

---

# PERKIRAAN SUMBER DAYA PENGEMBANGAN SISTEM INFORMASI MENGGUNAKAN COCOMO II (Studi Kasus : Pengembangan Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung)

<sup>1</sup>Dian Pitaloka, <sup>2</sup>Anie Rose Irawati, <sup>3</sup>Yohana Tri Utami

<sup>1,2,3</sup> Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam,  
Universitas Lampung

Jl. Prof. Dr. Ir. Sumantri Brojonegoro, RW.No: 1, Gedong Meneng, Kec. Rajabasa,  
Kota Bandar Lampung, Lampung, Indonesia

E-mail: [dian.pitaloka1017@students.unila.ac.id](mailto:dian.pitaloka1017@students.unila.ac.id), [anie.roseirawati@fmipa.unila.ac.id](mailto:anie.roseirawati@fmipa.unila.ac.id)  
[yohana.utami@fmipa.unila.ac.id](mailto:yohana.utami@fmipa.unila.ac.id)

## Abstract

*Success in working on a project should start with good planning. Inadequate planning will impact on cost, time, and human resources swelling so that it is not in accordance with the plan. This study implements the COCOMO II (Constructive Cost Model II) method to calculate the estimated resources needed in an information system development. This research uses Information System of English Proficiency Test (EPT) in Unit Pelaksana Teknis (UPT) Bahasa of the University of Lampung as a case study, which is currently in the development stage. Unit Pelaksana Teknis (UPT) Bahasa of the University of Lampung must know the resources needed to complete the information system development project in order to determine a budget plan. This method has the advantage of being able to clearly predict the time, cost, and resources to complete an information system project. The COCOMO II method is carried out by calculating the Scale Factor, Effort Multiplier, and Effort Estimation.*

**Keywords:** *Scale Factor, Effort Multiplier, Effort Estimation, Unadjusted Function Point, Source Lines Of Code*

## Abstrak

*Keberhasilan dalam mengerjakan suatu proyek software harus dimulai dengan perencanaan yang baik. Perencanaan yang kurang baik dapat berimbas pada pembengkakan sumber daya biaya, waktu, dan sumber daya manusia sehingga tidak sesuai dengan perencanaan. Penelitian ini mengimplementasikan metode COCOMO II (Constructive Cost Model II) untuk menghitung estimasi sumber daya yang dibutuhkan dalam pengembangan sistem informasi dengan studi kasus Sistem Informasi Pengelolaan Data English Proficiency Test (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung yang berada dalam tahap pengembangan. Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung harus mengetahui sumber daya yang diperlukan untuk menyelesaikan proyek pengembangan sistem informasi agar dapat menentukan rencana anggaran yang dijadikan sebagai acuan dalam pengembangan sistem. Metode COCOMO II memiliki kelebihan yaitu bisa memprediksi kebutuhan waktu, biaya, dan juga sumber daya untuk menyelesaikan proyek sistem informasi dengan jelas. Pengukuran dengan metode COCOMO II dilakukan dengan cara melakukan perhitungan Scale Factor, Effort Multiplier, dan Effort Estimation.*

**Kata Kunci:** *Scale Factor, Effort Multiplier, Effort Estimation, Unadjusted Function Point, Source Lines Of Code*

## 1. PENDAHULUAN

Keberhasilan dalam mengerjakan proyek harus dimulai dengan perencanaan yang baik. Perencanaan yang kurang baik akan menyebabkan efek bagi suatu proyek. Efek yang terjadi pada proyek dapat berupa pembengkakan sumber daya seperti biaya, waktu, dan sumber daya manusia yang dibutuhkan untuk mengembangkan proyek tidak sesuai dengan yang telah ditentukan di awal perencanaan. Untuk mengurangi resiko pembengkakan biaya proyek dibutuhkan estimasi yang tepat [3]. Estimasi pengembangan perangkat lunak yang meliputi estimasi biaya, waktu, dan sumber daya manusia yang dibutuhkan. Estimasi biaya pengembangan perangkat lunak merupakan proses prediksi biaya yang harus dikeluarkan dalam seluruh tahapan pembangunan perangkat lunak dengan menggunakan beberapa parameter [8].

Penelitian ini menggunakan studi kasus Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung yang digunakan untuk keperluan tes TOEFL. Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung sedang dalam tahap pengembangan. Pengembangan Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) dilakukan dengan penambahan fungsi grafik yang menampilkan ringkasan data secara statistik. Fungsi grafik diharapkan dapat memudahkan instansi Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung dalam pengambilan keputusan.

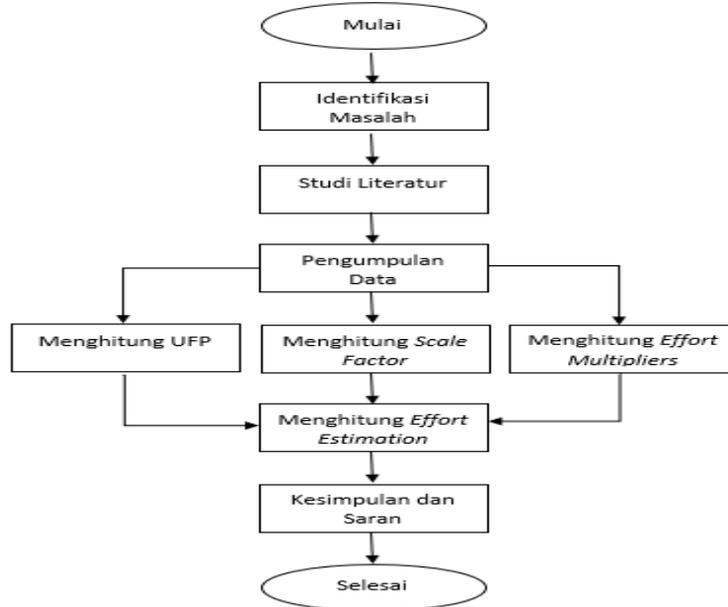
Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung harus mengetahui sumber daya yang diperlukan untuk menyelesaikan proyek Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT). Dengan mengetahui sumber daya yang tepat, Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung dapat menentukan rencana anggaran pengembangan sistem. Dengan mengetahui sumber daya yang diperlukan dapat menghindari adanya kekurangan atau kelebihan biaya, waktu, dan sumber daya manusia yang diperlukan.

Penelitian ini menggunakan metode COCOMO II (*Constructive Cost Model II*) guna mengetahui sumber daya berupa jumlah biaya, waktu, dan sumber daya manusia yang dibutuhkan dalam pengembangan Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung. COCOMO II adalah metode yang digunakan untuk menentukan estimasi sumber daya pengembangan sistem informasi yang menyediakan kerangka kerja untuk menghubungkan pengambilan keputusan dengan para *stakeholder* dalam proses pembuatan perangkat lunak [4].

## 2. METODOLOGI PENELITIAN

COCOMO II (*Constructive Cost Model II*) adalah metode perkiraan sumber daya yang obyektif untuk perencanaan dan pelaksanaan proyek-proyek perangkat lunak yang menghasilkan model estimasi yang dapat digunakan sebagai masukan atau acuan untuk estimasi pembangunan perangkat lunak berikutnya. COCOMO II merupakan hasil pengembangan dari COCOMO 81 yang dikembangkan oleh Barry

Boehm pada tahun 1981 dan dipublikasikan pada tahun 1997. Model ini memiliki kelebihan dapat memprediksi dengan jelas waktu, biaya, dan juga sumber daya untuk menyelesaikan proyek *software* [7]. Tahapan penelitian yang mengimplementasikan COCOMO II ini dijelaskan pada gambar 1 berikut.



Gambar 1. Diagram Alur Penelitian

### 2.1. Identifikasi Masalah

Unit Pelaksana Teknis (UPT) Balai Bahasa Universitas Lampung belum memiliki metode untuk menghitung estimasi biaya, waktu, dan sumber daya manusia perangkat lunak yang akan mereka kembangkan.

### 2.2. Studi Pustaka

Studi pustaka merupakan tahap pengumpulan literatur dan referensi dari *paper*, jurnal, dan buku untuk mendapatkan teori-teori dasar yang mendukung penelitian.

### 2.3. Pengumpulan Data

Pengumpulan data dilakukan dengan cara memberikan kuesioner yang diberikan kepada pengembang untuk mendapatkan penilaian *Scale Factor* dan *Effort Multiplier*. Data yang dikumpulkan digunakan untuk menghitung *Effort Estimation*.

### 2.4. Menghitung Unadjusted Function Point (UFP) dan *Source lines of Code* (SLOC)

*Unadjusted Function Point* (UFP) adalah metode pengukuran sekaligus satuan ukuran perangkat lunak dengan cara mengkuantifikasi fungsionalitas perangkat lunak yang disediakan untuk user berdasarkan pada desain logik [6]. Berikut ini adalah komponen *Function Point* menurut penelitian yang dilakukan oleh Maqdam dkk pada tahun 2019 [7].

Tabel 1. Komponen *Function Point*

| Komponen                             | Keterangan   |
|--------------------------------------|--|
| <i>External Input (EI)</i>           | <i>Input</i> berasal dari luar sistem, baik dari user maupun sistem lainnya yang selanjutnya digunakan untuk mengupdate <i>Internal Logical Files</i> .              |
| <i>External Output (EO)</i>          | <i>Output</i> merupakan data yang ditampilkan pada aplikasi untuk menyediakan Informasi kepada user, baik dalam bentuk laporan, tampilan di layar, pesan error, dst. |
| <i>External Inquiry (EQ)</i>         | <i>Inquiries eksternal</i> didefinisikan sebagai <i>input online</i> yang memicu respon dari <i>software</i> untuk menghasilkan <i>output online</i> .               |
| <i>Internal Logical File (ILF)</i>   | File logika internal merupakan data yang dikelompokkan secara logis, disimpan secara internal dan didapat dari input eksternal.                                      |
| <i>External Interface File (EIF)</i> | Data yang dikelompokkan secara logis namun berada diluar aplikasi yang menyediakan Informasi yang dibutuhkan aplikasi.   |

Nilai UFP ditentukan berdasarkan 5 komponen *function point* yaitu: *External Input (EI)*, *External Output (EO)*, *External Inquiry (EQ)*, *Internal Logical File (ILF)*, *External Interface File (EIF)*. Setelah setiap fungsi di analisis sesuai 5 komponen *function point*, maka UFP dihitung berdasarkan jumlah *Data Element Type (DET)*, *Record Element Type (RET)*, dan *Files Type References (FTR)*.

Nilai UFP yang telah didapatkan diubah ke dalam *Source lines of Code* (jumlah baris kode) menggunakan bahasa pemrograman berdasarkan QSM (*Quantitative Software Management*). QSM yang digunakan untuk mengkonversi UFP menjadi *Source lines of Code (SLOC)* dijelaskan pada tabel 2 [7].

Tabel 2. Bahasa Pemrograman Berdasarkan QSM

| Bahasa Pemrograman | QSM SLOC/FP Data |               |            |             |
|--------------------|------------------|---------------|------------|-------------|
|                    | <i>Average</i>   | <i>Median</i> | <i>Low</i> | <i>High</i> |
| FoxPro             | 36               | 35            | 34         | 38          |
| HTML *             | 34               | 40            | 14         | 48          |
| J2EE *             | 46               | 49            | 15         | 67          |
| Java *             | 53               | 53            | 14         | 134         |
| JavaScript *       | 47               | 53            | 31         | 63          |
| JCL *              | 62               | 48            | 25         | 221         |
| LINC II            | 29               | 30            | 22         | 38          |
| Bahasa Pemrograman | QSM SLOC/FP Data |               |            |             |
|                    | <i>Average</i>   | <i>Median</i> | <i>Low</i> | <i>High</i> |
| Lotus Notes *      | 23               | 21            | 19         | 40          |
| Natural *          | 40               | 34            | 34         | 53          |
| .NET *             | 57               | 60            | 53         | 60          |

|                |    |    |    |     |
|----------------|----|----|----|-----|
| Oracle *       | 37 | 40 | 17 | 60  |
| PACBASE *      | 35 | 32 | 22 | 60  |
| Perl *         | 24 | 15 | 15 | 60  |
| PL/I *         | 64 | 80 | 16 | 80  |
| PL/SQL *       | 37 | 35 | 13 | 60  |
| Powerbuilder * | 26 | 28 | 7  | 40  |
| REXX *         | 77 | 80 | 50 | 80  |
| Sabretalk *    | 70 | 66 | 45 | 109 |
| SAS *          | 38 | 37 | 22 | 55  |
| Siebel *       | 59 | 60 | 51 | 60  |
| SLOGAN *       | 75 | 75 | 74 | 75  |
| SQL *          | 21 | 21 | 13 | 37  |
| VB.NET *       | 52 | 60 | 26 | 60  |
| Visual Basic * | 42 | 44 | 20 | 60  |

Setelah itu nilai SLOC dibagi dengan 1000 untuk mendapatkan nilai KSLOC (*Kilo Source Line Of Code*) yang bisa dimasukkan dalam persamaan usaha.

## 2.5. Menghitung Scale Factor

*Scale Factor* (SF) adalah cara yang digunakan untuk menentukan usaha proyek dan karakteristik dari suatu proyek. Nilai *scale factor* dikumpulkan dari kuesioner yang sebelumnya telah diisi oleh pengembang yang terdiri dari 5 atribut yaitu *Precedentness* (PREC), *Development Flexibility* (FLEX), *Risk Resolution* (RESL), *Team Cohesion* (TEAM), *Process Maturity* (PMAT). Setelah didapatkan hasilnya, lalu dimasukkan dalam persamaan 1 untuk menghasilkan nilai faktor eksponen [7].

$$E = B + 0.01 \times SF \quad (1)$$

Keterangan:

E = faktor eksponen

B = nilai koefisien 0.91 (untuk COCOMO II.2000)

SF= total nilai *Scale Factor*

Berikut ini adalah 5 atribut *scale factor* menurut penelitian yang dilakukan oleh Aditya dkk pada tahun 2018 [1].

Tabel 3. Atribut *Scale Factor*

| <b>Scale Factor</b>         | <b>Deskripsi</b>   |
|-----------------------------|--|
| <i>Precedentness</i> (PREC) | Faktor skala ini merupakan faktor skala yang menggambarkan pengalaman masa lalu organisasi dengan proyek-proyek sejenis.   |
| <i>Flexibility</i> (FLEX)   | Faktor skala ini merupakan faktor skala yang menggambarkan kemampuan <i>klien</i> dalam menentukan tujuan dan mengkomunikasikan kebutuhan perangkat lunak kepada tim pengembang. |

|                                |  |
|--------------------------------|--|
| <i>Risk Resolution (RESL)</i>  | Faktor skala ini merupakan faktor skala yang menggambarkan seberapa sering perusahaan merencanakan manajemen resiko terhadap proyek yang dijalankan oleh perusahaan. Perencanaan manajemen resiko ini berguna untuk mengidentifikasi dan mencegah semua resiko yang mungkin terjadi. |
| <i>Team Cohesion (TEAM)</i>    | Faktor skala ini merupakan faktor skala yang menggambarkan seberapa baik tim pengembangan bekerja sama.  |
| <i>Process Maturity (PMAT)</i> | Faktor skala yang menggambarkan kematangan proses pengembangan perangkat lunak dalam organisasi. Hal ini didasarkan pada Model Kematangan Kemampuan Rekayasa Perangkat Lunak atau <i>Capability Maturity Model (CMM)</i> .   |

## 2.6. Menghitung Effort Multiplier

*Effort Multiplier* mempunyai 7 kategori dan 17 parameter dalam metode COCOMO II sub-model *Post- Architecture* yang digunakan untuk menentukan usaha nominal yang menggambarkan sistem informasi yang sedang dikembangkan. Berikut ini kategori dan atribut *Effort Multiplier* menurut penelitian yang dilakukan oleh Damayanti dkk pada tahun 2017 [3].

Tabel 4. Kategori dan Atribut *Effort Multiplier*

| No. | Kategori   | Atribut  |
|-----|--|--|
| 1.  | <i>RCPX (Product Reliability and Complexity)</i> | Penilaian <i>cost driver</i> terkait beberapa faktor dari produk yaitu :<br>Sejauh mana perangkat lunak menjalankan aplikasi sesuai fungsinya selama periode waktu tertentu ( <i>RELY</i> ).<br>Ukuran <i>database</i> yang digunakan yang dihitung menggunakan D/P ( <i>DATA</i> ).<br>Perangkat lunak dan perangkat keras dalam melakukan tugasnya seperti <i>platform</i> (arsitektur, sistem operasi, bahasa pemrograman dan antar muka yang terkait), sistem manajemen <i>database</i> , <i>browser</i> yang sesuai digunakan dalam menjalankan aplikasi ( <i>CPLX</i> ).<br>Kesesuaian dokumentasi proyek terhadap kebutuhan siklus hidup perangkat lunak ( <i>DOCU</i> ). |
| 2.  | <i>PERS (Personnel Capability)</i>               | Penilaian <i>cost driver</i> terkait beberapa kemampuan personel yaitu :<br>Kemampuan personel dalam analisis dan desain, efisiensi dan ketelitian, serta kemampuan untuk berkomunikasi dan bekerja sama. Dalam hal ini, dapat dinilai dari sertifikasi yang sudah didapatkan personel atau pengalaman kerja tim dalam suatu proyek ( <i>ACAP</i> ).<br>Kemampuan <i>programmer</i> dalam efisiensi penulisan kode program, ketelitian dan kemampuan untuk   |

|  |  |
|--|--|
|  | berkomunikasi dan bekerja sama sebagai sebuah tim. Seperti berapa banyak <i>programmer</i> tersebut terlibat dalam penyelesaian proyek (PCAP). Pergantian personel setiap tahun dalam menyelesaikan proyek. Semakin sedikit pergantian maka semakin tinggi skala (PCON).   |
| 3. <i>RUSE (Developed for Reusability)</i>     | <i>Cost driver</i> terkait tingkat upaya yang diperlukan untuk mengembangkan komponen yang digunakan kembali pada proyek-proyek yang sedang berjalan atau proyek di masa mendatang (RUSE).   |
| 4. <i>PDIF (Platform Difficulty)</i>           | Penilaian <i>cost driver</i> terkait beberapa kendala dalam pengembangan yaitu :<br>Persentase kendala waktu eksekusi yang diharapkan dapat digunakan pada sistem perangkat lunak (TIME).<br>Persentase tingkat kendala penyimpanan utama yang dikenakan pada sistem perangkat lunak (STOR).<br>Perubahan yang terjadi pada <i>hardware</i> dan <i>software</i> dalam kurun waktu tertentu (PVOL).                           |
| 5. <i>PREX (Personnel Experience)</i>          | Penilaian <i>cost driver</i> terkait beberapa pengalaman yaitu :<br>Pengalaman kerja tim pada suatu proyek pengembangan aplikasi sistem perangkat lunak atau subsistem (APEX).<br>Pemahaman tim dalam menggunakan platform, <i>interface database</i> , jaringan, <i>middleware</i> (PLEX).<br>Pengalaman tim dalam pemrograman dengan bahasa tertentu dan pemanfaatan CASE tool dalam mengembangkan perangkat lunak (LTEX). |
| 6. <i>FCIL (Facilities)</i>                    | Penilaian <i>cost driver</i> terkait penggunaan CASE tool dalam pengembangan perangkat lunak pada proyek, seperti dari mengubah kode yang sederhana menjadi terintegrasi (TOOL).<br>Cara komunikasi yang digunakan dalam pengembangan perangkat lunak (SITE).  |
| 7. <i>SCED (Required Development Schedule)</i> | Penilaian <i>cost driver</i> terkait tingkat persentase dari percepatan atau kemunduran jadwal terhadap jadwal suatu proyek yang telah ditetapkan sebelumnya (SCED).   |

Data yang telah didapatkan pada tahap pengumpulan data diolah untuk memperoleh hasil *Effort Estimation* yang dinyatakan dalam *Person-Month* (PM) berdasarkan persamaan 2.

$$PM = A \times (Size)^E \prod_{i=1}^{17} EM_i \quad (2)$$

A adalah nilai koefisien 2.94 untuk COCOMO II, 2000, *Size* adalah nilai KSLOC, E adalah nilai faktor eksponen (*Scale Factor*), dan EM adalah nilai dari *Effort Multiplier*. Hasil perhitungan tersebut kemudian digunakan untuk menghitung estimasi biaya, waktu, dan sumber daya manusia dalam pengembangan sistem informasi.

## 2.7. Menghitung Effort Estimation

Perhitungan *Effort Estimation* dilakukan untuk mengetahui estimasi biaya, waktu yang digunakan, dan jumlah sumber daya manusia yang diperlukan.

### 1. Perhitungan Waktu

Menghitung waktu yang diperlukan untuk mengembangkan sistem informasi dilakukan menggunakan persamaan 3 [7].

$$TDEV = C \times (PM)^{(D+0,2 \times (E-B))} \quad (3)$$

TDEV adalah *Time Development* atau waktu yang dinyatakan dalam hitungan bulan, PM adalah *Person-Month*, C adalah nilai 3,67 yang telah ditetapkan untuk COCOMO II, D adalah nilai 0,28 yang telah ditetapkan untuk COCOMO II, B adalah nilai 0,91 yang telah ditetapkan untuk COCOMO II, dan E adalah nilai faktor eksponen (*Scale Factor*) [7].

### 2. Perhitungan Orang (*Staff*)

Menghitung jumlah orang (*staff*) dapat menggunakan persamaan 4 [3].

$$Average\ Staff = PM/TDEV \quad (4)$$

*Average Staff* adalah jumlah pegawai dibutuhkan, PM adalah *Person-Month* dan TDEV adalah waktu (bulan).

### 3. Perhitungan Biaya per-Bulan

Total estimasi perhitungan biaya perbulan dapat dihitung menggunakan persamaan 5 [3].

$$Biaya\ per-Bulan = Average\ staff \times UMR \quad (5)$$

*Average Staff* adalah jumlah pegawai dibutuhkan dan UMR adalah upah minimum regional daerah.

### 4. Perhitungan Biaya Total

Menghitung total biaya keseluruhan dapat menggunakan rumus dari persamaan 6 [3].

$$Biaya\ total = Biaya\ per\ bulan \times TDEV \quad (6)$$

## 3. HASIL DAN PEMBAHASAN

### 3.1. Perhitungan *Unadjusted Function Point* (UFP) dan *Source Lines Of Code* (SLOC)

*Source Lines Of Code* (SLOC) dapat dihitung dengan mengetahui terlebih dahulu jumlah *Unadjusted Function Point* (UFP) dari sistem informasi. UFP didapatkan dari penelitian lain yaitu dari penelitian yang dilakukan oleh Yayan Budi Waskito. Terdapat dua UFP untuk Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung,

yaitu UFP dari seluruh fungsi sistem informasi sebelum dikembangkan dan UFP dari fungsi pengembangan sistem informasi. Tabel 5 merupakan UFP dari seluruh fungsi Sistem Informasi dan Tabel 6 adalah nilai UFP dari penambahan fungsi [9].

Tabel 5. UFP Seluruh Fungsi Sistem Informasi

| No. | Nama Fungsi                         | Tipe Fungsi | UFP Fungsi Transaksi | UFP Fungsi Data | UFP Total |
|-----|-------------------------------------|-------------|----------------------|-----------------|-----------|
| 1.  | Lihat informasi UPT                 | EQ          | 6                    | 20              | 26        |
| 2.  | Cek jadwal dan kuota EPT            | EQ          | 6                    | 25              | 31        |
| 3.  | Kirim pesan singkat                 | EI          | 3                    | 7               | 10        |
| 4.  | Cari hasil EPT                      | EQ          | 6                    | 37              | 43        |
| 5.  | Edit biodata diri                   | EI          | 6                    | 27              | 33        |
| 6.  | Daftar EPT                          | EI          | 6                    | 27              | 33        |
| 7.  | Cetak kartu peserta                 | EQ          | 7                    | 45              | 52        |
| 8.  | Lihat data nilai history EPT        | EQ          | 4                    | 10              | 14        |
| 9.  | Export laporan nilai peserta        | EQ          | 4                    | 12              | 16        |
| 10. | Lihat data nilai EPT                | EQ          | 4                    | 10              | 14        |
| 11. | Melihat data nilai peserta EPT      | EQ          | 7                    | 45              | 52        |
| 12. | Memasukan data nilai peserta EPT    | EI          | 6                    | 49              | 55        |
| 13. | Mengubah data nilai peserta EPT     | EI          | 6                    | 49              | 55        |
| 14. | Menghapus data nilai peserta EPT    | EI          | 6                    | 49              | 55        |
| 15. | Melihat data jadwal EPT             | EQ          | 7                    | 35              | 42        |
| 16. | Memasukan data jadwal EPT           | EI          | 6                    | 45              | 51        |
| 17. | Mengubah data jadwal EPT            | EI          | 6                    | 45              | 51        |
| 18. | Menghapus data jadwal EPT           | EI          | 6                    | 45              | 51        |
| 19. | Melihat data peserta EPT            | EQ          | 6                    | 20              | 26        |
| 20. | Memasukan data peserta EPT          | EI          | 4                    | 26              | 30        |
| 21. | Mengubah data peserta EPT           | EI          | 4                    | 26              | 30        |
| 22. | Menghapus data peserta EPT          | EI          | 4                    | 26              | 30        |
| No. | Nama Fungsi                         | Tipe Fungsi | UFP Fungsi Transaksi | UFP Fungsi Data | UFP Total |
| 23. | Verifikasi daftar ulang peserta EPT | EI          | 6                    | 37              | 43        |
| 24. | Melihat data akun ISEPT             | EQ          | 6                    | 20              | 26        |
| 25. | Memasukan data akun ISEPT           | EI          | 4                    | 26              | 30        |
| 26. | Mengubah data akun ISEPT            | EI          | 4                    | 26              | 30        |
| 27. | Menghapus data akun ISEPT           | EI          | 4                    | 26              | 30        |
| 28. | Melihat pesan guest                 | EO          | 4                    | 5               | 9         |
| 29. | Menghapus pesan guest               | EI          | 3                    | 7               | 10        |
| 30. | Melihat data layanan UPT            | EO          | 4                    | 5               | 9         |
| 31. | Memasukan data layanan UPT          | EI          | 3                    | 7               | 10        |

|                                |    |   |    |             |
|--------------------------------|----|---|----|-------------|
| 32. Mengubah data layanan UPT  | EI | 3 | 7  | 10          |
| 33. Menghapus data layanan UPT | EI | 3 | 7  | 10          |
| 34. Melihat data admin lain    | EO | 4 | 5  | 9           |
| 35. Memasukan data admin lain  | EI | 3 | 7  | 10          |
| 36. Mengubah data admin lain   | EI | 3 | 7  | 10          |
| 37. Menghapus data admin lain  | EI | 3 | 7  | 10          |
| 38. Melihat data pengumuman    | EQ | 4 | 10 | 14          |
| 39. Memasukan data pengumuman  | EI | 4 | 12 | 16          |
| 40. Mengubah data pengumuman   | EI | 4 | 12 | 16          |
| 41. Menghapus data pengumuman  | EI | 4 | 12 | 16          |
| 42. Melihat data event         | EQ | 4 | 10 | 14          |
| 43. Memasukan data event       | EI | 4 | 12 | 16          |
| 44. Mengubah data event        | EI | 4 | 12 | 16          |
| 45. Menghapus data event       | EI | 4 | 12 | 16          |
| 46. Melihat data staff         | EO | 4 | 5  | 9           |
| 47. Memasukan data staff       | EI | 3 | 7  | 10          |
| 48. Mengubah data staff        | EI | 3 | 7  | 10          |
| 49. Menghapus data staff       | EI | 3 | 7  | 10          |
| <b>Total UFP</b>               |    |   |    | <b>1219</b> |

Tabel 6. UFP Fungsi Tambahan Sistem Informasi

| No.       | Nama Fungsi                | Tipe Fungsi | UFP Fungsi Transaksi | UFP Fungsi Data | UFP Total |
|-----------|----------------------------|-------------|----------------------|-----------------|-----------|
| 1         | Melihat data nilai         | EQ          | 4                    | 10              | 14        |
| 2         | Melihat grafik universitas | EQ          | 6                    | 20              | 26        |
| 3         | Melihat grafik fakultas    | EQ          | 6                    | 20              | 26        |
| 4         | Melihat grafik jurusan     | EQ          | 6                    | 20              | 26        |
| 5         | Melihat grafik sarjana     | EQ          | 6                    | 20              | 26        |
| 6         | Melihat grafik diploma     | EQ          | 6                    | 20              | 26        |
| No.       | Nama Fungsi                | Tipe Fungsi | UFP Fungsi Transaksi | UFP Fungsi Data | UFP Total |
| 7         | Melihat grafik umum        | EQ          | 6                    | 20              | 26        |
| 8         | Export laporan nilai       | EQ          | 4                    | 12              | 16        |
| Total UFP |                            |             |                      |                 | 186       |

UFP pada sistem informasi ini dikalikan menggunakan QSM (*Quantitative Software Management*) SQL level data median dengan nilai 21. QSM level data median digunakan karena level data tersebut merupakan indikator yang dinilai paling akurat dibandingkan dengan level indikator yang lain (5), sehingga didapatkan nilai SLOC sebesar 29505. Nilai tersebut kemudian dibagi dengan 1000 untuk mendapatkan nilai KSLOC sehingga didapatkan angka 29,51.

### 3.2. Perhitungan *Scale Factor*

Nilai *Scale Factor* dikumpulkan dari kuesioner yang diisi oleh pengembang yang terdiri dari 5 atribut yaitu *Precedentness* (PREC), *Development Flexibility* (FLEX), *Risk Resolution* (RESL), *Team Cohesion* (TEAM), dan *Process Maturity* (PMAT).

Tabel 7. Penilaian *Scale Factor*

| Scale Factor | Hasil Penelitian |
|--------------|------------------|
| PREC         | 2,48             |
| FLEX         | 3,38             |
| RESL         | 2,83             |
| TEAM         | 0,55             |
| PMAT         | 1,56             |
| <b>Total</b> | <b>10,8</b>      |

Nilai dari *Scale Factor* (SF) telah diketahui yaitu sebesar 10,8. Nilai yang didapatkan dimasukkan kedalam persamaan 1 yang digunakan untuk menghitung jumlah faktor eksponen (E). Nilai faktor eksponen yang didapatkan untuk Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung adalah 1,02 dan termasuk kedalam skala disekonomi.

### 3.3. Perhitungan *Effort Multiplier*

Penilaian *Effort Multiplier* dalam penelitian ini menggunakan kuesioner penilaian yang berisi 17 faktor penggerak biaya (*cost driver*) dalam metode COCOMO II sub-model *Post- Architecture* yang diperlukan untuk pengembangan sebuah proyek perangkat lunak.

Tabel 8. Penilaian *Effort Multiplier*

| No. | <i>Effort Multiplier</i> | Responden | Hasil Penilaian |
|-----|--------------------------|-----------|-----------------|
| 1.  | RELY                     | High      | 1,10            |
| 2.  | DATA                     | Very High | 1,28            |
| 3.  | CPLX                     | Very High | 1,34            |
| 4.  | DOCU                     | Nominal   | 1,00            |
| 5.  | RUSE                     | Nominal   | 1,00            |
| 6.  | ACAP                     | Nominal   | 1,00            |
| 7.  | PCON                     | Very High | 0,81            |
| 8.  | TIME                     | Nominal   | 1,00            |
| 9.  | STOR                     | Nominal   | 1,00            |
| 10. | PVOL                     | Low       | 0,87            |
| 11. | PCAP                     | Nominal   | 1,00            |

|  |      |          |             |
|--|------|----------|-------------|
| 12.  | APEX | Nominal  | 1,00        |
| 13.  | PLEX | Very Low | 1,19        |
| 14.  | LTEX | Nominal  | 1,00        |
| 15.  | TOOL | Very Low | 1,17        |
| 16.  | SITE | Very Low | 1,22        |
| 17.  | SCED | Very Low | 1,43        |
| <b>Hasil Perkalian Total Effort Multiplier</b> |      |          | <b>3,23</b> |

### 3.4. Perhitungan Effort Estimation

*Effort Estimation* dinyatakan dalam *Person-Month* (PM). Perhitungan *Effort Estimation* dihitung menggunakan persamaan 2 sehingga menghasilkan nilai *Effort Estimation* sebesar 299,86. Perhitungan estimasi waktu yang diperlukan untuk mengembangkan sistem informasi dihitung menggunakan persamaan 3 dengan nilai PM=299,86 dan nilai E=1,02 sehingga didapatkan estimasi waktu sebesar 20 bulan. Setelah mendapatkan estimasi waktu yang dibutuhkan untuk mengembangkan sistem informasi, selanjutnya menghitung jumlah sumber daya manusia (*staff*) yang dibutuhkan untuk mengembangkan sistem informasi menggunakan persamaan 4 dengan nilai PM=299,86 dan nilai TDEV=20 sehingga diketahui jumlah sumber daya manusia (*staff*) yang dibutuhkan adalah **14,99** (dibulatkan menjadi 15 orang *staff*).

Perhitungan nilai estimasi biaya perbulan dihitung menggunakan persamaan 5 yaitu dengan mengalikan jumlah *staff* dengan UMP Lampung Tahun 2020 [5], sehingga didapatkan nilai estimasi biaya perbulan sebesar Rp36.480.023,6. Menghitung estimasi biaya total yang diperlukan menggunakan persamaan 6 yaitu dengan mengalikan biaya perbulan dengan *Time Development* (TDEV). Biaya total yang dibutuhkan untuk mengembangkan Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung yaitu sebesar Rp729.600.472.

## 4. SIMPULAN

Perhitungan estimasi waktu pengembangan Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung menggunakan metode COCOMO II menghasilkan estimasi waktu untuk mengerjakan proyek pengembangan sistem selama 20 bulan. Estimasi sumber daya manusia yang dibutuhkan dalam pengembangan sistem menggunakan metode COCOMO II adalah 15 orang *staff*. Hasil perhitungan estimasi biaya pengembangan menggunakan metode COCOMO II membutuhkan biaya sebesar Rp36.480.023,6 untuk setiap bulannya dan perhitungan estimasi biaya total yang dibutuhkan untuk mengerjakan pengembangan Sistem Informasi Pengelolaan Data *English Proficiency Test* (EPT) Unit Pelaksana Teknis (UPT) Bahasa Universitas Lampung selama 20 bulan menggunakan metode COCOMO II adalah sebesar Rp729.600.472.

## DAFTAR PUSTAKA

- [1] Aditya, P. M., Rachmadi, A., & Wicaksono, S. A., "Perhitungan Biaya Pembangunan Sistem Informasi Pariwisata Kota Sawahlunto Menggunakan Metode COCOMO II", *Jurnal Pengembangan Teknologi*

---

*Informasi Dan Ilmu Komputer (J-PTIHK) Universitas Brawijaya*, 2(11), 4785–4793, 2018.

- [2] Boehm, B., **“COCOMO II Model Definition Manual Version 2.1.s.I”**, University of Southern California, 2000.
- [3] Damayanti, D. E., Suprpto, & Perdanakusuma, A. R., **“Analisis Estimasi Biaya Pembuatan Perangkat Lunak Menggunakan Metode COCOMO II di Inagata Technosmith ( Studi Kasus : Sistem Informasi Monitoring dan Evaluasi Penerimaan Beasiswa Santri Berprestasi UIN Malang)”**, *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIHK) Universitas Brawijaya*, 1(10), 2017.
- [4] Fatonah, O. S., & Afrizal, Y., **Model Estimasi Biaya Perangkat Lunak Menggunakan Cocomo II (Studi Kasus PT. X)**. 40–46, 2013.
- [5] Gubernur Lampung, **“Keputusan Gubernur Lampung Nomor: G/776/V.07/HK/2019 Tentang Penetapan Upah Minimum Provinsi (UMP) Lampung Tahun 2020”**, 2019.
- [6] Laksamana, F. P., Perdanakusuma, A. R., & Saputra, M. C., **“Evaluasi Biaya Pengembangan Sistem Informasi Pengelolaan Arsip Surat ( SIPAS ) Menggunakan Function Point dan Object Point ( Studi Kasus : PT Sekawan Media Informatika)”**, *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIHK) Universitas Brawijaya*, 2(8), 2692–2701, 2018.
- [7] Maqдум, A. N., Perdanakusuma, A. R., & Putra, W. H. N., **“Implementasi Metode COCOMO II untuk Estimasi Biaya Pengembangan Perangkat Lunak di CV . Profile Image Studio”**, 3(6), 2019.
- [8] Suhartoyo, H., & Wijaya, T. A., **“Rancangan Estimasi Biaya dengan Teknik COCOMO II dan Neuro Fuzzy Studi Kasus : Sistem Informasi Rumah Sakit”**, *Inform*, 1(1). Retrieved from <http://ejournal.unitomo.ac.id/index.php/inform/article/view/215/118>, 2016.
- [9] Waskito, Y.B., **“Mengukur Nilai Fungsional Sistem Informasi Pengolahan Data English Proficiency Test (Ept) Universitas Lampung Menggunakan Metrik Function Point”**, Skripsi, Universitas Lampung, 2020.