

PAPER • OPEN ACCESS

Use Case Points (UCP) with 3 Point in Program Evaluation and Review Technique (PERT) to Estimate Effort Software

To cite this article: E Prayitno *et al* 2020 *J. Phys.: Conf. Ser.* **1641** 012008

View the [article online](#) for updates and enhancements.

You may also like

- [Low-level lasers affect uncoupling protein gene expression in skin and skeletal muscle tissues](#)
K S Canuto, L P S Sergio, F Paoli et al.
- [Evaluating spectral overlap with the degree of quenching in UCP luminescence energy transfer systems](#)
Letitia Burgess, Alex R Jones, Sam Hay et al.
- [Study of Electrolyte and Electrode Composition Changes vs Time in Aged Li-Ion Cells](#)
L. M. Thompson, J. E. Harlow, A. Eldesoky et al.



The Electrochemical Society
Advancing solid state & electrochemical science & technology

241st ECS Meeting

May 29 – June 2, 2022 Vancouver • BC • Canada

Abstract submission deadline: Dec 3, 2021

Connect. Engage. Champion. Empower. Accelerate.
We move science forward



Submit your abstract



Use Case Points (UCP) with 3 Point in Program Evaluation and Review Technique (PERT) to Estimate Effort Software

E Prayitno¹, J Siregar², Y N Dewi³, C Bachri⁴, L Indriyani⁵, S Ma'arif⁶

^{1,2,3,4}STMIK Nusa Mandiri, Jakarta, Indonesia

⁵Universitas Bina Sarana Informatika, Jakarta, Indonesia

⁶STMIK Mercusuar, Bekasi, Indonesia

E-mail: edhi.epo@nusamandiri.ac.id, juarni.jsr@nusamandiri.ac.id,
yumi.ymd@nusamandiri.ac.id, chaerul.cbc@nusamandiri.ac.id, luthfi.lfy@bsi.ac.id,
samsul@mercusuar.ac.id

Abstract. Software has an important influence in business to accelerate and manage business processes better. Software development projects and projects in general require precise estimates for good business management and running smoothly. The method of Use Case Points is one of the methods used to estimate object-based software utterances, but there are some disadvantages such as subjective assessment of Technical Complexity Factor (TCF) and Environmental Complexity Factor (ECF) for novice developers who are inexperienced. Three points from the Program Evaluation and Review Technique (PERT) are used at UCP to help estimate the two factors better by modifying both assessment factors. By using 3 PERT points in UCP, it is expected to help novice developers in estimating the size of the software as predicted by experienced developers. Results of the calculation of the estimated software information systems sales of goods in the Madrasah Istiqlal foundation obtained an effort value is 1,011 man-hours.

1. Introduction

Estimation is one of the initial stages of an important and fundamental system development planning activity for business continuity in the field of software engineering. For some people especially new in the software engineering, estimation is a difficult activity with high uncertainty due to assess something that is not sure the end result [1]. The following are problems with software project development, one of which is the estimation error and the importance of using formal estimation methodology, such as a report from the Standish Group Study 'CHAOS Summary 2018', which states that 36% of IT projects were successfully completed in a timely, cost-appropriate and appropriate manner [2]. Also a report issued by Capers Jones, Chief Scientist Emeritus Software Productivity Research (SPR) that estimates of formal costs are important factors to prevent project failure and the use of tools and methodologies for developing formal projects will double labor productivity compared to conventional use. Similar opinion is also given by Trendowicz appropriate cost estimation is one of the success factors for project managers in managing and completing projects well [3]. Some estimation of formal methods that can used to calculate software metrics, including Function Point (FP), Function Point Analysis (FPA), Delphi, Use Case Points (UCP), Cocomo, Analogy and etc. The following results show that the UCP estimation method has a good level of accuracy, such as UCP has a deviation of 6% [4], UCP has a deviation of 9% [5] and also 9% compared to experts' estimates with a deviation of 20% [6]. The use



of UCP method is quite difficult because it has weaknesses in assessing technical factors and environmental factors that are subjective so that it takes 3 points PERT to set the assessment approaching objective. With 3 points PERT will greatly help novice developers or developers who do not have a track record of software development to properly estimate the effort of a software.

2. Literature Review

2.1 Use Case Points (UCP)

The UCP method was developed by Gustav Karner in 1993, a method of estimating efforts based on the number and complexity of use cases [7]. Following are the steps to determine the estimated effort value using UCP, i.e.:

1. Calculate the value of Unadjusted Use Case Point (UUCP), by first determining the value of Unadjusted Actor Weights (UAW) and Unadjusted Use Case Weights (UUCW).

The Unadjusted Actor Weights (UAW) and Unadjusted Use Case Weights (UUCW) value is determined from the number of actors and use case classified i.e. simple, average, or complex multiplied by the weight factor value of each level according to the table and formula below:

Table 1. Types, Weights and Descriptions of Actors

Actor	Weights	Description
Simple	1	Defined with API
Medium	2	Interact via TCP / IP Protocol
Complex	3	Interact with the GUI or Web Page

Table 2. Types, Weights and Descriptions of Use Cases

Use Case	Weights	Description
Simple	5	Use ≤ 3 transaction
Medium	10	Use 4 to 7 transaction
Complex	15	Use > 7 transaction

$$\text{UUCP} = \text{UAW} + \text{UUCW} \quad (1)$$

2. Calculate the value of Technical Complexity Factor (TCF) and Environmental Complexity Factor (ECF). The value of Complexity factor are factors that directly influence the process of developing a software project. Below are the values for each complexity factor:

Table 3. Technical Factor and weights

No.	Technical Factor	Weights
1.	Distributed System Required	2
2.	Response Time is Important	1
3.	End User Efficiency	1
4.	Complex Internal Processing Required	1
5.	Reusable Code Must Be A Focus	1
6.	Installation easy	0.5
7.	Usability	0.5
8.	Cross-platform support	2
9.	Easy to change	1
10.	Highly concurrent	1
11.	Custom security	1
12.	Dependence on third-part code	1
13.	User training	1

Table 4. Environmental Factor and weights

No.	Technical Factor	Weights
1.	Familiarity with the Project	1.5
2.	Application Experience	0.5
3.	OO Programming Experience	1
4.	Lead Analyst Capability	0.5
5.	Motivation	1
6.	Stable Requirements	2
7.	Part Time Staff	-1
8.	Difficult Programming Language	-1

The values of the Technical Factor (TF) and Environmental Factor (EF) are multiplied by the weighted value of each. The weight of the value given on each factor depends on how much influence the factor is. Where value 0 means no affect, the value 3 means average, and the value of 5 means to give a big effect. The multiplication of values and weights are then aggregated to obtain a total Technical Factor and Environmental Factor, which is then used to obtain a Technical Complexity Factor (TCF) and Environmental Complexity Factor (ECF) based on the formula:

$$TCF = 0.6 + (0.01 \times TF) \quad (2)$$

$$ECF = 1.4 + (-0.03 \times EF) \quad (3)$$

3. And finally the Use Case Point (UCP) value is obtained by multiplying the UUCP value with TCF and ECF.

$$UCP = UUCP \times TCF \times ECF \quad (4)$$

2.2 Program Evaluation and Review Technique (PERT)

PERT is a useful method for reducing production delays and disruptions, as well as coordinating various parts of a work as a whole and speeding up project completion. The use of PERT method allows a job to be controlled and run regularly because the schedule and budget of a job have been determined before being carried out. The probability in PERT has three values is most optimistic estimate (o), most likely estimate (m), and most pessimistic estimate (the least possible estimate). The value most likely estimate (m) is weighted 4 times more than two estimates the other (optimistic (o) and pessimistic (p)). The best estimation (te) PERT is following the formula below: [8]

$$te = \frac{(o + 4m + p)}{6} \quad (5)$$

And following the graph of the beta distribution as below:

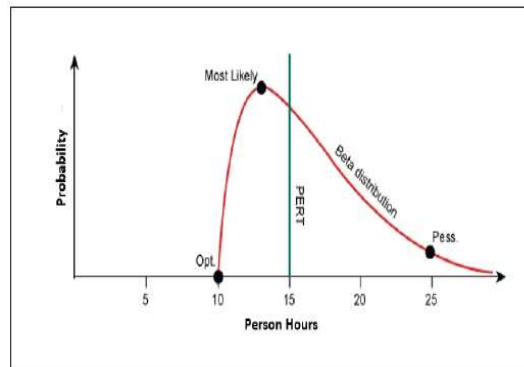


Figure 1. Beta distribution of the estimated 3 Points of PERT

3. Method

3.1 Data Research

Research data in this paper is taken from interview with Chairman of Madrasah Istiqlal Foundation to determine software system requirement which depicted in use case diagram. Use case metric that represents data contains number actors involved and functions provided by the system. And the questionnaire is used to complete the data needed in determining the value of TCF and ECF. Below is Use Case of the software information systems sales of goods Madrasah Istiqlal Foundation, that is sales use case, inventory use case, privileges acces use case and report use case.

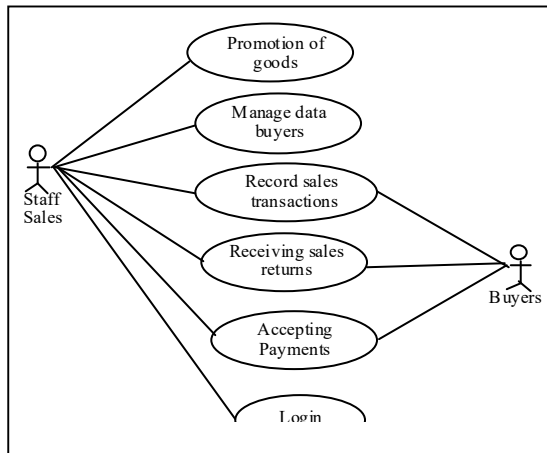


Figure 2. Sales Use Case

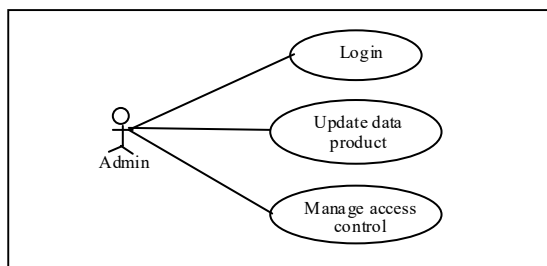


Figure 4. Privileges Acces Use Case

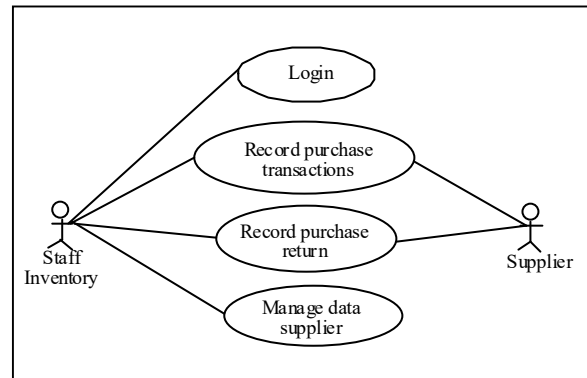


Figure 3. Inventory Use Case

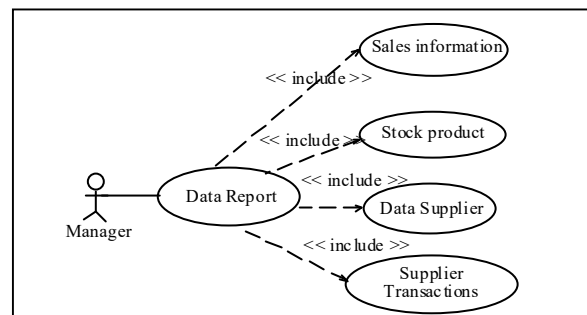


Figure 5. Report Use Case

3.2 Proposed method

Some background thoughts for combining UCP estimation method with 3 points PERT, i.e.:

1. Beginner software developers have difficulty determining the estimation of an exact value that describes the real conditions of software development.
2. Estimated values obtained by novice developers are expected to be close to estimates made by professional developers.
3. Helps novice developers easily determine effort estimates using UCP method combined with PERT.

This is an overview of proposed method use the UCP estimation method with 3 points PERT below:

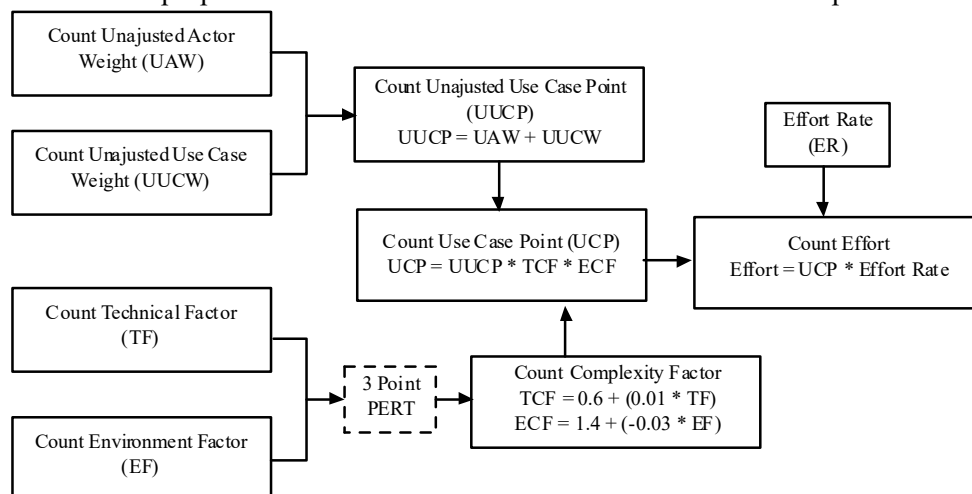


Figure 6. Combining UCP and 3 Point PERT to Estimate Effort Software

4. Results and Discussion

Software information systems sales of goods Madrasah Istiqlal Foundation is known there are 6 actors and 17 use cases. The steps is:

1. Classify the actors involved to produce Unadjusted Actor Weight (UAW) scores.

In this system actors interact through GUI or web pages, so that classified in complex.

Table 5. Count Value Unadjusted Actor Weight (UAW)

Actor Type	Description	Weighting Factor	Number	Result
Simple	External System with well-defined API	1	-	-
Average	External system using a protocol based interface as TCP/IP	2	-	-
Complex	Human interacts through Graphic User Interface (GUI) or Web Page	3	6	18
Unadjusted Actor Weight Total (UAW)				18

2. Classify the use case in the use case diagram to generate the value of Unadjusted Use Case Weighting (UUCW). Based on the results of the analysis, it is classified into 7 simple use cases and 10 average use cases.

Table 6. Count Value Unadjusted Use Case Weighting

Use Case Type	Description	Weighting Factor	Number	Result
Simple	1 – 3 transactions	5	7	35
Average	4 – 7 transactions	10	10	100
Complex	> 7 transactions	15	0	0
Unadjusted Use Case Weight Total (UUCW)				135

3. Finding the Unadjusted Use Case Points (UUCP) value by summing the UAW and UUCW values, so the UUCP value of Software information systems sales of goods is $18 + 135 = 153$.
4. Calculating the value of technical factors to obtain Technical Complexity Factor (TCF) value as shown in table 7.

$$TCF = 0.6 + (0.01 * TF) = 0.6 + (0.01 * 52,28) = 1.12$$

Table 7. Count Value Technical Factors (TF)

Factor Number	Description	Weight	Minimum Value (O) (0–5)	Likely Value (M) (0–5)	Maximum Value (L) (0–5)	Expected Value (O+4M+L)/6	UCP-PERT Weighted Value
T1	Distributed Systems	2,0	3	5	5	4,66	9,32
T2	Response time or throughput performance objectives	1,0	3	4	5	4	4,0
T3	End-user online efficiency	1,0	4	4	5	4,16	4,16
T4	Complex internal processing	1,0	2	4	5	3,83	3,83
T5	Reusability of code	1,0	2	2	4	2,33	2,33
T6	Easy to install	0,5	4	5	5	4,83	2,41
T7	Ease of use	0,5	4	4	5	4,16	2,08
T8	Portability	2,0	3	4	5	4	8,0
T9	Ease of change	1,0	2	3	5	3,16	3,16
T10	Concurrency	1,0	1	3	4	2,83	2,83
T11	Special security objectives included	1,0	1	2	3	2,0	2,16
T12	Direct access for third parties	1,0	4	5	5	4,83	4,83
T13	Special user training required	1,0	3	3	5	3,33	3,33
Nilai Tehnical Factor (TF)							52,28

5. Calculate the environmental factor value to obtain the value of Environmental Complexity Factor (ECF) as shown in table 8.

$$ECF = 1.4 + (- 0.03 * EF) = 1.4 + (- 0.03 * 22,74) = 0.72$$

Tabel 8. Count Value Environmental Factor (EF)

Factor Number	Description	Weight	Minimum Value (O) (0–5)	Likely Value (M) (0–5)	Maximum Value (L) (0–5)	Expected Value (O+4M+L)/6	UCP-PERT Weighted Value
E1	Familiarity with system development process being used	1,5	3	4	5	4	6
E2	Application experience	0,5	2	4	5	4	1,92
E3	Object-oriented experience	1,0	3	5	5	4,66	4,66
E4	Lead analyst capability	0,5	3	4	5	4	2
E5	Motivation	1,0	2	3	5	3,16	3,16
E6	Requirements stability	2,0	3	4	5	4	8
E7	Part time staff	-1,0	0	0	0	0	0
E8	Difficulty of programming language	-1,0	2	3	4	3	-3
Nilai Environmental Factor (EF)							22,74

6. UUCP, TCF and EF values are used to calculate Use Case Points value by the formula:

$$UCP = UUCP * TCF * EF = 153 * 1.12 * 0.72 = 123,4$$

7. Based on the research of Subriadi et al [9] for business applications in Indonesia, known value of Effort Rate (ER) is 8.2 man-hours.

8. Effort application obtained by using the formula:

$$\text{Effort} = UCP * ER = 123,4 * 8.2$$

$$\text{Effort} = 1,011 \text{ man-hours}$$

5. Conclusion

The combination method of Use Case Points (UCP) with 3 Points PERT provide solutions to estimation problem, particularly on Technical Complexity Factor (TCF) and Environmental Complexity Factor (ECF) factors. The value of UCP with 3 point PERT in software development of Madrasah Istiqlal Foundation is 123,4 and effort value for the business domain application is 1,011 man-hours. The results of effort estimation software use a combination of UCP methods with 3 points PERT approaching the value of the UCP method (can be greater or smaller) so that the results of this combination method cannot be said to be better or worse than other methods (especially UCP) until there is further research which is more accurate to answer this problem.

Acknowledgement

The authors would like to thank all member Yayasan Komunitas Open Source who have become participants in this research. The authors also would like to thank our family who provided the endless love and tireless support during the process.

References

- [1] S. McConnell, *Software Estimation, Demystifying the Black Art*. Washington, 2006.
- [2] Standish Group, "Chaos Reports," 2018.
- [3] A. Trendowicz, *Software Cost Estimation, Benchmarking, and Risk Assessment*. Springer-Verlag Berlin Heidelberg, 2013.
- [4] S. Nageswaran, "Test Effort Estimation Using Use Case Points," no. June, 2001.
- [5] E. R. Carroll, "Estimating Software Based on Use Case Points," 2005.
- [6] B. Anda, "Comparing Effort Estimates Based on Use Case Points with Expert Estimates," pp. 1–13, 2002.
- [7] G. Karner, "Resource Estimation for Objectory Projects," 1993.
- [8] wikipedia.org, "Program Evaluation and Review Technique (PERT)." [Online]. Available: <http://en.wikipedia.org/wiki/PERT>.
- [9] S. Subriadi, "Critical Review of The Effort Rate Value In Use Case Point Method for Estimating Software Development Effort," *J. Theor. Appl. Inf. Technol.*, vol. 59, no. 3, pp. 735–744, 2014.