or function points. For benchmarking and estimating purposes it often is useful to appreciate the relative size of a specific piece of software. It is feasible and practical to relate relative sizes to a absolute functional sizes expressed on a logarithmic scale.

The majority of projects (approx.90%) fall in the range 'small' to 'large'; i.e. larger than 30 function points but smaller than 3000 function points. Software cost estimation is a tedious process but it is necessary to keep track of project development progress. Generally , a more serious approach is used for medium and large projects. This paper throws light on various issues related to the estimation of small sized software projects.

Keywords : Estimation , COCOMO II ,SLOC , FP

1.Introduction

Software project development is a tedious and complex activity which should be carried out in a planned manner to achieve success. IT companies face a lot of problems and losses due to software cost overruns during project development process. There are number of issues related to cost overruns such as lack of knowledge related to client's requirements, technology change , incompetent team members , geographical distances etc.

Software cost estimation is an approximate judgment of cost required to complete a software project .It is defined as "A set of techniques and procedures that is used to derive the software cost estimate".

It is often measured in terms of efforts as Person – months/years. The overall cost of the project includes managerial cost , development cost and the cost of resources needed. Here , development cost also includes training cost and quality assurance cost.

Basically software cost estimation will never be exact science because there are too many variables such as human , political , environmental , technical .Software development process includes complex activities that are difficult to judge. It may also include different development environments which takes into account varied number of working hours of the team members such as 40+ hrs in North America ,35+hrs in Europe and again 40+hrs in India.

Software cost estimation is needed for the following purposes:-

1) Initial project bidding ,budgeting and planning
2) Cost control planning
3) Protects integrity of the project
4) Suffers from efforts and cost overruns
5) Faces problems due to unclear requirements, new design , new development tools and development.

In the last three decades, many quantitative software cost estimation models have been developed. They range from empirical models such as Boehm's COCOMO models [5] to *analytical* models such as those in [12, 9, 16].

Usually a software project is viewed as an amount of lines of source code generated during the development process. Hence , there are three types of software project in terms of the size namely small ,medium and large.

COCOMO (Constructive Cost Model) models

This family of models was proposed by Boehm [4, 5]. The models have been widely accepted in practice. In the COCOMOs, the code-size S is given in thousand LOC (KLOC) and Effort is in person-month.

A) Basic COCOMO: This model uses three sets of $\{a, b\}$ depending on the complexity of the software only:

(1) for simple, well-understood applications, $a = 2.4$, $b = 1.05$;
(2) for more complex systems, $a = 3.0$, $b = 1.15$;
(3) for embedded systems, $a = 3.6$, $b = 1.20$.

The basic COCOMO model is simple and easy to use. As many cost factors are not considered, it can only be used as a rough estimate.

B) Intermediate COCOMO and Detailed COCOMO :- In the intermediate COCOMO, a nominal effort estimation is obtained using the power function with three sets of {*a*, *b*}, with coefficient *a* being slightly different from that of the basic COCOMO:

(1) for simple, well-understood applications, $a = 3.2$, $b = 1.05$
(2) for more complex systems, $a = 3.0$, $b = 1.15$
(3) for embedded systems, $a = 2.8$, $b = 1.20$

2 .Sample Project Description and Scope

The sample project undertaken to find out various software cost estimation techniques used for estimating cost of small sized projects . As it is difficult to incept such projects using typical software development models such as waterfall or spiral.

The sample project has the following preferences :-

(1) The data collected through a rigorous survey is transferred to tables and an excel sheet .
(2) A model is built in JAVA using NETBEANS which applies Cocomo II estimates for small sized projects . Also an add-in of JAVA , JEXCELAPI is introduced so that linear regression can be carried out on the collected data.
(3) Since the project is small , detailed design or elaboration phase is not necessary.
(4) The project has time constraint.

3 . OBJECTIVES of the project

The set of objectives were proposed during project inception :-

- To identify the estimation processes resulting in systematic lowering of

  estimation errors in small sized projects.
- To identify which Software Cost Estimation model to use for such projects
- To identify factors to improve evaluation and training of people responsible for estimation
- To identify the factors that lead to estimation errors which will enable improved risk management.

4. RESEARCH METHODOLOGY

To meet the above stated objectives the following methodology is applied:-

a) A detailed study of recent literature was carried out to find the traditional approaches used to estimate the cost of software projects by critically analysing a variety of research papers published in international as well as national research journals , books on models and techniques used .
b) Structured interviews were also conducted for collecting the facts where IT people who were resistant to give any information in written . Similarly , record reviews were also taken into consideration for collecting factual information.
c) A rigorous survey was conducted where a compact questionnaire was prepared to collect relevant information about small sized projects and then distributed to project managers ,team leaders , senior application developers and few business analysts.
d) Then the data collected through the survey was pre-processed and made ready to use.
e) Then a model is developed using NETBEANS to calculate the cost of software projects using COCOMOII estimate.
f) Also linear regression is carried out on the collected sample data and results are validated.

5. Data collection and Result

A rigorous survey is conducted by submitting 85 questionnaires to various members involved in project development as well as in business analysis. Out of 85 questionnaires distributed , 57 filled questionnaire were obtained . 18 questionnaires are obtained fully filled while 39 were partially filled. But some of the data included was for medium and large size projects where KSLOC was more than 5 .So in all 10 projects' data is considered for this paper. The data then applied to the COCOMO II model and following results were obtained. PM is calculated by using formula

PM = A*size^E*EM

Here A,B and C are constants ,E is a exponent used for scaling factors while EM is used as a product of effort multipliers.

Table 1.0 The difference between actual efforts and estimated efforts

Sr. No.	Project-ID	KSLOC	Actual Efforts	Estimated Efforts
1	P01	4.85	119.5	6.49
2	P12	4.73	32.5	10.65
3	P21	4.49	40.3	9.23
4	P24	3.92	146.8	6.82
5	P30	4.87	121.9	5.28
6	P33	3.17	25.2	10.65
7	P37	2.92	8.4	2.43
8	P42	4.50	10.8	3.85
9	P49	4.85	352.8	72.92
10	P51	5.10	72.5	21.91

As shown in the above table, there is a vast difference between actual and estimated efforts. There are The following issues have been observed:-

1) Not much attention given to small sized projects as far as project planning is concerned. Usually price to win strategy is applied to such projects by most of the IT companies.
2) Story points/milestones were decided at the initial stages but not much heed given to it afterwards.
3) Usually small sized projects have fixed cost and complexity level is nominal.
4) Bottom approach is most suitable for small sized projects and review of activity based estimation must be considered to get accurate estimation.

Fig. 1.1 Comparison between Actual and Estimated efforts

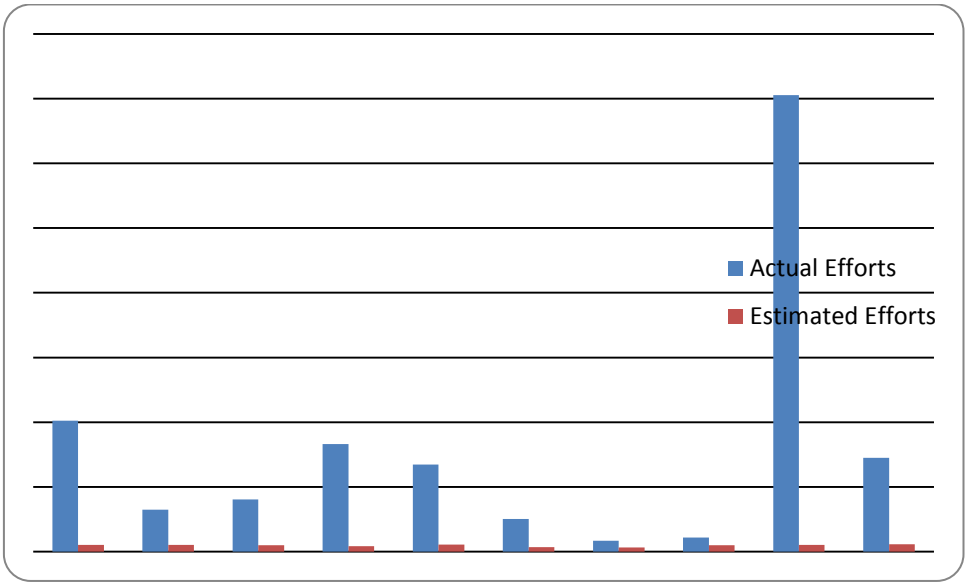
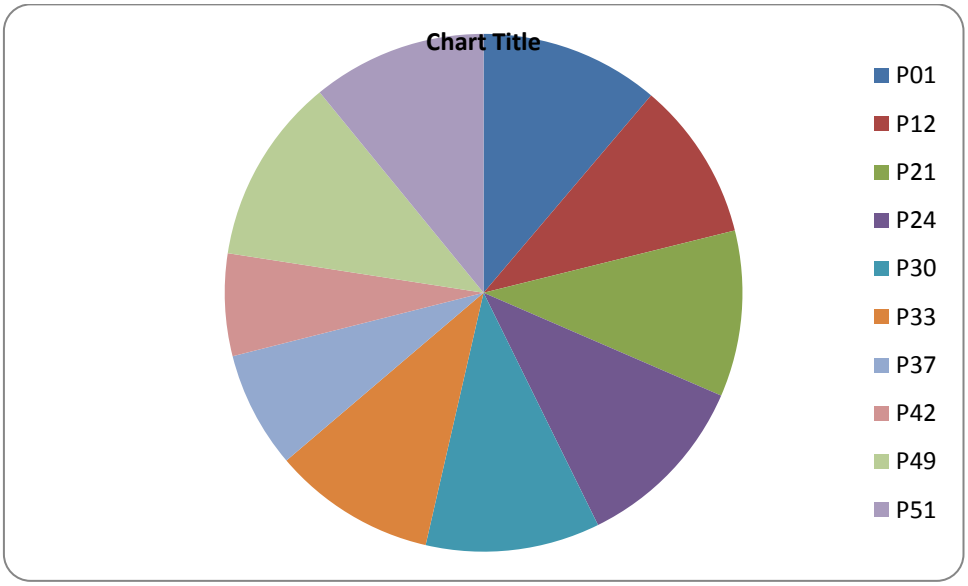


Fig. 1.2 Project wise Mean Relative Error



The above table shows vast difference between actual and estimated efforts as small size projects always suffer lack of planned activity . The following column chart maps the actual efforts and estimated efforts.

6. conclusion

It has been observed form the collected data that expert judgment is the most popular method used to predict the efforts for small sized projects. As well as analysis of actual and estimated efforts is hardly done after project completion. As the projects are small sized , milestones are considered to map project progress instead of phase wise approach in most of the IT companies. The results shown here are based on the factual data of 10 small sized projects collected through the survey Only COCOMO II model is applied to the collected data .Further calibration model should be applied to get more accuracy in the predictions.. The sample size is too small and thus inferences drawn here may be inadequate. So further studies with appropriate sample size and ANN , Naïve Bayesian Network as well as other data mining techniques should be applied in addition to COCOMO II for more appropriate results.

References

1. Agrawal M., Chari K., (2007) "Software Effort, Quality, and Cycle Time: A Study of CMM Level 5 Projects" IEEE Transactions on Software Engineering, Volume:33, Issue: 3, pp. 145 – 156.
2. Heemstra, F.J. "Software Cost Estimation". *http://core.ac.uk/*. N.p., 1992. Web. 21 June 2015.
3. Khatibi, Vahid, and Dayang Jawawi. "Software Cost Estimation Methods: A Review". *http://cisjournal.org/*. N.p., 2010. Web. 21 June 2015.
4. Boehm, B.W., *Improving Software Productivity*, Computer , pp 43-57, May 1987
5. B. W. Boehm, *Software engineering economics*, Englewood Cliffs, NJ: Prentice-Hall, 1981
6. Ashman R. (2004) "Project Estimation: A Simple Use-Case-Based Model" IT Professional Volume: 6, Issue: 4, pp. 40 – 44.
7. Attarzadeh, I., Hock, O.S.,(2009) "Using Enhancement Method to Improve Earned Value Index to Achieve an Accurate Project Time and Cost Estimation
8. Azath H., Wahidabanu R. S. D., (2011) "Efficient effort estimation system viz. function points and quality assurance coverage" IET Software, Volume: 6, Issue :4, pp. 335–341
9. L. H. Putnam, "A general empirical solution to the macro software sizing and estimating problem", *IEEE Trans. Soft. Eng*., July 1978, pp. 345-361.
10. Benediktsson O., Dalcher D., (2005) "Estimating size in incremental software development projects" IEE Proceedings online no. 20050019, Volume: 152, Issue:6, pp253-259.
11. Berger, J., O., (1993) "Statistical Decision Theory and Bayesian Analysis".Springer; 2nd edition Verlag, New York.
12. N. A. Parr, "An alternative to the Raleigh Curve Model for Software development effort", *IEEE on Software Eng.* May 1980.
13. 13.Boehm B, Abts C., Brown W., Chulani S., Clark B.k., Horowitz E , Madachy R.,Reifer D. j. and Steece B., (2000) "Software Cost Estimation with COCOMO II" Prentice Hall, Edition 1
14. Jones, T.C., Programming Productivity, McGraw-Hill, NY 1987.
15. Kansala, K., Integrating Risk Assessment with Cost Estimation, IEEE Software, May/June, 1997, pp. 61-68.
16. G. Cantone, A. Cimitile and U. De Carlini, "A comparison of models for software cost estimation and management of software projects", in *Computer Systems*: *Performance and Simulation*, Elisevier Science Publishers B.V., 1986.