

ESTIMATING WEB APPLICATION DEVELOPMENT EFFORT USING TEST ESTIMATION

Arti Bajaj¹, Dr. Gurvinder Kaur²

¹Research Scholar, Mewar University, ²Director, GNKITMS

Abstract: *In software development, software estimation defines the effort, size, cost and schedule for a specified software project in a specified environment, using different methods, tools and techniques. Planning and controlling of a project is not possible without a sound and reliable estimate. Various models for the prediction of effort and costs in software projects have been developed, including web applications. There is an increase in growth and demand of Web for completing business processes. The effort and cost estimation approach helps the companies to complete their web development projects within time and budget. In this paper we proposed a metric for cost estimation, to find out the optimized estimation based on various factors. This paper provides a systematic review of previous web estimation studies particularly focusing on estimation, factors and methods of estimation, tools and models of estimation. That's why there is a clear requisite for an estimation model for application development projects.*

Keywords: *Effort estimation, software cost estimation, software cost estimation Techniques*

I. INTRODUCTION

Test Estimation is an organization activity which approximates how long a task would take to complete. Estimating effort for the test is one of the foremost and important tasks in test management. Mainly two things comes around the mind i.e. first is how long will this testing take and second is how much will it cost? Estimate is based on various factors like resources, schedule, efforts and cost. Resources are essential to carry out any project tasks. Resources can be people, equipment, facilities, funding, or anything else capable of definition required for the completion of a project activity. Time is the most valued resource in a project. Every project has a deadline for delivery. A human service defines the knowledge and the experience of the team members. They affect to your estimation. For example, a team, whose associates have low testing skills, will take more time to finish the project than the one which has high testing skills. Cost defines the budget for the project. It means how much money it takes to finish the project.

II. LITERATURE REVIEW

Software project estimation is one of the challenging and important activities in software development. Proper planning and control is not possible without a sound and reliable estimate. The software industry doesn't estimate projects well and doesn't use estimates appropriately, because of this all suffer far more than as a result and we need to

focus some effort on improving the situation. Hareton Leung (2002) elaborates the concept of cost estimation in his paper "Software cost estimation". It is the process of predicting the effort required to develop software system. Many estimation models have been proposed over the last 30 years. He provides a general overview of software cost estimation methods including the recent advances in the field. As a number of these models rely on a software size estimate as input, firstly he provides an overview of common size metrics and then highlights the cost estimation models that have been proposed and used successfully.

In this paper based on the various models he concludes that almost no model can estimate the cost of software with a high degree of accuracy. The reasons are:

- There are a large number of interrelated factors that influence the software development process of a given development team and a large number of project attributes, such as number of user screens, volatility of system requirements and the use of reusable software components.
- The development environment is evolving continuously.
- The lack of measurement that truly reflects the complexity of a software system.

To produce a better estimate, we must improve our understanding of these project attributes and their causal relationships, model the impact of evolving environment, and develop effective ways of measuring software complexity.

GeetikaBatra(2013), in her paper "A Review on Cost and Effort Estimation Approach for Software Development" describes that the development of software in a success manner depends on accurate estimation, as numerous factors are responsible for the overall assessment of any project. Estimation of cost and effort is most complicated and challenging task in software industry. Many estimation models are introduced by the time, that concludes estimation is not a precise science and demanding of new methodologies should be proposed day by day.

III. APPROACHES TO TEST EFFORT ESTIMATION

There are two approaches which are used to estimate the efforts for the task:

3.1 Function point method

In this method, the Test Manager estimates Size, Duration, and Cost for the tasks, Lets discuss one by one:

(1) *Estimate the size for the task:* Based on the test specification, which we discussed earlier, the size of the task depends on functional size of the system under test. It reflects the amount of functionality that is relevant to the

user. The more number of functionality defines that the system is more complex.

No.	Module Name	Applicable Roles	Description	Weight age
1.	Balance Enquiry	Customer Manager	Customer: A customer can have several bank accounts. He/She can view balance of his/ her accounts. Manager: A manager can view the balance of all the customers who come under his administration.	2
2.	Fund Transfer	Customer Manager	Customer: A customer can transfer funds from his account to any destination account. Manager: A manager can transfer money from any source bank account to destination account.	3
3.	Mini Statement	Customer Manager	It will show last 5 transactions of an account. Customer: A customer can view mini-statement of only his own accounts Manager: A manager can view mini-statement of any account	2
4.	Customized Statement	Customer Manager	A modified statement allows you to filter and display transactions in an account based on date and transaction value. Customer: A customer can see modified-statement of only his own accounts Manager: A manager can view Customized -statement of any account	3
5.	Change Password	Manager Customer	Customer: Only a customer can change password of his own account. Manager: A manager also can change password of only his own account. He is not able to change passwords of his customers.	1
6.	New Customer	Manager	Manager: A manager can enhance a new customer. A manager can manage details like address, email, telephone of a customer.	2
7.	New Account	Manager	Currently system provides 2 types of accounts like Saving & Current. A customer can have several saving accounts. Manager: A manager can add a new account for a current customer.	3
8.	Edit Account	Manager	Manager: A manager can enhance account details for an existing account	1
9.	Delete Account	Manager	Manager: A manager can add and delete an account for a customer.	1

10.	Delete Customer	Manager	A customer can be deleted only if he/she has no active accounts. Manager: A manager can delete a customer.	1
11.	Deposit	Manager	Manager: A manager can credit money into any account. Generally done when cash is deposited at a bank branch.	2
12.	Withdrawal	Manager	Manager: A manager can extract money from any account. Mainly done when cash is withdrawn at a bank branch.	2

Functional Point is divided into three groups: (1) Complex (2) Medium & (3) Simple.

Complex: The term complex defines that the system comprises of multiple components that interact with each other.

Medium: It defines that the system is comprised with limited number of components.

Simple: Simple system composed of small components.

Based on the complex of software function the test manager has to give enough weightage to each functional point. For ex.

Group	Weight age
Complex	3
Medium	2
Simple	1

Based on the factors of function point the project is divided into various modules which is calculated as number of function points. For ex. A Bank websites are divided into 12 function points, the complexity of each function point can be calculated as follows:

(2) Estimate duration for the task

After categorizing the complexity of the function points, estimate the duration to test them. It defines how much time needs to finish the task.

Total Effort = Total Function Points * Estimate defined per function point

Where total effort defines the effort to completely test all the functions of the website and Total Function Points defines total modules of the website. Estimate defined per Function Points is the effort to complete one function points. It depends on the productivity of the member who will take in charge this task.

Suppose the project team of bank has estimated defined per Function Points of 5 hours/points. The estimation of total effort to test all the features is defined as:

Mode	Weight age	No. of function points	Total
Complex	3	3	9
Medium	2	5	10
Simple	1	4	4

Total number of function points is: 23

Estimate defined per point = 5 hr/ point

Total Effort (Person hours) = 23 * 5 = 115

(3) Estimate the cost for the task:

Based on this step the cost of the project can be calculated.

Let us suppose, on average the team salary for a project is Rs.1000 per hour. The time required for "Create Test Specs" task is 115 hours. Accordingly, the cost for the task is 1000 *

115= Rs.115000. Now we can calculate budget for other activities in WBS and arrive at overall budget for the project.

3.2 Three point Estimation

It is one of the techniques that can be used to estimate a task. The ease of the Three-point estimation makes it a very useful tool for a Project Manager that who wants to estimate. In three-point estimation, three values are formed initially for every task based on prior experience or best-guesses as follows:

- The Best case estimate or Optimal state (B)
- The Most likely Estimate or Most likely state (M)
- The Worst case Estimate or Worst case (W)

While estimating a task, the Test Manager wishes to provide three values, as specified above. The three values identified, estimate what happens in an optimal state, what is the most likely, or what is the worst case scenario.

Using an example, elaborate the use of above three values:

The total effort as per the function point methods is 115 person hours. Based on that calculation the estimate for three cases are as follows:

- The best case to complete this task is 80 person-hours (around 10 days). In this case, a talented team, they can finish the task in smallest time.
- The most likely case to complete this task is 115 person-hours (around 15 days). This is a normal case, A team have the enough resource and ability to complete the task
- The worst case to complete this task is 160 man-hours (around 20 days). The team needs to perform much more work because the team contains non-experienced persons.

The assigned value to each parameter is as below:

B=80 M=115 W= 160

The double-triangular distribution formula to calculate the effort is as follows-

$$E = (B+4*M+W)/6$$

$$E = (80+ 4* 115 +160) /6$$

$$E = 116.66$$

Parameter E is known as Weighted Average. It defines the estimation of the task "Create the test specification".

IV. FACTORS AFFECTING SOFTWARE TEST ESTIMATION

There are the following factors for test estimation:

1. Buffer Time

The estimation should include some buffer. But it is not realistic. To have a buffer in the estimation enables to cope with any delays that may occur. A buffer can helps to ensure maximum test coverage.

2. Bug Cycle

The test estimation also includes the bug cycle. The real test cycle may take more days than estimated. To escape this, consider the fact that test cycle depends on the stability of the build. If the build is not constant, then developers may need more time to fix and obviously, the testing cycle gets extended automatically.

3. Availability of Resources for Estimated Period

The test estimation should study all the leaves planned by the team members in the next few weeks or next few months. This will confirm that the estimations are realistic. The estimation should consider some secure number of resources for a test cycle. If the number of resources reduces then the estimation should be re-visited and restructured accordingly.

4. Parallel Testing

Parallel testing is a testing of multiple applications of one application concurrently to reduce the test time. Based on the previous versions of same product anyone can compare the output. This can make the testing task bit easier.

5. Initial stages of estimation

In early stages, team members re-visit the test estimations and make modification if needed. They cannot extend the estimation once freeze it unless there are major changes in requirement.

6. Time estimates

Experiences from past projects play an important role while preparing the time estimates. They can try to avoid all the difficulties or issues which they faced in past projects. Anyone can analyze how the previous estimates were and how much they helped to deliver the product on time.

7. Scope of Project

Team members must know about the end objective of the project and list of all final deliverables. Factors to be measured for small and large projects differ a lot. The Large project ordinarily includes setting up a test bed, generating test data, test scripts etc. Hereafter the estimations should be based on all these factors. While in small projects, usually the test cycle include test cases writing, execution and regression.

8. Load Testing

If anyone needs to put considerable time on performance testing then estimate accordingly. Estimations for projects, which include load testing, should be considered differently.

9. Team work

Based on the strengths and weaknesses of individuals working in a team, anyone can estimate testing tasks more precisely. However estimating one should consider the fact that all resources may not yield same productivity level. Some people can execute faster compared to others. However this is not a major factor, it adds up to the total delay in deliverables.

V. PROPOSED COST ESTIMATION METRIC (CEM)

The aspects affecting the cost of web applications have been categorized into following categories:

1. Technology Platform
2. Idea
3. User Interface
4. Localization
5. Features & Functionalities
6. System Administration
7. Developer Cost
8. Consultancy Cost

S.No.	Category	Aspects
1	Technology Platform	Is it robust in nature? Is it impacts on the budget in positive manner?
2	Idea	Is the idea contains all features and functionalities? Is it integrates with third party? It is increase the process & reduces the time? Is it automating few processes in particular function?
3	User Interface	Is the interface contains default standards like color, look and behavior? Is it convenient to add little change in labels, buttons and menus at a little cost?
4	Localization	How many locations you are targeting? How many information may vary based? How many languages support?
5	Features & Functionalities	What kind of features it has? Is the user prioritizing the features? Will it be helpful to eliminate the secondary features to reduce the budget?
6	System Administration	Is it incorporates hardware and software costs including maintenance? Is there consideration for Travel and training costs? Is it includes costs of networking and communications?
7	Developer Cost	Is the user Eliciting the requirements? How much effort is required to complete each activity? How much calendar time is needed to complete each activity? What is the total cost of each activity?
8	Consultancy Cost	Is it includes the cost of brainstorming? Is it defines the User Story Mapping? Is it precise description of the application?

The proposed metric is:

$$C_{et} = s/a \text{ (C being cost and et being estimation)}$$

Where

a = sum of total amounts of above aspects and

s = sum of positive aspects

If C_{et} is closer to 0.5, it indicates that cost estimation is

optimized. If C_{et} is low then it is not optimized and if C_{et} value is high then it is effectively optimized.

VI. CONCLUSION

As we proposed the estimation metric this contains eight factors did not present the analysis on a tool such as test link. This analysis represents the scope of our future work. In addition, we also plan to apply effort prediction to the Web applications and to compare the performance of several prediction techniques against human estimation. There's a serious need for sufficient, early stage effort prediction for Web development. As the use of the Web as a resource delivery environment increases, effort estimation can add expressively to the reduction of costs and time involved in developing Web applications.

REFERENCES

- [1] B.Boehm, R. Fairley, Software estimation perspectives, IEEE Software 17(6) (2000) 22-26.
- [2] E.J.Barry, T.Mukhopadhyay, S.A. Slaughter, Software project duration and effort: an empirical study, Information Technology and Management 3 (1-2)(2002) 113-136.
- [3] Heemstra, F.J., Software Cost Estimation. Information and Software Technology, 1992. 34(10): p. 627-639.
- [4] Jorgensen, M. and D.I.K. Sjoberg, The impact of customer expectation on software development effort estimates. To appear in Journal of Project Management, 2004. 22(4).
- [5] Kumari, Sweta and Pushkar, Shashank. Performance Analysis of the Software Cost Estimation Methods: A Review. International Journal of Advanced Research in Computer Science and Software Engineering,3(7), July - 2013, p. 229-238.
- [6] Kishore, Swapna and Naik, Rajesh (5 June 2001). Software Requirements and Estimation: McGraw Hill Education (India) Private Limited.
- [7] Litoriya, Ratnesh and Kothari, Abhay. An Efficient Approach for Agile Web Based Project Estimation: Agile MOW. Journal of Software Engineering and Applications, 2013, 6, 297-303.
- [8] Niazi, Adnan and Dai, Jian S. Product Cost Estimation: Technique Classification and Methodology Review, Journal of Manufacturing Science and Engineering MAY 2006.
- [9] Molokken Kietil and Jorgensen Maqne, "A Review of Surveys on Software Effort Estimation", 2003 IEEE.
- [10] Douqlis, Charles, Cost Benefit Discussion for knowledge-Based Estimation Tools, 1998.
- [11] International Function Point Users Group 2004. Function point counting practices manual, release 4.2.1.
- [12] Mendes, E., and Counsell, S., "Web Metrics Estimating Design and Authoring Effort", Proc. IFPUG 2001 Conference.