IOP Conf. Series: Materials Science and Engineering **434** (2018) 012062 doi:10.1088/1757-899X/434/1/012062

Estimated software measurement base on use case for online admission system

D Kurniadi[1]*, A Mulyani[1], Y Septiana[1] and H Aulawi[2]

[1] Department of Informatics, Sekolah Tinggi Teknologi Garut, Jalan Mayor Syamsu 1, Garut 44151, Indonesia
[2] Department of Industrial Engineering, Sekolah Tinggi Teknologi Garut, Jalan Mayor Syamsu 1, Garut 44151, Indonesia

*dede.kurniadi@sttgarut.ac.id

Abstract. This article aims to measure the size of the software Online Admission System using the method of Use Case Point (UCP). UCP method can help developers in the estimation of factor productivity development of software projects that include estimated time work, human resource needs, and the estimated cost of creating software. This method has the ability in providing the effort estimation approach which is needed to do a software development project. The measurement is carried out based on the number and complexity of software projects by analyzing the actor, use case, and a variety of technical factors as well as environmental factors. The result of the measurement is the software development project of online admission system belongs the medium category with the acquisition value of UCP 103.86, estimated cost 277,168,750 IDR and time work 1,454 hours

1. Introduction

Technological development has supported humans to improve their quality of life [1], one of which is information technology [2]. One of the factors that affect the success of the system development project is the optimal role of project leaders in the planning and execution of software development projects. Most software development projects can be said to have failed. One of the causes is because they cannot meet the set targets, such as insufficient software timing, inadequate human resource requirements, inappropriate prices, and software quality which is not according to user needs and expectations. Appropriate software size analysis and measurement can assist in solving software development project failure problems, especially in estimating software project development productivity factors, one of which can be done by measuring software size using Use Case Point (UCP) method, which is one of the approved software metrics to measure software size based on functionality [3–5].

Use Case Point is a method that can predict the estimated effort required to create a project based on the number and complexity of the use case owned by the software project. The advantages of the UCP method that can provide an estimation of effort close to the actual evaluation resulting from the software development experience include estimation of working time, human resource requirements, and development cost estimation [6,7]. This study aims to provide an overview of the measurement and analysis of UCP method usage in measuring the software development project. The case presented is the analysis and measurement of online admission system software project developed at Sekolah Tinggi Teknologi Garut.

Content from this work may be used under the terms of the Creative Commons Attribution 3.0 licence. Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.
Published under licence by IOP Publishing Ltd

2. Software measurement method

2.1. Related work of use case point method

The use case point (UCP) method is a method that can provide the effort estimation required to create a project development based on the number and complexity of the use case that belongs the software project [3]. This UCP method is significant for estimating effort, where the value of estimated effort is obtained from multiplication between UCP value and Effort Rate (ER) value [8]. The value of the Effort Rate (ER) is the value used as a variable to calculate effort. In other words, UCP is a method that can analyze the actor, use case, and various technical and environmental factors to become an equation [9]. The advantages of the use case point method that can provide estimation that is almost close to the actual estimation resulting from the experience of making or developing software. This is proven by several studies that have been done before and resulted in a statement that, UCP has a deviation of 6% [10], 9% [11], 16% [12], and other has a deviation of 20% [13].

2.2. Seven technique calculation use case point

The steps the perform the use case point calculation process are carried out by those formulated by Karner [3], which is divided into seven stages of calculation as follows:

2.2.1. Calculate Unadjusted Actor Weights (UAW). The first step in this stage is to define Unadjusted Actor Weights (UAW) (based on the level of complexity) as Simple with weight score 1 if API defines it, Average weight score 2 if it is interacting via TCP / IP Protocol or Complex weight score 3 if it interacts with GUI or Web Page.

$$\text{UAW} = \sum (\#\text{Actors} * \text{Weight Factor}) \quad (1)$$

2.2.2. Calculate Unadjusted Use Case Weights (UUCW). The way to calculate UUCW that is, each use case is divided into 3 groups namely Simple of weight score 5 if using less than 3 transactions/entities in database, Average weight score 10 if using 4 to 7 transactions/entities in database, and Complex weight score 15 if using more than 7 transactions/entities in the database, depending on the number of transactions performed.

$$\text{UUCW} = \sum (\#\text{Use Cases} * \text{Weight Factor}) \quad (2)$$

2.2.3. Calculate Unadjusted Use Case Point (UUCP). To calculate UUCP values, derived from summing UAW and UUCW values.

$$\text{UUCP} = \text{UAW} + \text{UUCW} \quad (3)$$

2.2.4. Calculate Technical Complexity Factor (TCF). The next step is to calculate the Technical Complexity Factor that will be used to calculate UCP. The weight of each score multiplies the scores in the technical factor. The weight of the score from the numbers 0 to 5 is given to each factor depends on how much the influence of the factor is. 0 means no influence, 3 means average, and 5 means give a significant influence. The multiplication result of score and weight is then summed to obtain the total Technical Factor (TF), which is then used to obtain Technical Complexity Factor (TCF).

$$\text{TCF} = 0.6 + (0.01 * \text{TFactor}) \quad (4)$$

2.2.5. Calculate Environmental Factor (EF). To calculate the value of UCP, the required value of EF obtained based on environmental factors that influence directly taking action against the working of software development projects. The scores on the environmental factor are multiplied by each weighted score. The weighted score given starts from the numbers 0 to 5 on each factor depends on how big the influence of the factor. 0 means no influence, 3 means average, and 5 means give a big influence. The

result of the multiplication of the score and weight are then summed to obtain the total Environmental Factor (EF).

$$EF = 1.4 + (-0.03 * EFactor) \quad (5)$$

2.2.6. Calculate Use Case Point (UCP). A score of Use Case Point (UCP) is obtained through multiplication of UUCP, TCF, and EF scores from the previous calculation process.

$$UCP = UUCP * TCF * EF \quad (6)$$

2.2.7. Calculate Estimate effort (E) in person-hours. After the calculation process from stage 1 to 6 is completed, the last calculation step is, calculate the effective effort (E) in person-hours (PH) by multiplying the specific score of Person Hour per UCP (PHperUCP) with the UCP from formula (6).

$$E = UCP * PHperUCP \quad (7)$$

The value of Effort rate is the ratio of the number of hours of a person per use case point based on projects in the past. If the project is a new project and no historical data which has been collected, then a score between 15 and 30 is used. However, when the project first generally used the number 20 [9].

3. Case study explanation

3.1. Use case Online Admission System

Online Admission System, in general, is part of the academic information system that exists in college, this system is useful in the management of new student enrolment. In the implementation of the Online Admission System, every college may have different business processes because it is tailored to the needs of each institution. As for the case of software measurement based on a use case to be used, that is by using Online Admission System in Sekolah Tinggi Teknologi Garut, along with the explanation of Use Case as in figure 1.

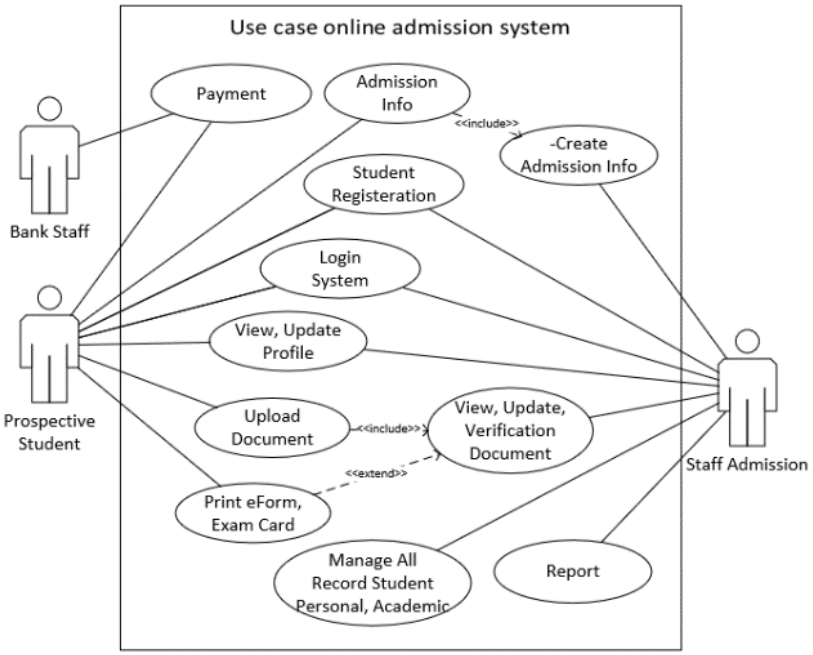


Figure 1. Use case diagram online admission system.

In figure 1, use case diagram online admission system consists of 3 actors namely prospective student, staff admission, and staff bank, meanwhile for the use case consists of 11 activities, they are:

- Admission info, use case activity which consists of acceptance schedule and information.
- Create admission info, use case activity to create schedule and information by the admission staff.
- Payment, use case activity to pay new student's registration that is done by the prospective student.
- Student registration, use case activity for online admission system account registration.
- Login system, use case activity to log in the system after account registration is verified.
- View, update profile, activity to fill in personal and academic activities of the prospective student.
- Upload document, use case activity to upload a required document for registration.
- View, update, verification document, use case activity in checking and validation processes of documents uploaded by the prospective student.
- Print form, exam card, use case activity to proceed the data and exam card printing.
- Manage all record student, use case activity to manage all the data of students who have done the online registration.
- The report, use case activity to report from the online admission system.

3.2. Step by step measurement process

3.2.1. Calculate Unadjusted Actor Weights (UAW). Based on the identification results of figure 1. All actors are categorized as complexes because they relate to Web Page, so they have weight 3, then the calculation result with the formula (1), they will generate the score UAW = 9, based on multiplication of weight score with a number of actors.

3.2.2. Calculate Unadjusted Use Case Weights (UUCW). The next step identifies 11 use case activities in figure 1 then classifies the complexity of each use case based on simple, average, or complex categories and calculates by using formula (2), so that the total score of UUCW = 85, the details can be seen in table 1.

Table 1. Identification Process UUCW.

Use case type	**Weight**	**#Use Case**	***∑ (#Use Cases * Weight)***
Simple	5	7	35
Average	10	2	20
Complex	15	2	30
Total UUCW			85

3.2.3. Calculate Unadjusted Use Case Point (UUCP). To calculate UUCP score using formula (3) by summing UAW score with UUCW score from previous calculation stage, UUCP = 9 + 85, score result is UUCP = 94.

3.2.4. Calculate Technical Complexity Factor (TCF). The TCF Calculation Process is identified based on the technical parameters of the use case, with its calculations and the determination of the weights presented in table 2.

Table 2. Calculation of technical complexity factor.

Technical Factor	Weight	Score	∑ (W* S)
T1-Distributed System Required	2	3	6
T2-Response Time is Important	1	5	5
T3-End User Efficiency	1	5	5
T4-Complex Internal Processing Required	1	1	1
T5-Reusable Code	1	2	2
T6-Installation easy	0.5	3	1.5
T7-Usability	0.5	4	2
T8-Portability	2	1	2
T9-Easy to change	1	4	4
T10-Highly concurrent	1	3	3
T11-Custom security	1	3	3
T12-Dependence on third-party code	1	1	1
T13-User training	1	1	1
Total Technical Complexity Factor (TFactor)			36.5

After determining the weighting of each factor then calculated using formula (4) based on TFactor in table 7, the calculation is TCF = 0.6 + (0.01 * 36.5), the result is TCF = 0.965.

3.2.5. Calculate Environmental Factor (EF). The detailed Environmental Factor calculations are presented in Table 3.

Table 3. Calculation environmental factor.

Environmental Factor	Weight	Score	∑ (W*S)
E1-Familiarity with PHP Framework	1.5	2	3
E2-Application Experience	0.5	2	1
E3-OOP Concept	1	3	3
E4-Leader Capabilities	0.5	3	1.5
E5-Strong Motivation	1	4	4
E6-Stable Requirements	2	3	6
E7-Part Time Workers	-1	5	-5
E8-Difficult Programming Language	-1	5	-5
Total Environmental Factor (EFactor)			8.5

Next step calculating the score of EF with formula (5), EF = 1.4 + (- 0.03 * 8.5), the result is EF = 1.145.

3.3.6. Calculate Use Case Point (UCP). Calculation of the UCP score is obtained based on formula (6) by multiplication of UUCP, TCF, and EF scores, then the result of UCP = 94*0.965*1.145 equals to 103.863 rounded to 103.86 for its UCP score.

3.3.7. Calculate Estimate effort (E) in person-hours. The final step is to calculate the estimated effort using formula (7). The score Person Hour per UCP (PHperUCP) is categorized as simple for the range of 1-20 hours, complex 21-40 hours, very complex more than 41 hours. Based on an interview with Project Leader, to make online admission system software to be included as a simple category with PHperUCP value = 14 hours, then EF value = 103.86*14, the result is EF = 1,454.04 hours and rounded to 1,454 hours.

4. Result and discussion

To simplify the process of calculating our software metrics, we categorize Use Case Point (UCP) values based on four software size categories in the previous study: Small if the score of UCP is less than 99,

Medium if the score of UCP is between 100 to 299, Large if the score of UCP is between 300 to 799, and Extreme if the score of UCP is more than 800 [14,15]. Based on the results of the calculations software size for Online Admission System case study, in the sixth step, it was obtained the UCP calculation of 103.86. Therefore, it can be concluded that the development of the Online Admission System software belongs to the Medium because the score is between 100 to 299. Meanwhile, to obtain the time prediction estimation of Online Admission System software development, it can be calculated based on working hour regulation in each region because sometimes there are institutions that apply time of working hours more than or less than regulation specified in general, but we assume the average working time in Garut is following the general time regulation of 5 working days per week with 8 hours per day. Based on the calculation in step seven using the formula (7), the estimated value of effort (EF) as the estimation effort for software development is 1,454 hours. If it is based on 8 hours working hour per day, then the estimated time effort is 1,454 hours divided by 8 hours so that it has a time of 181.75 days.

If it is calculated from the financial side based on information obtained from the project leader, it is stated that each IT worker has a different salary according to his expertise, with average for men/day is 1,525,000 IDR (the project holder determines this value based on interview results), so the estimated cost can be calculated (181.75 days * 1,525,000 IDR = 277,168,750 IDR). The result of the Online Admission System software development project has a project value of about 277,168,750 IDR.

5. Conclusions

Measurement of software size with software metrics such as Use Case Point measurement is based on the measurement of internal product attributes that can be measured only by the entity itself. Also, Use Case Point is the product measures part of the measurement of software regarding functionality where the functionality provided to the user. Measurements with Software size can reflect effort, cost, and productivity.

Measurement of software size using Use Case Point is beneficial in which software size can be measured based on use case diagram. Regarding measurement of online admission system software, the measurement indicates that the project has a medium software size with Use Case Point (UCP) score = 103.86 and has an estimated business to be developed at 1,454 hours and has a development cost of 277,168,750 IDR. Measuring the size of software with Use Case Points can help developer or management to make better decisions on how to handle projects, how long to develop and how much to invest. Also, measurement using the Use Case Point method, the determination of time estimation effort, the project value, and the number of people who are involved in the project can also be measured.

References

[1] Ramdhani M A, Aulawi H, Ikhwana A and Mauluddin Y 2017 Model of green technology adaptation in small and medium-sized tannery industry *Journal of Engineering and Applied Sciences* **12** pp 954–62

[2] Pamoragung A, Suryadi K and Ramdhani M A 2006 Enhancing the implementation of e-Government in indonesia through the high-quality of virtual community and knowledge portal *Proceedings of the European Conference on e-Government, ECEG* (Marburg, Germany) pp 341–8

[3] Karner G 1993 Resource Estimation for Objectory Projects *Object. Syst. SF AB* **17**

[4] Fenton N and Bieman J 2014 *Software metrics: a rigorous and practical approach* (CRC Press)

[5] Nassif A B, Capretz L F and Ho D 2014 Calibrating use case points *Companion Proceedings of the 36th International Conference on Software Engineering* pp 612–3

[6] Anda B, Dreiem H, Sjøberg D I K and Jørgensen M 2001 Estimating Software Development Effort Based on Use Cases — Experiences from Industry pp 487–502

[7] Mohagheghi P, Anda B and Conradi R Effort estimation of use cases for incremental large-scale software development *Proceedings. 27th International Conference on Software Engineering, 2005. ICSE 2005.* (IEEe) pp 303–11

[8] Usman M, Mendes E, Weidt F and Britto R 2014 Effort estimation in agile software development *Proceedings of the 10th International Conference on Predictive Models in Software Engineering - PROMISE '14* (New York, New York, USA: ACM Press) pp 82–91
[9] Clemmons R K 2006 Project Estimation With Use Case Points *J. Def. Softw. Eng.* 18–22
[10] Nageswaran S 2001 Test Effort Estimation Using Use Case Points *Quality Week* pp 1–6
[11] Carroll E R 2005 Estimating software based on use case points *Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications* pp 257–65
[12] Korolija N, Bojic D and Popovic J 2015 Analysis of task effort estimation accuracy based on use case point size *IET Softw.* **9** 166–73
[13] Anda B 2002 Comparing effort estimates based on use case points with expert estimates *Empir. Assess. Softw. Eng. (EASE 2002), Keele, UK*
[14] Iskandar K, Gaol F L, Soewito B and Hendric S W H L 2016 Software Size Measurement of Knowledge Management Portal with Use Case Point *2016 International Conference on Computer, Control, Informatics and Its Applications*
[15] Kurniadi D, Sasmoko, Warnars H L H S and Gaol F L 2017 Software size measurement of student information terminal with use case point *2017 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)* (Phuket, Thailand: IEEE) pp 164–9