

# Implementation of Linear Equations to Determine Weight of Unadjusted Use Case Weight in Use Case Points for Estimating Software Development Effort

SHOLIQ, APOL PRIBADI SUBRIADI, DESY GITAPRATAMA

**Abstract** - The estimates of the number of resources required in a software development project is an important part of the project planning stage. Estimates are activities that predict the output of a project by reviewing existing resources such as time, cost, risks to be faced, and the effort required to complete the project. Methods such as Use Case Points (UCP), which are based on the number of actors and use case complexity are popular as methods for estimating effort in software development projects. However, the weight calculation for the Unadjusted Use Case Weight (UUCW) as part of the UCP still has something called “not mathematically valid operations.” Therefore, it takes an alternative way to calculate the weight of the UUCW.

In this study, we implement a linear equation to determine the UUCW. The linear equation is derived from the value of the weight of UUCW on UCP original (UCP proposed by Karner). A linear line is created in which the x-axis is the center point of the transaction amount and the y-axis is the value of the weight of UUCW. Finally, we get the formula to determine UUCW, namely  $UUCW = \sum_{i=1}^n 1.428 x_i + 2.1429$ , where  $n$  is the number of use cases within system and  $x$  is the number of transactions within use case  $i$ .

The test on the linear equation of UUCW conducted using 17 test data from software development projects of small and medium scale showed that the formula of linear equations for UUCW (new formula) gives a better effort estimation result of the estimation using the original formula of UUCW. It had been characterized by deviation values using the new formula of UUCW smaller than the deviation value using the original formula of UUCW

**Keywords** - Use Case Points, Unadjusted Use Case Weight (UUCW), Effort Estimation, Software Development Project

## I. Introduction

As we know, software development project consists of five stages: initialization, planning, implementation, monitoring, and evaluation. At the planning stage, we must prepare very well. There are several important aspects that must be considered, one of which is to make estimates or forecasts of resources in terms of cost, time, and human resources. If these are not addressed, then a software development project is potentially a huge failure.

However, according to some previous studies there are methods that can be used for the calculation of estimated effort of software development projects, like the COCOMO, Expert Judgment, Function Point Size Estimates, Analogy, Neural Network, Fuzzy, and Use Case Points (UCP) [1]. Each of these methods has its own advantages and disadvantages. However, there is no single best method for projects because every characteristic of a software

development project is different. One of the methods of effort estimation in software development projects that is increasingly popular in the effort estimation techniques is the UCP method, first proposed by Gustav Karner in 1993 [2]. Studies show that in UCP the level of accuracy of estimate is pretty good, among other things: (1) estimating using UCP is better than estimating through experts, UCP deviation is 19% whereas experts deviation is 20% [3] (2) UCP deviation of 9% has been observed on approximately 200 software development projects [4].

In the UCP method, effort estimation is obtained by multiplying the value of the UCP with the value of the effort rate (ER). The value of UCP is defined by the number of transactions of the Use Case that are run by actors who play a role in the system. The complexity of the actor called the Unadjusted Actor Weight (UAW) is obtained from counting how many actors of each type of complexity (simple, average, or complex) multiplied by the weight of each type. While the complexity of the Use Case called the Unadjusted Use Case Weight (UUCW) is obtained by calculating the value of the Use Case of each type of complexity (simple, average, or complex) multiplied by the weight of each type. The UAW and UUCW then are summed to get the UUCP.

Furthermore, UUCW is obtained by the sum of all the weights of the complexity of the use case. While the weight of the complexity of a use case is obtained from the categorization as follows [2]: (1) if the transaction amount is less than 4, then a use case is categorized as “simple” so that the weight of complexity is 5; (2) if the transaction amount is between 4 to 7, then a use case is categorized as “average” so that the weight of complexity is 10; (3) and if the transaction amount is greater than 7, then a use case is categorized as “complex” so that the weight of complexity is 15.

Transformation of the number of transactions in a category to UUCW is called “not valid formally”, because the calculations are based on a measurement scale transformation algebraically unacceptable [5]. In UUCW calculation, the number of transactions (ratio scale) is transformed into one of the three categories (simple, medium, or complex), which is nominal scale, and then transformed again into a ratio scale. Therefore, the transformation of alternating (ratio-nominal-ratio) by Ochodek is called “not mathematically valid operations” [5].

This study aims to implement a linear equation to the UUCW calculation in order to be “mathematically valid operations.” Implementation of a linear equation is expected to get a simpler way to derive value of the UUCW.

## II. Literature Review

The UCP is an estimation technique for software development projects used mostly for forecasting the efforts that will be utilized. The UCP, first proposed by Gustav Karner in 1993, derived or adapted from Function Points Analysis with a purpose of providing simple estimation methods specific to object-oriented software projects [2]. Procedure of estimation using UCP is contened in these papers [1] [6].

In its development, this method experienced some improvement modifications, Frohnhoff and Engels applied UCP for large software projects, namely business-specific software for industrial applications. The accuracy of the results was obtained using the original UCP (Karner's UCP), which was 42%. Of course, this result was not realistic for business applications in the industry. Therefore, they provided a detailed description for charging the T-Factor and M-Factor (T-Factor relates to technical factors of the developed system, while M-Factor relates to the competence of the development team). In this way, the estimated revised UCP proposed by Frohnhoff and Engels had better accuracy than the original UCP, having a deviation of= 20% [7].

Other studies proposed modifications to UCP for specific things, for example, Use Case Points maintenance (UCPm) was proposed by Sergey Diev to estimate effort on software development projects that are part of a larger software project, an extension of the UCP of software estimation. It is to reflect the specifics of the maintenance phase of software life cycle. [8].

Other modifications are simplifying UCP. The simplification of UCP method has been done by Ochodek et al in which the first step is to determine the Unadjusted Actor Weights (UAW). The test data used 14 software development projects, which have been completed. Results of this study stated that the UAW calculation does not significantly influence the final results of the estimation or the UAW can be eliminated, so the effort estimation procedure becomes simpler [5].

In the study conducted by Azzeh M, UCP is used to estimate for global software development projects. This study suggests that the UCP can be used as an alternative to estimate software development projects on a global scale by modifying the environmental factors of the original UCP [9].

Modifications to the effort rate (ER) or the productivity index (PI) was made by Subriadi et al as a correction to the proposed ER Karner in the original UCP of 20 man hours. ER is a constant to gain effort value, where effort = UCP \* ER. In this study, the obtained ER = 8.2 man-hours on the basis that the productivity of software development can be better due to several reasons, namely: the presence of methods for software development, the existence of software engineering technology, the existence of the software components, and the availability of resources on the internet that can increase the productivity of the software development teams [5]. Likewise on the other empirically,

obtained ER=4.41 man-hours to the project website development [10].

In some of the previous studies, we have gained a few changes to the original UCP method to adjust the existing conditions. This study focuses on one of the improvements to the original UCP on the improvement UUCW formula.

## III. Research Methods

In this study, we implement a linear equation to determine the value of Unadjusted Use Case Weight (UUCW). The linear equation is derived from the value of the weight of UUCW on UCP original (UCP proposed by Karner), where weight = 5 for a transaction amount less than or equal to 3, weight = 10 for 4 to 7 transactions, and weight = 15 for transactions greater than or equal to 8. A linear line is created in which the x-axis is the center point of the transaction amount and the y-axis is the value of the weight of UUCW. Determination of the Center Point (CP) of the x-axis is given in Table 1, where CP = 2 for type = simple, which is obtained from the median of the number of transactions from 1 to 3. Also, for type = average, as transactions are between 4 to 7, the median is 5.5 (CP = 5.5). Likewise for type = complex, with a large number of transactions equaling to 8, we take the median as 9 (CP = 9).

Table 1 Determining center point as x-axis of linear equation

Type	Weight	∑Transaction	Center Point
Simple	5	1 to 3	2
Average	10	4 to7	5.5
Complex	15	≥8	9

After the midpoint or CP of the x axis and the y-axis is obtained using the value of the weight, the three points to make a line of linear equations obtained are P1 (2,5), P2 (5.5,10), and P3 (9,15). Thus, the linear equation to determine UUCW can be obtained.

After the linear equation is obtained, the next step is to test a new equation to determine the value of UUCW. Tests are carried out using data from 17 software projects that have been completed. Profiles of the 17 software projects are listed in Table 2.

Table 2. The projects used to test in this research

Project ID	Name of application/project
A	Multilevel marketing system
B	Sales system
C	Training management system
D	STNK electronic system
E	Labor management system
F	Online ticket management system
G	Building rental system
H	Mall search system
I	Cake sales system
J	Data dictionary system
K	System of lecturer credit scoring
L	Student administration system
M	Student information system
N	Final project monitoring system
O	Research documentation system
P	Mobile application school social network
Q	Web application school social network

Testing is focused by comparing the level of deviation of effort estimation between: (1) determining UUCW using linear equations and (2) determining UUCW using the method by Karner (original). Of course the level of deviation is obtained by comparing the effort estimation (UUCW by linear equations or original UUCW) with the actual effort that is obtained from a survey in the field.

#### IV. Result and Discussion

We need to recall the determination of UCCW in the original UCP method that has been discussed in step 2 of section 2.2 of this article. The value of UUCW is obtained from the sum of weights for all use cases that exist in the system. While weight can be worth 5, 10, or 15 depending on the number of transactions in each use case. If the transaction amount is less than or equal to 3, then the value of the weight = 5, if the transaction amount is 4 to 7, the value of the weight =10, and if the transaction amount is greater than or equal to 8, then the value of the weight =15.

Referring to Table 1 that the weight to determine UCCW that proposed by Karner can be used as a linear equation, when the number of use cases and the amount of the weight of every use case in a straight line drag to create a linear equation. The x-axis is the number of transactions in a use case and the y-axis is the weight of each use case. The values on the x-axis are the CP as presented in Table 3, while the values on the y-axis is the Weight. Therefore, we can make three points, namely P1(2.0,5.0), P2(5.5,10.0), and P3(9.0, 15.0) (see Table 1).

Table 3 The points for linear equation that center point of sum of transaction as x-axis and weight as y-axis

Point	CP as x-axis	Weight as y-axis
P1	2.0	5.0
P2	5.5	10.0
P3	9.0	15.0

Thus, the linear equation for determining weights for UCCW can be obtained as shown in Figure 1, where a linear equation is  $y = 1.428x + 2.1429$ . The formula derived for all the points are given as follows.

$$y_i = 1.428x_i + 2.1429 \dots \dots \dots (1)$$

Where,

$y_i$  = weight for a use case

$x_i$  = sum of transaction within a use case

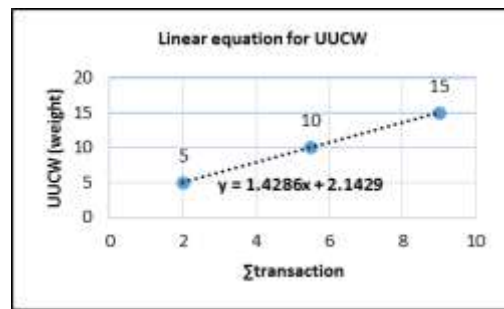


Figure 1 Linear equation to get the weight to determine UCCW

Next, to apply the formula of linear equations to obtain UCCW, we can develop the formula becomes:

Project_ID	Actual Effort	UUCW	UAW	UUCP	TCF	ECF	UCP	Estimated Effort (Karner)
I	2	3	4	5	6	7	8	9
A	3684	355	15	370	1.13	0.77	320.5	6410.3
B	1980	145	18	163	1.08	0.77	135.6	2711.0
C	3950	325	12	337	1.10	0.94	345.0	6900.6
D	1925	90	6	96	1.02	1.09	106.2	2124.9
E	2175	125	9	134	1.03	0.98	134.6	2692.1
F	2226	120	9	129	1.12	1.00	143.1	2862.3
G	2640	200	12	212	1.00	0.92	195.0	3900.8
H	2568	175	9	184	0.95	0.92	160.8	3216.3
I	3042	245	12	257	0.89	1.19	272.2	5443.8
J	1696	140	6	146	0.97	0.76	106.4	2127.4
K	2355	125	9	134	1.09	1.00	144.7	2893.3
L	2208	160	12	172	1.06	0.77	139.7	2794.5
M	3350	160	18	178	1.14	0.86	174.5	3490.2
N	3344	340	15	355	0.99	0.88	307.5	6150.4
O	4568	345	9	354	1.01	0.92	327.3	6546.2
P	2532	260	15	275	1.13	0.80	248.6	4972.0
Q	2340	175	15	190	1.14	0.62	134.3	2685.8

$$UUCW = \sum_{i=1}^n 1.428 x_i + 2.1429 \dots \dots (2)$$

Where,

$n$  = sum of use cases in system

$x_i$  = sum of transaction within a use case  $i$

Table 4 The actual effort and estimation effort for 17 project as tested data

#### Testing Results

In this section will be tested against the new formula of UUCW with 17 projects that have been known to actual effort (see Table 4). Test data is used as shown in Table 1, namely projects of small and medium-scale software that has been completed. Meanwhile, the actual effort of data obtained from a survey of 17 software projects are presented in Table 4 (column 2). Other data, about the effort estimation of the calculation includes UUCW, UAW, UUCP, UCP, TCF, ECF, UCP, and estimated effort are shown in Table 4 columns 3–9. Quantities presented in Table 4 were obtained using the formula of the original UCP Karner, which has been filed with the ER = 20 man-hours.

Next, we perform calculations using the new formula of UUCW that has been obtained. Table 5 shows the results of UUCW calculation with the new formula for the 17 software projects earlier

Table 5 UUCW using new formula

Project_ID	UUCW	Project_ID	UUCW
A	373.58	J	120.00
B	146.43	K	126.43
C	305.72	L	157.15
D	79.29	M	160.00
E	110.00	N	275.72
F	110.72	O	325.01
G	191.43	P	235.00
H	171.43	Q	160.72
I	206.43		

The values of the UAW, TCF, and ECF are unchanged (see Table 4). While other values are changed due to changes in the value of UUCW (new formula) is UUCP, UCP, and estimated effort. These values are presented in Table 6.

Furthermore, formula validation is done by comparing the level of error (deviation) results of estimated effort using the original formula of UUCW and the estimated effort using new formula of UUCW. Deviation formula used is given as follows:

$$Deviation = \frac{|ActualEffort - EstimationEffort|}{ActualEffort}$$

Table 7. Results of deviation calculations to estimated effort using the original formula and new formula of UUCW using ER = 20 man hours

Project_ID	Actual effort	Est. effort (original)	Est. effort (new)	Dev (original)	Dev (new)
1	2	3	4	5	6
A	3684	6410.3	6732.1	0.74*	0.83
B	1980	2711.0	2734.8	0.37*	0.38
C	3950	6900.6	6505.8	0.75	0.65
D	1925	2124.9	1887.8	0.10	0.02
E	2175	2692.1	2390.7	0.24	0.10
F	2226	2862.3	2656.4	0.29	0.19
G	2640	3900.8	3743.1	0.48	0.42
H	2568	3216.3	3153.9	0.25	0.23
I	3042	5443.8	4626.8	0.79	0.52
J	1696	2127.4	1836.0	0.25	0.08
K	2355	2893.3	2924.1	0.23*	0.24
L	2208	2794.5	2748.2	0.27	0.24
M	3350	3490.2	3490.2	0.04	0.04
N	3344	6150.4	5036.7	0.84	0.51
O	4568	6546.2	6176.5	0.43	0.35
P	2532	4972.0	4520.0	0.96	0.79
Q	2340	2685.8	2484.0	0.15	0.06
average				0.42	0.33

Using the deviation formula above each deviation of the estimated effort generated from the original formula (column 5) and the new formula (column 6) can be calculated. Results of calculating deviation are given in Table 7.

In Table 7 we can see that there are three of the 17 projects (marked with an asterisk \*) in which the deviation calculated using a new formula of UUCW is greater than that calculated using a original formula (projects A, B, and K), 1 project level it's deviations are equal (project M), and

13 projects showed that the their deviation using the new formula of UUCW is smaller than the their deviation using original formula (project C, D, E, F, G, H, I, J, L, N, O, P, and Q). While the average of deviation indicates that the calculation by using the new formula of UUCW gets smaller deviation than using the original formula of UUCW, the deviation to the new formula = 0.33 and the deviation using the original formula = 0.42. In the mean deviation obtained, a decrease of 0.09 (9%) is observed. It should be noted that the calculation of the estimated effort both UUCW original and a new formula using the ER = 20 man hours as proposed Karner [2].

In fact some previous studies may have used the values of ER other than ER = 20 man-hours proposed by Karner. This has been discussed in section 2.2 of this paper. Table 8 shows the deviation of estimated effort using the original formula of UUCW and the new formula of UUCW for ER = 15 man hours. From the table the obtained deviation for the estimated effort using original formula amounted to 0.17 and new formula amounted 0.16.

Table 8. Results of deviation calculations to estimated effort using the original formula and new formula of UUCW using ER = 15 man hours

Project_ID	Actual effort	Est. effort (original)	Est. effort (new)	Dev (original)	Dev (new)
A	3684	4807.7	5049.1	0.31	0.37
B	1980	2033.3	2051.1	0.03	0.04
C	3950	5175.4	4879.3	0.31	0.24
D	1925	1593.6	1415.9	0.17	0.26
E	2175	2019.0	1793.0	0.07	0.18
F	2226	2146.7	1992.3	0.04	0.10
G	2640	2925.6	2807.3	0.11	0.06
H	2568	2412.2	2365.4	0.06	0.08
I	3042	4082.8	3470.1	0.34	0.14
J	1696	1595.6	1377.0	0.06	0.19
K	2355	2169.9	2193.1	0.08	0.07
L	2208	2095.9	2061.1	0.05	0.07
M	3350	2617.7	2617.7	0.22	0.22
N	3344	4612.8	3777.5	0.38	0.13
O	4568	4909.6	4632.4	0.07	0.01
P	2532	3729.0	3390.0	0.47	0.34
Q	2340	2014.4	1863.0	0.14	0.20
average				0.17	0.16

Testing was also done for some other ER values. If we consider the deviation of the testing of various values of ER as given in Table 9 that to 7 times the test with various values of ER, 6 times the deviation of the estimated effort using a new formula of UUCW smaller than the deviation of the estimated effort using the original formula of UUCW (for ER = 36, 30, 24, 20, 15, and 10). In contrast only 1 time deviation of the estimated effort using the original formula of UUCW is smaller than the estimated effort using a new formula of UUCW for ER = 8.2 (see Table 9).

Table 9. The deviation of test using various Effort Rate

ER values (man hours)	Average of deviation	
	Original	New
36	1.56	1.39
30	1.13	1.00

ER values (man hours)	Average of deviation	
	Original	New
24	0.71	0.60
20	0.42	0.33
15	0.17	0.16
10	0.17	0.16
8.2	0.41*	0.45

## v. Conclusion

After testing and discussion of the results, we can conclude several things, among others: First, we have produced an additional way to calculate UUCW using linear equations as presented in equation 1, which simplifies the calculation as it improves the accuracy of results namely the effort estimation. From the test results of the new formula UUCW using 17 software projects of small and medium scale provide results that are average calculations of the estimated effort to get relatively better results as compared with the results derived using the original formula proposed by Karner.

Second, the test was also conducted on a variety of ER values as used in some previous studies. Seven times was examined by the ER = 36, 30, 24, 20, 15, 10, and 8.2 man hours showed that 6 times using a new formula UUCW gives better results than using the original formula. Third, further research can be done by extending the points for linear equations of UUCW in order to obtain better results.

## References

- [1] Vahid Khatibi and Dayang N. A. Jawawi, "Software Cost Estimation Methods: A Review," *Journal of Emerging Trends in Computing and Information Sciences*, vol. II, no. 1, pp. 21-29, 2011.
- [2] Gustav Karner, "Resource Estimation for Objectory Projects," *Objective Systems SF AB*, 1993.
- [3] Bente Anda, "Comparing effort estimates based on use cases with expert estimates," in *Empirical Assessment in Software Engineering (EASE)*, Keele UK, 2002, p. 13.
- [4] 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.
- [5] 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.
- [6] 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, January 2014.
- [7] Stephan Frohnhoff and Gregor Engels, "Revised Use Case Point Method- Effort Estimation in Development Projects for Business Application," in *Proceedings of the CONQUEST*, 2008.
- [8] Sergey Diev, "Software Estimation in the Maintenance Context," *Software Engineering Notes*, vol. 31, no. 2, pp. 1-8, March 2006.
- [9] Mohammad Azzeh, "Software Cost Estimation Based on Use Case Points for Global Software Development," in *5th International Conference on Computer Science and Information Technology (CSIT)*, 2013, pp. 214-218.
- [10] 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.