ICIC Express Letters
Volume **13**, Number **9**, September **2019**
 ISSN 1881-803X
pp. **781–788**

MEASUREMENT OF SIB MOBILE APPLICATION USING USE CASE POINTS METHOD

David David[1], Harco Leslie Hendric Spits Warnars[1], Ford Lumban Gaol[1]
Edi Abdurachman[1] and Benfano Soewito[2]

[1]Computer Science Department, BINUS Graduate Program – Doctor of Computer Science
[2]Computer Science Department, BINUS Graduate Program – Master of Computer Science
Bina Nusantara University
Jl. K. H. Syahdan No. 9, Kemanggisan, Palmerah, Jakarta 11480, Indonesia
{ david01; fgaol; edia; bsoewito }@binus.edu; spits.hendric@binus.ac.id

Received February 2019; accepted May 2019

Abstract. *The number of mobile applications is increasing every year because of the many needs of users. The project manager or project owner really needs a good calculation so that the use of resources can be done more efficiently. Use Case Points (UCP) is one method that can be used to measure the size of a software based on a use case diagram. Use case diagrams are usually made before the application begins to be developed into the application. In the calculation of UCP, there are 2 additional factors carried out: Technical Complexity Factor (TCF) and Environmental Factor (EF). This paper presents measurement software using use case points on the SIB mobile application. This application has never been measured both in terms of cost and size of the project. The results of the use case points method produce a UCP value of 129.86 and Effort Estimation (Effort) value of 1298.6 hours. With these results, the project manager gets a picture of the project so that the efficiency of the resources can be carried out properly.*
Keywords: Software metric, Use case points, Mobile application, Software size, Software cost

1. **Introduction.** In today's technology era, many new mobile applications are emerging. It is estimated that there are approximately 2,000,000 applications on the IOS platform that can be seen in the market [1]. This number is higher on the Android platform, which is around 2,100,000 [1]. This is because many smartphone users often use their smartphones for business and daily needs. The presence of applications on mobile also makes it easier for users to increase work productivity and reduce the cost needed. In the example of the business section, companies can change brochures on paper to e-brochures, thereby reducing printing costs. In addition, the information provided in the brochure can be replaced if needed.

Building a software needs a right cost estimation, because making software often exceeds the required cost [2]. Use case diagram is a technique that is widely used to draw business processes that are currently or will be developed [3]. Use case diagrams can be used for all software with large or small scope. Use case diagram is very suitable for object-oriented based systems because use case diagrams can divide all features based on objects in the system [4].

Suzuki Interactive Brochure (SIB) is one of the mobile applications on the Android platform that uses e-brochures to display car information. This application is intended to help employees promote cars, as well as being used to track the activity of each employee. This SIB application is created using C# Language and uses an object-oriented concept, but this application has never used software size calculation before. Using Content Management System (CMS) also can simplify and accelerate the project development

DOI: 10.24507/icicel.13.09.781

because it can change the content dynamically without changing the technical things in application [5].

It becomes very important for the project manager or project owner to calculate the size of the application that will be built so that they can prepare the resources needed to make the application. One way to calculate the amount of software is to use use case points method, where the software will be calculated based on each existing use case. With this calculation, it is expected to provide an overview of the estimated size of the software that is being made and can be used as a reference for further software development.

This paper will focus on the use of use case points where there are 7 steps taken to determine the approximate size of the software.

2. **Literature Review.** Use Case Points (UCP) was introduced by Karner in 1993 [6] where the purpose of this UCP was to estimate the size of a software. The concept of UCP is similar to function points, but UCP uses diagrams and uses object-oriented systems.

UCP calculations are divided into 7 steps which can be explained as follows.

1) Unadjusted Use Case Weights (UUCW)

   This stage is the first stage where each use case is labeled with simple, average, or complex based on the number of transactions in the use case. Label distribution is based on the number of transactions carried out. Simple label if the number of transactions is less than 3, average if the amount is between 3 and 7 transactions, and complex if there are more than 7 transactions. Each label has a weight of 5 for simple, 10 for average, and 15 for complex. UUCW is obtained by summing all the weights in each use case. The calculation formula can be seen below.

$$\text{UUCW} = \sum(\#\text{Number of Use Cases} \times \text{Weight Factor}) \tag{1}$$

2) Unadjusted Actor Weight (UAW)

   At this stage labels are carried out for each actor in the use case where the label is similar to UUCW with the following conditions.

Table 1. Unadjusted actor weight

Actor complexity	Example	Weight factor
Simple	A system with defined API	1
Average	A system interacting through protocol	2
Complex	A user interacting through GUI	3

   The total of UAW is the sum of all weights for each actor. The calculation formula can be seen below.

$$\text{UAW} = \sum(\#\text{Number of Actors} \times \text{Weight Factor}) \tag{2}$$

3) Unadjusted Use Case Points (UUCP)

   At this stage UUCP is calculated by summing the results of the Unadjusted Use Case Weight (UUCW) and Unadjusted Actor Weight (UAW). The formula for the calculation can be seen below.

$$\text{UUCP} = \text{UUCW} + \text{UAW} \tag{3}$$

4) Technical Complexity Factor (TCF)

   At this stage TCF calculations are carried out by looking at 13 factors, each of which has a predetermined weight. The 13 factors can be seen in Table 2.

Table 2. Technical complexity factor

Factor	Description	Weight
T1	Distributed system	2.0
T2	Response time or throughput performance objectives	1.0
T3	End user efficiency	1.0
T4	Complex internal processing	1.0
T5	Code must be reusable	1.0
T6	Easy to install	0.5
T7	Easy to use	0.5
T8	Portable	2.0
T9	Easy to change	1.0
T10	Concurrent	1.0
T11	Includes special security objectives	1.0
T12	Provides direct access for third parties	1.0
T13	Special user training facilities are required	1.0

Each factor will be given a value between 0 (irrelevant) to 5 (very important). Then the value will be multiplied by each weight and added to get the Technical Factor (TF). Calculation of TF can be calculated using the formula below.

$$\text{TF} = \sum(\text{Weight} \times \text{Rate Value}) \tag{4}$$

The calculated TF value will be used to get the TCF value using the formula below.

$$\text{TCF} = 0.6 + (0.01 \times \text{TF}) \tag{5}$$

5) Environmental Factor (EF)

At this stage the calculation of the environment is carried out which affects the size of the software. This calculation is similar to TCF but with different criteria and weights and can be seen in Table 3.

Table 3. Environmental factor

Factor	Description	Weight
F1	Familiar with the project model that is used	1.5
F2	Application experience	0.5
F3	Object-oriented experience	1.0
F4	Lead analyst capability	0.5
F5	Motivation	1.0
F6	Stable requirements	2.0
F7	Part-time staff	−1.0
F8	Difficult programming language	−1.0

Each factor is given the same value as TCF, which is 0 (irrelevant) to 5 (very important). Then the value will be multiplied by each weight and added to get EFactor. EFactor values can be calculated using the formula below.

$$\text{EFactor} = \sum(\text{Weight} \times \text{Rate Value}) \tag{6}$$

The calculated EFactor value will be used to get the value from EF using the formula below.

$$\text{EF} = 1.4 + (-0.03 \times \text{EFactor}) \tag{7}$$

6) Use Case Points (UCP)
   At this stage, the value of Use Case Points (UCP) is calculated by multiplying the values obtained from UUCP, TCF, and EF. The formula for calculating UCP can be seen below.

$$\text{UCP} = \text{UUCP} \times \text{TCF} \times \text{EF} \tag{8}$$

7) Effort Estimation (Effort)
   At this stage the value for the Effort Estimation (Effort) is calculated by multiplying the UCP value by Person Hour per UCP (PH). Effort estimation can be calculated using the formula below.

$$\text{Effort} = \text{UCP} \times \text{PH} \tag{9}$$

3. **Research Method.** SIB is an Android application created to facilitate employees in promoting goods. This application provides features for viewing brochures, calculating credit simulations, and tracking records for each customer that has been visited. This application makes employees often visit prospective buyers to offer products that are sold in order to get a bonus. The examples of application display can be seen in Figure 1. Figure 1 displays pages for car selection, credit simulation, location search, and visitor records.

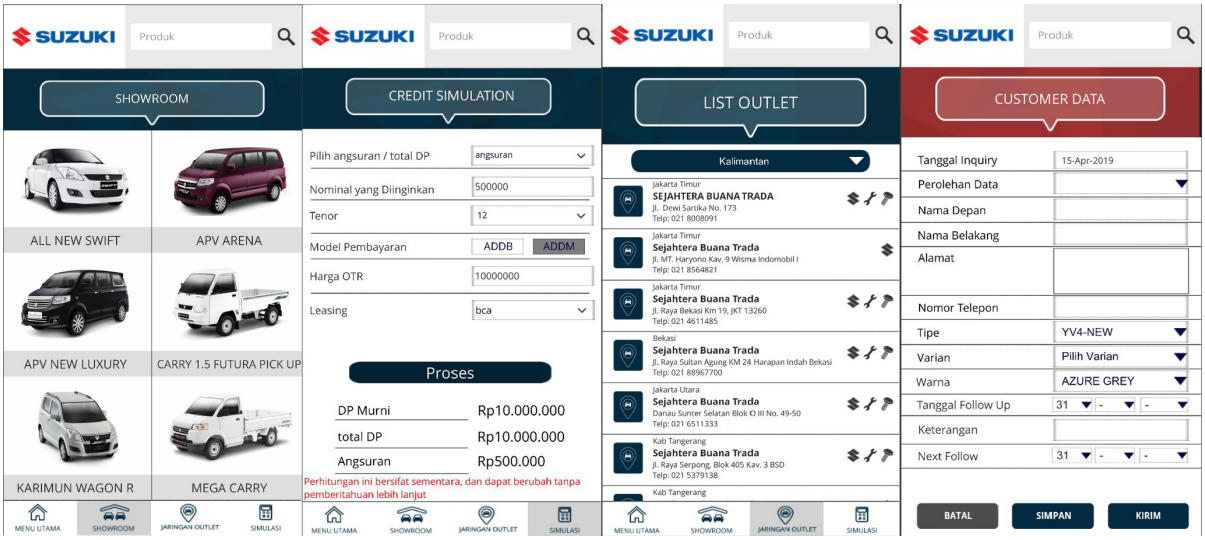

FIGURE 1. Screenshot of SIB mobile application

Use case diagram from SIB can be seen in Figure 2, where there are 2 actors: users and admin. There are also 16 use cases where 7 of them are extended use cases.

Here is an explanation for each use case.

1) Update Data Master: Admin will be able to update master data and will update all data across application.
2) Login: This page will be used to log in to the application for users and admin. ID and password are required to log in.
3) Show Car: Users will be shown the list available cars.
4) View Detail Car: Users can see the detail of the chosen cars such as body, feature, and detail parts.
5) Show About: Users can see the company journey.
6) Show Location: Users can search and view the service location center.
7) View Detail Location: Users can see the detail of the location and call the center.
8) Show Simulation Credit: Users can simulate the credit cost for buying the car.
9) Save Simulation: Users can save the simulation and share it to friends.
10) Show Brochure: Users can show the traditional brochure.

11) Open Customer Tracking: System will show list customer and their last report.
12) Add New Customer: Users can add new customer.
13) Add New Report: Users can update the existing customer report and send it to the headquarter.
14) Open Settings Interface: System will open the settings interface.
15) Change Password: Users can change the password.
16) Change Credit Parameter: Users can add or change credit parameter for simulation.

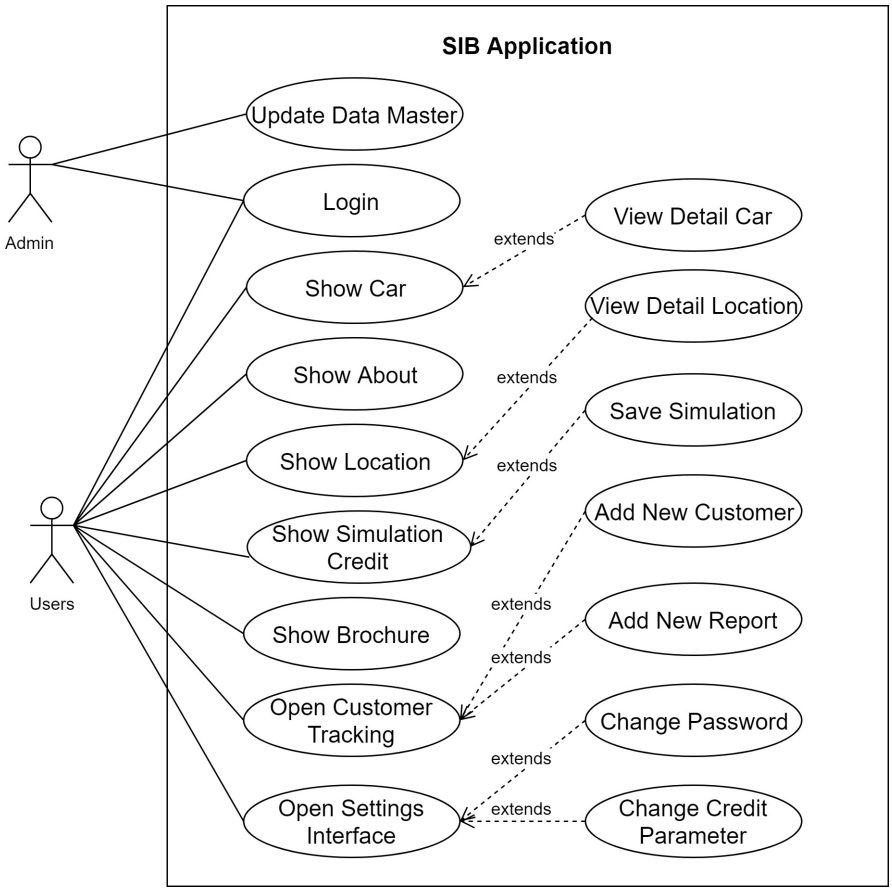

FIGURE 2. Use case diagram of SIB mobile application

SIB application has never been measured using any method. To calculate the cost and size of the software, the use case points are used to estimate the cost and size of the software.

The first step is to calculate the value of the Unadjusted Use Case Weight (UUCW). The calculation results can be seen in Table 4, where each use case is filled with a value of 5, 10 or 15 depending on the level of complexity of the use case. Based on Formula (1), the result from UUCW is 130.

The second step is calculating Unadjusted Actor Weight (UAW). Because all actors use the GUI interface, the two actors fall into the complex category and have a weight of 3 for each actor. Based on Formula (2), the UAW value is $2 \times 3 = 6$.

After obtaining UUCW and UAW values, the next process is to calculate UUCP by using Formula (3) and the UUCP value is $130 + 6 = 136$.

The fourth step is the calculation of the Technical Complexity Factor (TCF) where there are 13 factors that will be given a value based on each existing weight. The TCF calculation results can be seen in Table 5.

Based on Formula (5), the technical complexity factor value is $0.6 + (0.01 \times 53) = 1.13$.

The fifth step is to calculate the value of the environmental factor with details that can be seen in Table 6.

Table 4. Unadjusted use case weight result

Use case	Category	Weight
Update Data Master	Complex	15
Login	Simple	5
Show Car	Average	10
View Detail Car	Complex	15
Show About	Simple	5
Show Location	Simple	5
View Detail Location	Average	10
Show Simulation Credit	Complex	15
Save Simulation	Simple	5
Show Brochure	Simple	5
Open Customer Tracking	Simple	5
Add New Customer	Simple	5
Add New Report	Average	10
Open Settings Interface	Simple	5
Change Password	Simple	5
Change Credit Parameter	Average	10
Unadjusted Use Case Weight (UUCW)		130

Table 5. Technical complexity factor result

Factor	Description	Weight	Value	Weight × Value
T1	Distributed system	2.0	3	6.0
T2	Response time or throughput performance objectives	1.0	5	5.0
T3	End user efficiency	1.0	4	4.0
T4	Complex internal processing	1.0	3	3.0
T5	Code must be reusable	1.0	4	4.0
T6	Easy to install	0.5	5	2.5
T7	Easy to use	0.5	5	2.5
T8	Portable	2.0	5	10.0
T9	Easy to change	1.0	4	4.0
T10	Concurrent	1.0	1	1.0
T11	Includes special security objectives	1.0	3	3.0
T12	Provides direct access for third parties	1.0	5	5.0
T13	Special user training facilities are required	1.0	3	3.0
Total of Technical Factor (TF)				53.0

Based on Formula (7), the environmental factor value is $1.4 + (-0.03 \times 18.5) = 0.845$.

The sixth step is to calculate the value of Use Case Points (UCP) which is calculated using Formula (8). The results obtained are $136 \times 1.13 \times 0.845 = 129.86$.

The final step is to calculate the value of the Effort Estimation (Effort) by multiplying the UCP value with Person Hour per use case point (PH). Person hour is obtained by calculating the time needed to complete a use case. This value depends on the type of project and the number of people involved. For small projects have a value of 1 to 20, while for the project medium has a value of 21-40, and for large projects has a value of more than 40 [7]. The project manager gives a value of 10 for the project. Based on Formula (9), the Effort value ranges from $129.86 \times 10 = 1298.6$ hours.

Table 6. Environmental factor

Factor	Description	Weight	Value	Weight × Value
F1	Familiar with the project model that is used	1.5	3	4.5
F2	Application experience	0.5	3	1.5
F3	Object-oriented experience	1.0	4	4.0
F4	Lead analyst capability	0.5	5	2.5
F5	Motivation	1.0	3	3.0
F6	Stable requirements	2.0	4	8.0
F7	Part-time staff	−1.0	2	−2.0
F8	Difficult programming language	−1.0	3	−3.0
Total of EFactor				18.5

4. **Result and Discussion.** After calculating all the necessary calculations, the size of the SIB software can be determined by looking at the results of the Use Case Points (UCP) number. Based on Table 7 [7], the size of this application is medium because the UCP value of this SIB is 129.86 and is in the range 100-299 in the medium category.

Table 7. Software size category

Category	Use case points
Small	< 100
Medium	100-299
Large	300-799
Extreme	> 799

To calculate the cost value of this project, we must calculate the amount of salary and the time needed to complete the project. The time needed is obtained from the Effort, which is 1298.6 hours. The working hours calculated are 8 hours per day and 5 days per week, so the time taken is 1298.6/8/5 = 32.47 weeks or around 8.11 months. According to a survey conducted by consultant Persol Indonesia with Kelly Service [8], the average programmer salary per month is IDR 8,000,000 with experience of 3-8 years. Based on these data, it can be calculated that the required cost is 8.11 × 8,000,000 = IDR 64,880,000. The summary of the calculation can be seen in Table 8.

Table 8. Summary of SIB software size measurement

Measurement items	SIB application
Unadjusted Use Case Weight (UUCW)	130
Unadjusted Actor Weight (UAW)	6
Unadjusted Use Case Points (UUCP)	136
Technical Complexity Factor (TCF)	1.13
Environmental Factor (EF)	0.845
Use Case Points (UCP)	129.86
Person Hour per UCP (PH)	10
Software size	Medium
Effort Estimation (hours)	1298.6
Effort Estimation (weeks)	32.47
Effort Estimation (months)	8.11
Average monthly programmer in Indonesia	8,000,000
Project value	64,880,000

5. **Conclusion.** This paper discusses SIB mobile application software size calculation using Use Case Points (UCP). UCP calculates software functionality based on use case diagrams. Based on the results of the calculation, SIB is classified as a medium size with a UCP value of 129.86. The estimated effort required is around 1298.6 hours, equivalent to 8.11 months. And the total cost for making this application is IDR 64,880,000. This data can be combined with other methods such as function points to get values from different aspects for the same project. With the addition of these data, the project manager or project owner can analyze the resources needed for the project so that they can optimize the use of existing resources.

REFERENCES

[1] *Number of Apps Available in Leading App Stores as of 3rd Quarter 2018*, https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores, 2018.
[2] D. Kashyap, D. Shukla and A. K. Misra, Refining the use case classification for use case point method for software effort estimation, *Proc. of Int. Conf. on Recent Trends in Information, Telecommunication and Computing (ITC)*, pp.183-191, 2014.
[3] R. K. Clemmons, Project estimation with use case points, *Journal of Defense Software Engineering*, pp.18-22, 2006.
[4] R. Klimek and P. Szwed, Formal analysis of use case diagrams, *The Computer Science Journal*, vol.11, pp.115-131, 2010.
[5] Sholiq, T. Sutanto, A. P. Widodo and W. 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, 2014.
[6] G. Karner, Resource estimation for objectory projects, *Object. Syst. SF AB*, vol.17, 1993.
[7] IT_BINUS_Department, *BINUS IT Software Development Documentation*, Jakarta, 2016.
[8] A. D. Putera, *How Much is the Salary of Employees Who Work in the Automotive and IT Industry*, https://ekonomi.kompas.com/read/2018/05/02/125944826/berapa-besar-gaji-karyawan-yang-kerja-di-industri-otomotif-dan-it, 2018.