

# Factor of Success in Software Development: Effort Prediction

Akshika Aneja

Assistant professor

Department of Computer Science

GNDU, Amritsar

India

## ABSTRACT

Effort prediction is a vital aspect in software progress, and any abnormalities have an impact on the success of the project. Both under-estimation and over-estimation impacts project execution and lessens the probability of accomplishment of the project. However, under-estimation could have a greater influence due to greater chances of project swamped in terms of effort and cost. It also has effects at the organizational level. The characteristic of effort prediction has always been a challenging task. The need for effective software metrics and improved estimation skills coupled with the dynamic alterations in the software segment pose a bigger challenge.

**Keywords:-** Software Development

## I. INTRODUCTION

Effort estimation is a significant element of software development projects, and the accomplishment of a project depends on the correctness with which we are able to calculate software development effort [1]. When actual effort is greater than estimated effort, it effects the project under implementation in terms of price, plan, client contentment, project/organizational repute, and cost-effectiveness. Together under-estimation and over-estimation of effort have impact on the project over the entire globe.

The need for software metrics to keep pace with the changing aspects of the software arena is one of the important tasks in software effort prediction [2]. Various challenges in project size and estimation include web-based development, evolutionary development, model-driven development, net-centric systems.

## II. ESTIMATION OF SOFTWARE EFFORT

The estimation of effort in a software progress project helps the project manager in taking significant judgments, and in resource planning. Under-estimation of effort leads to cost overruns and Over-estimation of effort results in non-optimal resource allocations [3]. When cost of projects increases, organizations have a tendency to look at to minimize the costs in the project which could tend to avoiding or lessening tasks in the project life cycle that were originally planned. These measures of minimization could affect the class of the software product [4].

viewpoint of cause-effect association. The estimation or forecast for a definite phenomenon will have an effect on other processes. There is high possibility of under-estimation or over-estimation, if the whole series of links is not considered. Prediction and size are inadequately accepted through organizations, When projects face problems in some cases it is too late to do a proper estimation. Hence, they stress the importance of effort estimation and recommend development of the prediction process through a closed cycle of prediction, scheduling, size, and periodical development of estimates [5]. In overall, correct prediction of effort in any project is not a simple job considering the inherent difficulties.

As compared to other area of software development more promises have been broken in the area of software estimation, but we always claim promises since it is essential to live [6].

The abstract environment of software united with dynamically varying requirements could make prediction of software size more challenging which in turn effects the correctness of software progress effort. Moreover, absence of clarity in necessities, rareness of each project, high customer hopes, organizational dynamics, and insufficient assessment of risks are some factors which effect software progress effort. These adopt importance since they impact the understanding of project requirements, alterations in design and development processes, value of deliverables, and resource allocation mixture which effects in an impact of the organization and the project.

It has been a constant test to consider various factors and correctly

Effort Prediction is essential to be observed from the

predict software development effort [6]. The significance related with software effort prediction has directed to initiation of intensive research in this area and formation of formal effort estimation models. Various Effort estimation models could be categorized as empirical models, regression-based models, theory-based approaches, and machine learning techniques [7]. Software development organizations accept global delivery models relating software development teams distributed across the globe. Moreover, aspects such as standardization of project practices, deployment of tools / automation, design and code reusability, could effect the software development effort necessity and facilitate in successful completion of the project.

### III. EFFORT ESTIMATION PRECISION

The probability of accomplishment (P(s)) in a project could be expressed as a function as given below:  $P(s) = f(Eacc)$  where Eacc is the Effort prediction accuracy

Effort prediction accuracy is expressed as  $Eacc = \Delta Ep / Ep$  where  $\Delta Ep$  is the deviation from predicted effort. This in turn is stated as  $\Delta Ep = Ea - Ep$  with  $Ea$  being the actual effort and  $Ep$  being the predicted effort)

The level of accuracy in effort prediction and its impact on the success of a project is shown below:

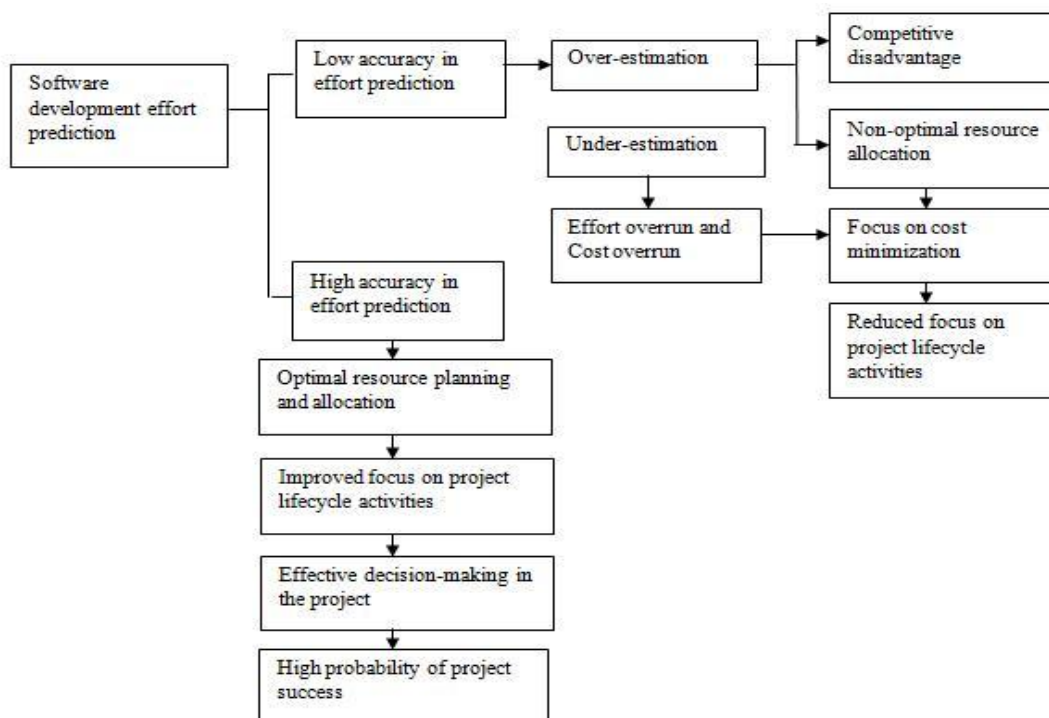


Fig.1. Effect of effort estimation on success of project

### IV. ANALYSIS OF RESEARCH

The incompetence to have high correctness of estimates associated with software development effort and cost has been described in various sources such as reports provided by project management consultancy organizations, prediction-related surveys, case studies relating to failure of projects. Three studies conducted over a period of time and in different countries focus the estimation correctness factor by signifying that the proportion of projects completed beyond budget are 61 percent, 70 percent and 63 percent respectively [9].

According to a study conducted in more than 100 software development organizations by Cutter Consortium in 2008, 48 percent of software organizations have either cancelled or abandoned projects in the previous three years on account of considerable budget or cost overruns. According to the Cutter Consortium report, only 37% of the surveyed organizations had a budget success rate of 70% or more in their projects, and only 10% of the surveyed organizations had a budget success rate of more than 90%. Further, the Cutter Consortium report also refers to Boehm and Valerdi's project estimation performance data of 8000 projects in 350 organizations in 2006 which indicates that 19% of projects were cancelled before completion and 46% of projects had budget and schedule overruns [2].

## V. CONCLUSION

The paper discussed the aspect of project sizing and effort estimation and how these could have a significant impact on the success of a project. It is critical to understand the requirements and customer expectations and appropriately consider them during the process of software sizing and effort estimation. In addition, estimation of software development effort needs to consider aspects such as distributed development models, agile software development approaches, and cloud-based scenarios for solution development. The various research surveys and their analysis discussed in the paper depict the significance of effort estimation accuracy and their impact on the project in terms of budget overruns. The occurrence of overruns in the project in terms of effort and cost due to inaccurate effort estimation does not only cause an impact at the project or program or portfolio levels, but can have far-reaching consequences at the organizational level.

## REFERENCES

- [1] D. Milic and C. Wohin, "Distribution patterns of effort estimation," *IEEE Conference Proceedings of Euromicro 2004, Track on Software Process and Product Improvement*, 2004, pp.422-429.
- [2] B.Boehm, B.Clark, T.Tan, R.Madachy, and W.Rosa, "Future software sizing metrics and estimation challenges," [Online]. Available: <http://www.sercuarc.org/>
- [3] T. Menzies, K. Jacky, and E. Kocaguneli, "On the value of ensemble effort estimation", *Journal of IEEE Transactions on Software Engineering*, " 2012, vol. 38, pp.1403-1416.
- [4] A. Bakir, B. Turban, and A. Bener, "A comparative study for estimating software development effort intervals," *Software Quality Journal*, 2011, vol. 19, pp.537-552.
- [5] L.Buglione and C.Ebert, "Estimation tools and techniques," *IEEE Software*, 2011.
- [6] E.M.Bennatan, "A fresh look at software project estimation: Promises impossible to keep," *Agile Product & Project Management Advisory Service Executive Update*, Cutter Consortium, vol.9.
- [7] P.R. Dinesh, M.S. Kevin, and S.H. Girish, "Software effort estimation using a neural network ensemble," *The Journal of Computer Information Systems*, 2013, vol. 53, pp.49-58.
- [8] V.M.K. Hari, et.al., "SEEP: A toolbox for software effort estimation using soft computing techniques," *International Journal of Computer Applications*, 2011, vol. 31, pp.12-19.
- [9] K. Molokken and M. Jorgensen, "A review of surveys on software effort estimation," *Proceedings of the 2003 International Symposium on Empirical Software Engineering, ISESE '03*.