

Analisis Ekonomi Perangkat Lunak Dalam Pengembangan Sistem Informasi Pondok Pesantren Islamic Center Elkisi (SIP-E)

Economic Analysis of Software Development for the Information System in Pondok Pesantren Islamic Center eLKISI (SIP-e)

Hamzah Hutomo¹, M. Ainul Yaqin²

^{1,2} Program Studi Magister Teknik Informatika, Universitas Islam Negeri Maulana Malik Ibrahim Malang
e-mail: ¹ 230605210002@ student.uin-malang.ac.id ² yaqinov@tit.uin-malang.ac.id

Abstrak

Sistem Informasi Pondok Pesantren Islamic Center eLKISI (SIP-e) adalah platform yang dirancang untuk mengelola aspek administratif dan pendidikan di lembaga pendidikan Islam. Untuk meningkatkan kualitas pendidikan dan efisiensi, SIP-e mengadopsi teknologi informasi. Namun, dengan perkembangan teknologi dan kebutuhan yang kompleks, evaluasi ekonomi dari pengembangan dan pemeliharaan perangkat lunak diperlukan. Analisis ekonomi ini penting agar SIP-e dapat memahami dan mengelola biaya terkait, membuat keputusan alokasi sumber daya yang lebih cerdas, memperkirakan biaya, dan mengevaluasi nilai investasi dalam teknologi informasi. Metode Constructive Cost Model II (COCOMO II) digunakan dalam analisis ini karena terbukti fleksibel dan akurat dalam estimasi biaya pembangunan perangkat lunak. Hasil perhitungan menunjukkan estimasi usaha sebesar 60,91 person-months, dengan durasi pengerjaan 11 bulan dan kebutuhan SDM 6 orang. Berdasarkan Upah Minimum Provinsi Jawa Timur sebesar Rp 2.165.244, biaya proyek per bulan diperkirakan Rp 12.991.464, dengan total biaya selama 11 bulan sebesar Rp 142.906.104.

Kata kunci: COCOMO II, estimasi biaya, sistem informasi, pengembangan perangkat lunak, Pondok Pesantren.

Abstract

The Islamic Center eLKISI Boarding School Information System (SIP-e) is a platform designed to manage administrative and educational aspects of an Islamic educational institution. To enhance the quality of education and management efficiency, SIP-e has adopted information technology. However, with technological advancements and complex needs, an in-depth evaluation of the economic aspects of software development and maintenance is necessary. This economic analysis is crucial for SIP-e to understand and manage associated costs, make smarter resource allocation decisions, estimate expenses, and evaluate the long-term value of IT investments. The Constructive Cost Model II (COCOMO II) method was used for this analysis due to its proven flexibility and accuracy in estimating software development costs. The calculations showed an effort estimate of 60.91 person-months, with a project duration of 11 months and a need for 6 personnel. Based on the East Java Provincial Minimum Wage of Rp 2,165,244, the monthly project cost is estimated at Rp 12,991,464, with a total cost of Rp 142,906,104 over 11 months.

Keywords: COCOMO II, cost estimation, information system, software development, Boarding School.

1. PENDAHULUAN

Sistem Informasi Pondok Pesantren Islamic Center eLKISI (SIP-e) merupakan sebuah Sistem Informasi dari sebuah lembaga pendidikan Islam yang berkomitmen untuk memberikan pendidikan dan pembinaan kepada para santri dalam bidang ilmu pengetahuan umum dan agama. Sebagai bagian dari upaya untuk meningkatkan kualitas pendidikan dan efisiensi pengelolaan, SIP-e telah melihat pentingnya pemanfaatan teknologi informasi dalam mendukung berbagai aktivitas administratif, pembelajaran, dan manajemen pesantren secara keseluruhan.

SIP-e yang telah dibangun memungkinkan untuk pengelolaan data santri, keuangan, administrasi, serta kegiatan pembelajaran. Namun, seiring dengan perkembangan teknologi dan

kompleksitas kebutuhan pengelolaan pesantren, SIP-e menyadari bahwa perlu adanya evaluasi mendalam terhadap aspek ekonomi dari pengembangan dan pemeliharaan perangkat lunak yang digunakan dalam sistem informasi mereka.

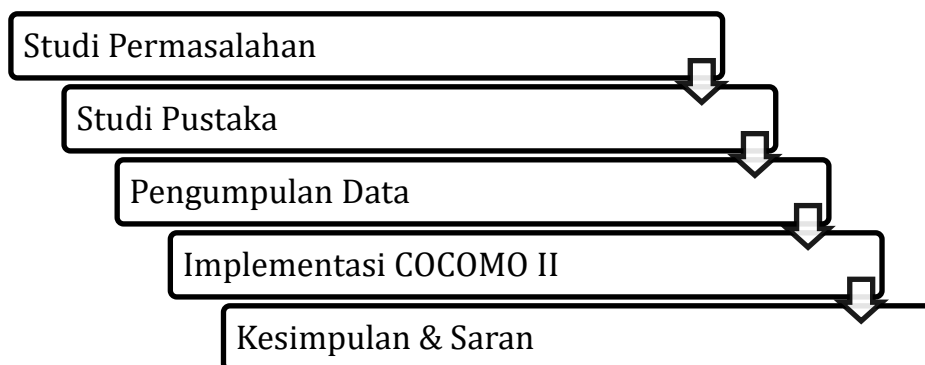
Analisis ekonomi perangkat lunak menjadi penting dalam konteks ini karena akan membantu SIP-e dalam memahami dan mengelola biaya yang terkait dengan pengembangan, operasional, dan pemeliharaan sistem informasi. Dengan memahami aspek ekonomi dari perangkat lunak yang digunakan, SIP-e dapat membuat keputusan yang lebih cerdas dalam alokasi sumber daya, memperkirakan biaya yang terkait dengan perangkat lunak, serta mengevaluasi nilai jangka panjang dari investasi dalam teknologi informasi.

Tujuan dari analisis ini adalah untuk memberikan wawasan yang mendalam kepada Pondok Pesantren Islamic Center eLKISI tentang bagaimana pengelolaan ekonomi perangkat lunak dapat berkontribusi pada keseluruhan efisiensi dan efektivitas sistem informasi mereka. Dengan memahami nilai ekonomi dari setiap aspek pengembangan sistem informasi, SIP-e akan dapat membuat keputusan yang lebih terinformasi tentang alokasi anggaran, pemilihan vendor perangkat lunak, serta strategi jangka panjang untuk pemeliharaan dan peningkatan sistem informasi mereka.

Dalam konteks pengembangan perangkat lunak, pendekatan yang umum digunakan untuk memperkirakan estimasi biaya dan waktu proyek adalah melalui penggunaan metode Constructive Cost Model II (COCOMO II). Metode ini telah terbukti memberikan fleksibilitas dan akurasi yang tinggi dalam estimasi biaya pembangunan perangkat lunak[1]. COCOMO II merupakan salah satu model algoritmik yang sering digunakan dalam perencanaan proyek untuk membentuk anggaran dan menentukan jadwal proyek[2]. Keunggulan COCOMO II adalah kemampuannya untuk mengakomodasi proyek dengan skala yang beragam, baik proyek dengan cakupan besar maupun kecil[2][3][4].

2. METODE PENELITIAN

Penelitian ini menggunakan beberapa tahapan. Tahapan yang dilaksanakan pada penelitian ini. Gambar 1 adalah tahapan penelitian yang dilakukan pada penelitian ini.



Gambar 1. Tahapan Penelitian

1. Studi Permasalahan
Tahap ini merupakan tahap identifikasi permasalahan yang didapatkan dalam penelitian[5]. Sehingga dalam tahap ini dapat memberikan arah yang jelas dalam penelitian
2. Studi Literatur
Dengan permasalahan yang diketahui, dilakukan studi literatur untuk dapat menyelesaikan permasalahan. Tahap ini dilakukan dengan mencari referensi dengan teori dan metode yang mendukung proses penelitian[6].

3. Pengumpulan data
Tahap ini dilakukan pengumpulan data pendukung yang diperlukan untuk menjadi bahan dalam menjalankan penelitian[7].
4. Implementasi COCOMO II
Dalam implementasi COCOMO II, diberlakukan dalam beberapa tahap, yaitu: Mencari nilai UFP (*Unadjusted Function Point*), Mencari Nilai *Scale Factor* dan *Effort Multiplier*, Mencari Estimasi Biaya,
5. Kesimpulan dan Saran
Tahap ini mengevaluasi sistem yang sudah dibuat setelah melakukan pengujian serta mengambil kesimpulan yang telah dilakukan[8].

COCOMO II

Metode *Cost Constructive Model* (COCOMO) merupakan suatu pendekatan yang terstruktur dalam mengestimasi biaya dan penjadwalan proyek pengembangan perangkat lunak. Dalam konteks ini, persamaan matematis digunakan untuk menghitung jumlah sumber daya manusia, waktu yang diperlukan, dan biaya yang terkait untuk menyelesaikan suatu proyek[9][10]. COCOMO memberikan dukungan yang terkini bagi berbagai jenis bisnis perangkat lunak, termasuk perangkat lunak berbasis objek, pengembangan ulang, dan aplikasi komersial. Metode ini dapat diterapkan pada berbagai tahap proyek, mulai dari perencanaan awal hingga implementasi.

Manfaat dari Metode COCOMO meliputi[3]

- a. Penetapan Anggaran Perusahaan: Membantu perusahaan dalam menetapkan anggaran yang tepat untuk biaya perancangan perangkat lunak.
- b. Fasilitasi Tawaran Kontrak yang Kompetitif: Memungkinkan perusahaan untuk menawarkan kontrak yang kompetitif kepada klien atau pihak lain yang terlibat dalam proyek.
- c. Menentukan Efektivitas Organisasi: Memungkinkan organisasi untuk menentukan strategi yang paling efektif dalam mengalokasikan sumber daya dan mengelola proyek perangkat lunak.

Proses estimasi biaya dengan menggunakan Metode COCOMO II melibatkan beberapa langkah yang harus ditempuh secara sistematis[2].

- a. Mencari nilai UFP (*Unadjusted Function Point*)
Function Point adalah metode untuk mengukur kompleksitas suatu sistem dengan memperhitungkan jumlah total dari tipe berkas, input dan output data, serta kontrol eksternal. Perhitungan dilakukan dengan mengklasifikasikan tipe-tipe fungsi pengguna yang ada dalam *Data Flow Diagram* (DFD) maupun dalam struktur database yang direpresentasikan oleh *Entity-Relationship Diagram* (ER). Komponen data dalam sistem kemudian digunakan untuk menetapkan bobot kompleksitas sesuai dengan aturan yang telah ditetapkan untuk *Data Element Types* (DET), *File Types Referenced* (FTR), dan *Record Element Types* (RET), yang akan diuraikan lebih lanjut.

Tabel 1. Komponen nilai UFP

Komponen	Keterangan
External Input (EI)	Merupakan nilai yang diberikan pada setiap data yang dimasukkan oleh pengguna, yang dianalisis dari DFD maupun dari Database.
External Output (EO)	Menggambarkan nilai output yang dihasilkan setelah menerima input dari pengguna, yang dianalisis dari DFD maupun Database.
Internal Logical File (ILF)	Merujuk pada data yang disimpan dalam sistem, termasuk data yang dibuat, disimpan, atau dipelihara oleh sistem itu sendiri.

External Interface File (EIF)	Berkas eksternal yang dibagikan dari sistem lain yang terintegrasi dengan sistem saat ini.
External Inquiry (EQ)	Mewakili nilai input yang menghasilkan output secara instan dalam sistem.

b. Mencari Nilai *Scale Factor* Dan *Effort Multiplier*

Dalam mencari nilai *scale factor* terdapat parameter penilaian untuk setiap bobot yang ada pada *scale factor*, penilaian tersebut dilakukan dengan cara memberikan kuisioner kepada beberapa responden, responden disini merupakan tim pengembang perangkat lunak[4].

Tabel 2. Nilai Scale Factor dan Effort Multiplier

Scale Factor	Deskripsi
PREC	Faktor skala ini mencerminkan tingkat pengalaman masa lalu organisasi dalam menangani proyek-proyek serupa.
FLEX	Faktor skala ini mencerminkan kemampuan klien dalam menetapkan tujuan dan mengomunikasikan kebutuhan perangkat lunak kepada tim pengembang.
RESL	Faktor skala ini menggambarkan seberapa sering perusahaan melakukan perencanaan manajemen risiko terhadap proyek yang mereka jalankan. Perencanaan manajemen risiko bertujuan untuk mengidentifikasi dan mencegah kemungkinan risiko yang dapat timbul selama proses pengembangan perangkat lunak.
TEAM	Faktor skala ini mencerminkan tingkat kerja sama dan kualitas kerja tim pengembangan.
PMAT	Faktor skala ini mencerminkan tingkat kematangan proses pengembangan perangkat lunak dalam organisasi. Hal ini berdasarkan pada Model Kematangan Kemampuan Rekayasa Perangkat Lunak (Capability Maturity Model) yang memiliki 5 tingkatan.

c. Mencari Estimasi Biaya

Estimasi dilakukan dalam beberapa tahap dengan menggunakan beberapa persamaan[2][11].

1) Estimasi Usaha (*Person-Month*)[12]

Langkah pertama adalah mencari estimasi usaha, yang dihitung berdasarkan ukuran proyek, pengalaman tim, dan faktor skala yang terkait.

$$PM = A \times (size)^E \times \prod_{i=1}^n EM_i \quad \dots \text{pers (1)}$$

Nilai A merupakan nilai ketentuan yang ada pada jurnal COCOMO dengan nilai sebesar 2.94, size merupakan nilai UFP yang telah dikonversikan menjadi KLOC, E merupakan nilai dari Scale Factor, dan EM merupakan nilai rata-rata dari Effort Multiplier.

2) Durasi Proyek (*Time to Develop*)

Setelah itu, estimasi waktu yang dibutuhkan untuk menyelesaikan proyek dihitung berdasarkan estimasi usaha, faktor skala, dan faktor jadwal yang dipertimbangkan.

$$TDEV = [A \times PM^{(0.33+0.2 \times (E-1.01))}] \quad \dots \text{pers (2)}$$

Dari rumus diatas diketahui nilai A adalah 3.0 sesuai ketentuan dari COCOMO II, PM merupakan nilai dari estimasi usaha, nilai E merupakan nilai dari Scale Factor, dan Sced merupakan nilai dari driver Schedule yang terdapat pada kuisioner effort multiplier.

3) Estimasi SDM (*Average Staffing*)

Selanjutnya, estimasi jumlah karyawan yang dibutuhkan dalam proyek dihitung berdasarkan estimasi usaha dan estimasi waktu pengerjaan proyek.

$$\text{Average Staffing} = PM \times TDEV \quad \dots \text{pers (3)}$$

Dari rumus diatas dapat diketahui PM merupakan nilai dari estimasi usaha dan TDEV merupakan nilai estimasi durasi pengerjaan proyek.

4) Estimasi Biaya:

$$\text{Biaya Total} = \text{Average Staffing} \times \text{Average Cost Labor} \quad \dots \text{pers (4)}$$

Langkah terakhir adalah menghitung biaya total proyek, yang diperoleh dari perkalian estimasi jumlah karyawan yang dibutuhkan dengan standar biaya gaji per karyawan.

3. HASIL DAN PEMBAHASAN

Analisis Masalah

Masalah yang dihadapi oleh Pondok Pesantren Islamic Center eLKISI (SIP-e) adalah kesulitan dalam mengelola biaya dan waktu yang terkait dengan pengembangan dan pemeliharaan sistem informasi mereka. Meskipun SIP-e telah membangun sistem informasi untuk mengelola data santri, keuangan, administrasi, dan kegiatan pembelajaran, masih terdapat keterbatasan dalam pengelolaan dan efisiensi sistem informasi tersebut. Untuk mengatasi masalah ini, SIP-e dapat menggunakan metode Constructive Cost Model II (COCOMO II) untuk melakukan estimasi biaya dan waktu proyek secara akurat. COCOMO II telah terbukti memberikan fleksibilitas dan akurasi yang tinggi dalam estimasi biaya pembangunan perangkat lunak. Model ini juga dapat mengakomodasi proyek dengan skala yang beragam, baik proyek dengan cakupan besar maupun kecil. Dengan menggunakan COCOMO II, SIP-e dapat memperkirakan biaya yang terkait dengan pengembangan, operasional, dan pemeliharaan sistem informasi mereka dengan lebih tepat. Estimasi biaya yang akurat akan membantu SIP-e dalam mengalokasikan sumber daya dengan lebih efisien, menghindari anggaran yang melebihi batas, dan menjaga proyek agar tetap sesuai dengan jadwal yang telah ditetapkan. Selain itu, COCOMO II juga akan membantu SIP-e dalam membuat keputusan yang lebih cerdas tentang alokasi anggaran, pemilihan vendor perangkat lunak, dan strategi pemeliharaan jangka panjang. Dengan demikian, penggunaan COCOMO II akan menjadi solusi yang tepat untuk meningkatkan efisiensi dan efektivitas pengelolaan sistem informasi SIP-e.

Implementasi COCOMO II

Dalam penyelesaian masalah, digunakan implementasi COCOMO II yang dibutuhkan beberapa langkah untuk didapatkan estimasi harga dari Sistem Informasi Pondok Pesantren Islamic Center eLKISI (SIP-e).

1. Analisis Data

Untuk langkah pertama adalah memetakan fitur yang digunakan dalam aplikasi, seperti tabel

Tabel 3. Data Fitur Aplikasi SIP-e

Elemen Data	Fitur yang Disediakan
Data Guru	Profil guru, hak akses guru
Data Santri	Profil santri
Data Foto	Foto santri, Foto kegiatan santri, Foto bukti sertifikat, Foto bukti pelanggaran, Foto bukti sakit
Data Rekap Hafalan	Jumlah hafalan hadist setiap santri, Jumlah hafalan al-qur'an setiap santri, Sertifikat hafalan setiap santri, Rekap hafalan hadist seluruh santri
Data Rekap Kesehatan	Rekam medis setiap santri, Rekap pemeriksaan keseluruhan santri
Data Rekap Sanksi	Rekam pelanggaran setiap santri, Rekap pelanggaran keseluruhan santri

Data Rekap Prestasi

Rekam prestasi setiap santri, Rekap prestasi keseluruhan santri

Data tersebut dianalisis termasuk pada kategori yang mana dalam metode COCOMO II, maka didapatkan tabel berikut sebagai klasifikasi data untuk selanjutnya dapat dihitung dengan persamaan yang digunakan dalam metode ini.

Tabel 4. Klasifikasi Data

No.	Fitur	DET	RET	FTR	EI	EO	EQ	ILF	EIF
1	Data Guru	2	1	0	1	0	0	0	0
2	Data Santri	2	1	0	0	0	0	1	0
3	Data Foto	4	0	0	0	0	0	0	0
4	Data Rekap Hafalan	4	0	0	0	0	0	0	0
5	Data Rekap Kesehatan	2	0	0	0	0	0	0	0
6	Data Rekap Sanksi	2	0	0	0	0	0	0	0
7	Data Rekap Prestasi	2	0	0	0	0	0	0	0

Berdasarkan Tabel 4 diketahui bahwa:

a. DET (*Data Element Types*):

Menunjukkan jumlah elemen data yang diperlukan untuk diolah atau disimpan oleh fitur tersebut.

b. RET (*Record Element Types*):

Menunjukkan jumlah tipe catatan atau rekaman yang digunakan atau dihasilkan oleh fitur tersebut.

c. FTR (*File Types Referenced*):

Menunjukkan jumlah file eksternal yang digunakan atau diakses oleh fitur tersebut.

d. EI (*External Input*):

Menunjukkan fungsi-fungsi atau fitur-fitur yang menerima data dari luar sistem untuk diproses.

e. EO (*External Output*):

Menunjukkan fungsi-fungsi atau fitur-fitur yang menghasilkan data untuk dikirimkan keluar sistem.

f. EQ (*External Inquiry*):

Menunjukkan fungsi-fungsi atau fitur-fitur yang menerima masukan dari pengguna dan menghasilkan keluaran seketika.

g. ILF (*Internal Logical File*):

Menunjukkan data yang disimpan dalam sistem dan dikelola oleh fungsi-fungsi atau komponen-komponen dalam sistem itu sendiri.

h. EIF (*External Interface File*):

Menunjukkan file eksternal yang digunakan atau diakses oleh sistem tetapi dikendalikan oleh sistem lain.

Dari klasifikasi data, dapat dihitung nilai UFP (*Unadjusted Function Point*) dengan menambahkan seluruh klasifikasi data.

$$Total\ UFP = \Sigma(DET) + \Sigma(RET) + \Sigma(FTR) + \Sigma(EI) + \Sigma(EO) + \Sigma(EQ) + \Sigma(ILF) + \Sigma(EIF) \dots pers\ (5)$$

Menggunakan data dari tabel sebelumnya, kita dapat menghitung total UFP:

$$\begin{aligned}
Total\ UFP &= (2 + 2 + 4 + 4 + 2 + 2 + 2) + (1 + 1 + 0 + 0 + 0 + 0 + 0 + 0) \\
&\quad + (0 + 0 + 0 + 0 + 0 + 0 + 0 + 0) \\
&\quad + (1 + 0 + 0 + 0 + 0 + 0 + 0 + 0) \\
&\quad + (0 + 0 + 0 + 0 + 0 + 0 + 0 + 0) \\
&\quad + (0 + 0 + 0 + 0 + 0 + 0 + 0 + 0) \\
&\quad + (0 + 1 + 0 + 0 + 0 + 0 + 0 + 0) \\
&\quad + (0 + 0 + 0 + 0 + 0 + 0 + 0 + 0) \\
Total\ UFP &= 19 + 2 + 1 + 1 \\
Total\ UFP &= 23\ UFP
\end{aligned}$$

2. Perhitungan Scale Factor & Effort Multiplier

Pertama ditentukan dulu beberapa faktor dalam aplikasi SIP-e, faktor-faktor ini merupakan aspek yang dijelaskan dalam metode COCOMO II untuk diimplementasikan pada analisis aplikasi. Faktor-faktor tersebut seperti tabel 5.

Tabel 5. Analisis Scale Factor

Faktor	Keterangan	Nilai
<i>Precedentedness (PREC)</i>	Low	0.91
<i>Development Flexibility (FLEX)</i>	Nominal	1.0
<i>Risk Resolution (RESL)</i>	Nominal	1.0
<i>Team Cohesion (TEAM)</i>	Nominal	1.0
<i>Process Maturity (PMAT)</i>	Nominal	1.0
Total		4.91

Precedentedness (PREC): Faktor ini menggambarkan seberapa sering proyek ini telah dijalankan sebelumnya. Semakin sering proyek dilakukan, semakin tinggi tingkat *precedentedness*. Untuk proyek yang sering dilakukan sebelumnya, PREC memiliki nilai rendah.

- Nilai DET dan RET yang relatif kecil menunjukkan bahwa proyek ini memiliki karakteristik yang baru atau jarang dilakukan sebelumnya. Oleh karena itu, kita dapat menetapkan nilai PREC sebagai Low (0.91).
- Development Flexibility (FLEX)*: Faktor ini menggambarkan seberapa fleksibel proses pengembangan. Semakin fleksibel prosesnya, semakin tinggi nilai FLEX.
- Risk Resolution (RESL)*: Faktor ini menggambarkan seberapa sering resiko telah diatasi sebelumnya. Semakin sering resiko diatasi, semakin tinggi nilai RESL.
- Team Cohesion (TEAM)*: Faktor ini menggambarkan seberapa kuat hubungan tim pengembangan. Semakin kuat hubungannya, semakin tinggi nilai TEAM.
- Process Maturity (PMAT)*: Faktor ini menggambarkan seberapa matang proses pengembangan dalam organisasi. Semakin matang, semakin tinggi nilai PMAT.

Dalam persamaan untuk menganalisis Scale Faktor, digunakan notasi E. Faktor-faktor tersebut dianalisis melalui persamaan berikut:

$$\begin{aligned}
E &= B + 0.01 \times SF \\
E &= 0.91 + 0.01 \times 4.91 \\
E &= 0.91 + 0.05 \\
E &= 0.96
\end{aligned}$$

Keterangan:

B = 0.91 (untuk COCOMO II)

SF = nilai *Scale Factor*

Selain *Scale Factor*, diperlukan penilaian *Effort Multiplier* sebelum dilakukan analisis mengenai nilai sebuah aplikasi. Tabel 6 adalah hasil analisis *Effort Multiplier*.

Tabel 6. Analisis Scale Factor

Faktor	Nilai	Keterangan
<i>RELY (Required Software Reliability)</i>	1.15	High
<i>DATA (Database Size)</i>	1.15	High
<i>CPLX (Product Complexity)</i>	1.15	High
<i>DOCU (Documentation Match to Lifecycle Needs)</i>	1.0	Nominal
<i>RUSE (Required Reusability)</i>	1.0	Nominal
<i>ACAP (Analyst Capability)</i>	1.0	Nominal
<i>PCON (Personnel Continuity)</i>	1.15	High
<i>STOR (Platform Experience)</i>	1.0	Nominal
<i>PVOL (Platform Volatility)</i>	1.0	Nominal
<i>PCAP (Personnel Capability)</i>	1.0	Nominal
<i>AEXP/APEX (Applications Experience/Applied Expertise)</i>	1.0	Nominal
<i>PEXP/PLEX (Platform Experience/Programming Language Experience)</i>	1.0	Nominal
<i>LTEX (Language and Tool Experience)</i>	1.0	Nominal
<i>TOOL (Use of Software Tools)</i>	1.0	Nominal
<i>SITE (Multisite Development)</i>	1.0	Nominal
<i>SCED (Schedule)</i>	1.0	Nominal
Total	16.6	

- 1) *RELY (Required Software Reliability)*:
Karena sistem ini mungkin membutuhkan data yang akurat dan dapat diandalkan untuk mengelola informasi santri, maka reliabilitas yang dibutuhkan mungkin cukup tinggi.
- 2) *DATA (Database Size)*:
Data yang disimpan dalam proyek ini mungkin cukup besar karena mencakup profil santri, rekam hafalan, kesehatan, sanksi, dan prestasi.
- 3) *CPLX (Product Complexity)*:
Fitur-fitur yang diberikan termasuk manajemen data, manajemen foto, dan rekapitulasi data yang mungkin memiliki kompleksitas yang sedang.
- 4) *DOCU (Documentation Match to Lifecycle Needs)*:
Dokumentasi yang memadai mungkin diperlukan untuk proyek ini karena dapat membantu pemeliharaan sistem di masa mendatang.
- 5) *RUSE (Required Reusability)*:

Karena proyek ini mungkin unik dan tidak terlalu tergantung pada komponen yang dapat digunakan kembali, nilai RUSE dapat dianggap sebagai Nominal.

- 6) ACAP (*Analyst Capability*):
Tim yang mengembangkan proyek ini mungkin membutuhkan kemampuan analisis yang cukup baik untuk memahami kebutuhan pengguna. Namun, karena proyek ini tidak terlalu besar, nilai ACAP dapat dianggap sebagai Nominal.
- 7) PCON (*Personnel Continuity*):
Karena proyek ini mungkin memiliki kebutuhan yang berkelanjutan terhadap personil, nilai PCON dapat dianggap sebagai Nominal atau High.
- 8) STOR (*Platform Experience*):
Pengalaman dengan platform pengembangan yang digunakan mungkin cukup penting untuk memastikan kelancaran pengembangan. Oleh karena itu, nilai STOR dapat dianggap sebagai Nominal atau High.
- 9) PVOL (*Platform Volatility*):
Karena pengembangan platform tidak terlalu sering berubah, nilai PVOL dapat dianggap sebagai Nominal.
- 10) PCAP (*Personnel Capability*):
Kemampuan personil dalam mengembangkan proyek ini mungkin cukup penting untuk menjamin kualitasnya. Namun, karena proyek ini tidak terlalu besar, nilai PCAP dapat dianggap sebagai Nominal.
- 11) AEXP/APEX (*Applications Experience/Applied Expertise*):
Pengalaman dalam pengembangan aplikasi serupa mungkin penting untuk memastikan proyek ini berhasil. Namun, karena proyek ini tidak terlalu besar, nilai AEXP/APEX dapat dianggap sebagai Nominal.
- 12) PEXP/PLEX (*Platform Experience/Programming Language Experience*):
Pengalaman dalam menggunakan platform dan bahasa pemrograman yang digunakan mungkin penting untuk kesuksesan proyek. Namun, karena proyek ini tidak terlalu besar, nilai PEXP/PLEX dapat dianggap sebagai Nominal.
- 13) LTEX (*Language and Tool Experience*):
Pengalaman dengan bahasa pemrograman dan alat pengembangan mungkin cukup penting untuk memastikan kelancaran pengembangan. Namun, karena proyek ini tidak terlalu besar, nilai LTEX dapat dianggap sebagai Nominal.
- 14) TOOL (*Use of Software Tools*):
Penggunaan alat bantu pengembangan mungkin penting untuk meningkatkan efisiensi pengembangan. Namun, karena proyek ini tidak terlalu besar, nilai TOOL dapat dianggap sebagai Nominal.
- 15) SITE (*Multisite Development*):
Pengembangan di beberapa lokasi mungkin tidak diperlukan untuk proyek ini. Oleh karena itu, nilai SITE dapat dianggap sebagai Nominal.
- 16) SCED (*Schedule*):
Penjadwalan proyek ini mungkin penting untuk dipertimbangkan dengan baik untuk memastikan proyek selesai tepat waktu. Namun, karena proyek ini tidak terlalu besar, nilai SCED dapat dianggap sebagai Nominal.

3. Estimasi Biaya

Sesuai dengan langkah-langkah yang dijelaskan sebelumnya, berikut perhitungan yang dilakukan:

- a. Estimasi Usaha (*Person-Month*):

$$PM = A \times (Size)^E \times \prod_{i=1}^{17} EMI$$

$$PM = 2,94 \times 24^{0,96} \times \frac{16,6}{17}$$

$$PM = 2,94 \times 21,14 \times 0,98$$

$$PM = 60,91$$

- b. Durasi Proyek (*Time to Develop*):

$$TDEV = [A \times PM^{(0.33+0.2 \times (E-1.01))}]$$

$$TDEV = [3 \times 60,91^{(0.33+0.2 \times (0,96-1.01))}]$$

$$TDEV = 11,17$$

Berarti Estimasi waktu 11 bulan

- c. Estimasi SDM (*Average Staffing*):

$$\text{Average Staffing} = \frac{PM}{TDEV} = \frac{60,91}{11} = 5,53$$

dibulatkan menjadi 6, sehingga rata-rata pekerja yang dibutuhkan adalah 6 orang

- d. Estimasi Biaya:

$$\text{Biaya Total} = \text{Average Staffing} \times \text{Average Cost Labor}$$

$$\text{Biaya Total} = 6 \times 2.165.244 \text{ (UMP Jawa Timur)}$$

$$\text{Biaya Total} = 12.991.664$$

Karena proyek diestimasi 11 bulan, maka nilai biaya total dikalikan 11 bulan, hasilnya adalah sebagai berikut:

$$\text{Biaya Total} = 12.991.664 \times 11 = \text{Rp } 142.906.104$$

4. KESIMPULAN

Analisis ekonomi perangkat lunak dalam pengembangan Sistem Informasi Pondok Pesantren Islamic Center eLKISI (SIP-e) menunjukkan pentingnya pemahaman mendalam mengenai biaya dan sumber daya yang dibutuhkan untuk memastikan keberlanjutan dan efisiensi sistem informasi. Dengan menggunakan metode *Constructive Cost Model II* (COCOMO II), SIP-e mampu memperkirakan estimasi usaha sebesar 60,91 person-months, dengan durasi proyek 11 bulan dan kebutuhan SDM sebanyak 6 orang. Estimasi biaya berdasarkan Upah Minimum Provinsi (UMP) Jawa Timur sebesar Rp 2.165.244 menunjukkan bahwa biaya bulanan proyek untuk gaji 6 orang adalah Rp 12.991.464, dengan total biaya selama 11 bulan mencapai Rp 142.906.104. Hasil analisis ini memberikan wawasan berharga bagi SIP-e dalam membuat keputusan yang lebih terinformasi terkait alokasi anggaran, pemilihan vendor perangkat lunak, serta strategi jangka panjang untuk pemeliharaan dan peningkatan sistem informasi mereka.

DAFTAR PUSTAKA

- [1] A. D. P. Ariyanto, L. Azizah, and U. L. Yuhana, "Analisis Metode Estimasi Biaya pada Perangkat Lunak Beserta Faktor-Faktor yang Mempengaruhi : A Systematic Literature Review," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 4, pp. 699–708, 2022, doi: 10.25126/jtiik.2021864611.
- [2] A. Puspaningrum, F. P. B. Muhammad, and E. Mulyani, "Optimasi Koefisien COCOMO II Menggunakan Algoritma Kelelawar untuk Meningkatkan Akurasi Estimasi Biaya dan Waktu Pengembangan Perangkat Lunak," *J. IKRAITH-INFORMATIKA*, vol. 5, no. 3, p. 103, 2021.
- [3] R. Parlika, D. C. M. Wijaya, H. Khariono, and R. A. Fernanda, "Studi literatur perbandingan antara metode LOC, COCOMO, FPA dalam ranah software metric," *J. Pendidik. Inform. dan Sains*, vol. 9, no. 1, p. 66, 2020, doi: 10.31571/saintek.v9i1.1697.

- [4] E. F. Harsono, “Sistem Aplikasi Penentu Estimasi Biaya Pengembangan Perangkat Lunak Menggunakan Metode COCOMO II,” Universitas Dinamika, 2020.
- [5] J. Madre, H. Yudi Sukmono, and S. Gunawan, “Perancangan Sistem Informasi Berbasis Website Sebagai Salah Satu Media Promosi Pada Perusahaan,” *J. Ind. Manuf. Eng.*, vol. 5, no. 2, 2021, doi: 10.31289/jime.v5i2.5594.
- [6] L. Angela and E. Erandaru, “STUDI PERBANDINGAN TEORI DAN PRAKTEK PROSES PERANCANGAN UI/UX di ARYANNA,” *J. DKV Adiwarna*, vol. 1, p. 10, 2022.
- [7] R. Rusliyawati and R. Nuraini, “Sistem Pendukung Keputusan Pemilihan Vendor IT Menggunakan Metode Perbandingan Eksponensial (MPE),” *Insearch Inf. Syst. Res. J.*, vol. 2, no. 02, pp. 90–98, 2022, doi: 10.15548/isrj.v2i02.4382.
- [8] Y. Khadaffi, J. Jupriyadi, and W. Kurnia, “Aplikasi Smart School Untuk Kebutuhan Guru Di Era New Normal (Studi Kasus : Sma Negeri 1 Krui),” *J. Teknol. dan Sist. Inf.*, vol. 2, no. 2, p. 15, 2021, doi: 10.33365/jtsi.v2i2.866.
- [9] R. Permana, A. Abdilah, Fuad Nur Hasan, and Mahmud Syarif, “Estimation Effort Pengembangan Software Inventory PT. Infinity Global Mandiri Menggunakan Metode Use Case Point,” *J. RESTIKOM Ris. Tek. Inform. dan Komput.*, vol. 5, no. 2, pp. 73–84, 2023, doi: 10.52005/restikom.v5i2.144.
- [10] L. M. Rompas, “Analisis Proporsi Sumber Daya Pada Proyek Konstruksi (Studi Kasus Kota Manado),” *J. TEKNO*, vol. 20, no. 82, pp. 1189–1194, 2022.
- [11] L. Indriyani, “Perbandingan Metode Cocomo II Dan Metode Analogy Untuk Estimasi Effort Pengembangan Software,” *J. Tek. Komput. AMIK BSI*, vol. 6, no. 2, 2020, doi: 10.31294/jtk.v4i2.
- [12] E. Handoyo, R. R. Isnanto, and A. Primaraka, “Estimasi Biaya Pembuatan Perangkat Lunak Menggunakan COCOMO II Pada Sistem Informasi Pelaporan Kegiatan Pembangunan.”