Determining the Factors Affecting the Accuracy of Effort Estimates for Different Application and Task Types

Saqib Bukhari
Faculty of Information Technology
University of Central Punjab
Lahore, Pakistan
saqib.bukhari@ucp.edu.pk

Ali Afzal Malik
Faculty of Information Technology
University of Central Punjab
Lahore, Pakistan
drali@ucp.edu.pk

Abstract—**An important asset in the skill set of any software project manager is the ability to somewhat accurately estimate the effort required to develop a software application. Acquiring this asset, however, requires a thorough understanding of the factors that may affect the accuracy of these estimates. This paper presents the results of an empirical study conducted to determine the causes of variation in the accuracy of effort estimations for different application and task types. A Pakistani software house that specializes in developing financial transaction processing applications is chosen for this empirical study. Actual and estimated values for software development effort are gathered and analyzed for four different types of applications – web-based, database, parallel processing, and telephony – each having six different types of tasks i.e. business-development, new features, usability, security, support, and performance. Over 1000 data points are considered. Analysis of the results reveals, for instance, that the effort for web-based applications is mostly underestimated while the effort for telephony applications is mostly overestimated. The underestimation in web-based applications is usually due to a failure to account for the learning curve associated with rapidly changing web technologies while the overestimation in telephony applications is usually due to a failure to account for the usage of third-party components.**

Keywords- estimation accuracy; effort estimation; empirical study; project management

I. INTRODUCTION

One of the most important stages in the software development life cycle is the application inception stage. In this stage, the scope of the application is defined, the stakeholders are identified, and the plan for the entire application is laid out. The scope of the application indicates the boundary of the system, the stakeholders provide the functional and non-functional requirements, and the project plan provides a detailed roadmap of the application.

A project plan consists of, among other things, the application's schedule, resource allocations, and effort estimates of application tasks. The application schedule provides a timeline for the entire application including the start and end dates for various tasks and milestones. These dates are determined using the resources allocated to and the effort estimated for these tasks.

Since plans and estimates are future-oriented, it is likely that the actual effort and schedule of a task will differ from the estimated effort and planned schedule. The greater this difference, the more difficult it becomes to manage the application. Therefore, an important asset in the skill set of any software project manager is the ability to somewhat accurately estimate the effort required to develop a software application. Acquiring this asset, however, requires a thorough understanding of the factors that may affect the accuracy of these estimates.

This paper presents the results of an empirical study conducted to determine the causes of variation in the accuracy of effort estimations for different application and task types. A Pakistani software house that specializes in developing financial transaction processing applications is chosen for this empirical study. Actual and Estimated values for software development effort are gathered and analyzed for four different types of applications – web-based, database, parallel processing, and telephony – each having six different types of tasks i.e. business-development, new features, usability, security, support, and performance.

Rest of the paper is organized as follows. Section II explains the empirical setting of our study. Section III summarizes the results of data collection and includes a discussion on the salient aspects of these results. Related work is presented in Section IV while Section V concludes this paper by summarizing the main findings and proposing avenues for further research in this area.

II. EMPIRICAL SETTING

Effort data – estimates and actual values – are gathered from a well-established Pakistani software house that specializes in developing financial applications for the prepaid card industry. This software house has a staff of over 400 software professionals. Applications developed by this software house involve financial transactions with banks. Due

978-0-7695-4927-9/12 $26.00 © 2012 IEEE
DOI 10.1109/FIT.2012.16

to the business critical nature of these applications, quality is of utmost importance and there is a little room for bugs.

In order to analyze the causes of variation in the accuracy of effort estimates, we look at four different types of applications developed by the chosen software house i.e. web-based, database, parallel processing, and telephony. Web-based applications are responsible for client interaction. Database applications are responsible for storing and retrieving sensitive data. They comprise stored procedures, DB functions, and triggers. Parallel processing applications deal with batch processing. They include the core business logic. Telephony applications use Interactive Voice Response (IVR) to perform different cardholder-related operations.

Each application can have 6 types of tasks i.e. business development, new feature, performance, security, usability, and support. Business development tasks are solely initiated by the clients to enhance their specific application. New feature tasks are initiated internally by the software developers to include new generic features in their applications. Performance tasks deal with performance improvement of the application. Security tasks deal with application security to make it more reliable and trustworthy. These tasks are initiated to secure critical financial transactions against various attacks such as SQL Blind, Cross Site Scripting, Cross Frame Scripting, etc. Usability tasks are related to the Graphical User Interface (GUI). These tasks deal with modifications or amendments at the front-end. Support tasks are used by clients to report defects or configuration changes required in their existing applications.

It is not the case that all four types of applications involve all six types of tasks. Database applications, for instance, do not involve usability tasks. Similarly, parallel processing applications do not involve security tasks. Table I shows the relationship between application types and task types.

TABLE I. RELATIONSHIP BETWEEN APPLICATION AND TASK TYPES

Application /Task	Web-Based	Database	Parallel Processing	Telephony
Business Development	✔	✔	✔	✔
New Feature	✔	✔	✔	✔
Performance	✔	✔	✔	✔
Security	✔	✔	X	✔
Support	✔	✔	✔	✔
Usability	✔	X	X	X

Over 700 data points (i.e. tasks) are analyzed to extract an equal number of estimated effort and actual effort pairs. Fig. 1 shows the distribution of these data points for different application types. The composition of the data points for each application type with respect to task types is shown in Fig. 2 through Fig. 5.

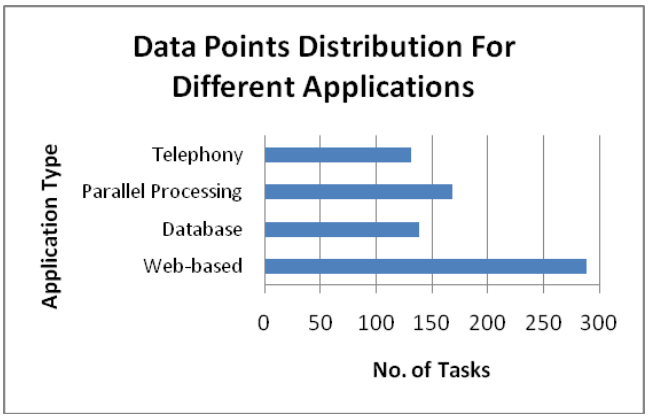

Figure 1. Data points distribution for different applications

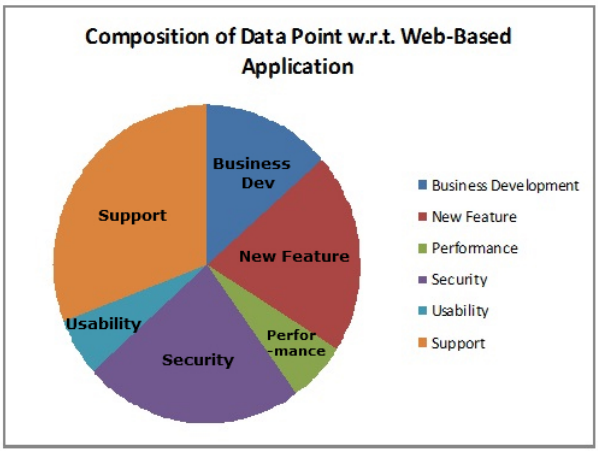

Figure 2. Composition of data points w.r.t. web-based application

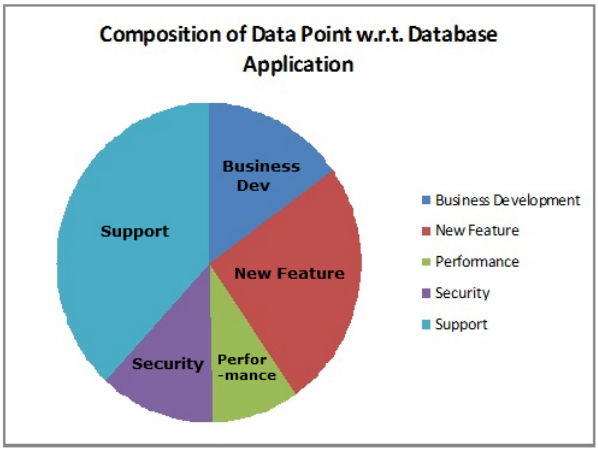

Figure 3. Composition of data points w.r.t. database application

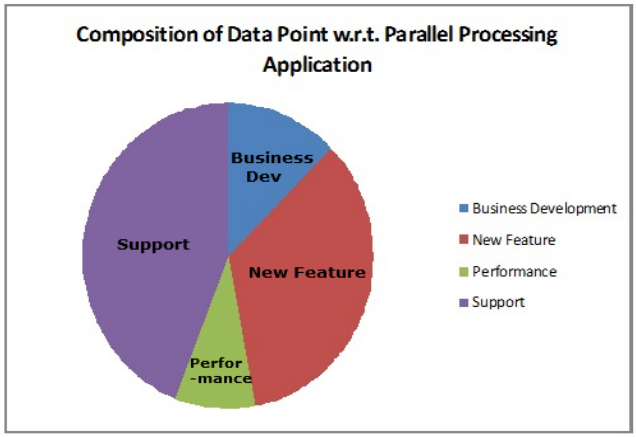

Figure 4. Composition of data points w.r.t. parallel processing application

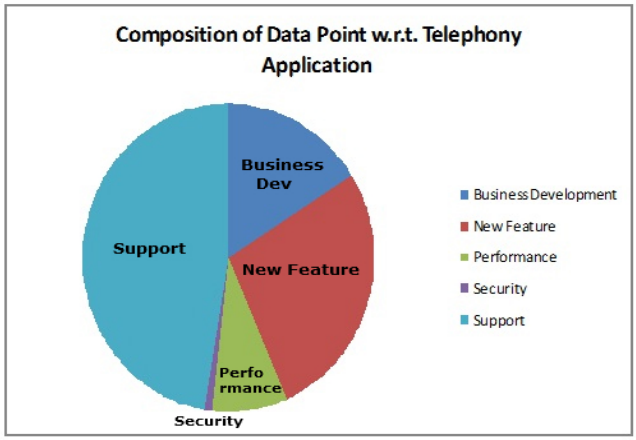

Figure 5. Composition of data points w.r.t. telephony application

III. RESULTS AND DISCUSSION

TABLE II. ESTIMATION ACCURACY FOR WEB-BASED APPLICATIONS

Sr #	Task Type	% Over Estimated (OE)	% Under Estimated (UE)	% Same	OE/UE
1	Business Development	17.39	69.57	13.04	0.25
2	New Features	15.66	71.08	13.25	0.22
3	Performance	9.52	85.71	4.76	0.11
4	Security	3.74	83.18	13.08	0.04
5	Support	6.60	83.96	9.43	0.08
6	Usability	7.69	84.62	7.69	0.09

Table II shows the estimation accuracy results for web-based applications. The last column (OE/UE) clearly indicates that all types of tasks in this type of applications are mostly underestimated. This is usually due to a failure to account for the learning curve associated with rapidly changing web technologies. Another major factor that could cause this underestimation is the frequent failure to consider the effort required to ensure the streamless execution of these web applications on different browsers (e.g. IE, Chrome, Mozilla Firefox etc) and browser versions (e.g. IE V8, V9 etc).

TABLE III. ESTIMATION ACCURACY FOR DATABASES APPLICATIONS

S #	Task Type	% Over Estimated (OE)	% Under Estimated (UE)	% Same	OE/UE
1	Business Development	76.92	15.38	7.69	5.00
2	New Features	76.47	11.76	11.76	6.50
3	Performance	87.50	12.50	0.00	7.00
4	Security	84.62	7.69	7.69	11.00
5	Support	75.86	17.24	6.90	4.40

Table III shows the estimation accuracy results for database applications. The last column (OE/UE) clearly indicates that all types of tasks in this type of applications are mostly overestimated. Investigation revealed that the main reason for this overestimation is the failure to account for the actual experience level of the resources involved in developing this type of applications. The resources actually involved in the development of these applications are very experienced whereas the estimates are based on resources with average experience.

TABLE IV. ESTIMATION ACCURACY FOR PARALLEL PROCESSING APPLICATIONS

S #	Task Type	% Over Estimated (OE)	% Under Estimated (UE)	% Same	OE/UE
1	Business Development	7.14	85.71	7.14	0.08
2	New Features	18.79	74.14	6.90	0.26
3	Performance	11.11	77.78	11.11	0.14
4	Support	20.63	71.43	7.94	0.29

Table IV shows the estimation accuracy results for parallel processing applications. The last column (OE/UE) clearly indicates that all types of tasks in this type of applications are mostly underestimated. The main reason for this underestimation was found to be a failure to perform proper impact analysis. Since this type of applications deals with the core business logic, performing proper impact analysis is extremely important. Failure to perform proper impact analysis at the initial stage leads to rework in the later stages.

TABLE V. ESTIMATION ACCURACY FOR TELEPHONY APPLICATIONS

S #	Task Type	% Over Estimated (OE)	% Under Estimated (UE)	% Same	OE/UE
1	Business Development	75.00	12.50	12.50	6.00
2	New Features	75.86	10.34	13.79	7.33
3	Performance	85.71	14.29	0.00	6.00
4	Security	100.00	0.00	0.00	-
5	Support	86.44	8.47	5.08	10.20

Table V shows the estimation accuracy results for telephony applications. The last column (OE/UE) clearly

indicates that all types of tasks in this type of applications are mostly overestimated. Closer examination of the tasks in this type of applications revealed that this overestimation was a primarily due to a failure to account for the usage of third-party components. These pre-built third-party components for IVR required only configuration.

IV. Related Work

Software Effort Estimation plays a vital role in the software industry to deliver quality product with the desired quality while meeting the delivery deadline.

Several authors have contributed in software estimation and analysis about the factors that lead to incorrect software estimations. Some authors worked on the theme that how accurate are the subjective estimates of task's durations for a project. They have gathered the data to compare the actual results with the estimated ones in terms of duration. When the result is analyzed from the data they have collected, it is concluded that there is over estimation in majority of tasks but the mean error is almost 1 percent which is under estimate. The solution they have suggested is to make Work Breakdown structure when tasks are being estimated [1].

Hughes worked on another methodology instead of work breakdown structure as stated in [1], he used expert judgment in [2] and emphasized it for software estimation. For this purpose a survey was conducted and resulted with a conclusion of modification in the existing system thus on it estimation was predicted.

We also found work being done on comparison of different software estimation techniques. In [3] Boehm, Abts and Chulani compared several software estimation techniques and models like model-based, expertise-based, learning-based, dynamics-based, regression-based to conclude that which model and technique is the most appropriate one to be used for estimations. After analyzing the result they were of the view that there is no technique that can be said as the best for estimation, as every technique have its own advantages. But in [4] Chulani, Boehm and Steece compared only 2 techniques COCOMO 2 and Bayesian and they did the analysis on complete software projects of different software houses and concluded that for multiple regressions Bayesian is better technique for estimation.

Shepperd and Schofield presented result of estimation with a new software algorithmic approach i.e. analogy [5]. The paper pointed out the disadvantages in the approaches i.e. COCOMO and FP (Function Points) and a remedy with an automated technique. The conclusion was that there must be more than one software estimation models to estimate accurately on a project.

Instead of all these efforts some are of the view that it all depends on the size of the project and the flexibility in it. These are the main factors involved in predicting software estimations [6]. Some researches are done to analyze that why errors are generated in software estimations. What are the factors that lead to predict wrong estimations? For this purpose when they gathered data they found that there are multiple factors that impact on wrong software estimation e.g.; analysis, data collected etc.

Jorgensen has a different point of view, according to him the main factor that affects estimation is the person who is responsible for making estimations. His skills, competencies have a major role in prediction. The solution and result of their study is that only Project Manager should not make estimations, there should be collaboration among resources while estimating [7].

In this paper, we have chosen an organization that deals with payment industry and chosen four different projects for prediction accuracy comparison of effort estimation and analyzed the result that how the nature of the project affects the overall estimations. In all projects there are different task's comparison that are also performed on tasks w.r.t. projects.

V. Conclusions and Future Work

Although there are many factors that are included to make correct/ accurate estimations but we have also analyzed that software estimation vary from project to project. We are of the view that only Project Manager cannot predict the accurate project estimation and there should be different roles that lead to project estimation. The analysis of each project plays an important role in software estimation. Four different types of projects are chosen to analyze the factors that are involved in making accurate and incorrect estimations. In web-based projects there is overall under estimation because it involves repeated changes with an urgent notice and thus the estimation predicted is under estimated. For the other projects the estimation trends are different than web project. However there is a huge research to be done that involves all the major areas that can lead software estimation to become more accurate and reliable.

The future work that can be done in this area is to gather several different natured projects and analyze the situation of estimation in them. Different natured tasks can also be worked out to analyze the estimation level. Some work can also be done by targeting different organization's estimated data and perform analysis on it. This can be helpful for all type of organizations not only prepaid card industry.

Acknowledgment

We are thankful to Ms. Reham Tufail for helping us in gathering data for this empirical study.

References

[1] J. Hill, L.C Thomas, D.E Allen, "Experts' estimates of task durations in software development projects," International Journal of Project Management, Vol 18, pp. 13-21, Feb.,2000.

[2] R.T. Hughes,"Expertjudgement as an estimating method," vol 38, issue 2, pp.[67 - 75], 1996.

[3] B. Boehm, C. Abts, S. Chulani,"Software development cost estimation approaches — A survey," vol 10, issue 1-4, pp. [177 - 205], 2000.

[4] S. Chulani, B. Boehm, B. Steece, "Bayesian analysis of empirical software engineering cost models,"Centre for Software Eng; vol 25, issue 4, pp.[573 - 583],Aug., 2002.

[5] M. Shepperd, C.Schofield, B. Kitchenham,"Effort estimation using analogy," ICSE '96 Proceedings of the 18th international conference on Software engineering, pp. [170 - 178], 1996.

[6] H.Otto,K. “Estimating factors affecting project task size in product development-an empirical study,” vol 53, issue 1, pp. 86 – 94, Feb., 2006.

[7] M.Jorgensen, K.Molokken, “Reasons for software effort estimation error: impact of respondent role, information collection approach, and data analysis method,” vol 30, issue 12, pp. 993 – 1007, Dec., 2004.