

Code Quotation Generator

Kashyap Kurani¹, Dhvani Patel², Sameep Patani³ and Deepti Pawar⁴

¹⁻⁴ Shah and Anchor Kutchhi Engineering College, Mumbai, India

Email: kashyap.kurani_19@sakec.ac.in, {dhvani.patel_19, sameep.patani_19, deepti.pawar}@sakec.ac.in

Abstract— Project managers are assisted by software cost models and effort approximations in allocating resources, managing schedules and budgets, and creating cutting-edge procedures that enable projects to be finished on time and within budget. If expenses and labor are shown to be questionable in software projects, appropriate times may be overlooked, while expectant predictions may result in some resource loss. These concerns are crucial in the context of online development and extremely difficult given that web projects have tight deadlines and variable opportunity. Early projects may not always cover all components of a new project when used as a foundation for cost assessment because software projects are constantly changing in nature. Regression analysis or mathematical sources are used to build the preliminary software estimation models. The purpose of this study is to find a proper platform where one can find all the software prices and be able to give an estimated cost for the same.

Index Terms— Cost estimation, Software development, Effort estimation, COCOMO model, Cost analysis, Financial planning.

I. INTRODUCTION

Software cost estimation pertains to cost budget preparation, manpower demand, human employment, and process setup, which is a crucial preliminary task to increase software development efficiency and assist the software project implementation by low price sensibly. Traditional cost assessment involves more physical labor, however there are many issues with it: it can be difficult to get data for software project measurement; measurement guidelines and standards may not be consistent; it may be challenging to guarantee the accuracy and timeliness of measurement data; and communication and coordination between project team members may not be effective. However, the decision support system is employed to support semi-structured and unstructured decision-making issues and practically provides executives with the necessary basis for decision-making. Consequently, this article will go over cost estimation techniques and make decisions. Support application model based on software development cost estimation moving ahead, and in accordance with various cost estimation techniques, will design a more than one person estimating decision support system to assist managers performing software development cost estimation.

A. Cost And Effort Estimation Model:

Exploration on effort and cost estimation of software development has been plentiful and expanded since the end of the Seventies. This area is still very much buzzing, as shown by the numerous works existing in the literature. In this area, experts have extensively investigated the topic, in relation to both approximation approach and investigation approach. These studies were carried out in both industrial and academic contexts. A number of cost estimation methods exist and they can be classified into three main categories. These categories are:

Expert Judgment: In expert judgment, a software project estimates the cost and effort established on chronological data and related projects to estimate software. This technique is very subjective and it absences adjustments and thus, cannot be returnable.

Algorithmic Models: This model is a widespread category in the effort and cost estimation technique. These prototypes include FPSE, LM, MM, COCOMO, SLIM, SEER-SEM, SLOC and PM .

Machine Learning: Newly, these techniques are being used in aggregation or as replacements to algorithmic models. These methods include neuro-fuzzy, genetic algorithms, neural networks, fuzzy logic and regression trees.

II. RELATED WORK

M. Ruhe et al. and authors argue that accurate cost estimation is crucial for effective project planning and management, and that traditional methods of cost estimation are often inaccurate and overly simplistic [1]. The Author proposes a method based on a combination of the COCOMO model and the Function Point Analysis (FPA) technique. COCOMO is a software cost estimation model that takes into account the size, complexity, and development environment of a project. FPA is a technique for measuring the functionality provided by a software application, which is then used to estimate the effort required for development and maintenance. The paper also outlines the steps involved in the proposed method, including the identification of functional requirements, the estimation of complexity and development environment factors, and the calculation of effort required for development and maintenance.

The author Jun Li et al presents the development and evaluation of a decision support system (DSS) for software project cost estimation. The DSS is designed to assist project managers and developers in estimating the cost of software projects by providing them with accurate and reliable estimates [2]. It describes the development process of the DSS, which is based on the cost estimation models and algorithms used in the COCOMO (Constructive Cost Model) approach. The DSS incorporates a user-friendly interface that allows users to input project-specific information, such as project size, complexity, and team experience. The system then uses this information to generate estimates for project cost, effort, and duration. The author presented the results of the evaluation of the DSS, which was conducted using data from real-world software projects. The evaluation showed that the DSS provides accurate and reliable estimates.

Sayed Mohsin et al. presents a case study in which the proposed approach is applied to estimate the effort and cost of developing a real-world web application [3]. The results of the case study show that the proposed approach is effective in providing accurate estimates, and it outperforms traditional estimation approaches. Author also discusses the advantages of using the FMT approach, such as its flexibility, simplicity, and adaptability to different development environments. The authors suggest that the FMT approach can be used as a reliable and practical tool for web application estimation in various contexts.

III. MOTIVATION

To provide a more accurate and efficient way for companies to generate quotes for their clients. By predicting the price of a project accurately, companies can better manage their resources and ensure that they are not underquoting or overquoting for their services. This can also help improve customer satisfaction and reduce the risk of disputes or renegotiations over pricing. Additionally, accurate price prediction can also help companies stay competitive in the market and make more informed business decisions.

IV. PROBLEM STATEMENT

In the modern era, software quality has emerged as a key economic advantage. Appropriate project management and software development methods are necessary for creating high-quality software applications. One of the most important first stages in any new software project for project managers is cost estimation. To be able to submit competing bids and continue to succeed in the market, accurate cost estimates are a crucial component. Overestimations and underestimations of anticipated expenses have a big impact on a company's reputation and competitiveness. Even though research on software cost estimation techniques began in the early 1960s, the industry still reports difficulties in arriving at accurate cost estimates. Creating precise cost estimation models has been the subject of extensive study over the past few decades. However, surveys show that only a small number of businesses actively use them. There are several pragmatic justifications against using cost estimation methods. Software and price prediction have become increasingly important topics in today's fast-paced digital world. With the rise of big data, machine learning, and artificial intelligence, predicting prices and trends has

become more accurate and efficient than ever before. However, despite the advances in technology, there are still challenges to overcome when it comes to predicting software prices accurately. One challenge is the constantly changing market conditions and the rapid evolution of technology. With new software solutions emerging all the time, it can be difficult to predict which products will be successful and which will fail. Moreover, the pricing strategy of a particular software product can be influenced by various factors such as market competition, customer preferences, and marketing initiatives, among others.

V. PROPOSED APPROACH

Website design costs can vary depending on the type, size, and complexity of your site. Our website design cost calculator simplifies the process of determining costs, so we can budget for our website design project.

The calculator guides us through every step of the budgeting process. It's pre-populated with common site design elements, so all one has to do is determine and enter each cost that applies to your project. Since all of these elements are broken down, the calculator can help to prevent one from forgetting important elements.

Once we enter the costs, the calculator does all of the math, reducing the chance for human errors. It helps to ensure that the project budget is accurate, so one can plan out its finances and make sure that they have all of the money necessary to complete the project.

The calculator can also be used to see how changing elements of a site will affect the project cost. One might experiment with reducing or increasing the number of website pages, or with adding in extra design elements. The cost estimator can help to better understand everything that goes into website design, so that we are well-prepared for everything that the project will be involved with.

VI. FLOWCHART

The COCOMO software estimating model is among the most widely utilised in the world. Based on the size of the programme, COCOMO forecasts the time and effort required to produce a software product. As we can see in Fig.1. the steps required by this paradigm are:

Step 1: Estimate Size

The first step in using the COCOMO 2 model is to estimate the size of the software project in terms of lines of code or function points. This is done by analyzing the requirements of the software project and determining the functionality that needs to be developed. Once the size is estimated, it is used to calculate the effort and schedule required for the project.

Step 2: Rate Scale Factors and Effort Multipliers

The next step is to rate the scale factors and effort multipliers that are applicable to the software project. The scale factors account for various project attributes such as product complexity, development experience, and team cohesion, while the effort multipliers account for environmental factors such as development tools, documentation requirements, and language and platform experience. The rating of these factors is done on a scale of 0.5 to 1.5, with 1.0 being the default value. Choose a group of 15 multiplying factors from the project's numerous characteristics.

Step 3: Estimate Effort and Schedule

Using the size estimate and the ratings of scale factors and effort multipliers, the COCOMO 2 model calculates the effort and schedule required to complete the software project. The effort is calculated in person-months, and the schedule is calculated in months. This step involves using the COCOMO 2 formula to calculate the effort and schedule based on the size and ratings. Multiply the initial estimate by all the multiplying factors, or the values from steps 1 and 2, to determine the effort estimate.

Step 4: Need for Further Adjustments

In some cases, the effort and schedule estimates obtained from the COCOMO 2 model may need to be adjusted further based on project-specific factors that are not accounted for by the scale factors and effort multipliers. These factors may include team turnover rates, requirements volatility, and technology maturity. This step involves reviewing the estimates obtained from the model and making any necessary adjustments.

Step 5: Allocate Effort to Schedule

The final step in using the COCOMO 2 model is to allocate the effort to the schedule. This involves breaking down the effort estimate into specific activities and assigning them to specific time periods in the schedule. This

step helps to ensure that the project is completed on time and within budget by providing a detailed plan for the development process.

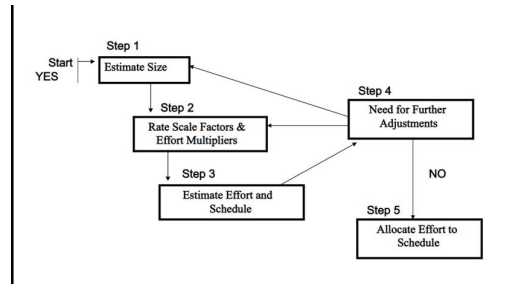


Figure 1. COCOMO II Estimate Development Scenario

In COCOMO, projects are categorized into three types: Organic, SemiDetached and Embedded.

A. Organic:

A development project can be classified as being of the organic kind if it involves creating an application programme that is widely understood, the project team is reasonably small, and the team members have prior experience creating initiatives that use similar methodologies. Simple business systems, simple inventory management systems, and data processing technologies are a few examples of this type of project.

B. Semidetached:

A project that involves both experienced and inexperienced workers might be considered as a semidetached kind. Despite having limited expertise in related systems, team members might not be familiar with all facets of the order being produced. One example of a semi-detached system is the creation of a new operating system (OS), database management system (DBMS), and intricate inventory management system.

C. Embedded:

A project is considered to be of the embedded type if the software being created is tightly tied to sophisticated hardware or if there are strict rules about the operational manner. ATM and air traffic control.

VII. EVALUATION RESULT

The following cost is estimated for different types of projects/websites using our “Code Quotation Generator” while taking into consideration Types and purpose of project/website:

A. Factors By Types And Purpose :-

a. Personal Website and Blog:-

The cost of personal websites and blogs will be under 30k. The price will depend on the number of pages and features you want to add. This includes the domain name and hosting fees for a year.

b. Brochure (Landing web page):-

The cost will be determined by the number of pages, the complexity of the design, and whether you want us to write the content for you. A brochure usually has about 10-12 pages and can cost anywhere from \$200 to \$500 depending on what extras you need.

B. Cost Predictions Of Various Different Projects Estimated:

a. Corporate:-

A typical corporate website consists of more than 50 web pages and has a lot of other things to cover. It usually includes products/services, a homepage, an about us section, and company blog contact information. For corporate websites, the hosting provider, database, and other maintenance systems must be even more sophisticated. The price of web development for these sites runs between Rs.15,000 and 20k.

b. Educative:-

The price of an educational website will be dependent on the size of your company. The more complex your business is, the more options there are for you to choose from and this will affect the price. Some companies will

charge more for a website that can handle large amounts of traffic and more users, while others may have different rates based on the package you choose.

c. Web Portal:-

The price of any web portal will go up to \$10,000 a year, which is far less than the cost of a full-time employee. \$1,000 or more depending on the complexity of the project. An experienced developer will be able to create a simple web portal with an attractive design in just a few days.

VIII. CONCLUSION

This paper appraised the frame of research on effort and cost estimation models for web applications by scrutinizing procedures, datasets, and research types. Despite many revisions, there is no strong suggestion that there is a certified method or a set of verified approaches for estimating the effort and cost of web applications. All of the performances used are tailored forms of systems taken from traditional software engineering, and there is discussion about what scope drivers should be used to originate approximations. Additionally, new size metrics are being technologically advanced and tailored from current approaches Object Points, Web Objects, Data Web Points, etc. Other frequently used size drivers in traditional software development such as Lines of Code (LOC) are hardly used in Web Application Effort and Cost Estimation Models.

The proposed case study displays the significance of the output result and the proposals comprising the premise of linear algebra. Each proposal has a principle and the proposed approach proposes details of the Association rule.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to all those who have supported and contributed to the successful completion of my project on "Code Quotation Estimator." First and foremost, I would like to thank my project supervisor Deepti Pawar for her valuable guidance, support, and encouragement throughout the project.

I would also like to extend my gratitude to all the Teachers from SHAH & ANCHOR KUTCHHI ENGINEERING COLLEGE who generously shared their insights, knowledge, and expertise with me, which were invaluable in developing this project. Additionally, I am grateful to my friends and family for their unwavering support and understanding throughout the duration of the project.

REFERENCES

- [1] M. Ruhe, R. Jeffery and I. Wiecek, "Cost estimation for web applications," 25th International Conference on Software Engineering, 2003. Proceedings., Portland, OR, USA, 2003, pp. 285-294, doi: 10.1109/ICSE.2003.1201208.
- [2] J. Li, J. Lin, Y. Jin and Q. Chen, "Development of the Decision Support System for Software Project Cost Estimation," 2008 International Symposium on Information Science and Engineering, Shanghai, China, 2008, pp. 299-302, doi: 10.1109/ISISE.2008.270.
- [3] S. M. Reza, M. M. Rahman, M. H. Parvez, M. S. Kaiser and S. Al Mamun, "Innovative approach in web application effort & cost estimation using functional measurement type," 2015 International Conference on Electrical Engineering and Information Communication Technology (ICEEICT), Savar, Bangladesh, 2015, pp. 1-7, doi: 10.1109/ICEEICT.2015.7307462.
- [4] L.C. Briand, K. El Emam, F. Bomarius, COBRA: A Hybrid Method for Software Cost Estimation, Benchmarking, and Risk Assessment. 8 &(. April 1998, pp. 390-399.
- [5] S.D. Conte, H.E. Dunsmore, V. Y. Shen. &&. The Benjamin/Cummings Publishing Company, Inc., 1986.
- [6] D. Reifer. Web-Development: Estimating Quick-Time-toMarket Software, vol. 17, no. 8, November/December 2000, pp. 57-64.
- [7] D. Galorath and R. Evans, Software Estimation and Risk Management, Auerbach, 2006.
- [8] D. Garmus and D. Herron, Function Point Analysis, Addison Wesley, 2001.
- [9] J. Johnson, My Life Is Failure, Standish Group Intl., 2016.
- [10] C. Jones, Estimating Software Costs, McGraw Hill, 1998.
- [11] M. Jorgensen, A review of studies on expert estimation of software development effort, J. Systems and Software 70 (2004) 37-60.
- [12] S. Mc Connell, Software Estimation, Microsoft Press, 2006.
- [13] J. Poulin, Measuring Software Reuse, Addison Wesley, 1996
- [14] L. Putnam and W. Myers, Measuring for Excellence, Yourdon Press, 1992.
- [15] R. Stutzke, Estimating Software-Intensive Systems, Addison Wesley, 2005.
- [16] A. Trendowicz and R. Jeffery, Software Project Effort Estimation, Springer, 2014.

- [17] True Planning User Manual, Price Systems, 2002-2016.
- [18] R. Valerdi, The Constructive Systems Engineering Cost Model (COSYSMO), VDM Verlag, 2008.