MODEL TO DETERMINE COST ESTIMATION FOR SOFTWARE DEVELOPMENT PROJECTS OF SMALL AND MEDIUM SCALES USING USE CASE POINTS

by Teguh Sutanto

Submission date: 29-Sep-2022 10:57AM (UTC+0700)
Submission ID: 1911825664
File name: Copy_of_A_MODEL_TO_DETERMINE_COST_ESTIMATION_FOR.pdf (254.33K)
Word count: 4645
Character count: 23582

Journal of Theoretical and Applied Information Technology
10th March 2016. Vol.85. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A MODEL TO DETERMINE COST ESTIMATION FOR SOFTWARE DEVELOPMENT PROJECTS OF SMALL AND MEDIUM SCALES USING USE CASE POINTS

[1]SHOLIQ, [2]ARIFIN PUJI WIDODO, [3]TEGUH SUTANTO, [4]APOL PRIBADI SUBRIADI
[1 & 4)]Department of Information Systems, Sepuluh Nopember Institute of Technology (ITS), Indonesia
E-mail: sholiq@is.its.ac.id; sholiq99@yahoo.com

[2 & 3)]Department of Information System, STMIK STIKOM Surabaya, Indonesia

ABSTRACT

The purpose of this study was to develop a cost estimation model for software development projects of small and medium scale. The model was derived from the Use Case Points, which is usually used to estimate software development effort. The development of our cost estimation model was based on the need for a reference to the estimated costs for software development projects, particularly software development projects of small–medium scale. The cost estimation was used to estimate the allocation of resources spent on covering personnel resources, money, and time to complete the project. The result of this study was a cost estimation model for software development projects of small–medium scale that had been tested with four software projects that had been completed. Testing of the model was performed using data from four projects got level deviations between the estimated cost and the actual cost amounted to 6.89%.
Keywords: *Use Case Points, Software Development Project, Cost Estimation Model, Effort Estimation*

1. INTRODUCTION

The use of information technology (IT) in Indonesia has been getting higher, Business Monitor International (BMI) predicts that spending information technology (IT) in Indonesia in 2014 amounted to IDR 69.7 trillion up by 12.3% compared to the year 2013 [1]. BMI believed that the factors driving the growth of the IT market in Indonesia, one of which was due to the increase in the amount of use and computer ownership. Moreover, the high number of IT spending due to increased awareness of the use of IT in government, business, and general consumers to support the performance. Also the present of a variety of multimedia devices encourage higher information technology spending. The software spending in 2014 was estimated to approximately IDR 9.2 trillion [1].

In Indonesia, the procurement of goods and services to government agencies stipulated in the presidential decree number 54 of 2010, then refined by a presidential decree number 70 of 2012. In the presidential decree, the term "goods" has been defined as objects either tangible or intangible and movable or immovable, and those that can be traded, worn, used, or exploited by the users [2] [3]. Considering this definition, software has been categorized as goods, so that the procurement of software for government agencies is treated equally with the procurement of other goods in general. According to the presidential decree number 70 of 2012, the procurement of goods and services is done by the Committing Officer (CO) [3].

The procurement of goods and services by CO begins with planning the procurement, which includes the following: 1) define the technical specifications of goods or services, 2) set the owner estimate cost (OEC), and 3) create a draft contract. For the procurement of customized software or software that is not sold in the market in general, determining OEC is the most difficult activity. Unfortunately, not available to the market price for customized software; also, there is no standard method that can be used for reference.

According to a report by Standish Group Study (CHAOS) in 2012, until 2013, only 39% of IT projects were successful [4]. Failure of majority of the software development projects was due to a lack of proper planning. In other words, the causes of the failure were the lack of

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

accuracy of the estimates [5]. The level of uncertainty regarding the estimation of effort and lack of data on the value of the distribution of effort at every stage of a software development project can make it difficult for the project manager to plan staffing and other resources [5].

Furthermore, to obtain the cost of a customized software development project, we need to first know how much effort is required for the development of the project. One of the effort estimation methods that is widely used is the Use Case Points (UCP); this method estimates the amount of effort based on the complexity of the use case [6].

Some previous studies on UCP have reported the following results: (1) comparison between the effort estimated using UCP and the actual effort has a degree of deviation of 19%, while estimates by experts have a degree of deviation of 20% [7], and (2) two other studies reported that UCP had a degree of deviation of 6% [8] and 9% [9] compared with the actual effort. Thus, it seems that UCP can be reliably used for effort estimation. The focus of this study is to determine how UCP can be applied to obtain OEC or estimated cost for software development projects.

2. LITERATURE REVIEW

2.1. Related Work

Today, the UCP method is popular as an effort estimation technique (effort required) for software development. As reported by several studies, UCP provides fairly good estimates of the level of accuracy. In few studies, it was found that UCP promises reliability for estimating software development, for example: (1) estimates made using UCP were reportedly better than those made by experts [7], (2) estimations made using UCP had a degree of deviation of 9% in one study [9], and (3) estimations made using UCP had a degree of deviation of 6% in another study [8].

Several other studies have been reported on a drastic modification of UCP in comparison with the original UCP: (a) Frohnhoff and Engels applied UCP to project larger software, which was special software by provide a detailed description for charging the T and M factors. The T factor represents the technical factors of the system that is being developed, whereas the M factor is related to the competence of the team developing the system. The estimations made using the modified UCP, as proposed by Frohnhoff and Engels, resulted in better accuracy with a degree of deviation of 20%. It would have been better if this was compared with the original UCP, having a degree of deviation of 42% [10]. (b) UCP maintain (UCPm), proposed by Sergey Diev, has been used to estimate the effort at a software development where it was part of software development projects more large [11]. (c) The UCP method was simplified by Ochodek et al; as per this, step 1 involves calculating unadjusted actor weight (UAW). Test data was used by 14 software development projects. Results of this study stated that the UAW calculation does not significantly influence the final result or the UAW estimation can be eliminated, so the effort estimation procedure using UCP becomes much simpler [12].

So far most of the estimates made using UCP limited effort to obtain effort of a software development project. Most researchers have not yet reached to estimating the costs involved. Nevertheless, in two studies, UCP has been used to estimate the costs involved in the development of software projects of large and medium scales [13] and enterprise resource planning software [14]. In these studies, the activities of software development were categorized into three main activity phases namely: software development, ongoing activity, and testing and quality assurance phases. Also, the distribution of effort per activity (and per subactivity) was presented along with its percentage, but how the percentage of effort per activity was obtained was not presented in detail.

2.2. Use Case Points

UCP was first proposed by Gustav Karner in 1993 and was developed from Function Point Analysis for object-oriented applications [6] [15]. The calculation process of UCP requires use case diagrams and its descriptions. The use case description contains the steps on how a use case is executed. These steps are known as the term "transaction" of use case. UCP calculation involves seven steps:

Step 1: Calculating UAW. Actors of each use case are categorized into simple, medium, or complex. Each actor is set into groups; the criteria for this are presented in Table 1.

Journal of Theoretical and Applied Information Technology
10th March 2016. Vol.85. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 1. Classification of Actor

Actor Category	Description	Actor Weight
Simple	If actor interacts application through Application Programming Interface (API), as Command Prompt.	1
Medium	If actor interacts application through protocol, as TCP/IP.	2
Complex	If actor interacts through Graphic User Interface (GUI) or Web Page.	3

UAW is obtained as the sum of the weights of each actor (formula 1).

$$UAW = \sum_{i=1}^{n} ActorWeight_i \quad (1)$$

where, n = number of actor and ActorWeight = weight of each actor category (see Table 1)

Step 2: Calculating Unadjusted Use Case Weight (UUCW). UUCW expresses the use case complexity that is measured by the number of transaction in a use case. Each use case in the system is categorized into simple, medium, or complex; the criteria for this are presented in *Table 2*. UUCW is obtained as sum of the weights of each use case (formula 2).

Table 2. Classification of Use Case

Use case category	Description	Use Case Weight
Simple	A use case has 3 or less transactions including alternative transactions.	5
Medium	A use case has 3 to 7 transactions including alternative transactions.	10
Complex	A use case has more than 7 transactions including alternative transactions.	15

$$UUCW = \sum_{i=1}^{n} UseCaseWeight_i \quad (2)$$

where, n = number of use case and UseCaseWeight = weight of each use case category (see Table 2)

Step 3: Calculating Unadjusted Use Case Point (UUCP). UUCP is obtained as the sum of UUCW and UAW (formula 3).

$$UUCP = UUCW + UAW \quad (3)$$

Step 4: Calculating Technical Complexity Factor (TCF). TCF is used to estimate the software size in order to consider the technical considerations of the system. It is determined by assigning a score of between 0 (non-relevant factor) to 5 (important factor) for each of the 13 technical factors listed in Table 3.

Table 3. Technical Factor Weight

Ti	Technical Factor	Weight
T1	Required Distributed Systems	2
T2	Response Time Is Important	1
T3	End User Efficiency	1
T4	Required Complex Internal Processing	1
T5	Reusable code to Focus	1
T6	Installation Easy	0.5
T7	Usability	0.5
T8	Cross-Platform Support	2
T9	Easy To Change	1
T10	Highly Concurrent	1
T11	Custom Security	1
T12	Dependence On Third-Part Code	1
T13	User Training	1

This score is multiplied by the weighted value assigned to each factor. TF is obtained as the sum of multiplying score and weight (formula 4).

$$TF = \sum_{1}^{13} Score_i * Weight_i \quad (4)$$

TF is used to obtain the value of TCF (formula 5):

$$TCF = 0.6 + (0.01 * TF) \quad (5)$$

Step 5: Calculating Environmental Complexity Factor (ECF). ECF is another factor that is applied to estimate the software size by taking into account the environmental considerations of the system. It is determined by assigning a score of between 0 (no experience) to (expert) for each of the 8 environmental factors listed in Table 4.

Journal of Theoretical and Applied Information Technology
10th March 2016. Vol.85. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 4. Environmental Factor Weight

Ei	Environmental Factor	Weight
E1	Familiarity with the Project	1.5
E2	Application Experience	0.5
E3	OO Programming Experience	1
E4	Lead Analyst Capability	0.5
E5	Motivation	1
E6	Stable Requirements	2
E7	Part Time Staff	-1
E8	Difficulty Programming Language	-1

Similar to the calculation of TCF, this score is multiplied by a weighted value of each factor. This score is multiplied by the weighted value assigned to each factor. EF is obtained as the sum of multiplying score and weight (formula 6). Furthermore, the value of EF is used to obtain ECF. The formula for the calculation of ECF is given below (formula 7).

$$EF = \sum_{1}^{8} Score_i \,.\, Weight_i \quad (6)$$

$$ECF = 1.4 + (-0.03 * EF) \quad (7)$$

Step 6: Calculating UCP. UCP is obtained by multiplying UUCP, TCF, and ECF (formula 8).

$$UCP = UUCP * TCF * ECF \quad (8)$$

Step 7: The final step in the UCP method involves calculating the effort. The value of effort is obtained by multiplying the value of UCP and the constant ER in staff hours/UCP (formula 9). Researchers typically use a value of ER that is equal to 20 staff hours/UCP, as proposed by Karner [6]. The value of ER can also be 8.2 for small- and medium-scale business applications [16] or 4.4 for the development of websites using a template or component [17].

$$Effort = UCP * ER \quad (9)$$

3. RESEARCH METHODS

There are five steps involved:

1. Identify the problem and propose a model.
2. Collected data from several projects have been completed.
3. Calculate effort.
4. Determine the distribution of effort.
5. Determine the rate per activity.
6. Testing and validation of models.

3.1. Identify the problem and propose a model

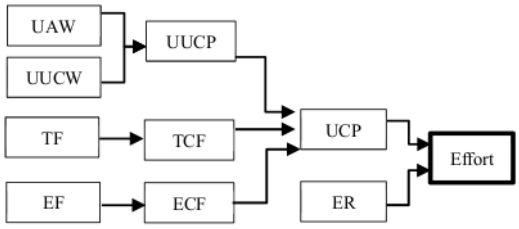

Figure 1. Summary of the procedure to determine the effort in software development using UCP

The procedure for determining the effort has been described in Section 2.2. It can be summarized in a chart form as presented in Figure 1. Thus, as described in a previous section, a model to determine the cost in these study will derive from the procedure for determining the effort (in man-hours) using UCP.

Some abbreviations used in Figure 1 are as follows: UAW = Unadjusted Actor Weight, UUCW =Unadjusted Use Case Weight, UUCP = Unadjusted Use Case Point, TF = Technical Factor, TCF=Technical Complexity Factor, EF= Environmental Factor, ECF=Environmental Complexity Factor, UCP=Use Case Points, and ER=Effort Rate. The effort is the labor and time required to complete software projects (in man-hours).

As we know that in software development, the cost is more influenced by the professional workforce. Whereas the procedure UCP (Figure 1) can be used to obtain the effort in man-hours, so that it can be developed to obtain estimates of the cost of software development by extending some components such as the distribution of effort per activity, pay rates per activity, and other components, which will be discussed in section 4.1.

3.2. Collect data for software projects earlier.

Data was collected through interviews and questionnaires to the 4 projects of small–medium software. Interviews were conducted with the project manager and the team of software developers. The size of the software project was estimated based on several parameters, namely the number of programmers, settlement time duration, and number of lines of code [18]. Category of the project size can be seen in Table 5.

Journal of Theoretical and Applied Information Technology
10th March 2016. Vol.85. No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 5. Category of the project size

Category	∑Programmer	Time required	∑lines
Trivial	1	1-4 week	500
Small	1	1-6 month	1K-2K
Medium	2-5	1-2 year	5K-50K
Large	5-20	2-3 year	50K-100K
Very large	100-1K	4-5 year	1M
Extra large	2K-5K	5-10 year	1M-10M

After conducting the interview, to obtain data related to the financial amount spent on each stage of the project, questionnaires were administered to the project manager and the team of software developers. The questionnaire contained activities that were performed during the software development phase; the questionnaire contained questions on the number of workers involved and time spent to complete the project. The data collected contained the following, among others: project name, user identity, project type and size, project cost, completion time, personnel count, activities undertaken for each project, and effort and cost for each activity.

3.3. Calculate effort with UCP

The effort to each sample projects will calculated using the procedure of UCP as given in section 2.2.

3.4. Determine the distribution of effort

Distribution of effort used refer to this paper[19]. Distribution of effort (in percent) is used to distribute the effort estimation is generated using UCP. To estimate the effort using UCP produces an overall effort for software development projects.

This step produces output: effort per activity of software development projects. While activities for software development projects are found in this paper [13].

3.5. Determine the pay rate per activity

Pay rates were determined considering the standard salary of IT workers or software developers to whom the task has been officially designated or based on a survey by credible institutions.

3.6. Testing and validation of models

Testing and validation were performed with four test data to determine the magnitude of the degree of deviation between the estimated cost and effort and the actual cost and effort.

4. RESULT AND DISCUSSION

4.1. Proposed Model

The proposed model for determining the cost of software development is given in Figure 2. This model is derived from a model or procedure for effort determination using UCP (as presented in Figure 1) by adding three components: the percentage of effort per activity, pay rates per activity, and other costs.

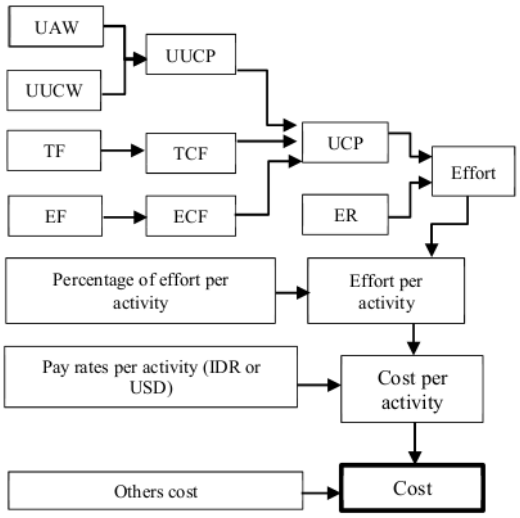

Figure 2. The proposed model for determining software development costs

To obtain software development costs, the effort established using UCP would be distributed to each activity using a benchmark that has been obtained from the survey [19]. The process was carried out to obtain the effort per activity. Furthermore, the effort per activity was multiplied by the pay rates per activity to obtain the cost per activity. Finally, the cost of software development projects was obtained by summing the cost per activity with other costs.

The effort distribution of software development was grouped into three phases: software development, ongoing activity, and

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

testing and quality assurance phases. The percentage of the three phases can be seen in this paper [19]. It can be noted that the three phases corresponded to 72.6% for software development, 17.5% for ongoing activity, and 9.9% for testing and quality assurance.

4.2. Testing and validation of models

Testing and validation was performed to determine the degree of deviation between the cost estimation using models and the actual cost. Tests used test data that has been collected from the previous stage. Test data is a software project that has been completed. Characteristics of the four projects that used the software test data are given in Table 6.

Table 6. Characteristics of test data

ID	Project Name	Time required (month)	Actual cost (IDR)
A	Industrial Registration	2	44,300,000
B	Industrial Business License	2	47,080,000
C	Principle Approval	2	46,800,000
D	Certificate of Company Registration	3	91,500,000

The first step in this test was to obtain the effort of each project using the procedure stated in Section 2.2 or Figure 1. The amount of UCP and effort for all four example projects is shown in Table 7. Effort determination was obtained using formula 9 and ER = 8.2 [16].

Table 7. Amount of UCP and effort for four example projects

Project ID	UUCP	TCF	ECF	UCP	Effort
A	552	0.97	0.86	460.5	3776
B	567	0.97	0.86	473.0	3879
C	552	0.92	0.95	482.4	3956
D	498	1.075	0.935	501.5	4113

Using the percentage of effort distribution as this paper [19], the effort distribution for the four example projects is presented in Table 8.

Table 8. Effort Distribution for four example projects

Phases/ Activities	%	Project ID			
		A	B	C	D
Software development					
Needs analysis	1.6	62	64	65	68
Specification	7.5	283	290	296	308
Design	6.0	225	231	236	245
implementation	52.0	1965	2018	2058	2140
Integrated testing	7.0	263	270	275	286
Acceptance & installation	5.5	206	211	216	224
Ongoing activity					
Project management	3.8	144	148	150	156
Configuration management	4.3	161	166	169	176
Quality assurance	0.9	35	36	37	39
Documentation	8.4	317	325	332	345
Training & technical support	1.0	39	40	41	43
Evaluation & testing	2.0	76	79	80	83
Total of Effort	**100**	3776	3879	3956	4113

By providing the pay rate for each activity (Table 9), we obtained costs as presented in Table 10. The pay rates in Table 9 were obtained from a document of the Indonesia Salary Guide 2011-2012, which was released by Kelly Services [20]. It was used list of salary in 2011 because the project data used in this study occurred between 2010 and 2012. As shown in Table 9, the value of pay rate is obtained from 55% of item in the list of Indonesia Salary Guide 2011-2012 for each corresponding position.

Table 9. Pay rate per activity

Activities	Rate per hours (IDR)
Needs analysis	17,187.5
Specification	17,187.5
Design	10,312.5
implementation	10,312.5
Integrated testing	10,312.5
Acceptance & installation	10,312.5
Project management	34,375.0
Configuration management	34,375.0
Quality assurance	10,312.5
Documentation	10,312.5
Training & technical support	10,312.5
Evaluation & testing	10,312.5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 10. Cost for fourth example projects

Phases/ Activities	Cost estimation of Project (from effort)			
	A	B	C	D
Software development				
Needs analysis	4.857.436	4.989.934	5.088.987	5.290.951
Specification	2.320.127	2.383.415	2.430.727	2.527.194
Design	20.261.386	20.814.067	21.227.236	22.069.672
implementation	2.711.053	2.785.004	2.840.287	2.953.009
Integrated testing	2.123.075	2.180.988	2.224.281	2.312.555
Acceptance & installation	1.070.013	1.099.200	1.121.020	1.165.509
Ongoing activity				
Project management	4.936.892	5.071.558	5.172.231	5.377.499
Configuration management	5.540.761	5.691.899	5.804.886	6.035.262
Quality assurance	365.500	375.469	382.923	398.120
Documentation	3.267.248	3.356.370	3.422.996	3.558.843
Training & technical support	403.639	414.649	422.880	439.662
Evaluation & testing	788.208	809.708	825.781	858.554
Total of cost	48.645.336	49.972.261	50.964.235	52.986.829

Furthermore, Table 11 shows a comparison between the estimated and actual costs for four example projects (A–D). The estimated cost was obtained from a total of cost (Table 10) for each project. The mean deviation for the test data of example projects (A–D) was 6.89%, where for all the test data, the estimated cost was greater than the actual cost. It was in line with several previous studies on effort estimation that showed the degree of deviation to be 6% [8], 9% [9], or 19% [7]. The advantages of this study compared with previous ones are that this study estimates cost, whereas the previous ones were still at the stage of effort estimation.

Table 11. Cost and deviation for fourth example projects

ID	Actual Cost	Estimation Cost From Effort	Others Cost	Estimation Cost	Deviation	%
A	44,300,000	48.645.336	0	48.645.336	4.345.336	9,81%
B	47,080,000	49.972.261	0	49.972.261	2.892.261	6,14%
C	46,800,000	50.964.235	0	50.964.235	4.164.235	8,90%
D	91,500,000	52.986.829	41.000.000	93.986.829	2.486.829	2,72%
				Average	3.472.165	6,89%

5. CONCLUSION

Based on the process and steps that have been performed in this study, we conclude the following: (1) cost estimation models for software development projects of small–medium scale can be developed using UCP and (2) Testing of the model was performed using data from four projects got level deviations between the estimated cost and the actual cost amounted to 6.89%. This shows us a model that has been generated has a sufficient degree of accuracy.

Nevertheless, this model needs to be tested further by using different test data and test data amount to more. The next study is recommended to test the model produced this study to focus on the data project specific software, such as applications of government, business applications, web applications, or other applications. In addition, it is also necessary to study further to the level of the framework of determining the cost of a software project and not just a model.

REFRENCES:

[1] BMI. (2014) Indonesia Information Technology Report. [Online]. http://store.businessmonitor.com/indonesia-information-technology-report.html

[2] Presiden_RI, Peraturan presiden nomor 54 tahun 2010. Jakarta, 2010.

[3] Presiden_RI, Peraturan presiden nomor 70 tahun 2012. Jakarta, 2012.

[4] Group Standish. (2013) news. [Online]. www.standishgroup.com/news

[5] Thomas Tan, Barry Boehm, and Brad Clark, "An Investigation on Application Domains for Software Effort Distribution Patterns," Transition, vol. 12, pp. 0-20, 2011.

[6] Gustav Karner, "Resource Estimation for Objectory Projects," Objective Systems SF AB, 1993.

[7] Bente Anda, "Comparing effort estimat 25 based on use cases with expert estimates," in

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Empirical Assessment in Software Engineering (EASE), Keele UK, 2002, p. 13.

[8] Suresh Nageswaran. (2001, June) Test_Effort_Estimation.pdf. [Online]. http://www.cognizant.com/cognizantcommunity/presentations/Test_Effort_Estimation.pdf

[9] Edward R Carrol, "Estimating Software Based on Use Case Points," in Object Oriented Programming Systems Languages and Applications (OOPSLA) Conference, San Diego, 2005, pp. 257–265.

[10] Stephan Frohnhoff and Gregor Engels, "Revised Use Case Point Method- Effort Estimation in Development Projects for Business Application," in Proceedings of the CONQUEST, 2008.

[11] Sergey Diev, "Software Estimation in the Maintenance," Software Engineering Notes, vol. 31, no. 2, pp. 1-8, March 2006.

[12] M Ochodek, J Nawrocki, and K Kwarciak, "Simplifying effort estimation based on Use Case Points," Information and Software Technology, vol. 53, no. 3, pp. 200-213, 2011.

[13] Kassem Shaleh, "Effort and Cost Allocation in Medium to Large Software Development Project," International Journal of Computer, vol. 5, no. 1, pp. 74-79, 2011.

[14] Grandys Frieska Prassida, Achmad Holil Noor Ali, and Sholiq, "Estimasi Biaya Pembuatan Modul Enterprise Resource Planning (ERP) Untuk Unit Bisnis Pabrik Gula Di PT. Perkebunan XYZ Dengan Metode Use Case Point," Jurnal Teknik Pomits, vol. 1, no. 1, pp. 1-6, 2012.

[15] Gustavo Ibarra Ibarra and Patrícia Vilain, "Software Estimation Based on Use Case Size," in Brazilian Symposium on Software Engineering, 2010, pp. 178-187.

[16] Apol Pribadi Subriadi, Sholiq, and Puji Agustin Ningrum, "Critical Review of The Effort Rate Value In Use Case Point Method for Estimating Software Development Effort," Journal of Theoretical and Applied Information Technology, vol. 59, no. 3, pp. 735-744, 2014.

[17] Sholiq, Teguh Sutanto, Arifin Puji Widodo, and Wahyu Kurniawan, "Effort Rate on Use Case Point Method for Effort Estimation of Website Development," Journal of Theoretical and Applied Information Technology, vol. 63, no. 1, pp. 209-218, May 2014.

[18] Donna L. Johnson,.: LOGOS International, IncNashville, 2006.

[19] Putu Linda Primandari and Sholiq, "Effort Distribution to Estimate Cost in Small to Medium Software Development Project with Use Case Points," in Procedia Computer Science, vol. 72, 2015, pp. 78-85.

[20] Kelly. (2011) Indonesia Salary Guide 2011-2012. [Online]. http://www.kellyservices.co.id/res/content/id/services/en/docs/indonesia_salary_guide_2011_12.pdf

MODEL TO DETERMINE COST ESTIMATION FOR SOFTWARE DEVELOPMENT PROJECTS OF SMALL AND MEDIUM SCALES USING USE CASE POINTS

ORIGINALITY REPORT

19%	13%	17%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	eprints.perbanas.ac.id Internet Source	4%
2	www.semanticscholar.org Internet Source	2%
3	www.airccse.org Internet Source	2%
4	Mudasir Manzoor Kirmani, Abdul Wahid. "Impact of Modification Made in Re-UCP on Software Effort Estimation", Journal of Software Engineering and Applications, 2015 Publication	1%
5	Sholiq, Arifin Puji Widodo, Teguh Sutanto, Apol Pribadi Subriadi. "Effort Distribution per Activities for Small Software Development Project Uses Prototype Model", 2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 2018 Publication	1%

6	text-id.123dok.com Internet Source	1 %
7	Ibarra, Gustavo Bestetti, and Patricia Vilain. "Software Estimation Based on Use Case Size", 2010 Brazilian Symposium on Software Engineering, 2010. Publication	1 %
8	1library.net Internet Source	1 %
9	"Lower Urinary Tract Symptoms, Sexual Function, and the Quality of Life of Married Women with Urinary Incontinence", Journal of Comprehensive Nursing Research and Care, 2019 Publication	1 %
10	E Prayitno, J Siregar, Y N Dewi, C Bachri, L Indriyani, S Ma'arif. "Use Case Points (UCP) with 3 Point in Program Evaluation and Review Technique (PERT) to Estimate Effort Software", Journal of Physics: Conference Series, 2020 Publication	<1 %
11	adoc.pub Internet Source	<1 %
12	Imam Kurniawan, Arry Akhmad Arman, Sukrisno Mardiyanto. "Development of analogy-based estimation method for	<1 %

	software development cost estimation in government agencies", 2017 6th International Conference on Electrical Engineering and Informatics (ICEEI), 2017 Publication	
13	static1.1.sqspcdn.com Internet Source	<1 %
14	www.igi-global.com Internet Source	<1 %
15	Shofiyah Al Idrus, Wahyu Nur Hidayat, Achmad Hamdan. "NAYA as a Tool of Software Cost Automatic Analysis", 2020 4th International Conference on Vocational Education and Training (ICOVET), 2020 Publication	<1 %
16	Sholiq Sholiq, Pandu Satrio Hutomo, Ariani Dwi Wulandari, Apol Pribadi Subriadi, Anisah Herdiyanti, Eko Wahyu Tyas Darmaningrat. "Testing of Owner Estimate Cost Model with Android-based Application", 2019 3rd International Conference on Informatics and Computational Sciences (ICICoS), 2019 Publication	<1 %
17	"Software Engineering Application in Informatics", Springer Science and Business Media LLC, 2021 Publication	<1 %

18	Nassif, Ali Bou, Danny Ho, and Luiz Fernando Capretz. "Towards an early software estimation using log-linear regression and a multilayer perceptron model", Journal of Systems and Software, 2013. Publication	<1 %
19	Anupama Kaushik, Nisha Choudhary, Priyanka. "Chapter 2 Software Cost Estimation Using LSTM-RNN", Springer Science and Business Media LLC, 2021 Publication	<1 %
20	Anandi Hira, Barry Boehm. "Combatting Use Case Points' Limitations with COCOMO(R) II and Relative Difficulty", 2016 23rd Asia-Pacific Software Engineering Conference (APSEC), 2016 Publication	<1 %
21	Sarwosri, Muhammad Jabir Al Haiyan, Mujahid Husein, Aditya Putra Ferza. "The development of method of the enhancement of Technical Factor (TF) and Environmental Factor (EF) to the Use Case Point (UCP) to calculate the estimation of software's effort", 2016 International Conference on Information & Communication Technology and Systems (ICTS), 2016 Publication	<1 %

m.scirp.org

22	Internet Source	<1%
23	Luís M. Alves, Pedro Ribeiro, Ricardo J. Machado. "Chapter 5 Architectural Element Points: Estimating Software Development Effort by Analysis of Logical Architectures", Springer Science and Business Media LLC, 2016 Publication	<1%
24	www.kemenkeu.go.id Internet Source	<1%
25	www.simula.no Internet Source	<1%
26	Benrahman, Ford Lumban Gaol, Harco Leslie Hendric Spits Warners, Edi Abdurahman, Benfano Soewito. "Using Use Case Points Method In a Health Surveillance Application Estimation", 2019 Fourth International Conference on Informatics and Computing (ICIC), 2019 Publication	<1%
27	Renny Sari Dewi, Riyanarto Sarno. "Software Effort Estimation Using Early COSMIC to Substitute Use Case Weight", 2020 International Seminar on Application for Technology of Information and Communication (iSemantic), 2020 Publication	<1%

28	Yue Xie, Jilian Guo, Anwei Shen. "Chapter 2 Use Case Points Method of Software Size Measurement Based on Fuzzy Inference", Springer Science and Business Media LLC, 2015 Publication	<1 %
29	arxiv-export-lb.library.cornell.edu Internet Source	<1 %
30	blog.mgallen.com Internet Source	<1 %
31	groups.uni-paderborn.de Internet Source	<1 %
32	"Pervasive Computing and Social Networking", Springer Science and Business Media LLC, 2022 Publication	<1 %
33	Andy Effendi, Reina Setiawan, Zulfany Erlisa Rasjid. "Adjustment Factor for Use Case Point Software Effort Estimation (Study Case: Student Desk Portal)", Procedia Computer Science, 2019 Publication	<1 %
34	Anureet Kaur, Kulwant Kaur. "Chapter 14 Effort Estimation for Mobile Applications Using Use Case Point (UCP)", Springer Science and Business Media LLC, 2019 Publication	<1 %

35	Mourad Badri, Linda Badri, William Flageol, Fadel Toure. "Investigating the Accuracy of Test Code Size Prediction using Use Case Metrics and Machine Learning Algorithms", Proceedings of the 2017 International Conference on Machine Learning and Soft Computing - ICMLSC '17, 2017 Publication	<1 %
36	ir.kabarak.ac.ke Internet Source	<1 %
37	www.europment.org Internet Source	<1 %
38	Apol Pribadi Subriadi, Sholiq, Valeriana Lukitosari, Rochma Permatasari. "The Comparison Analysis Of Estimation Effort Among Software Development Using Function Point Method", 2018 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 2018 Publication	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		