

Perbandingan Metode Cocomo II Dan Metode Analogy Untuk Estimasi Effort Pengembangan Software

Luthfi Indriyani¹

¹ Universitas Bina Sarana Informatika / Teknologi Komputer
¹Luthfi.lfy@bsi.ac.id

Diterima	Direvisi	Disetujui
10/04/2020	14/05/2020	29/07/2020

Abstrak- Estimasi sangat berpengaruh besar terhadap sebuah perencanaan baik itu perencanaan membangun rumah, gedung, jalan raya, sampai perencanaan pembuatan suatu perangkat lunak (*software*). Estimasi biaya dan usaha proyek merupakan suatu kegiatan pengaturan sumber daya dalam mencapai tujuan dan sasaran dari proyek, sehingga proyek dapat berjalan sesuai dengan tahapan dan target yang dikehendaki. Estimasi biaya proyek merupakan perkiraan waktu, biaya serta jumlah pegawai yang mengerjakan proyek. Estimasi biaya pada penelitian ini menggunakan metode COCOMO II. Metode COCOMO II dan metode analogi sebagai pembanding. Kedua metode tersebut menyediakan sebuah teknik estimasi biaya yang bertujuan untuk mengetahui berapa lamanya proses pengembangan perangkat lunak serta berapa jumlah pegawai yang dibutuhkan untuk menyelesaikan proyek perangkat lunak tersebut. Untuk menggambarkan karakteristik proyek, metode COCOMO II menggunakan kuesioner *Scale Driver* dan *Effort Multipliers* yang di bagikan kepada seluruh tim proyek. Untuk menggambarkan kompleksitas sistem yang akan dikerjakan, menggunakan nilai *Unadjusted Function Point* di setiap proses penggunaan aplikasi pada PT. Fan Integrasi Teknologi. Sedangkan Metode Analogy menggunakan nilai *Unadjusted Function Point* untuk diubah kedalam *function point* yang nanti nya akan dihitung untuk menghasilkan perhitungan waktu, tenaga kerja dan biaya proyek.

Kata kunci : Estimasi Effort, metode Cocomo II, metode Analogy

Abstract - Estimation is a big influence on a planning whether it is planning building houses, buildings, highways, until planning the creation of a software (*software*). Estimated cost and effort of the project is a resource management activity in achieving the objectives and objectives of the project, so that the project can run according to the desired stages and targets. Project cost estimation is an estimate of the time, cost and number of employees working on the project. The cost estimate of this study is using the COCOMO II method. Method of COCOMO II and methods of analogy as a comparator. Both methods provide a cost estimation technique aimed at knowing how long the software development process is and how many employees are required to complete the software project. To illustrate the characteristics of the project, the COCOMO II method uses the Scale Driver and Effort Multipliers questionnaires that are shared with the entire project team. To illustrate the complexity of the system to be done, use the value of Unadjusted Function Point in each application process using PT. Fan integration Technology. While the analogy method uses the value of Unadjusted Function Point to be changed into a Function Point that will be calculated to produce time, labor and project costs calculation.

Keywords: Estimation Effort,, Cocomo II method, Analogy method

PENDAHULUAN

Dalam sebuah perencanaan perlu adanya sebuah alokasi pendanaan yang dapat menentukan kualitas dari sebuah proyek perangkat lunak. Estimasi sangat berpengaruh besar terhadap sebuah perencanaan baik itu perencanaan membangun rumah, gedung, jalan raya, sampai perencanaan pembuatan suatu perangkat lunak.

Software cost estimation merupakan sebuah analisis yang ditujukan bagi para manajer dalam penentuan proyek TI dilihat dari perkiraan biaya yang dibutuhkan untuk perencanaan dan pengerjaan proyek tersebut. Analisis ini merupakan bagian dari manajemen proyek system informasi yang digunakan untuk mendukung pengambilan keputusan di perusahaan terkait proyek TI yang akan diterima dan dikerjakan oleh perusahaan. Dalam pelaksanaannya, estimasi biaya perangkat lunak dapat menggunakan

bantuan dari sistem lain seperti *Expert Systems* dan *Fuzzy Logic*, atau dengan menerapkan metode khusus untuk *software cost estimation* (Onah Siti Fatonah, Yasmi Afrizal,2016).

Sisi penting estimasi dalam perencanaan proyek adalah munculnya jadwal serta anggaran yang tepat, meski tidak sepenuhnya sebuah estimasi akan berakhir dengan tepat. Tetapi, tanpa sebuah estimasi dalam pelaksanaan proyek perangkat lunak maka dapat dikatakan bahwa proyek perangkat lunak tersebut adalah sebuah *blind project*. Yang diibaratkan seperti seorang buta yang harus berjalan di sebuah jalan raya yang sangat ramai (Soetam Rizky, 2011).

Estimasi biaya perangkat lunak (PL) adalah proses memperkirakan biaya yang digunakan untuk membangun atau memelihara suatu PL. Model estimasi biaya PL menggunakan tahapan pengembangan PL sebagai dasar penentuan parameter estimasi biaya. Model estimasi biaya PL dibedakan menjadi dua jenis, yaitu model *algorithmic* dan *model non-algorithmic*. Model *algorithmic* membutuhkan masukan seperti estimasi yang akurat dari atribut yang digunakan seperti *Line Of Code* (LOC), kompleksitas dan antar muka PL. *Model non-algorithmic* dibuat berdasarkan algoritma *soft computing* seperti jaringan syaraf tiruan, fuzzy, dan algoritma genetika. *Soft computing* [3] merupakan metode pengolahan data yang bersifat tidak pasti, impresisi dan dapat diimplementasikan dengan biaya yang murah. Ketiga algoritma *soft computing* tersebut dapat dihibridasi satu sama lain untuk menghasilkan model estimasi PL yang lebih baik bila ditinjau dari tingkat kesalahan [1,2], seperti hibridasi COCOMO II dengan *logika fuzzy*, *algoritma fuzzy-genetic*, dan *neuro-fuzzy* (Riska Arinta dkk, 2012) .

Menurut Andi Pressman (2020, p1), Perangkat Lunak Komputer adalah produk yang dibangun oleh seorang tenaga profesional dan bisa dikembangkan dalam jangka waktu yang panjang, meliputi program yang akan dieksekusi oleh computer dengan ukuran dan arsi tektur tertentu. Konten yang berada di dalamnya merupakan implementasi dari eksekusi program komputer, bisa berupa informasi deskriptif maupun informasi virtual.

COCOMO adalah sebuah model yang didesain oleh Barry Boehm untuk memperoleh perkiraan dari jumlah orang-bulan yang diperlukan untuk mengembangkan suatu produk perangkat lunak. Satu hasil observasi yang paling penting dalam model ini adalah bahwa motivasi dari tiap orang yang terlibat ditempatkan sebagai titik berat. Hal ini menunjukkan bahwa kepemimpinan dan kerja sama tim merupakan sesuatu yang penting, namun demikian poin pada bagian ini sering diabaikan.

Metode Analogy adalah metode yang menyimpan hasil observasi pada proyek-proyek yang telah lalu, seperti: upaya yang dibutuhkan untuk mengembangkan proyek, platform pemrograman dan

sebagainya. Ketika terdapat proyek baru, proyek tersebut diidentifikasi berdasarkan nilai masing-masing parameternya. Selanjutnya dicari kemiripannya dengan proyek-proyek lama. Proyek lama yang mirip dianggap sebagai proyek analog dan estimasi upaya dilakukan dengan cara menghitung upaya rata-rata dari upaya proyek-proyek analog (Sarno dkk,2010). Beberapa tahapan estimasi dengan menggunakan metode Analogy, adalah:

1. Pengukuran parameter-parameter dari proyek baru sesuai dengan basis data (*dataset*) yang akan digunakan.
2. Penentuan similaritas antara proyek baru dengan proyek-proyek lama dalam dataset.
3. Pemilihan proyek analog (mirip) dengan proyek baru.
4. Estimasi upaya proyek baru.

Metode COCOMO II menghasilkan estimasi yang lebih lama dibanding dengan estimasi perkiraan, karena metode Cocomo II menghitung estimasi secara terstruktur sehingga menghasilkan perkiraan jumlah pegawai yang mengerjakan, waktu untuk menyelesaikan dan total biaya yang dibutuhkan untuk Sistem Informasi Monitoring dan Evaluasi Penerimaan Beasiswa Santri Berprestasi UIN Malang yang lebih akurat dibandingkan dengan menggunakan metode Perkiraan (Damayanti,dkk,2017).

Pada COCOMO II *fase Application Composition*, hasil estimasi yang didapatkan tidak berbeda jauh dengan hasil estimasi dengan menggunakan *basic COCOMO I*, karena tidak memperhitungkan faktor atau hal-hal yang dapat mempengaruhi hasil estimasi yang dilakukan pada fase lanjutan. Hasil estimasi *fase Early Design* mirip dengan hasil perhitungan Intermediate COCOMO karena ada tujuh *cost driver* baru yang teridentifikasi. Pada *fase Post-Architectural* hasil estimasi meningkat hampir dua kali lipat dari perhitungan fase sebelumnya. Ini menunjukkan aplikasi yang dianalisis membutuhkan usaha yang tinggi untuk melakukan pengembangan lebih lanjut (Baqir habib Muhammad dan anita hidayati, 2015).

Penelitian terhadap model COCOMO II selanjutnya ditunjukkan pada penelitian mengenai pengembangan cost driver model COCOMO II dengan memodifikasi nilai atribut analysis capability untuk estimasi usaha perangkat lunak. Tujuan dari penelitian tersebut adalah untuk memodifikasi nilai atribut analysis capability pada aspek cost driver COCOMO II serta membandingkan akurasi hasil estimasi usaha pembangunan perangkat lunak yang menggunakan model COCOMO II dengan hasil estimasi usaha pembangunan perangkat lunak yang menggunakan model COCOMO modifikasi serta mengevaluasi performansi hasil COCOMO modifikasi (Sri Andayani, L. Anang Setiyo,2012).

Dari metode yang ada, metode yang sesuai dengan permasalahan estimasi biaya dalam penulisan ini dengan menggunakan metode COCOMO II dan Analogi. Dari kedua metode yang dipilih akan

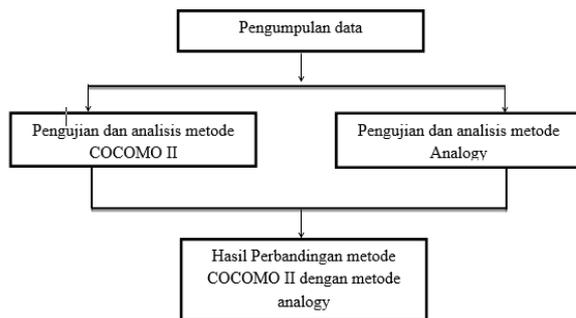
dibandingkan metode mana yang tepat dalam mengestimasi permasalahan tersebut. Yang dapat diketahui berapa hasil perhitungan dari masing-masing metode untuk mengestimasi biaya, waktu dan orang- bulan.

METODOLOGI PENELITIAN

Pada penelitian ini jenis penelitian yang diambil adalah eksperimen komparatif yaitu dengan membandingkan dua metode yang berbeda dengan melihat hasil estimasi upaya menggunakan metode COCOMO II dan metode Analogy.

Penelitian ini bertujuan untuk melakukan komparasi dan evaluasi metode COCOMO II dan metode Analogy untuk mengetahui metode apa yang memiliki tingkat keakuratan yang lebih baik dalam mengestimasi pengembangan aplikasi pada unit kerja di PT Fan Integrasi Teknologi. Jenis data yang digunakan adalah *source code* dalam pembuatan dan pengembangan aplikasi .

Penelitian ini menggunakan beberapa langkah untuk mendapatkan hasil yang diinginkan, berikut langkah-langkah dalam menyelesaikan penelitian ini :



Gambar 1. Tahapan Kerangka Pemikiran Penulis

HASIL DAN PEMBAHASAN

Data set yang digunakan dalam penelitian ini adalah data yang diperoleh berupa pengembangan aplikasi survey di PT. Fan Integrasi Teknologi aplikasi yang diperoleh menggambarkan tahapan-tahapan penggunaan aplikasi survey serta mengolah data melalui kuisisioner yang disebarkan kepada 3 responden sebagai pengelola aplikasi dan sebagai pengguna aplikasi tersebut.

Dari hasil kuisisioner dilakukan perhitungan untuk memperoleh data yang dapat dihitung menggunakan metode COCOMO II dan Metode Analogy. Untuk menghasilkan perhitungan metode COCOMO II dan Metode Analogy penulis mengusulkan langkah-langkah penyelesaian metode:

1. Studi Literatur

Studi literatur dengan mengumpulkan data yang diperoleh dari observasi yang dilakukan dengan mengamati system pada software aplikasi survey pada PT Fan Integrasi Teknologi melalui proses atau

tahap-tahapan penggunaannya, kemudian dilakukan sebuah wawancara atau penyebaran kuisisioner kebeberapa pengguna aplikasi tersebut untuk kemudian data di olah dan di hitung untuk menghasilkan perhitungan dari Metode yang dipilih. Selain itu juga mencari sumber terkait melalui media buku, e-book, artikel ilmiah, yang berasal dari sumber internet diantaranya studi literatur metode COCOMO II dan metode Analogy untuk menyelesaikan masalah estimasi biaya pengembangan perangkat lunak.

2. Pengujian Metode COCOMO II

Tahap Penggunaan Model estimasi *Effort* dengan Metode Cocomo II adalah sebagai berikut:

a. Analisa *Function Point*

Function Point mengukur proyek perangkat lunak dengan mengkuantifikasi kegunaan pemrosesan informasi yang berhubungan dengan tipe berkas, keluaran, masukan data atau kontrol eksternal. Nilai pada pembobotan tiap fungsi pada Unjusted *Function Point* didapatkan dari hasil analisa data flow diagram dari rancangan sistem enterprise Resource Planing Perusahaan Perkebunan. Tidak seperti perhitungan pada umumnya, perhitungan *function point* pada COCOMO II tidak melibatkan *degree of influence* atau *value adjustment factor*. Salah satu analisa perhitungan *function point* berdasarkan analisa data flow diagram dan arsitektur aplikasi pada unit bisnis manajemen lahan menghasilkan identifikasi tiap fungsi yaitu : tidak terdapat eksternal interface files dan eksternal inquiries pada modul manajemen lahan. EIF tidak ditemukan dalam sistem, yakni sebuah proses informasi yang merujuk pada database diluar batasan aplikasi, sedangkan EQ sebagai sebuah proses dasar yang mengirim data atau kontrol informasi yang berada diluar batasan aplikasi, juga tidak ditemukan dalam proses aplikasi (Alifi dkk, 2012).

Untuk *Function Point* mengukur proyek perangkat lunak dengan menghitung kegunaan pemrosesan informasi yang berhubungan dengan tipe berkas, keluaran, masukan data atau control eksternal. Lima fungsi pengguna tersebut perlu diidentifikasi terlebih dahulu. Tabel 1 menjelaskan lima tipe fungsi pengguna (*User*) dalam estimasi berdasarkan *function point*.

Tabel 1 Tipe *Function Point*

ILF	EIF	EI	EQ	EO
Data User	-	Update User	-	Data User
Data Area	-	Update Area	-	Data Area

Bobot Kompleksitas dari masing-masing dinilai sesuai aturan DET,RET, dan FTR untuk setiap fungsi pemrosesan informasi. berikut adalah penjabaran perhitungan untuk bobot kompleksitas tiap fungsi pemrosesan sebagai berikut:

Tabel 2. Perhitungan RET, DET dan FTR

Fungsi	RET	DET	FTR
ILF	Data user	Id User	Data User
	Data Area	Id Area	Data Area
	Total 2 RET	Total 2 DET	Total 2 FTR
EIF	-	0	0
	Total 0 RET	Total 0 DET	Total 0 FTR
EI	-	Id user	Data User
		password	
		username	
		id area	data area
		area	
	Total 0 RET	Total 5 DET	Total 2 FTR
EQ	-	0	0
	Total 0 RET	Total 0 DET	Total 0 FTR
EO	-	ID user	Data User
		username	
		status	
		nama	
		Id Area	data area
		nama	
		area	
Total 0 RET	Total 7 DET	Total 2 FTR	

Sehingga rangkuman dari semua perhitungan UFP Manajemen survey dijabarkan pada table dibawah ini.

Fungsi	Jml	Perhitungan Bobot Kompleksitas			Bobot Kompleksitas	Nilai Kompleksitas	UFP
		R	D	F			
		E	E	T			
ILF	2	2	2	2	low	7	14
EIF	0	0	0	0	-	0	0
EI	2	0	5	2	low	3	6
EO	0	0	0	0	-	0	0
EQ	2	0	7	2	Avg	4	8
Total UFP							28

Tabel 3. Rangkuman Perhitungan DET, RET dan FTR

b. Perhitungan Total UFP Aplikasi Survei

Hasil setiap fungsi dari masing-masing 8 sub unit yang ada di PT. Fan Integrasi Teknologi diakumulasi sehingga mendapatkan hasil UFP Keseluruhan. Nilai UFP yang diperoleh, akan dikalikan dengan nilai *Quantitative Software Management* dimana nilai ini merupakan standar konversi *Source Line Of Code* untuk bahasa pemrograman berbasis objek dengan konversi bernilai 29 untuk mendapatkan nilai SLOC sesuai dengan bahasa pemrograman yang digunakan. Setelah itu nilai SLOC dibagi dengan 1000 untuk mendapatkan nilai KSLOC.

Tabel 4. Perhitungan *Unjust Fuction Point* Seluruh Modul

Sub Modul	ILF	EI	EI	EI	EQ	UF	SLOC	KSLOC
Dashboard	7	5	0	3	0	15	795	0,795
User Management	7	0	3	0	4	14	742	0,742
Data Maturity Level	7	0	3	0	5	15	795	0,795
Data P2k3	7	0	3	0	4	14	742	0,742

History Maturity Level	7	0	3	0	4	14	742	0,742
Form Maturity Level	10	0	3	0	7	20	1060	1,06
Form P2k3	7	0	3	0	4	14	742	0,742
Laporan Inpeksi	7	0	3	0	4	14	742	0,742
						120	6360	6,36

c. Perhitungan Faktor Eksponen

Untuk mendapatkan nilai *Scale Driver*, terdapat 5 parameter pengukuran untuk submodel *Early Design*. Penilaian dilakukan dengan pengisian kuesioner *Scale Driver* oleh tim proyek. Tabel 5 merupakan deskripsi parameter *Scale Driver*.

Tabel 5. Perhitungan *Scale Driver* dari TIM Proyek PT. FAN INTEGRASI TEKNOLOGI

SF	R1	R2	R3	Nilai
PREC	3,72	3,72	2,48	3,306667
FLEX	2,03	3,04	2,03	2,366667
RESL	4,24	4,24	4,24	4,24
TEAM	2,19	1,1	3,29	2,193333
PMAT	4,68	4,68	4,68	4,68
Total				16,78666667

Berikut merupakan perhitungan persamaan factor Eksponen:

$$E = B + 0.01 \times SF$$

Keterangan:

$$B = 0.91 \text{ (For COCOMO II.2000)}$$

SF = Nilai *Scale Driver*

Jawaban:

$$E = 0.91 + (0.01 \times 16,786)$$

$$E = 1,0779$$

b. Mendapatkan Nilai Effort Multiplier

Untuk mendapatkan penilaian *Effort Multiplier*, metode yang digunakan sama dengan mendapatkan nilai pada *Scale Driver* yaitu dengan pengisian kuesioner *Effort Multiplier* yang berisi 7 parameter, oleh tim proyek. Tabel 6 merupakan parameter *Effort Multiplier*.

Tabel 6. *Effort Multiplier*

CD	R1	R2	R3	Nilai
RCPX	1	1,91	1	1,303333
PERS	0,63	0,63	0,63	0,63
RUSE	1	1	1	1
PDIF	1	1	0,89	0,963333
PREX	0,87	0,87	0,87	0,87
FCIL	1	1	1	1
SCED	1	1	1	1
Total Penilaian <i>Effort Multipliers</i>				0,966667

Keterangan :

R = Responden

VL = *Very Low*, L = *Low*, N = *Normal*,

H = *High*, VH = *Very High*, EH = *Extra High*

Dari tabel 6 dapat dihasilkan *Effort Multipliers* sebesar **0,966667**.

c. Perhitungan Usaha (Person Month)

Dalam menyelesaikan perhitungan usaha maka dihitung persamaan sebagai berikut:

$$PM = A \times (SIZE)^E \times \prod_{i=1}^7 EM_i$$

Keterangan:

$$A = 2.94 \text{ (for COCOMO II.2000)}$$

E = Eksponen

EM = *Effort Multipliers*

SIZE = KLOC

Jawaban:

$$PM = A \times (SIZE)^E \times \prod_{i=1}^7 EM_i$$

$$PM = 2.94 + (6,36^{1,0779}) \times 0,9667$$

$$PM = 10,0411$$

Diketahui bahwa nilai A = 2.94 (for COCOMO II.2000). Size merupakan nilai keseluruhan UFP dalam satuan KSLOC yaitu sebesar 6,36. Size dipangkatkan dengan E yang merupakan Faktor Eksponen. Nilai E sebesar 1,0779. EM merupakan nilai rata-rata nilai *Effort Multiplier* sebesar 0,9667. Maka didapatkan nilai PM sebesar **10,0411**.

d. Perhitungan Estimasi Biaya

Setelah mendapatkan estimasi usaha yang dinyatakan dengan *Person-Month*. Dari estimasi usaha tersebut akan dimasukkan ke dalam persamaan estimasi biaya sehingga menghasilkan perkiraan waktu, orang serta biaya yang dibutuhkan untuk menyelesaikan proyek. Nilai PM dimasukkan kedalam persamaan berikut:

$$TDEV = 3,67 \times (PM)^{(0,28 + 0,2 \times (E - 0,91))}$$

Jawaban:

$$TDEV = [3,67 \times 10,0411^{(0,28 + 0,2 \times (1,0779 - 0,91))}]$$

$$TDEV = 7,5649$$

Dimana nilai PM adalah 9,2614, nilai E adalah 1,0775, maka didapatkan nilai durasi proyek adalah

7,3743 bulan atau 7 bulan. Kemudian untuk mendapatkan nilai *average staffing* digunakan persamaan berikut.

$$\begin{aligned} \text{Average Staff} &= \text{PM/TDEV} \\ \text{Average Staff} &= 10,0411 / 7,5649 \\ &= \mathbf{1,3273} \end{aligned}$$

Maka didapatkan perkiraan jumlah pegawai yang dibutuhkan untuk menyelesaikan proyek survei di PT. Fan Integrasi Teknologi sebanyak 1,3273 orang atau 2 orang. Maka total estimasi biaya pembuatan aplikasi survei dalam satuan rupiah digunakan persamaan berikut ini.

$$\begin{aligned} \text{Biaya Total} &= \text{Avg Staff} \times \text{Avg Labor Cost} \\ \text{Biaya Total} &= 2 \times 5.000.000 \\ &= \text{Rp } 10.000.000 \end{aligned}$$

Atau jika digenapkan menjadi 2 orang x Rp 5.000.000, maka Biaya total pengerjaan sebesar Rp 10.000.000.

3. Pengujian Metode Analogy

Untuk menentukan Estimasi Biaya suatu pengembangan perangkat lunak menggunakan metode Analogy terdapat beberapa teknik yang dapat dilakukan dengan tahap berikut:

a. Penentuan 3D Function Point

Adalah dengan identifikasi fungsi-fungsi sebagai parameter proyek disesuaikan dengan permintaan pemakai Antara lain: *outputs, inquiries, inputs, files, interfaces, transformations, transitions*. Setelah masing-masing fungsi dikelompokkan dan dihitung, kemudian diberi bobot sesuai dengan tingkat kompleksitasnya. Nilai total seluruh fungsi disebut nilai *Un-adjusted Function Points* (UFP). Kemudian identifikasi karakteristik aplikasi (factor kompleksitas teknis). Nilai FP dihitung dengan mengalikan nilai UFP dan nilai faktor kompleksitas teknis (*Adjusted Factor/AF*). Selanjutnya nilai FP yang telah diketahui dapat dikonversi ke jumlah *Source Lines of Code* (SLOC) yang ekuivalen. Konversi dapat dilakukan dengan menggunakan Tabel ekuivalensi bahasa pemrograman pada table

Tabel 7. Ekuivalensi Bahasa Pemrograman

Bahasa	SLOC per <i>Function Point</i>
Basic Visual	32
C	128
C++	55
Cobol (ANSI 85)	91
HTML 3.0	15

JAVA	53
Visual C++	34
Visual Basic	29

Tabel 8. Studi Kasus PT. Fan Integrasi Teknologi

Business	Jml	Kompleksitas	Bobot	UFP
OUTPUT	5	Sederhana	4	20
	1	Rata-rata	5	5
	1	Kompleks	7	7
	7			32
INQUERIES	1	Sederhana	3	3
	0	Rata-rata	4	0
	0	Kompleks	6	0
	3			3
INPUT	7	Sederhana	3	21
	0	Rata-rata	4	0
	0	Kompleks	6	0
	7			21
FILE	7	Sederhana	7	49
	1	Rata-rata	10	10
	0	Kompleks	15	0
	8			59
INTERFACE	1	Sederhana	5	5
	0	Rata-rata	7	0
	0	Kompleks	10	0
	1			5
TRANSFOR MATION	0	Sederhana	7	0

	0	Rata-rata	10	0
	0	Kompleks	15	0
				0
TRANSITIO				
N	0	Sederhana	N/A	0
	0	Rata-rata	3	0
	0	Kompleks	N/A	0
				0
Total Undjusted Function Point (UFP)				120

b. Kalkulasi penggunaan perangkat lunak yang ada dan komponen-komponen serta pustaka komersial.

Tabel 9. Tabel Kompleksitas Teknis

Komponen Teknis	Nilai
Backup dan recovery dapat dipercaya	5
Komunikasi data	4
Fungsi distribusi	4
Performansi	4
Lingkungan operasional	5
Layar interaktif untuk input	4
Data entry on-line	5
Online update	5
Kompleksitas interface	3
Bisa digunakan kembali (reusability)	3
Kompleksitas proses	3
Kemudahan dalam install	3
Memiliki banyak site	3
Mudah digunakan	4

Total Komponen Teknis (TFC)	55
------------------------------------	-----------

c. Estimasi Effort (E) dengan menggunakan metode Analogy

Untuk mendapatkan estimasi *Effort* terlebih dahulu kita harus mencari jumlah *function point*. Perhitungan *Function Point* sebagai Berikut :

$$FP = UFP \times (0,65 + 0,01 \times TFC)$$

$$= 144$$

Hasil FP 144, jika di buat untuk *Source line of per function point* maka di konversi sesuai dengan bahasa pemograman yang digunakan. Jika menggunakan Bahasa JAVA maka dikonversi SLOC adalah 53. Jika dibuat KLOC dibagi 1000. Jadi jika terdapat FP 144, maka rata-rata SLOC adalah:

$$SLOC = 144 \times 14 = 7632$$

$$KLOC = 7,632$$

d. Penentuan Waktu

Sebelum menentukan waktu pengerjaan terlebih dahulu mencari nilai *Effort* untuk dimasukkan kedalam rumus penentuan waktu pengerjaan. Dalam pengerjaan proyek ini tergolong proyek kecil dan termasuk *basic semi detached*. Rumus mencari *Effort* sebagai berikut:

$$Effort = a \times SIZE + b$$

$$= 3,0 \times 7,632 + 1,05$$

$$= 23,946$$

Setelah mendapatkan nilai *Effort* sebesar 23,946 maka di masukkan kedalam rumus penentuan waktu dengan rumus sebagai berikut:

$$Td = SM \times (E)0,33$$

Keterangan:

Td = Waktu yang diperlukan

SM = *Schedule Multiple*

E = *Effort*

Jawaban :

$$td = 3,10 \times (23,946) 0,33$$

$$= 8,7859$$

maka waktu yang dibutuhkan adalah 8,7859 atau 9 bulan.

e. Penentuan Person Month

Dalam menentukan *person month* digunakan sebuah persamaan rumus sebagai berikut:

$$MP = E / td$$

$$MP = 23,946 / 8,7859$$

$$MP = 2,72$$

Maka jumlah orang yang dibutuhkan untuk pemograman sebanyak 3 orang.

f. Penentuan Biaya Proyek

Estimasi biaya Tenaga kerja untuk pemograman dalam satuan rupiah digunakan persamaan berikut ini:

$$\begin{aligned} \text{Cmb} &= \text{UR} * \text{MP} \\ \text{Cmb} &= \text{Rp. } 5.000.000 * 3 \\ &= \text{Rp } 15.000.000 \end{aligned}$$

Hasil Perbandingan Metode COCOMO II dan Metode Analogy

Penulis mencoba membandingkan kedua hasil metode yang telah dilakukan pada estimasi biaya kerangka pengembangan perangkat lunak dokumentasi data pada PT. Fan Integritas Teknologi dengan data yang sama dan perbandingannya penulis buat dalam tabel.

Tabel 10. Perbandingan Hasil Estimasi Biaya

	Metode Cocomo II	Metode Analogy
Waktu pengerjaan proyek	8 Bulan	9 Bulan
Jumlah pegawai yang mengerjakan proyek	2 Orang	3 Orang
Jumlah biaya dalam satuan bulan	Rp. 10.000.000	Rp. 15.000.000
Biaya total	Rp. 80.000.000	Rp. 135.000.000

KESIMPULAN

Berdasarkan hasil penelitian ini maka dapat diambil kesimpulan bahwa:

Metode estimasi biaya yang didapatkan dari hasil penelitian ini dapat diketahui waktu pengerjaan untuk metode COCOMO II selama 8 bulan, dengan tenaga kerja 2 orang dengan biaya yang dikeluarkan perbulan Rp 6.600.000 untuk biaya tenaga kerja, dengan total biaya Rp 52.800.000 untuk dapat menyelesaikan proyek pengembangan perangkat lunak pada PT. Fan Integrasi Teknologi.

Metode Analogy menghasilkan estimasi biaya pengembangan perangkat lunak dengan perkiraan waktu 9 bulan, dengan Tenaga kerja 3 orang, dan biaya yg dikeluarkan setiap bulan sekitar Rp 9.900.000 untuk tenaga kerja, dengan total Biaya 89.100.000 untuk dapat menyelesaikan proyek pengembangan perangkat lunak pada PT. Fan Integrasi Teknologi.

Hasil analisa dari kedua metode estimasi biaya menunjukkan bahwa metode Analogy menunjukkan waktu pengerjaan yang cukup lama dengan biaya yang cukup mahal. Dibandingkan dengan metode COCOMO II. Hal ini dikarenakan metode COCOMO II menghitung estimasi biaya dengan terstruktur menggunakan UFP, Tabel *Scale driver*, table *Effort Multiplier*. Sedangkan metode Analogy menghitung nilai FP yang selanjutnya dimasukkan kedalam nilai perkiraan waktu, tenaga kerja dan biaya.

REFERENSI

- Alifi Adia Pranatha, Achmad Holil Noor Ali, dan S. (2012). Analisis Perkiraan Biaya Pembuatan Enterprise Resource Planning (Erp) Modul Pabrik Gula Di Perusahaan Perkebunan Dengan Metode Cocomo Ii. 1(1), 1–6.
- Andi, R. P.-Y., & 2002, undefined. (2020). Rekayasa Perangkat Lunak Pendekatan Praktisi (Buku Satu). Diglib.Amikom.Ac.Id. <http://diglib.amikom.ac.id/upload/Rekayasa-Perangkat-Lunak-Pendekatan-Praktis.pdf>
- Arinta, R., Yusuf, D., & Widyastutik, D. (2012). Manajemen Risiko Pada Model Estimasi Biaya Perangkat Lunak. JUTI: Jurnal Ilmiah Teknologi Informasi, 10(1), 48. <https://doi.org/10.12962/j24068535.v10i1.a30>
- Baqir, H. M., & Hidayati, A. (2015). Perbandingan Hasil Analisis Aplikasi dengan Metode COCOMO I dan II. Multinetics, 1(2), 6. <https://doi.org/10.32722/vol1.no2.2015.pp6-11>
- Damayanti, D. E. (2017). Analisis Estimasi Biaya Pembuatan Perangkat Lunak Menggunakan Metode COCOMO II di Inagata Technosmith (Studi Kasus: Sistem Informasi Monitoring dan ... 1(10). <http://repository.ub.ac.id/7747/>
- Fatonah, O. S., & Afrizal, Y. (2016). Model Estimasi Biaya Perangkat Lunak Menggunakan Cocomo Ii (Studi Kasus Pt. X). Informatika, 1–7.
- Khoiro MHerlambang, A. D. S. M. (2018). Tampilan Evaluasi Biaya Pengembangan Perangkat Lunak Dengan Menggunakan Metode Cocomo II (Studi Kasus: PT DOT Indonesia). <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/2592/956>
- Rizky, S. (2011). konsep dasar rekayasa perangkat lunak (sofan amri (ed.)).
- Rosmala, D. (2010). 1_1_2010a.Pdf. Pengembangan Aplikasi Perkiraan Biaya Proyek IT Dengan Menggunakan Metode Cocomo, 1 vol 1, 52–61.
- Sarno, R., Buliali, J. L., & Maimunah, S. (2010). Pengembangan Metode Analogy Untuk Estimasi Biaya Rancang Bangun Perangkat Lunak. MAKARA of Technology Series, 6(2), 46–55. <https://doi.org/10.7454/mst.v6i2.72>