

# Aplikasi Estimasi Biaya Pembangunan *Software* Menggunakan Metode *Function Point Analysis* (FPA) Dengan Platform Android

Hendrawan<sup>1</sup>, Pareza Alam Jusia<sup>1</sup>, Errissya Rasywir<sup>2</sup>, Yovi Pratama<sup>2</sup>

<sup>1</sup>Sistem Informasi, STIKOM Dinamika Bangsa, Jambi, Indonesia

<sup>2</sup>Teknik Informatika, STIKOM Dinamika Bangsa, Jambi, Indonesia

Email: <sup>1</sup>hendrawan.stikom@gmail.com, <sup>2</sup>parezaalam@gmail.com, <sup>3</sup>errissya.rasywir@gmail.com, <sup>4</sup>yovi.pratama@gmail.com

## Abstrak

Sebuah proyek pembangunan aplikasi (*software*) pasti membutuhkan alokasi biaya proyek. Alokasi biaya proyek sangat penting dalam melakukan perencanaan, pengolahan serta manajemen dalam pembangunan *software*. Sehingga perlu mengevaluasi akurasi estimasi biaya tersebut dan dapat digunakan sebagai salah satu indikator penentu keberhasilan suatu pekerjaan pembangunan aplikasi. Penelitian mengenai perbaikan metode estimasi biaya *software* yang sudah dilakukan menunjukkan bahwa metode *Function Point Analysis* (FPA) mempunyai hasil yang baik. Hasil pengujian dengan menggunakan 20 data proyek, nilai RMSE dari harga estimasi dengan harga real dilapangan yang diterima adalah sebesar Rp 2,271,539.69. Sedangkan nilai error yang diperoleh 0.498929 setelah diujikan pada sistem estimasi biaya *software* menggunakan metode *Function Point Analysis* (FPA) berbasis Android. Artinya sangat besar perbedaan atau selisih harga antara nilai estimasi ketika diujicobakan pada data real proyek *software* di lapangan.

**Kata Kunci:** Android, FPA, *Software*, Estimasi Biaya, Aplikasi

## 1. PENDAHULUAN

Sebuah proyek pembangunan perangkat lunak (*software*) membutuhkan alokasi sumber daya dan biaya proyek. Alokasi sumber daya dan biaya proyek merupakan aspek penting perencanaan dan manajemen proyek pembangunan *software* [1], [2]. Dengan demikian akurasi estimasi biaya pembangunan *software* merupakan salah satu penentu keberhasilan suatu proyek [1], [3]. Saat ini terdapat beberapa teknik estimasi biaya perangkat lunak, diantaranya adalah COCOMO dan SLIM dengan kelebihan dan kekurangan masing-masing metode [4], [5]. Teknik tersebut dapat dikelompokkan dalam tiga kategori berdasarkan, yaitu: *judgement* ahli, model algoritma dan *analogi* [6]. Berdasarkan penelitian terdahulu, ternyata ada beberapa kelemahan pada metode algoritmik, antara lain lamanya waktu eksekusi [3], [7]. Meskipun banyak penelitian menghasilkan estimasi biaya perangkat lunak mengacu pada metode algoritmik, namun metode *judgement* ahli lebih banyak digunakan [8]–[10]. Metode ini didasarkan pada pengalaman manajer pada proyek yang serupa, kompetensi, obyektivitas dan persepsi dari ahli [11], [12]. Penelitian mengenai perbaikan metode estimasi biaya pembangunan aplikasi berdasarkan kategori analogi sudah dilakukan seperti pada metode *analogy based estimation* dan menghasilkan nilai yang lebih baik [1], [13]. Namun, pada penelitian yang melakukan estimasi biaya *software* dengan metode model algoritma, menunjukkan bahwa metode *Function Point Analysis* (FPA) lebih baik lagi [2], [10].

Berdasarkan hal tersebut, maka pada penelitian ini melakukan pembangunan aplikasi estimasi biaya pembangunan *software* menggunakan metode *Function Point Analysis* (FPA) berbasis Android [14]. Dengan aplikasi berbasis Android ini, diharapkan aplikasi ini nantinya dapat memudahkan pihak developer dan *client* dalam melakukan negosiasi biaya pembangunan *software* dimanapun dan kapanpun. Selain membangun sebuah *software* yang mampu melakukan estimasi biaya perangkat lunak, pada penelitian ini dilakukan eksperimen riset pada estimasi *Cost Estimation* pada *Software Development* dengan menggunakan parameter evaluasi *Magnitude of Relative Error* (MRE) dan *Mean Magnitude of Relative Error* (MMRE) [15]. Berdasarkan permasalahan tersebut peneliti melakukan implementasi aplikasi estimasi biaya pembangunan *software* menggunakan metode *Function Point Analysis* (FPA) berbasis Android [14] serta melakukan evaluasi dan analisis terhadap output aplikasi estimasi biaya pembangunan *software* menggunakan metode *Function Point Analysis* (FPA) berbasis Android ketika diujicobakan pada data real proyek *software* di lapangan.

## 2. METODOLOGI PENELITIAN

### 2.1 Analisis Sistem Estimasi Biaya *Software*

Estimasi Biaya *Software* adalah proses memprediksi biaya yang diperlukan dalam pengembangan sistem *software* [10]. Perkiraan harga *software* mengacu pada prediksi jumlah biaya, waktu dan staffing yang diperlukan untuk membangun sistem *software* [1]. Banyak pemodelan perkiraan *software* yang berdasarkan ukuran *software* itu sendiri. Berikut beberapa standar ukuran *software* [1]:

1. *Lines of code*: Baris kode program.
2. *Function point*: pengukuran berdasarkan fungsionalitas dari program.
3. *Object point*: pengukuran berdasarkan jumlah objek.
4. *Feature point*: perluasan dari *function point* sebagai kelas baru.
5. *Software science*: pengukuran berdasarkan rumus metrik Halstead.

Estimasi biaya pembangunan *software* dapat diklasifikasikan menjadi metode pemodelan algoritmik dan non algoritmik. Pemodelan algoritmik diturunkan dari analisis *statistical* dari data proyek terdahulu[11]. Contoh pemodelan algoritmik yaitu model linear, model multiplikatif, dan model *power function*.

## 2.2 Function Point Analysis (FPA)

Metode yang digunakan untuk melakukan estimasi biaya *software* dalam penelitian ini adalah FPA. Metode ini digunakan untuk mengukur LOC, *cost*, *effort*, *resource* dan durasi *software*. Berikut cara menghitung FPA antara lain [13]:

### 1. Menghitung Crude Function Points (CFP)

**Tabel 1.** Komponen Penghitungan CFP

No	Komponen	Keterangan
1	External Input (EI)	Interace untuk memasukkan data pada aplikasi
2	External Output (EO)	Output yang dihasilkan
3	External Inquiry (EQ)	Query terhadap data yang tersimpan
4	Internal logical File (ILF)	Logik penyimpanan data
5	External Interface File (EIF)	Komunikasi data pada perangkat/ mesin yang lain

**Tabel 2.** Function Point Complexity Weight

ILF/EIF	RET	1-19	20-50	51+
	1	Low	Low	Average
	2-5	Low	Average	High
	6+	Average	High	High

**Tabel 3.** Komponen Standar Nilai Weight (FTR).

Komponen	Low	Average	High
EI	3	4	6
EO	4	5	7
EQ	3	4	6
ILF	7	10	15
EIF	5	7	10

**Tabel 4.** Komponen CFP dan Nilai Level Kompleksitas

Tipe	Level Kompleksitas									CFP
	Sederhana (low)			Medium (average)			Kompleks (High)			
	DET	FTR	RET	DET	FTR	RET	DET	FTR	RET	
EI	?	3	?	?	4	?	?	6	?	?
EO	?	4	?	?	5	?	?	7	?	?
EQ	?	3	?	?	4	?	?	6	?	?
ILF	?	7	?	?	10	?	?	15	?	?
EIF	?	5	?	?	7	?	?	10	?	?
	<b>TOTAL</b>									?

Pada tabel 4 terdapat nilai yakni *Data Element Types* (DET), *File Type References* (FTR) dan *Record Element Types* (RET).

### 2. Menghitung Relative Complexity Adjustment Factor (RCAF)

Penilaian dengan nilai 0 = Tidak Pengaruh; 1 = Insidental; 2 = Moderat; 3 = Rata-rata; 4 = Signifikan; 5 = *Essential*.

**Tabel 5.** Value Adjusment Factor

Id	Faktor Karakteristik	Bobot	Keterangan
C1	Data Communications	3	Kompleksitas Komunikasi Data
C2	Distributed Function	4	Kompleksitas Pemrosesan Terdistribusi
C3	Performance Objectives	3	Kompleksitas Performance
C4	Heavily used configuration	4	Kompleksitas Konfigurasi
C5	Transaction Rate	3	Frekuensi penggunaan <i>software</i>
C6	On-line Data entry	3	Frekuensi Input Data
C7	End user efficiency	4	Kemudahan Penggunaan Bagi user
C8	On-line Update	3	Frekuensi Update data
C9	Complex Processing	3	Kompleksitas <i>processing</i> data
C10	Reusability	5	Kemungkinan <i>Reusable</i> Kode Program
C11	installation ease	5	Kemudahan dalam Instalasi

C12	<i>operational ease</i>	3	Kemudahan ( <i>back up recovery</i> )
C13	<i>Multiple Sites</i>	3	Software dibuat untuk multi organisasi
C14	<i>Facilitate change</i>	4	Kompleksitas Fleksibel

Karakteristik tersebut dibuat oleh *Function Point Internasional User Group* (IFPUG) atau disebut juga *Total Degree of Influence* (TDI).

### 3. Menghitung *Function Point* (FP)

Menghitung TCA (*Technical Complexity Adjustment*), dengan rumus [13]:

$$TCA = 0.65 + 0.01 * TDI \dots\dots\dots (1)$$

$$FP = CFP \times TCA \dots\dots\dots (2)$$

Untuk mengukur LOC dari FP cukup mengalikan FP dengan nilai faktor produktivitas bahasa pemrograman yang dibuat oleh Capers Jones. Angka 0.65 dan 0.01 adalah konstanta dari *Function Point Internasional User Group* (IFPUG). Rumus FP [13].

### 2.3 Melakukan Analisis Proses

Pada tahap ini dilakukan analisis kebutuhan estimasi biaya pembangunan *software* dengan Metode *Function Point Analysis* (FPA) untuk mendukung proses penilaian tersebut meliputi analisis kebutuhan fungsional dan analisis kebutuhan data. Proses ini juga termasuk memodelkan perhitungan *Function Point Analysis* (FPA) kedalam *source code* sistem yang dibangun. Kemudian *programmer* akan mendaftarkan kebutuhan fungsional dan non fungsional yang terdapat dalam pembangunan sistem Estimasi biaya pembangunan *software* dengan menggunakan Metode *Function Point Analysis* (FPA) sekaligus menentukan rancangan aliran data dan perancangan kelasnya

### 2.4 Memodelkan Sistem dengan UML

Pada tahap ini dibuat pemodelan kebutuhan sistem dengan menggunakan diagram UML. Kegiatan ini akan menghasilkan beberapa pemodelan untuk pembangunan sistem Estimasi biaya pembangunan *software* dalam bentuk *usecase diagram*, *activity diagram* dan *class diagram*.

### 2.5 Membangun Sistem Estimasi Biaya *Software* dengan metode FPA

Setelah dirancang pemodelan sistem estimasi *software* FPA menggunakan berbagai diagram pemodelan seperti yang disebutkan pada sub 2.5 di atas, maka kegiatan dilanjutkan dengan membangun sistem. Sistem atau aplikasi estimasi biaya *software* ini dibangun menggunakan platform android. Hal ini ditujukan untuk memudahkan dalam mengestimasi biaya.

### 2.6 Evaluasi Sistem Estimasi Biaya *Software*

Metode yang digunakan untuk melakukan evaluasi sistem menggunakan nilai RMSE (*root mean squared error*) dan nilai *error*, berikut rumus mencari nilai RMSE dan nilai *error*:

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (p_i - a_i)^2}{n}} \dots\dots\dots (3)$$

$$Error (\%) = \frac{\sum_{i=1}^n \frac{p_i - a_i}{a_i}}{n} \times 100 \% \dots\dots\dots (4)$$

Dengan nilai  $p_i$  adalah nilai dugaan output ulangan ke  $i$ ,  $a_i$  adalah nilai real output ulangan ke  $i$  dan nilai  $n$ =jumlah contoh data. Dalam penelitian ini terdapat 20 data pembangunan perangkat lunak yang diujicobakan pada Sistem Estimasi Biaya *Software* menggunakan metode FPA. Data proyek ini diperoleh dari salah satu *software house* di Kota Jambi.

## 3. ANALISA DAN PEMBAHASAN

### 3.1 Analisis Hasil Sistem Estimasi Biaya dengan FPA

Hasil dari riset ini adalah Sistem Estimasi Biaya dengan FPA yang dibangun menggunakan platform Android. Skema pemodelan Sistem Estimasi Biaya dengan FPA yang dibangun antara lain memiliki rancangan antara lain dapat dilihat pada bagian 3.2 mengenai *Usecase List* Sistem Estimasi Biaya dengan FPA, bagian 3.3 mengenai Diagram Aktivitas dan *usecase* Sistem Estimasi Biaya dengan FPA.

### 3.2 Analisis *Usecase List* Sistem Estimasi Biaya dengan FPA

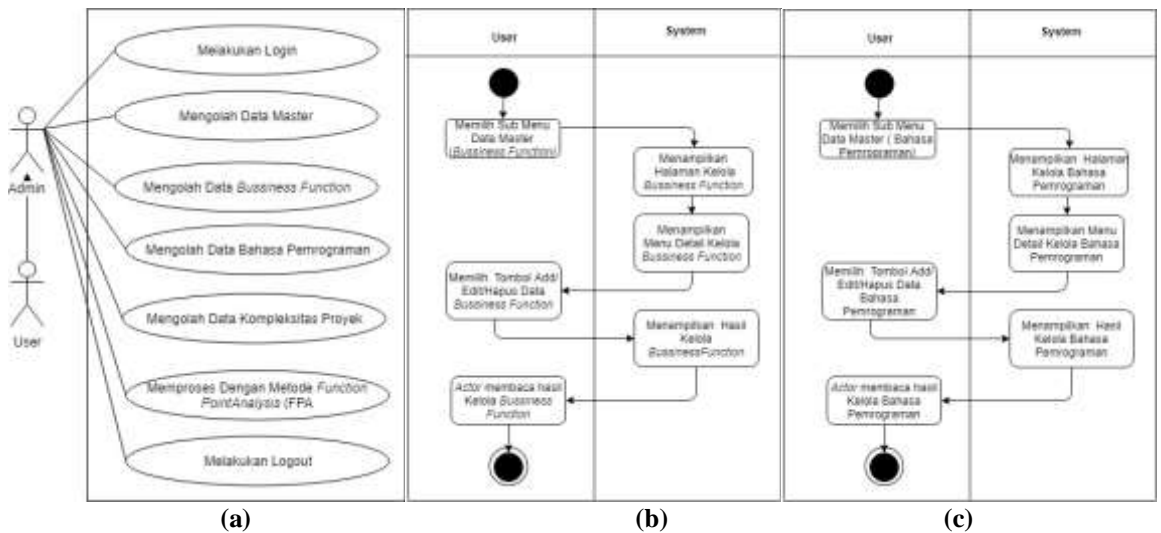
Tabel 6. *Usecase List* Sistem Estimasi Biaya dengan FPA

No	Use Case	Deskripsi
1	Melakukan Login	Dilakukan oleh <i>actor</i> untuk melakukan login sistem
2	Mengolah Data Master	Dilakukan oleh <i>actor</i> untuk Mengolah Data Master
3.	Mengolah Data <i>Bussiness Function</i>	Dilakukan oleh <i>actor</i> untuk Mengolah Data <i>Bussiness Function</i>
4	Mengolah Data Bahasa Pemrograman	Dilakukan oleh <i>actor</i> untuk Mengolah Bahasa Pemrograman
5	Mengolah Data Kompleksitas Proyek	Dilakukan oleh <i>actor</i> Mengolah Data Kompleksitas Proyek

Memproses Dengan Metode <i>Function Point Analysis</i> (FPA)	Dilakukan oleh <i>actor</i> untuk Memproses Dengan Metode <i>Function Point Analysis</i> (FPA)
6 Melakukan Logout	Dilakukan oleh <i>actor</i> untuk melakukan logout sistem

Tabel 6 diatas merupakan list interaksi yang dilakukan oleh user terhadap Sistem Estimasi Biaya dengan FPA. Interaksi yang dilakukan antara lain Melakukan *Login*, Mengolah Data Master, Mengolah Data *Bussiness Function*, Mengolah Data Bahasa Pemrograman, Mengolah Data Kompleksitas Proyek, Memproses Dengan Metode *Function Point Analysis* (FPA) Melakukan *Logout*.

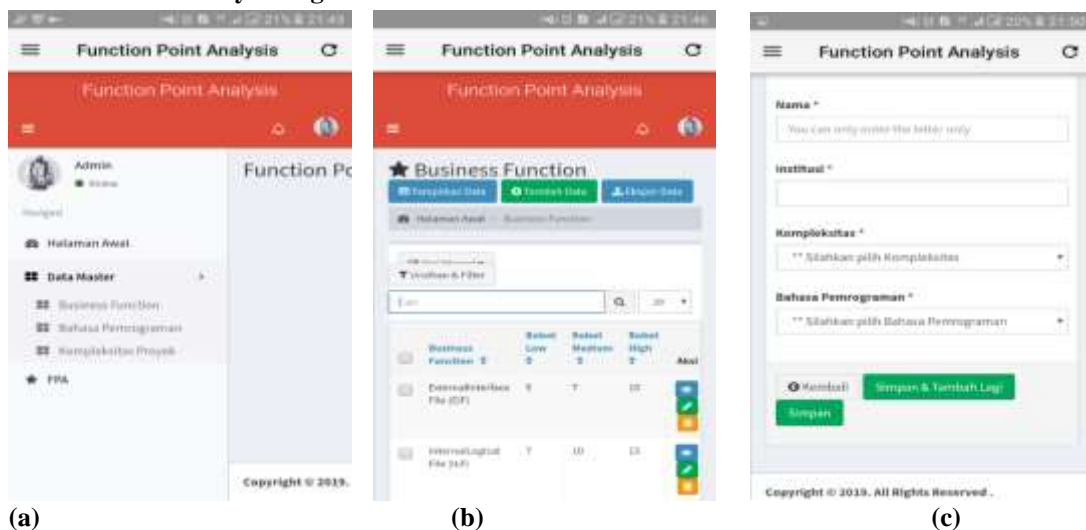
**3.3 Diagram Aktivitas dan usecase Sistem Estimasi Biaya dengan FPA**



**Gambar 1.** (a) *Usecase Diagram* Sistem Estimasi Biaya dengan FPA Gambar *Activity Diagram* Sistem Estimasi Biaya dengan FPA (b) Mengolah *Bussiness Function* dan, (c) Mengolah Bahasa Pemrograman

Bagan di atas merupakan bagan sebagian dari perancangan Sistem Estimasi Biaya dengan FPA. Pada Gambar 1 merupakan skema interaksi user dengan Sistem Estimasi Biaya dengan FPA. Pada Gambar 1 terdapat dua cuplikan diagram aktivitas atau *activity diagram* sistem estimasi biaya dengan FPA. Gambar 2 bagian (b) merupakan *Activity Diagram* Mengolah Data Master Bagian *Bussiness Function*. Di mana aktivitas mengolah data master merupakan terdiri dari proses tambah (*add*), ubah (*edit*), hapus (*delete*) data *Bussiness Function*. Begitu juga pada gambar bagian (c) *Activity Diagram* Mengolah Data Master Bagian Bahasa Pemrograman juga terdapat detail proses tambah (*add*), ubah (*edit*), hapus (*delete*). Selain dua diagram aktivitas di atas terdapat juga diagram aktivitas untuk aktivitas lainnya pada Sistem Estimasi Biaya dengan FPA.

**3.4 Analisis Sistem Estimasi Biaya dengan FPA**



**Gambar 2.** Gambar Tampilan Sistem Estimasi Biaya dengan FPA (a) Dashboard Aplikasi Dengan Daftar Menu Utama, (b) Tampilan Detail *Bussiness Function* dan (c) Tampilan Proses Input *Function Point Analysis*.

Gambar 2 adalah beberapa cuplikan dari Sistem Estimasi Biaya dengan FPA berbasis Android. Bagian (a) merupakan *dashboard* aplikasi dengan daftar menu utama yang berisi menu Data Master yang terdiri dari Bussiness Funcyion, Bahasa Pemrograman dan Kompleksitas Proyek. Gambar bagian (b) merupakan tampilan Detail *Bussiness Function*, pada laman ini terdapat menu tambah data dan ekspor data. Beberapa parameter inputan *business function* dapat ditambahkan antara lain dan (c) Tampilan Proses Input *Function Point Analysis*.

### 3.5 Analisis Pengujian Sistem Estimasi Biaya dengan FPA

Pada bagian ini akan dipaparkan hasil pengujian dan evaluasi pada menu utama dari Sistem Estimasi Biaya dengan FPA. Dari hasil pengujian Sistem Estimasi Biaya dengan FPA dengan menggunakan metode *blackbox testing*. Pengujian yang dilakukan pada daftar menu utama yakni yang terdiri dari Uji Menu Login, Uji Menu Data Master, Uji Menu *Bussiness Function*, Uji Menu Bahasa Pemrograman, Uji Menu Data Kompleksitas Proyek, Uji Menu *Function Point Analysis* (FPA) dan Uji Menu *Logout* mampu menghasilkan hasil yang sesuai dengan diharapkan seperti proses perancangan. Setelah sistem dipastikan benar-benar berhasil jalan dan tidak ada *syntax error* dan *logic error*, dilakukan evaluasi sistem menggunakan 20 data proyek. Berikut hasil komparasi estimasi dengan nilai *real*.

**Tabel 7.** Evaluasi Sistem Estimasi Biaya dengan FPA

No	$p_i$	$p_i$	$p_i - a_i$	$(p_i - a_i)^2$	$p_i - a_i/a_i$
1	8,201,340	6,000,000	2,201,340	4,845,897,795,600	0.36689
2	12,020,303	10,000,000	2,020,303	4,081,624,211,809	0.2020303
3	6,505,093	4,000,000	2,505,093	6,275,490,938,649	0.62627325
4	14,459,459	12,000,000	2,459,459	6,048,938,572,681	0.204954917
5	7,639,003	5,000,000	2,639,003	6,964,336,834,009	0.5278006
6	8,309,409	6,000,000	2,309,409	5,333,369,929,281	0.3849015
7	9,239,302	7,000,000	2,239,302	5,014,473,447,204	0.319900286
8	11,480,502	9,000,000	2,480,502	6,152,890,172,004	0.275611333
9	12,203,120	10,000,000	2,203,120	4,853,737,734,400	0.220312
10	5,249,490	3,000,000	2,249,490	5,060,205,260,100	0.74983
11	6,209,232	4,000,000	2,209,232	4,880,706,029,824	0.552308
12	3,109,487	2,000,000	1,109,487	1,230,961,403,169	0.5547435
13	5,323,429	3,000,000	2,323,429	5,398,322,318,041	0.774476333
14	3,193,209	2,000,000	1,193,209	1,423,747,717,681	0.5966045
15	7,957,928	5,000,000	2,957,928	8,749,338,053,184	0.5915856
16	8,948,029	6,000,000	2,948,029	8,690,874,984,841	0.491338167
17	11,029,321	9,000,000	2,029,321	4,118,143,721,041	0.225480111
18	4,019,324	2,000,000	2,019,324	4,077,669,416,976	19662
19	5,232,242	3,000,000	2,232,242	4,982,904,346,564	0.744080667
20	6,239,245	4,000,000	2,239,245	5,014,218,170,025	0.55981125
<b>Sigma <math>(p_i - a_i)^2/20</math></b>				103,197,851,057,083	9.978594313
<b>Akar Sigma <math>(p_i - a_i)^2/20</math></b>				5,159,892,552,854.15	0.498929716
<b>RMSE</b>				2,271,539.69	
<b>Error</b>					0.498929716

Dari tabel 7 mengenai hasil evaluasi Sistem Estimasi Biaya dengan metode FPA didapat beberapa perhitungan nilai error dan RMSE. Nilai RMSE dari harga estimasi dengan harga real dilapangan yang diterima adalah sebesar Rp 2,271,539.69. Sedangkan nilai error yang diperoleh 0.498929.

## 4. KESIMPULAN

Penelitian ini melakukan pembangunan aplikasi estimasi biaya pembangunan *software* menggunakan metode *Function Point Analysis* (FPA) berbasis Android. Dengan aplikasi berbasis Android ini, diharapkan aplikasi ini nantinya dapat memudahkan pihak developer dan *client* dalam melakukan negosiasi biaya pembangunan *software* dimanapun dan kapanpun. Selain membangun sebuah *software* yang mampu melakukan estimasi biaya perangkat lunak, pada penelitian ini dilakukan eksperimen riset pada estimasi *Cost Estimation* pada *Software Development* dengan menggunakan parameter evaluasi RMSE dan *Error*. Hasil pengujian dengan menggunakan 20 data proyek, nilai RMSE dari harga estimasi dengan harga real dilapangan yang diterima adalah sebesar Rp 2,271,539.69. Sedangkan nilai error yang diperoleh 0.498929 setelah diujikan pada sistem estimasi biaya *software* menggunakan metode *Function Point Analysis* (FPA) berbasis Android. Artinya sangat besar perbedaan atau selisih harga antara nilai estimasi ketika diujicobakan pada data real proyek *software* di lapangan.

## REFERENCES

- [1] Y. Pratama and E. Rasywir, "Automatic Cost Estimation Analysis on Datawarehouse Project with Modified Analogy Based Method," in *Proceedings of 2018 International Conference on Electrical Engineering and Computer Science, ICECOS 2018*, 2019, pp. 171-176.
- [2] F. Fachrudin and Y. Pratama, "Eksperimen Seleksi Fitur Pada Parameter Proyek Untuk Software Effort Estimation dengan K-Nearest Neighbor,"

- J. Inform. J. Pengemb. IT*, vol. 2, no. 2, pp. 53–62, 2017.
- [3] R. Popli, “Cost and Effort Estimation in Agile Software Development,” in *International Conference on Reliability, Optimization and Information Technology*, 2014, pp. 57–61.
- [4] S. Kumari and S. Pushkar, “Comparison and Analysis of Different Software Cost Estimation Methods,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, no. 1, pp. 153–157, 2013.
- [5] S. Kaur, S. Assistant, J. Kaur, S. Faculty, N. Chandigarh, and S. Singh, “Effect of Data Preprocessing on Software Effort Estimation,” *Int. J. Comput. Appl.*, vol. 69, no. 25, pp. 975–8887, 2013.
- [6] M. R. Borroek, E. Rasywir, Y. Pratama, Fachruddin, and M. Istoningtyas, “Analysis on Knowledge Layer Application for Knowledge Based System,” in *Proceedings of 2018 International Conference on Electrical Engineering and Computer Science, ICECOS 2018*, 2019, pp. 177–182.
- [7] R. Samo and J. Sidabutar, “Comparison of Different Neural Network Architectures for Software Cost Estimation,” in *International Conference on Computer, Control, Informatics and Its Applications Comparison*, 2015, pp. 68–73.
- [8] N. Mittas and L. Angelis, “LSEbA: Least squares regression and estimation by analogy in a semi-parametric model for Software Cost Estimation,” *Empir. Softw. Eng.*, vol. 15, no. 5, pp. 523–555, 2010.
- [9] J. Huang, Y. F. Li, and M. Xie, “An empirical analysis of data preprocessing for machine learning-based software cost estimation,” *Inf. Softw. Technol.*, vol. 67, pp. 108–127, 2015.
- [10] S. KaurSehra, J. Kaur, and S. Singh Sehra, “Effect of Data Preprocessing on Software Effort Estimation,” *Int. J. Comput. Appl.*, vol. 69, no. 25, pp. 29–32, 2013.
- [11] J. Wen, S. Li, Z. Lin, Y. Hu, and C. Huang, “Systematic literature review of machine learning based software development effort estimation models,” *Inf. Softw. Technol.*, vol. 54, no. 1, pp. 41–59, 2012.
- [12] J. W. S. L. L. Tang, “Software, Improve Analogy-based Estimation, Effort Principal, Using Analysis, Components Weighting,” in *Software Engineering Conference*, 2011, vol. 16, no. 2, pp. 11–12.
- [13] K. Sangeetha and P. P. Dalal, “Analysis of Software Estimation Method : Function point and Use case point,” *IJISET*, vol. 2, no. 11, pp. 880–884, 2015.
- [14] W. Enck, D. Oceau, and P. McDaniel, “A Study of Android Application Security,” 2011.
- [15] E. Rasywir, Y. Pratama, Hendrawan, and M. Istoningtyas, “Removal of Modulo as Hashing Modification Process in Essay Scoring System Using Rabin-Karp,” *2018 Int. Conf. Electr. Eng. Comput. Sci.*, pp. 159–164, 2018.